Improving surface reconstruction in Shape from Shading using easy-to-set boundary conditions

L. Governi*, R. Furferi, L. Puggelli and Y. Volpe

Department of Industrial Engineering, University of Florence, via S. Marta, 50139 Firenze, Italy
E-mail: lapo.governi@unifi.it
E-mail: rocco.furferi@unifi.it
E-mail: luca.puggelli@unifi.it
E-mail: yary.volpe@unifi.it
* Corresponding author

Abstract: Minimization techniques are commonly adopted methodologies for retrieving a 3D surface starting from its shaded representation (image), i.e. for solving the widely known shape from shading (SFS) problem. Unfortunately, depending on the imaged object to be reconstructed, retrieved surfaces often result to be completely different from the expected ones. In recent years, a number of interactive methods have been explored with the aim of improving surface reconstruction; however, since most of these methods require user interaction performed on a tentative reconstructed surface which often is significantly different from the desired one, it is advisable to increase the quality of the surface, to be further processed, as much as possible. Inspired by such techniques, the present work describes a new method for interactive retrieving of shaded object surface. The proposed approach is meant to recover the expected surface by using easy-to-set boundary conditions, so that the human-computer interaction primarily takes place prior to the surface retrieval. The method, tested on a set of case studies, proves to be effective in achieving sufficiently accurate reconstruction of scenes with both front and side illumination.

Keywords: Shape from shading; minimization techniques; computational vision; boundary conditions; human-computer interaction.


Biographical notes: Governi L. PhD in Machine design and Construction (2002) - University of Florence, Italy. Graduated M.Sc. in Mechanical Engineering - University of Florence. After working as a post-doctoral researcher at the Department of Mechanics and Industrial Technologies of the University of Florence, in 2005 he assumed a Faculty position as Assistant Professor for the courses of “Reverse Engineering” and “Design and modelling methods”. His main scientific interests are: machine vision and reverse engineering, colorimetry, tools and methods for product design and development. He is author and co-author of many publications printed in international journals and participated to a number of international conferences. Some latest publications described techniques oriented towards the 3D reconstruction from orthographic views, vision-based product and process assessment and spline-based approximation of
L. Governi et al.

point clouds.

Furferi R., PhD in Machine design and Construction (2005) - University of Florence, Italy. Graduated M.Sc. in Mechanical Engineering - University of Florence. After working as a post-doctoral researcher at the Department of Mechanics and Industrial Technologies of the University of Florence, in 2008 he assumed a Faculty position as Assistant Professor for the course “Mechanical Drafting”. His main scientific interests are: development of artificial vision systems for industrial and textile control, artificial neural networks, colorimetry, reverse engineering and rapid prototyping. He is author and co-author of many publications printed in journals and presented on international conferences. Some latest publications described methods for color assessment of textiles, algorithms for 3D reconstruction of objects from orthographic views and ANN-based systems for industrial process.

Puggelli L., PhD candidate in “Design and development of industrial products and processes” - University of Florence, Italy. Graduated M.Sc. in Mechanical Engineering (2011) - University of Florence. His Doctorate Research deals with 2D to 3D conversion, Surface Reconstruction from Single Images, Shape from Shading, Reverse Engineering and Rapid Prototyping.

Volpe Y., PhD in Machine design and Construction (2006) - University of Florence, Italy. Graduated M.Sc. in Mechanical Engineering - University of Florence. He is currently working as Assistant Professor at the Department of Mechanics and Industrial Technologies - University of Florence. He is also Adjunct Professor of the course “Computational Graphics” from the Engineering Faculty of the University of Florence. His main scientific interests are: Computer Aided Design, Image Processing, Virtual Prototyping, FE simulation, Reverse Engineering and Rapid Prototyping. He is author and co-author of many publications printed in international journals and participated to a number of international conferences. Some latest publications described techniques for comfort-oriented design, machine vision-based systems for industrial processes and spline-based approximation of point clouds.

1 Introduction

As widely recognized, Shape from Shading (SFS) is an inverse problem of computer vision that leads to the reconstruction of three-dimensional surfaces represented on a single image, by analysing, pixel-by-pixel, the brightness (or shading) of the scene (Zhang et al. (1999); Durou et al. (2008); Horn (1970)).

The problem, known since late 60s (Zhang et al. (1999); Durou et al. (2008); Horn (1970); Rindfleisch (1966)), can be presented in terms of reconstruction of the normal map of the unknown surface, by using a simplified formulation, once the following assumptions are made: 1. the image representing the shapes to be reconstructed is assumed to be the result of the orthogonal projection of the scene on the focal plane of the observer (i.e. focal length is set at infinity); 2. the surface of the object is assumed homogeneous and perfectly diffusing (i.e. is a Lambertian surface); 3. such a surface is required not to present hidden parts (undercuts); 4. light source is supposed to be placed at infinity (parallel light); 5. the reference system $\Sigma_{xyz}$, that maps the three-dimensional reconstruction space, is set so that the plane $xy$ ($\Pi_{xy}$) lies on the focal plane and the $z$ axis heads towards the observer.
Improving surface reconstruction

If all of these hypotheses are verified, the surface brightness at each point, related to the surface normal vector, is expressed by the (well known) equation:

\[ \vec{L} \cdot \vec{N} = \frac{1}{\rho} I \]  

where: \( \vec{L} = [l_x, l_y, l_z] \) is the unit-vector opposed to light direction, \( \vec{N} = [n_x, n_y, n_z] \) is the outward unit-length vector normal to the surface, \( \rho \) is the albedo and \( I \) is the brightness.

This equation, where vector \( \vec{N} \) is the unknown, is usually expressed using surface gradient \( \nabla z \), resulting in the following non-linear partial differential equation (PDE):

\[ \frac{1}{\rho} I \sqrt{1 + |\nabla z|^2} + (l_x, l_y) \cdot \nabla z - l_z = 0 \]

Since 70s to nowadays, many methods have been explored in scientific literature in order to achieve the surface reconstruction from single shaded images (i.e. to provide a close solution to eqs. 1 and 2). As suggested in many works (Zhang et al. (1999); Durou et al. (2008)), such methods can be classified in four main different categories: direct (Horn (1970); Rindfleisch (1966); Rouy and Tourin (1992); Kimmel and Bruckstein (1992); Prados et al. (2006); Tankus et al. (2005, 2006)), minimization (Ikeuchi and Horn (1981); Brooks and Horn (1985); Frankot and Chellappa (1988); Horn (1989); Zheng and Chellappa (1991); Lee and Kuo (1993); Saito and Tsunashima (1994); Worthington and Hancock (1999); Daniel and Durou (2000); Courteille et al. (2006); Wu et al. (2008); Huang and Smith (2009); Di Angelo and Di Stefano (2012)), local approximation (Pentland (1984); Lee and Rosenfeld (1985)) and linear approximation approaches (Pentland (1988); Tsai and Shah (1994)).

Due to their flexibility and to extremely wide area of applicability, minimization techniques resulted to be among the most explored for solving SFS problem. Such techniques suppose that the expected surface is the one that minimizes a suitable functional, usually comprising the error between the (iteratively) reconstructed surface and the actual one. One of the most relevant advantages of these methods is that they are extremely robust in presence of image noise or imprecise settings - e.g. guessed light direction when unknown (Zhang et al. (1999); Durou et al. (2008); Worthington and Hancock (1999); Huang and Smith (2009)).

More precisely, the functional to be minimized is usually composed by the sum of several contributions, often called “constraints”, each one pulling the solution towards the respect of specific requirements. The main constraints, widely used in literature, are the following: brightness constraint, smoothness constraint and integrability constraint (Zhang et al. (1999); Durou et al. (2008)).

The first one, also known as “variation-to-data” constraint, requires that the reconstructed surface generates an image as close as possible to the input one under the same light conditions (i.e. the difference between the image of the reconstructed surface and the original image is required to be negligible). This component is always required in all of the minimization techniques.

Smoothness constraint implies that slope of the reconstructed surface changes gradually from a pixel to its neighbourhood, so that the solution results the smoothest possible. While, on one side, this constraint should avoid the surface to be irregular, on the other hand it can lead to over-smoothing (Daniel and Durou (2000)).
Integrability constraint, requires the solution to respect the principle of integrability (i.e. the surface height at any point needs to be independent from the path of integration), so that the obtained surface is “physically valid” (Horn (1989); Zheng and Chellappa (1991); Frankot and Chellappa (1988)).

Once the functional has been built, two main different strategies can be followed: 1. directly minimize the functional (Daniel and Durou (2000)) or 2. minimize the associated Euler function defined as the gradient of the functional itself (Ikeuchi and Horn (1981); Brooks and Horn (1985); Horn (1989); Zheng and Chellappa (1991); Frankot and Chellappa (1988)). While the direct approach usually requires a great computational effort and convergence results extremely slow, quite the opposite, indirect method results in faster solution convergence. Unfortunately for both methods the optimized solution not always corresponds to the global minimum of the functional (in other words the iterative process falls into functional local minima). In order to avoid this weak solution, a straightforward method is to carry out a global minimization processes, using heuristic techniques such as simulated annealing (Courteille et al. (2006)) or genetic algorithms (Saito and Tsunashima (1994)). However, time required for achieving a complete convergence is so long that the use of global methods is seldom feasible. In order to overcome some of the limitations of typical minimization techniques, usually operating in a completely-automatic way (i.e. without any intervention from user), in the last years a number of works have been proposed implementing user interaction, in order to guide the solution towards desired reconstructed surfaces. These approaches have been proposed with particular reference to situations where a more “practical” technique is necessary in order to obtain a 3D model qualitatively corresponding to a starting image which often considerably differs from an ideal one. In such works, users are allowed to manually impose a set of conditions to be respected by the final surface, such as height of local maxima, suitable initialization (Daniel and Durou (2000)) and/or others. Alternatively, once a preliminary solution has been obtained using an automatic procedure, users may manually modify the surface (often in terms of slope) in some particularly significant regions (Wu et al. (2008)) - i.e. to re-model the retrieved surface. In the first case, height needs to be known in a number of points, which is anything but obvious since surface height is the unknown to be found! In the second one, the time required for adjusting the results is the main drawback; moreover the reconstruction process results to be considerably “hand-crafted”.

Inspired by the most recent interactive procedures, the present work proposes a novel method for single image-based surface reconstruction. While the minimization is performed using classical minimization approaches, the reconstruction is guided by interactively imposing a number of easy-to-set boundary conditions (such as identification of regions for surface maxima and/or minima, slope conditions in correspondence of surface boundaries, etc.). In this way, the reconstructed surface is forced to respect a number of user supplied morphological characteristics. Once the interactive setting phase is completed, the reconstruction proceeds in a completely automatic way. From this point of view, the proposed method is meant to provide a reconstructed surface which might be successively refined, but which is considerably more similar to the final one compared to the solutions obtained by alternative techniques. The paper is organized as follows. In section 2 the development of the novel method is described step by step. The obtained results with reference to a set of case studies are shown and analysed in section 3. Finally, concluding remarks are provided in section 4.
2 Method

With the aim of providing a new method for surface reconstruction by using an interactive minimization process, the proposed work has been carried out according to the following steps: initial definition of the functional to be minimized, interactive boundary conditions setting, minimization process and surface reconstruction.

2.1 Functional Definition

The functional to be minimized can be defined (into the surface reconstruction domain $D$) as a linear combination of brightness ($B$) and smoothness ($S$) constraints:

$$
E = B + \lambda S = \sum_{i \in D} \left( \frac{1}{\rho} I_i - \vec{N}_i^T \cdot \vec{L} \right)^2 + \lambda \sum_{\{i,j\} \in D} \left( \vec{N}_i - \vec{N}_j \right)^2
$$

where:

- $i$ is the pixel index;
- $j$ is the index of a generic pixel belonging to the 4-neighbourhood of pixel $i$;
- $I_i$ is the brightness of pixel $i$ (range $[0 \rightarrow 1]$);
- $\vec{N}_i = [n_{i,x}, n_{i,y}, n_{i,z}]$, $\vec{N}_j = [n_{j,x}, n_{j,y}, n_{j,z}]$ are the unit length vectors normal to the surface (unknown of the optimization problem) in positions $i$ and $j$, respectively;
- $\lambda$ is a regularization factor for smoothness constraint (weight).

If, on one hand, the above functional formulation involves a large number of unknowns (i.e. the 3 elements of $\vec{N}_i$ for each pixel), on the other hand its expression is straightforward: since both the constraints ($B$ and $S$) are quadratic, the resulting functional is a quadratic form too. Let $\Phi$ be the column vector containing the elements of all $\vec{N}_i$ defined as follows:

$$
\Phi = [n_{1,x}, n_{1,y}, n_{1,z}, n_{2,x}, n_{2,y}, n_{2,z}, \ldots, n_{k,x}, n_{k,y}, n_{k,z}]
$$

where $k$ is the overall number of pixels. As a result, the functional can be rewritten in a matrix form:

$$
E = \frac{1}{2} \Phi^T A \Phi + \Phi^T b + c
$$
whose minimization can be carried out by minimizing its gradient:

$$\nabla E = A\Phi + b$$  \hspace{1cm} (6)

where $A$ (size $3k \times 3k$) results to be a symmetric matrix. The indirect minimization of the functional expressed in eq. 6 can be accomplished by applying well known linear methods such as Jacobi, Gauss-Seidel or Successive-Over-Relaxation (SOR); these methods are known to allow a very fast convergence to the optimized solution. Among such methods, Gauss-Seidel iteration with SOR (Ikeuchi and Horn (1981); Brooks and Horn (1985); Horn (1989)) has been used, since is proved to be particularly efficient for solving linear minimization problems. Unfortunately, the complexity of this process depends on the size of the input image: the larger is the number of unknowns $3k$, the larger are the required iterations to allow a complete convergence. Furthermore, the straightforward execution of SOR algorithm, without any boundary condition, may lead to incorrect reconstructed surface. Fortunately, as stated above, a proper set of boundary conditions is just the one the authors intend to impose in order to provide a better reconstruction and additionally to reduce the computational time.

2.2 Interactive boundary conditions setting

Once the matrix form of the functional has been built, the user is allowed to set different kinds of boundary conditions as described below. Such conditions are applied using typical image-processing based procedures (Liverani et al. (2010); Carfagni et al. (2011); Furferi et al. (2011,b)).

The effect of boundary conditions produces a considerable reduction in the number of unknowns. As a consequence, the reduced matrix formulation of the gradient to be minimized is provided by:

$$\nabla E_r = A_r\Phi + b_r$$  \hspace{1cm} (7)

For the sake of clarity, the interactive imposition of boundary conditions is explained referring to an image of a hemisphere with $\vec{L} = [-0.433, -0.25, 0.866]^T$ obtained with a light positioned with altitude angle of $60^\circ$ and azimuth angle of $30^\circ$.

**Background boundary condition**

User can set vertical unit length normal vector (only “normal” from now on) for all the pixels belonging to the background of the scene, so that it will be perfectly horizontal once reconstructed. While this operation is apparently unnecessary, it can rather reduce the dimension of the problem since the number of unknowns is considerably decreased (see Figure 2).

$$\vec{N}_i = [0 \hspace{0.2cm} 0 \hspace{0.2cm} 1] \forall i \in \text{background}$$  \hspace{1cm} (8)

**Silhouette contour boundary condition**

When the object represented may be clearly separated from the background, user is allowed to set the value of the normal around its silhouette (i.e. on its outline, see Figure 3),
Improving surface reconstruction

**Figure 2** Normal set by flat, horizontal background boundary condition. (to facilitate image comprehension only a subset of normals is shown)

as outward-pointing. This assumption is valid for all real objects, and it is meant to facilitate the rough reconstruction of the overall volume of the shape (Ikeuchi and Horn (1981); Brooks and Horn (1985)). Since the outline is obviously discrete, assuming it has been previously detected and vectorized in a succession of points stored in an array \( P_j = [x_j, y_j] \), this step can be carried out as follows:

- the angular coefficient \( t_j \) of the direction tangent to the outline for each of its points \( P_j = [x_j, y_j] \) is evaluated as the difference quotient between \( P_{j+1} = [x_{j+1}, y_{j+1}] \) and \( P_{j-1} = [x_{j-1}, y_{j-1}] \):

  \[
  t_j = \frac{y_{j+1} - y_{j-1}}{x_{j+1} - x_{j-1}}
  \]

  \( \text{(9)} \)

- the angular coefficient \( m_j \) of the \( j^{th} \) normal is obtained as:

  \[
  m_j = -\frac{1}{t_j}
  \]

  \( \text{(10)} \)

In order to avoid instabilities during surface height (\( z \)) recovering process (see Section 3), all of the components along \( z \)-axis should not be null. For this reason, a constant small positive value \( \epsilon \) is assigned to the third component of each imposed normal.

**White points boundary condition**

Analogously to the boundary conditions taken for the background points, user can set a specific normal vector for all the white points in the image (Ikeuchi and Horn (1981); Brooks and Horn (1985)); in fact, since for such points the brightness level of the image reaches its maximum value (equal to 1), the only viable solution is the one characterized by surface normal coincident with the unit vector \( \hat{L} \) (or, in other words by the unit vector opposed to light direction, as depicted in Figure 4).
Figure 3  Normals set by using silhouette contour boundary condition (to facilitate image comprehension only a subset of normals is shown)

Figure 4  Normals set by white points boundary condition (to facilitate image comprehension only a subset of normals is shown)

Morphology-based boundary condition

The last boundary condition to be set is referred to the morphology of the shape to be reconstructed; in fact, all the previously described conditions do not take into account concave-convex ambiguity, one of the most common issues in SFS problems. Accordingly, an appositely devised procedure has been carried out. In particular, users are required to specify, for a number of white regions in the original image (possibly for all of them), which ones correspond to local maxima or minima (in terms of surface height) figuring out the final shape as seen from an observer located in correspondence of the light source. Once such points are selected, the algorithm automatically evaluates a proper set of normals in the neighbourhood of the regions as explained below. In order to better explain the devised procedure, it is useful to define the following set of notations (see Figure 5):

- $\vec{l}_{xy}$: projection of light direction $\vec{l}$, (with $\vec{l} = -\vec{L}$), on $\Pi_{xy}$;
- $\theta$: angle between $\vec{l}_{xy}$ and $x$-axis (azimuth);
Improving surface reconstruction

- \( \delta \): angle between \( \vec{L} \) and \( z \)-axis;
- \( \gamma \): altitude of \( \vec{L} \) i.e. the angle complementary to \( \delta \);
- \( \vec{r} \): normal to the plane \( \Pi \) defined by \( \vec{L} \) and \( z \)-axis, so that \( \vec{r} = \frac{\vec{L} \times \vec{z}}{\text{norm}(\vec{L} \times \vec{z})} \);
- \( \Sigma_{x'y'z'} \): coordinate system obtained rotating \( \Sigma_{xyz} \) around \( \vec{r} \) by the angle \( \delta \) (i.e. \( z \)-axis is coincident with \( \vec{L} \));
- \( \Pi_{x'y'} \): plane defined by \( x' \) and \( y' \) axis in \( \Sigma_{x'y'z'} \).

Since images are discrete representations of real scenes, maxima (or minima) regions are necessarily represented by finite areas, generally described by more than one pixel. Since the normal of each white pixel inevitably coincides with \( \vec{L} \), as previously stated, each white region defines a portion of plane \( \Pi \) (see Figure 5) delimited by the border of the white region. Starting from this observation, referring to \( \Sigma_{x'y'z'} \) reference system, it is possible to define the normal \( \vec{N}' \) for all the pixels defining the border of each local maximum (or minimum) region. In particular, the projections \( n'_{x} \) and \( n'_{y} \) on the plane \( \Pi \) of these normals are orthogonal to the border and point outwards in case of local maxima and inwards otherwise. As a consequence, it is possible to determine both the sign and the direction of such normals projections (i.e. the ratio \( m \) between \( y' \) and \( x' \) components).

Since every normal \( \vec{N}' \) must satisfy the irradiance equation (eq. 1), it is possible to write in the reference system \( \Sigma_{x'y'z'} \):

\[
\vec{N}'^T \cdot \vec{L}' = I
\]  

(11)

where:

\[
\vec{L}' = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\]

(12)

\[
\vec{N}' = \begin{bmatrix} n'_{x} \\ n'_{y} \\ n'_{z} \end{bmatrix} = \begin{bmatrix} n_{x'} \\ m \cdot n_{x'} \\ n_{z'} \end{bmatrix}
\]

(13)

combining eq.11 with eqs. 12 and 13:

\[
n_{z'} = I
\]

(14)

moreover, by definition:

\[
\| \vec{N}' \| = \sqrt{n'^2_{x'} + (m \cdot n_{x'})^2 + I^2} = 1
\]

(15)

therefore:

\[
n_{x'} = \pm \sqrt{\frac{1 - I^2}{1 + m^2}}
\]

(16)
finally:

\[
\vec{N}' = \begin{bmatrix}
\pm \sqrt{\frac{1-I^2}{1+m^2}} \\
\pm m \cdot \sqrt{\frac{1-I^2}{1+m^2}}
\end{bmatrix}
\]  

(17)
Once the normals $\vec{N}'$ are evaluated, the actual value of $\vec{N}$ in the scene reference system is easily obtained as follows:

$$\vec{N} = R(-\delta, \vec{r}) \cdot \vec{N}' = R(-\delta, \vec{r}) \cdot \left[ \pm \frac{1}{\sqrt{1+m^2}} \begin{pmatrix} \pm \sqrt{\frac{1-I^2}{1+m^2}} \\ \mp \frac{m}{\sqrt{1+m^2}} \end{pmatrix} \right]$$

(18)

where $R(\delta, \vec{r})$ is the matrix that maps the inverse rotation along axis $\vec{r}$ by the angle $\delta$. For each $\vec{N}$ defined in eq. 18, the only unknown is the parameter $m$. With the aim of determining such parameter, the knowledge of the coordinates of the actual pixels defining the white areas borders (as seen by an observer positioned in correspondence to the light source) is required. As a consequence the following further steps are necessary:

1. Extraction of coordinates $(x, y)$ of the pixels defining the border of the considered area on the image.

2. Rotation of the points with coordinates $(x, y)$ by the angle $\theta$ around $z$. As a consequence it is possible to express the border coordinates in a new reference system $\tilde{\Sigma}_{\tilde{x}\tilde{y}z}$ (note that $z \equiv \tilde{z}$) where $\tilde{\Sigma}_{xy}$ is aligned with $\tilde{x}$. In such a reference system it is possible to write (see Figure 6a):

$$\nabla z(\tilde{x}, \tilde{y}) = [\tan(\gamma), 0]$$

(19)

3. Recovering of a relative height $z_{rel}$ referring to the minimum height value, which corresponds to the point of minimum $x$-coordinate $\tilde{x}_{min}$ in the rotated configuration (see Figure 6a). Accordingly is possible to state that:

$$z_{rel} = \tan(\gamma) \cdot \Delta \tilde{x}$$

(20)

where:

$$\Delta \tilde{x} = \tilde{x}_i - \tilde{x}_{min}$$

(21)

This assumption is true only if border points and white points lie on the same plane; even if this condition is not satisfied (border pixels are not white), the relative slope between each one of them and $\Pi_i$ proves to be practically negligible.

4. Rotation of the coordinates $(x, y, z_{rel})$ by the angle $\delta$ along $\vec{r}$ (see Figure 6b); the resulting coordinates define the outline in the reference system $\Sigma_{x'y'z'}$, i.e.:

$$\begin{bmatrix} x' \\ y' \\ z'_{rel} \end{bmatrix} = R(\delta, \vec{r}) \cdot \begin{bmatrix} x \\ y \\ z_{rel} \end{bmatrix}$$

(22)

It has to be noticed that, since all the analysed points were supposed to lay on the same plane $\Pi_i$, whose normal coincides with $z$-axis, at the end of these transformations, each $z$ coordinate must result null as depicted in Figure 6b.

5. Once the $(x', y')$ coordinates are obtained, the evaluation of $m$ proceeds as seen previously (eqs. 9 and 10), so that normals to be set are evaluated (see Figure 7).
2.3 Normal map and height evaluation

Once the reduced formulation (see eq. 7) of the gradient to be minimized is obtained, the normal map recovering is carried out by using Gauss-Seidel with SOR minimization.
Improving surface reconstruction technique. The obtained normal map is then processed in order to recover the depth map of the surface. As suggested by Wu et al. (2008), this step can be optimally performed by minimizing the following energy function:

\[ E_2 = \sum_{i,j} ((z_i - z_j) - q_{ij})^2 \]  

where \( z_i \) and \( z_j \) are respectively the heights relative to pixel \( i \) and \( j \), while \( q_{ij} \) is the relative height between the two points, calculated by fitting an osculating arc between the two normals \( \vec{N}_i \) and \( \vec{N}_j \). Actually, the obtained surface does not correspond exactly to the normal map. This is due to the fact that in eq. 23 a strong condition of integrability is imposed. However, this method proves to be a smart way of applying the integrability constraint (Horn (1989); Zheng and Chellappa (1991); Frankot and Chellappa (1988)) thus allowing a good retrieval of the final surface. In Figure 8 the final result of the procedure for the exemplificative image is shown.

Figure 8  Recovered hemisphere

3 Case studies

In order to test the performance of the proposed approach, results of shape reconstruction for a series of case studies are shown and compared with the ones obtained without setting boundary conditions.

3.1 Donut

The first case study consists of a synthetic image (size 192 × 192 pixels) of a frontally illuminated donut (see Figure 9a), obtained as the top view of the correspondent surface depicted in Figure 9b. Without setting any boundary condition and using a regularization factor \( \lambda \) equal to 0.01, the reconstructed surface (see Figure 9c) strongly differs from the ground truth one. By interactively setting the boundary conditions (as depicted in Figure 9d) and using the same regularization factor, the obtained surface (Figure 9e) is much more similar to the real shape.
Analogously, Figure 10c shows the recovered shape obtained starting from the image of a synthetic donut with oblique light direction ($\vec{L} = [-0.433, -0.25, 0.866]^T$) shown in Figure 10a. The regularization factor $\lambda$ is set equal to 0.1. It has to be noticed that the input image is characterized by a number of black pixels for which SFS fails in surface
Improving surface reconstruction

reconstruction. In Figure 10b the result obtained without setting boundary conditions is shown; also in this case the obtained shape results to be inaccurate.

Figure 10  (a) synthetic image; (b) surface reconstructed without boundary conditions; (c) surface reconstructed with the proposed method

3.2 Matlab® peaks

The second selected case study is a synthetic image (Figure 11a) of frontally illuminated Matlab® Peaks surface (Figure 11b). The size of the input image is $220 \times 220$ pixels; the regularization factor $\lambda$ is set equal to 0.01. Differently from previous case studies, the scene does not represent an object distinct from the background; it is most like a mountainous surface instead, characterized by 3 peaks and 2 valleys. Accordingly, in this case, both background and silhouette contour boundary conditions are not imposed. The result obtained without setting any boundary conditions (see Figure 11c) is unsatisfactory, quite the opposite the one obtained using the proposed method (see Figure 11d) is very close to the ground truth surface.
Figure 11 (a) synthetic image; (b) ground truth surface; (c) surface reconstructed without boundary conditions; (d) surface reconstructed with the proposed method

3.3 Coin

The last case study is a real image (photograph) of the bas-relief impressed on the face of a coin (see Figure 12a). Since in this case the light direction is unknown, the value of the vector \( \vec{L} \) has been estimated to be \( \vec{L} = [0.408, 0.408, 0.816]^T \) by comparing the coin image with a set of differently illuminated spheres. The size of the input image is 353 \( \times \) 274 pixels and the regularization factor \( \lambda \) is set equal to 0.05. Differently from the previous case study, all possible boundary conditions are set. Being a real world case-study, the input image is unescapably affected by noise, so that the reconstruction is not trivial. Nevertheless, once the boundary conditions are properly set, the minimization procedure allows a good reconstruction (see Figure 12c), while the unconstrained solution is completely wrong (see Figure 12b).

3.4 Boundary conditions contributions

In order to better define the contribution brought by different boundary conditions, a number of tests have been carried out; the relevance of the Morphology Based boundary condition has been particularly investigated since its definition is one of the main improvements to the SFS method provided by the authors in the present work. The analysis has been performed by alternatively using the following four sets of boundary conditions:
Improving surface reconstruction

Figure 12  (a) real image; (b) surface reconstructed without boundary conditions; (c) surface reconstructed with the proposed method

(a) Background and White Points boundary conditions (basic set);
(b) Basic set plus Silhouette Contour boundary condition;
(c) Basic set plus Morphology Based boundary condition;
(d) All the boundary conditions.

Background and White Points boundary conditions do not contribute to guide the solution process towards the expected solution; but, rather, their merit is to reduce (sometimes significantly) the overall number of unknowns. Accordingly the cases (b) and (c) are the ones to be investigated in order to spot the individual contribution of, respectively, Silhouette Contour and Morphology Based boundary conditions to the surface retrieval. Finally, the last case (d) is useful to determine the importance of the simultaneous use of both the boundary conditions.
Two case studies, “Peaks” and “Donut”, are examined in this section and the results obtained by imposing the established sets of boundary conditions are discussed. For both cases, the starting point consists of a synthetic image obtained using Matlab® and choosing a frontal illumination.

As explained in Section 3.2 for “Peaks” case only the sets (a) and (c) are applicable (see Table 1).

<table>
<thead>
<tr>
<th>Set case</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peaks</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>Donut</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

The analysis of the obtained surface (see Figure 13) highlights the importance of the Morphology Based boundary condition; in fact the surface obtained by imposing set (a) is considerably different from the ground-truth. Conversely, the surface obtained by adding Morphology Based boundary condition closely resembles the expected one.

Refering to the “Donut” case, the results obtained using all the 4 possible sets can be investigated. Results obtained by imposing sets (a) and (b) are markedly different from ground truth. Once again, the outcome can be improved by imposing the Morphology Based boundary condition (i.e. set (c)): its contribution to provide satisfactory results is clearly shown in Figure 14. In any case, the best results are achieved if all the boundary conditions are contemporarily chosen (set (d)).
Figure 14 results obtained using basic set (a), basic set plus Silhouette Contour boundary condition (b), basic set plus Morphology Based boundary condition (c), and all boundary conditions (d)

4 Conclusion

The present work described a novel method for single image based surface reconstruction using minimization approaches. The reconstruction is performed by setting a number of easy-to-set boundary conditions chosen by the user. In particular, background, silhouette contour, singular point and morphology-based boundary conditions are set by means of human-computer interaction, so that the functional to be minimized is properly modified and the surface retrieval is assisted.

The devised method relies on the use of classical minimization techniques since they are recognized to be one of the most suitable ways for solving SFS problem, especially when the image representing the surface to be reconstructed is noisy. Though a number of interactive methods can be found in scientific literature, interaction is typically needed after the reconstruction is accomplished; in other words, users are allowed to re-model the retrieved surface after a first-attempt reconstruction is performed. Conversely, in the proposed method user interaction is purposeful for allowing a better surface retrieval so that more realistic surfaces are reconstructed. As a consequence, the need of possible surface post-processing is strongly reduced even though, mostly in case of real world images, it cannot generally be avoided.

The method has been tested against a set of case studies (synthetic and “real” images) and results have been compared with unconstrained methods. The proposed approach proved to be effective for surface retrieval both for frontally and laterally illuminated scenes. Note that the surfaces reconstructed by means of the proposed methodology are not claimed to be dimensionally accurate with respect to the ground truth (nor this kind of accuracy is the
goal of this work); the effectiveness of the methodology is rather evaluated by qualitatively comparing the obtained results with the ones coming from alternative approaches.

Future work will be addressed to increase the number of test cases, with particular regard to noisy images; this will allow to stress method’s possible drawbacks and to conceive possible improvements. Moreover, the authors are aimed to implement the proposed interactive method using non-linear minimization techniques; this will allow to add non-linear constraints such as the integrability one. At the same time, further investigation of SOR method will be accomplished by implementing different kind of initialization. By a way of example, pre-processing of images by using filtering could allow a preliminary reconstruction of the surface to be used for initializing the minimization. Furthermore, since SOR method is known to be error-prone, especially for lateral illumination, the implementation of different minimization algorithms will be investigated.

Acknowledgement

Authors wish to thanks Matteo Baldini for his contribute in supporting the implementation and testing of some of the algorithms proposed in the present work.

References


Improving surface reconstruction


