Numerical Methods for Shape-from-shading: A New Survey with Benchmarks

Jean-Denis Durou, Maurizio Falcone, Manuela Sagona

Jean-Denis Durou, Maurizio Falcone, Manuela Sagona. Numerical Methods for Shape-from-shading: A New Survey with Benchmarks. Computer Vision and Image Understanding, 2008, 109 (1), pp.22–43. hal-04587417

Numerical Methods for Shape-from-shading: A New Survey with Benchmarks ⋆

Jean-Denis Durou [a]

[a] *Institut de Recherche en Informatique de Toulouse, Université Paul Sabatier*
118, route de Narbonne, 31062 Toulouse Cedex 4, France.

Maurizio Falcone and Manuela Sagona [b]

[b] *Dipartimento di Matematica, Università di Roma "La Sapienza"*
Piazzale Aldo Moro, 2, 00185 Roma, Italy.

Abstract

Many algorithms have been suggested for the shape-from-shading problem, and some years have passed since the publication of the survey paper by Zhang *et al.* [1]. In this new survey paper, we try to update their presentation including some recent methods which seem to be particularly representative of three classes of methods: methods based on partial differential equations, methods using optimization, and methods approximating the image irradiance equation. One of the goals of this paper is to set the comparison of these methods on a firm basis. To this end, we provide a brief description of each method, highlighting its basic assumptions and mathematical properties. Moreover, we propose some numerical benchmarks in order to compare the methods in terms of their efficiency and accuracy in the reconstruction of surfaces corresponding to synthetic, as well as to real images.

Key words:
shape-from-shading, eikonal equation, numerical methods, algorithms comparison.

⋆ This work was partially supported by the MURST Funds "Scientific Computing" and by the TMR Network "Viscosity Solutions and Applications" (contract FMRX-CT98-0234).
We wish to thank Frédéric Courteille for his help in the development of the website `www.irit.fr/sfs`.

Email addresses: `durou@irit.fr` (Jean-Denis Durou),
`{falcone,sagona}@mat.uniroma1.it` (Maurizio Falcone and Manuela Sagona).

Preprint submitted to Elsevier Science *22 October 2007*

1 Introduction

A great number of papers on the shape-from-shading (SFS) problem have appeared since the publication of the classical study by Horn and Brooks [2]. This is probably due to the fact that, despite the simplicity of its formulation, the SFS problem deserves analysis and new approaches, since a global method for its resolution under realistic assumptions is still lacking. Many technical questions (*e.g.*, the uniqueness of solutions without continuity assumptions) remain open. Moreover, new mathematical tools and numerical techniques have appeared in the last decade, so it seems to us appropriate to update the scenario. Some of the new methods involve non-smooth solutions and several types of boundary conditions, guarantee convergence to an approximate solution under rather broad assumptions, are reasonably fast and in some cases can be extended to deal with dark shadows (*i.e.*, black spots) in the image.

In this paper, following Zhang *et al.* [1], we review a number of methods for the SFS problem which seem to be the most representative in three classes: methods of resolution of partial differential equations, methods using minimization, and methods approximating the image irradiance equation. We survey the methods, taking into account the mathematical formulation behind the problem, the tools that are used to compute the solution, their features and the assumptions needed by each of them. After this analysis, we select three methods (one per class) which can be applied to the orthographic SFS problem.

Another goal of this paper is to set up a collection of significant tests which can be used to compare the methods, giving users more precise information about their accuracy in terms of several quantitative indicators. To this end we have chosen a set of different types of images which includes smooth and non-smooth, as well as synthetic and real images. It should be noted that four of those images have already been proposed in [1]. We have also tried to define a rigorous methodology to compare the algorithms.

All this information will be useful to the reader who has to choose between different SFS methods. Moreover, a companion web site of this paper is available (`www.irit.fr/sfs`), that was constructed by the authors. It contains an SFS bibliography with more than 1100 BibTex entries, a web interface allowing the tests presented in this paper to be performed on-line, and a portable platform with all the codes used for the tests. We hope that this effort will be appreciated by the community and that many other authors will contribute to the construction of that web site, increasing little by little the number of benchmarks and providing their codes.

To the best of our knowledge, this is one of the first attempts to compare the algorithms in such a precise way, apart from the survey paper by Zhang *et al.* [1], where the comparisons are mainly of a qualitative nature. It should also be mentioned that in other papers, a description of some existing methods can be found, *e.g.* [3–7].

The paper is organized as follows. In Section 2, we present a brief outline of our model problem, the orthographic SFS problem. Section 3 is devoted to a short description of the main classes of methods and of what we think are the most representative algorithms in every class. In Section 4, three methods (one per class) are selected for the tests. In Section 5, we describe our methodology to compare these methods, and give the mathematical definitions of the errors, as well as some hints at reconstructing the physical quantities of the model (normals and greylevels), starting from the shapes computed by every algorithm. Section 6 contains the tests including error tables, figures and comments. Finally, in Section 7 we summarize the results of our analysis.

2 Problem Formulation

2.1 Shape-from-Shading Equations

We start by giving a brief outline of the SFS problem and introducing the basic assumptions. We attach to the camera a three-dimensional coordinate system $(Oxyz)$, such that Oxy coincides with the image plane and Oz coincides with the optical axis. Under the assumption of orthographic projection, the visible part of the scene is, up to a scale factor, a graph $z = u(\mathbf{x})$, where $\mathbf{x} = (x, y)$ is an image point. As is well known [2], the SFS problem can be modeled by the "image irradiance equation":

$$R(\mathbf{n}(\mathbf{x})) = I(\mathbf{x}), \tag{1}$$

where $I(\mathbf{x})$ is the greylevel measured in the image at point $\mathbf{x}$ (in fact, $I(\mathbf{x})$ is the irradiance at point $\mathbf{x}$, but both quantities are proportional) and $R(\mathbf{n}(\mathbf{x}))$ is the reflectance function, giving the value of the light re-emitted by the surface as a function of its orientation *i.e.*, of the unit normal $\mathbf{n}(\mathbf{x})$ to the surface at point $(\mathbf{x}, u(\mathbf{x}))$. This normal can easily be expressed as:

$$\mathbf{n}(\mathbf{x}) = \frac{1}{\sqrt{1 + p(\mathbf{x})^2 + q(\mathbf{x})^2}} (-p(\mathbf{x}), -q(\mathbf{x}), 1), \tag{2}$$

where $p = \partial u/\partial x$ and $q = \partial u/\partial y$, so that $\nabla u(\mathbf{x}) = (p(\mathbf{x}), q(\mathbf{x}))$. Irradiance function I is the datum in the model since it is measured at each pixel of the image, for example in terms of a greylevel (from 0 to 255). To construct a continuous model, we will assume that I takes real values in the interval $[0, 1]$.

Height function u, which is the unknown of the problem, has to be reconstructed on a compact domain $\Omega \subset \mathbb{R}^2$, called the "reconstruction domain". Assume that there is a unique light source at infinity whose direction is indicated by the unit vector $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$. Also assume for simplicity that $\boldsymbol{\omega}$ is given (in some works, $\boldsymbol{\omega}$ as well is considered as unknown, see *e.g.* [8,9], even if this new problem is sometimes ill-posed [10]). Recalling that, for a Lambertian surface of uniform albedo equal to 1, $R(\mathbf{n}(\mathbf{x})) = \boldsymbol{\omega} \cdot \mathbf{n}(\mathbf{x})$, Eq. (1) can be written, using (2):

$$I(\mathbf{x})\sqrt{1 + |\nabla u(\mathbf{x})|^2} + (\omega_1, \omega_2) \cdot \nabla u(\mathbf{x}) - \omega_3 = 0 \qquad \text{for } \mathbf{x} \in \Omega, \tag{3}$$

which is a first order non-linear partial differential equation (PDE) of the Hamilton-Jacobi type. Points $\mathbf{x} \in \Omega$ such that $I(\mathbf{x})$ is maximal correspond to the particular situation where $\boldsymbol{\omega}$ and $\mathbf{n}(\mathbf{x})$ point in the same direction: these points are usually called "singular points".

Let us mention that Eq. (3) is not the most general equation of SFS [11]: since real materials are not purely Lambertian, some publications are concerned with non-Lambertian SFS problems [12–14]; moreover, the situation is more complex in the presence of other lighting models [15,16] or when the interreflections are taken into account [17,18]. We will also consider the equation which appears in most of the papers and corresponds to frontal light source at infinity *i.e.*, $\boldsymbol{\omega} = (0, 0, 1)$. Then (3) becomes the "eikonal equation":

$$|\nabla u(\mathbf{x})| = f(\mathbf{x}) \qquad \text{for } \mathbf{x} \in \Omega, \tag{4}$$

where:

$$f(\mathbf{x}) = \sqrt{\frac{1}{I(\mathbf{x})^2} - 1}. \tag{5}$$

In the last few years, new models have appeared. The main goal of those models is to modify the classical assumptions, in order to deal with real-life applications. In fact, although the classical SFS problem has attracted many researchers for years, its impact on applications has been rather limited. The major modification that has been considered is to replace the usual assumption that the projection of scene points during a photographic process is orthographic, with the more realistic assumption of perspective projection. Very few papers had considered perspective projection [19–23], and none of them had established the equation of this more realistic model. The new model problem is called "perspective shape-from-shading" (PSFS). Recently, three papers have established almost simultaneously the PSFS equation [24–26], which is still a non-linear PDE. Moreover, in [22,27], the light source is no longer assumed to be located at infinity, but at the center of projection.

2.2 Additional Equations

Equations (3) or (4) are sometimes complemented with boundary conditions on $\partial\Omega$ or with additional information to select a unique solution. One can set up a boundary value problem which imposes either a value to the solution u (Dirichlet type boundary condition), or a value on the normal derivative (Neumann type boundary condition), or a so-called "state constraint" boundary condition where one imposes an equation to be satisfied on the boundary. For an image containing an "occluding boundary", it is usual to use it as boundary $\partial\Omega$ of the reconstruction domain Ω. For example, in Fig. 1, if the part of the image representing the object in greylevels ("silhouette") is Ω, then $\partial\Omega$ coincides with the occluding boundary.

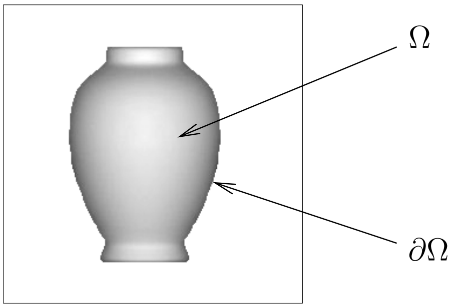


Fig. 1. Object with occluding boundary, which might be used as boundary $\partial\Omega$.

A current choice is to consider Dirichlet type boundary conditions in order to take into account (at least) two different possibilities. The first corresponds to the assumption that the surface is standing on a flat background *i.e.*, we set:

$$u(\mathbf{x}) = 0 \qquad \text{for } \mathbf{x} \in \partial\Omega. \tag{6}$$

The second possibility occurs when the height of the surface on the occluding boundary is known. This situation leads to the more general condition:

$$u(\mathbf{x}) = g(\mathbf{x}) \qquad \text{for } \mathbf{x} \in \partial\Omega. \tag{7}$$

The solution of Eq. (3) or of the Dirichlet problems (3)-(6) or (3)-(7) will give the surface corresponding to greylevel $I(\mathbf{x})$ measured in Ω. Let us now review the methods of resolution of these problems.

3 Review of Shape-from-Shading Methods

A large number of SFS methods which use a great variety of mathematical tools are available. We classify the methods into three classes: methods of resolution of PDEs, methods using optimization, and methods approximating the image irradiance equation (a very similar classification is proposed in [1]).

3.1 Methods of Resolution of PDEs

Equations (3) and (4) have attracted much attention in the research community in PDEs for their wide range of applications. In the framework of the SFS problem, several methods of resolution have been tested: characteristic strips expansion, approximation of viscosity solutions, etc.

3.1.1 Characteristic Strips Expansion

The first mention of 3D reconstruction using photometric cues is due to the Dutch astronomer Van Diggelen [28]. The first resolution was suggested by Rindfleisch [29], who demonstrated that if the photometric behaviour of a surface follows certain properties, then the shape can be expressed as an integral along a set of convergent straight lines. He implemented this computation on images of the Moon, claiming that its surface verifies the necessary photometric properties reasonably well. Later, Horn suggested calling this problem "shape-from-shading", and showed that the resolution proposed by Rindfleisch in a particular case could be generalized, while still using the characteristic strips expansion [30], under the following two conditions: (i) the function u has to be of class C^2; (ii) the 5-uplet (x, y, u, p, q) has to be known at every point of a curve called the "initial curve", which means in fact that two boundary conditions are needed simultaneously, one on u (Dirichlet boundary condition) and the other on (p, q) (Neumann boundary condition). The "characteristic lines" (which are the lines along which the integration has to be performed) can be of any form in the image plane, and this differs from the case studied by Rindfleisch.

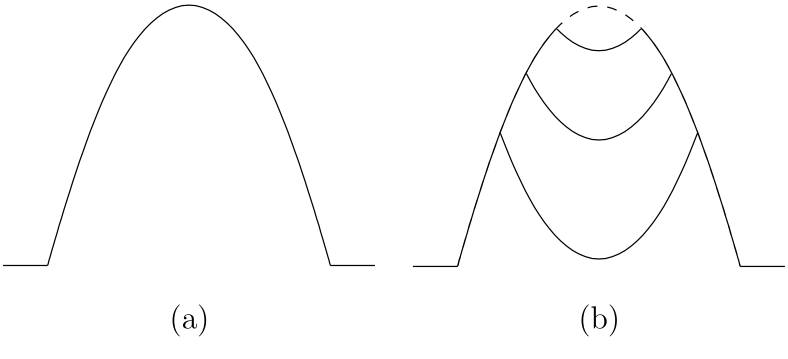
Besides the inherent defect of error accumulation, which is typical of every method of resolution using integration, the determination of these characteristic lines is a new problem in itself, since they also are defined through integration. Therefore, the accuracy of boundary conditions is much more crucial than for other methods. It follows that a certain number of obstacles must be overcome, *e.g.* the crossing of characteristic lines, which should normally occur only at singular points, or the presence of holes in Ω, which must be filled using secondary lines [30].

Finally, this method has been essentially used for the theoretical study of the number of solutions of class C^2 of the eikonal equation: a number of uniqueness results have been provided (see [31–34] and the references therein).

3.1.2 Approximation of Viscosity Solutions

Starting from the paper by Rouy and Tourin [35], the most recent approach to the resolution of SFS uses the notion of "viscosity solutions" to first order PDEs, see *e.g.* [36]. To give an idea, these are almost-everywhere solutions (a.e. solutions) which can be obtained as the limit in a family of solutions for regularized second order problems (the so-called "vanishing viscosity" method). These solutions are typically Lipschitz continuous solutions (but discontinuous viscosity solutions have also been considered in the literature, *cf.* [36]). The development of the theory of viscosity solutions for Hamilton-Jacobi type equations provides a good framework for the analysis of the SFS problem.

Moreover, several algorithms have been proposed to compute viscosity solutions. Finite difference numerical methods have been used in [35,37] for the resolution of (4) and generalized to the resolution of (3) in [38]. Similar results have been obtained by Oliensis and Dupuis [39] with an algorithm based on the Markov Chain approximation. Unfortunately, the Dirichlet problem (3)-(7) can have several "weak solutions" in the viscosity sense and also several classical solutions (due to the so-called "concave/convex ambiguity", see [30]). As an example, all the surfaces represented in Fig. 2 are viscosity solutions of the same equations (4)-(6), which is a particular case of (3)-(7). The solution represented in Fig. 2-a is the maximal solution and is smooth. All the non-smooth a.e. solutions which can be obtained by a reflection with respect to a horizontal axis, are still admissible weak solutions (*cf.* Fig. 2-b). In this example, the lack of uniqueness of the viscosity solutions is due to the existence of a singular point, where the right hand side of (4) vanishes. An additional effort is then needed to define which the preferable solution since the lack of uniqueness is also a big drawback when trying to compute a numerical solution. In order to circumvent these difficulties, the problem is usually solved by adding some information such as the height at each singular point [37].



(a) (b)

Fig. 2. Illustration of the concave/convex ambiguity: (a) maximal solution and (b) a.e. solutions giving the same image.

More recently, an attempt has been made to eliminate the need for *a priori* additional information. In recent results in the theory of viscosity solutions, the "maximal solution" without additional information apart from the equation was characterized, as was the construction of an algorithm which converges to that solution. A result by Ishii and Ramaswamy [40] applied to SFS guarantees that if function I is continuous and the number of singular points is finite, then a unique maximal solution (in the viscosity sense) of (3)-(7) exists. It should be noted that their result on the characterization of the maximal solution does not apply to the general situation when the set of singular points has a positive measure (this is the case, for example, of a flat roof). More general uniqueness results for maximal solutions of (4)-(7) or (3)-(7) have been recently obtained by Camilli *et al.* [41,42]. Several papers have followed this approach providing different algorithms to compute the maximal solution, which has been shown to be unique, see *e.g.* [43–45] and the references therein.

Since the PSFS equation is a first order non-linear PDE as well, some of the methods have been adapted to the perspective model. Tankus *et al.* apply [46,47] a modification of the fast marching finite difference method for the eikonal equation, originally proposed by Kimmel and Sethian [48], and show that PSFS can be used successfully on medical images for reconstruction of organs. Prados *et al.* propose [27,49] two algorithms and show convergence. They prove an existence and uniqueness result for the PSFS equation coupled to state constraint boundary condition, in the case where the light source is located at the center of projection. Moreover, they use a control theoretical interpretation of the equation to build their approximation schemes (as in [41]), showing convergence under some restrictive assumptions. An interesting application to reconstruction of faces validates their approach (for this application, see also [50]).

We conclude this section by mentioning a few other extensions in the framework of viscosity solutions. All the theoretical results mentioned above use the regularity of the greylevel function I, which is supposed to be (at least) continuous. Naturally, real images do not fit that assumption, even in the case of Lambertian objects. The continuity assumption for I has been removed in papers by Kain and Ostrov [51] and by Prados and Faugeras [52]. Both these papers also contain a scheme and some numerical examples. Finally, the paper by Falcone *et al.* [53] deals with the case of an oblique light source with black shadows in the image and gives a convergent scheme in the framework of viscosity solutions.

3.1.3 Other PDE Methods

Another approach which produces a global solution to SFS consists in the search for equal-height contours, originally proposed by Bruckstein [54] and later re-introduced by Kimmel and Bruckstein [55,56]. The method consists of two major steps: the computation of weighted distance functions from all the singular points, using a level set method, and the merging of these surfaces. The algorithm can compute a global solution (which is an a.e. solution) of eikonal equation (4) in the reconstruction domain, only combining the local solutions obtained during the first step. Interestingly, this method has been extended to the case of near light source [22].

Finally, the idea of solving the eikonal equation using a power series expansion at a singular point, in the case of a greylevel function of class C^∞, has been introduced by Bruss [57] and has been extended to the analytical greylevel functions by Durou and Piau, which could exhibit a "non-visible deformation" *i.e.*, a continuous family of analytical shapes giving the same image [58]. This is an important theoretical result but, nevertheless, no algorithm has been derived from this method of resolution.

3.1.4 Boundary Conditions

The use of PDE methods for the resolution of the SFS problem leads necessarily to the definition of some sort of boundary conditions. This is one of the differences with respect to the methods using optimization, since for those methods boundary conditions can be imposed but are not compulsory. A detailed analysis of the well-posedness of the boundary value problem for nonlinear PDEs in the framework of weak solutions (in the viscosity sense) can be found in Barles' book [36] and in the references therein. It is important to note that the addition of boundary conditions does not solve the concave/convex ambiguity and that in practical applications boundary conditions are seldom known.

The choice between the different types of boundary conditions is a question of appropriateness and simplicity, or depends on the additional information available on the object (if any). The Dirichlet boundary condition is typically used when the object is standing on a flat background and the surfaces meets the background at $\partial\Omega$, or if the height on $\partial\Omega$ is known (or assumed, for example by symmetry). Neumann boundary conditions correspond to $\partial u/\partial\nu(\mathbf{x}) = m(\mathbf{x})$, where $\nu(\cdot)$ represents the outward normal to domain Ω. A typical use of it is when we know (or we presume) that the level curves of the surface are orthogonal to the boundary $\partial\Omega$ or to a subset of it where we simply choose $m(\mathbf{x}) = 0$. The Neumann boundary condition gives more freedom in the computation since it only imposes the value of a derivative and does not fix the height of the surface at the boundary. Naturally, also this condition modifies the surface. State constraint boundary conditions differ from the previous ones since they do not impose a value either for the height or for its normal derivative. In this respect, it has been interpreted as a "no boundary condition" choice [59], although this interpretation is rather superficial. In fact, a real function u bounded and uniformly continuous is said to be a "state constraint viscosity solution" if and only if it is a subsolution (in the viscosity sense) in Ω and a supersolution in $\overline{\Omega}$ (*i.e.*, up to the boundary). It can also be stated as a Dirichlet boundary condition, simply setting $g(\mathbf{x}) = g_\mathrm{c}$ on $\partial\Omega$, where g_c is a constant, provided:

$$g_\mathrm{c} > \max_{\mathbf{x}\in\Omega} \{u(\mathbf{x})\}\,. \tag{8}$$

Note that in our problem, this is a mild assumption since we can easily fix an upper bound for the height of the object. The effect of the state constraint boundary condition is to produce solutions that grow inwards from the boundary $\partial\Omega$. This choice can be appropriate in some situations and wrong in other situations, in any case also this boundary condition affects the computation.

3.2 Methods Using Optimization

The second class of algorithms which have been suggested are optimization methods based on the variational approach. Note that these algorithms can work in the most general case of Eq. (1), contrary to the PDE methods. In this class of methods, three basic ingredients must be chosen: the unknowns, the functional which has to be optimized (in fact, minimized), and the minimization method. Surprisingly, a certain number of papers falling within the domain of optimization do not clearly show the implications of these choices, so that the choice of a functional is sometimes only guided by considerations on convergence. Even in [60], which is a major reference in the field, the discussion of several functionals is based on the possibility of finding an algorithm that converges towards a minimum. Indeed, it is possible *a priori* to freely combine any functional and any minimization algorithm.

3.2.1 Unknowns

The first difficulty encountered in the SFS problem is the choice of the unknowns. The natural unknown *i.e.*, height u, is rarely used [61]: problems of convergence [60] or of slowness [62] are mentioned. Many papers dealing with optimization use $p = \partial u/\partial x$ and $q = \partial u/\partial y$ as unknowns (see *e.g.* [63,60,64,62]), because u appears in the image irradiance equation only through its first derivatives; however, if u is supposed to be of class C^2, p and q are two non-independent functions, since:

$$\frac{\partial p}{\partial y} = \frac{\partial q}{\partial x}. \tag{9}$$

Other unknowns have been used: the three unknowns (u, p, q) have been dealt with simultaneously [3,65]; the stereographic coordinates of the normal [66,67] present the great interest of being bounded on the occluding boundaries, contrary to (p, q); the normal itself is sometimes used [68,69]. Finally, several model-based approaches have been proposed, such as quadratic models [70], triangular elements [71], deformable models [72] or B-splines [73].

When dealing with other unknowns than u itself, an additional problem is then to compute u. For instance, using (p, q) as unknowns, the following equations must be solved:

$$\begin{cases} \dfrac{\partial u}{\partial x} = p, \\ \dfrac{\partial u}{\partial y} = q, \end{cases} \tag{10}$$

which give (9) by elimination of u. Nevertheless, since Eqs. (10) are linear in u, the problem of integration is much easier to solve than the SFS problem.

3.2.2 Functionals

The only quantity which has to be minimized, whatever the unknowns, is the "brightness error". Using u as unknown, defining function r so that $R(\mathbf{n}(\mathbf{x})) = r(p(\mathbf{x}), q(\mathbf{x}))$, and using least square error, this term can be expressed as:

$$\mathcal{F}_1(u) = \int_{\mathbf{x}\in\Omega} \left[r(\partial u/\partial x(\mathbf{x}), \partial u/\partial y(\mathbf{x})) - I(\mathbf{x})\right]^2 d\mathbf{x}. \tag{11}$$

Using (p, q), a strictly equivalent functional to $\mathcal{F}_1(u)$ is:

$$\mathcal{F}_2\left(p, q, \mu\right) = \int_{\mathbf{x}\in\Omega} \left[r(p(\mathbf{x}), q(\mathbf{x})) - I(\mathbf{x})\right]^2 d\mathbf{x} + \int_{\mathbf{x}\in\Omega} \mu(\mathbf{x}) \left[\frac{\partial p(\mathbf{x})}{\partial y} - \frac{\partial q(\mathbf{x})}{\partial x}\right] d\mathbf{x}, \tag{12}$$

where μ is a Lagrange multiplier *i.e.*, a new unknown [60]. This last functional is often approximated by another one (for numerical reasons) where the constraint term becomes a least square penalty term, often called the "integrability term" [60]:

$$\mathcal{F}_3\left(p, q\right) = \int_{\mathbf{x}\in\Omega} \left[r(p(\mathbf{x}), q(\mathbf{x})) - I(\mathbf{x})\right]^2 d\mathbf{x} + \lambda_{\mathrm{i}} \int_{\mathbf{x}\in\Omega} \left[\frac{\partial p(\mathbf{x})}{\partial y} - \frac{\partial q(\mathbf{x})}{\partial x}\right]^2 d\mathbf{x}. \tag{13}$$

Several discrete implementations of this functional have been proposed [63,60]. A problem is that it is parametric, contrary to $\mathcal{F}_1(u)$ and $\mathcal{F}_2\left(p, q, \mu\right)$, depending on parameter λ_{i} which is called the "integrability factor". Another way of translating functional (12), without introducing any parameter, consists in imposing the hard constraint (9) in an iterative process: this leads to the well-known method by Frankot and Chellappa [64], where integrability is forced at each step. Symmetrically, it has recently been proposed in [69] to impose Eq. (1) at each step of an iterative optimization method. Another least square penalty term which has been much used is the "smoothness term" [61]:

$$\mathcal{F}_4\left(p,q\right) = \int_{\mathbf{x}\in\Omega} \left[r(p(\mathbf{x}),q(\mathbf{x})) - I(\mathbf{x})\right]^2 d\mathbf{x} + \lambda_{\mathrm{s}} \int_{\mathbf{x}\in\Omega} \left[\left|\nabla p(\mathbf{x})\right|^2 + \left|\nabla q(\mathbf{x})\right|^2\right] d\mathbf{x}, \tag{14}$$

where λ_{s} is a second parameter given the name of "smoothing factor". Incidentally, one critical point of the optimization approach is that, given a noise-less image with assumptions satisfied, the true surface will minimize functionals $\mathcal{F}_1(u)$ and $\mathcal{F}_2(p,q,\mu)$, but the same may not be said of the "improvement" that results in functional $\mathcal{F}_4(p,q)$. This can be avoided by progressively decreasing λ_{s} as long as the brightness error decreases [3,61]. Of course, some authors have used the functional combining both least square penalty terms [65,62]:

$$\begin{aligned}\mathcal{F}_5\left(p,q\right) = \int_{\mathbf{x}\in\Omega} \left[r(p(\mathbf{x}),q(\mathbf{x})) - I(\mathbf{x})\right]\,^2 d\mathbf{x} + \lambda_{\mathrm{i}} \int_{\mathbf{x}\in\Omega} \left[\frac{\partial p(\mathbf{x})}{\partial y} - \frac{\partial q(\mathbf{x})}{\partial x}\right]^2 d\mathbf{x} \\ + \lambda_{\mathrm{s}} \int_{\mathbf{x}\in\Omega} \left[\left|\nabla p(\mathbf{x})\right|^2 + \left|\nabla q(\mathbf{x})\right|^2\right] d\mathbf{x}.\end{aligned} \tag{15}$$

Finally, some other least square penalty terms have been used, as for example the "image intensity gradient constraint" [74] and, beside the classical least square estimator, robust estimators have also been used [75].

3.2.3 Methods of Minimization

When a given functional is chosen, two main strategies to find its minimum exist, as recalled in [65]: either the Euler equations associated with the functional are solved, or the functional is directly minimized. The first strategy has been used much more often than the second one (see *e.g.* [63,66,60,64,3]), since it is easier to implement and generally faster, but its main drawback is possible divergence [76], because convergence is hard to prove for a Jacobi iteration. Nevertheless, on the one hand, such a method is proved to be convergent in [67] and, on the other hand, linearizing the reflectance function renders the associated Euler equations linear [71], thus avoiding possible problems of divergence. Conjugate gradient descent has been used as a method of direct minimization in [65,61], providing results of rather good quality, but convergence is not guaranteed, contrary to the classical gradient descent method combined with line search [62].

The approximate solutions computed are typically local minima of the functional. To obtain a global minimum, a global optimization algorithm has to be used, typically a stochastic algorithm like simulated annealing [77,73] or genetic algorithms [78]. Naturally, the price to pay is a longer CPU time for the computation. This problem of a long computing time has been partially solved either using multiresolution [65,72,77] or dealing with a parametric model with few parameters [73]. Finally, Courteille *et al.* [26] have extended some minimization algorithms to PSFS, showing some applications to documents digitization.

3.3 Methods Approximating the Image Irradiance Equation

There exists a third class of SFS methods, whose name designates their common feature, recognizing that all of them make an approximation of the image irradiance equation. In [1], they are classified into two sub-classes: local methods and linear methods.

3.3.1 Local Methods

Local methods make the computation of the normal at each point in the image, independently of the same computation for the other points, but they need a strong assumption on the observed surface, which is generally difficult to justify. Without that assumption, these methods would be unfeasible. The usual assumption which is made on the surface is that it is locally spherical [79–81], except in [82], where it is assumed to be locally cylindrical, but the latter paper is concerned with radarclinometry, which is quite different from SFS. The consequence of making this very strong assumption is that the obtained normals are usually very far from being integrable, so that the computed shapes are often very bad, except if the observed shape exactly satisfies the local assumption *i.e.*, if the shape is a part of a sphere or of a cylinder.

3.3.2 Linear Methods

Linear methods make either a global or a local linear approximation of the reflectance function. Global linear approximation means that the same approximation is used for all points in the image [83–86], as is also the case for an already cited optimization method [71] (this proves how difficult a *strict* classification of the SFS methods is). Obviously, a global approximation of the reflectance function cannot reasonably be used, unless the normal is quasi-invariant on the whole surface, even if Pentland proved in [83] that, the greater the angle between the observer's direction and the lighting direction, the better this approximation. Consequently, it appears that this approach is a little limited, even if the implementations which follow from it are convincing, using either finite differences [84–86] or Fourier transform [83].

Local linear approximation of the reflectance function is used in [87], a method of resolution that will be discussed in detail in Section 4.2.3. Now, let us select one method per class for the tests.

4 Selection of Three Shape-from-Shading Methods for the Tests

After the survey, in which a number of methods have been classified and commented, we have now to select a few methods, in order to test their performances. We first discuss the opportuneness of making such a selection. After that, we detail the three selected methods in a common formalism and give the subtleties that make them work.

4.1 Selection of Shape-from-Shading Methods

4.1.1 Discussion

Selecting several SFS methods belonging to the same class, in order to numerically compare them, is much easier than selecting methods belonging to different classes, as in [1] or in the present work, because the mathematical tools that are behind them are very different, and they do not always require the same data. Whereas a survey has to be as exhaustive as possible, a selection must be significant. Thus, one method at least per class must be selected *i.e.*, three methods at least. Ideally, the best method of each class should be selected, but such a ranking strongly depends on which benchmarks are used. Thus, rather than selecting the methods according to their performances, we aim at selecting methods which can be compared *i.e.*, methods that require the same data. For example, the method described in [88] requires the height at each singular point, contrary to many other methods: would it make sense to conclude that this method works better than another one?

4.1.2 Selection of One Shape-from-Shading Method per Class

First Class. A big problem with the characteristics method proposed by Horn [30] is the uneven sampling of the image due to the characteristic strips going their own way. The method by Falcone and Sagona [43] is chosen since it is an approximation method which provably converges to the “maximal solution” (in the viscosity sense) and does not require any additional knowledge on the surface, as for example the height at singular points. Kimmel and Bruckstein’s method [56] had been originally selected as a second representative of the first class, but the merging process is allowed to continue after the first solution is computed, which means that it is quite difficult to define a stopping rule. Moreover, since the merging process is separated (offline) and uses a dynamic list of local solutions, this makes it quite difficult to compare this algorithm with others in terms of CPU time. This shows, if necessary, that implementing a method, just using the description given in a paper, is not technically evident.

Second Class. Most of the methods using optimization require knowledge of the normal in $\partial\Omega$, which is of no use for the selected methods of both other classes, but Daniel and Durou's method does not require this knowledge, because it uses functional $\mathcal{F}_5(p,q)$ with two penalty terms. This renders the minimization problem well-posed in the absence of boundary condition on (p,q) [89], even if in [32], the use of a smoothing term is considered as superfluous to avoid infinite ambiguity. This is a first argument to select that method as representative of the second class. Moreover, even if the methods proposed in [65], which use the same functional $\mathcal{F}_5(p,q)$, are of good quality, mostly in terms of CPU time, they are not provably convergent, contrary to Daniel and Durou's method.

Third Class. Amongst the methods approximating the image irradiance equation, the only method that can be applied to a variety of situations is that of Tsai and Shah [87], since that method uses local linearization. In [1], it is one of the six methods which are compared. Even if it numerically obtains the worst overall rank (total error of 59.3, to be compared to 41.3 for the best method, which shows that the scores are very close), it is qualitatively the best method on synthetic images, while the greatest part of the scores is obtained on real images, for which all the reconstructions are of very bad quality. This method, which is also iterative, sometimes diverges, but that can be avoided by stopping the iteration after a fixed number of steps, as recommended by the authors.

4.1.3 Comparison with the Survey by Zhang et al.

It can be argued that two of the three methods that we select were devised by us, and that the source code of the third selected method is freely available [1]. Moreover, it can be argued that in the survey by Zhang *et al.*, twice as many methods are tested than in ours *i.e.*, six methods. Although this survey was extremely useful to us, the selection of the methods which are numerically tested is open to criticism. First, only one of the six methods was devised by the authors, even if they claim the following: "We manually selected parameters for each algorithm in order to obtain the best results". Second, it is obvious that the third class (methods approximating the image irradiance equation) is much over-represented, with four methods out of six (Lee and Kuo's [71], Lee and Rosenfeld's [80], Pentland's [90], Tsai and Shah's [87]), even if Lee and Kuo's method can also be considered as a minimization method (the other minimization method is that of Zheng and Chellappa [74]). Finally, Bichsel and Pentland's method [88], which is the only PDE method, requires the

---

[1] The source codes of the survey by Zhang *et al.* are available by anonymous ftp under the `pub/tech_paper/survey` directory, at `eustis.cs.ucf.edu` (132.170.108.42).

height at each singular point.

4.2 Description of the Three Selected Methods

Now we describe the fully discrete schemes of the three selected methods, discretizing the image with a regular square mesh of size δ. A node of this mesh, or pixel, is designated by (i,j): it corresponds to the point $\mathbf{x}_{i,j}$ of coordinates $(i\,\delta, j\,\delta)$ in system (Oxy). The set of pixels (i,j) such that $\mathbf{x}_{i,j} \in \Omega$ is designated by D and its number of elements by $N = \mathrm{card}(D)$. We define also the three following subsets of D: a pixel (i,j) of D is in $\overline{D}$ if $(i+1,j)$ and $(i,j+1)$ are in D; it is in D_{in} if its four nearest neighbours $(i+1,j)$, $(i,j+1)$, $(i-1,j)$ and $(i,j-1)$ are in D; it is in D_{bd} if it is outside D_{in}.

4.2.1 Falcone and Sagona's Method

First we introduce the semi-Lagrangian approximation for (4), (6) or (4), (7). Note that this method works on the more general case (3), (7) (see [53]). Here and in the sequel we will assume for simplicity that $u \geq 0$ in Ω. This is not restrictive, since Eq. (4) depends only on ∇u and we can always add to u the minimum value of $u(\mathbf{x})$ on Ω to satisfy that requirement. In order to obtain an approximation scheme in the form of a fixed point problem, it is useful to introduce a change in the unknown $v(\mathbf{x}) = 1 - \exp\left(-u(\mathbf{x})\right)$. Note that by definition $0 \leq v \leq 1$. The SFS problems (4), (6) or (4), (7) for the new unknown v are:

$$\begin{cases} v(\mathbf{x}) + \max\limits_{\mathbf{a}\in B_2(0,1)} \left\{ -\dfrac{\mathbf{a}}{f(\mathbf{x})} \cdot \nabla v(\mathbf{x}) - 1 \right\} = 0 & \text{for } \mathbf{x} \in \Omega, \\ v(\mathbf{x}) = 0 \quad \text{or} \quad v(\mathbf{x}) = 1 - \exp\left(-g(\mathbf{x})\right) & \text{for } \mathbf{x} \in \partial\Omega, \end{cases} \tag{16}$$

where f is given by (5) and $B_2(0,1)$ is the $\mathbb{R}^2$ unit ball. It is known that (16) has a unique continuous viscosity solution provided f is bounded and never vanishes in Ω [91]. We look for a solution v in the space of piecewise affine functions which are linear on the cells (P^1 finite element approximation):

$$\begin{cases} v(\mathbf{x}_{i,j}) = \min\limits_{\mathbf{a}\in B_2(0,1)} \left\{ \exp(-h)\, v\left(\mathbf{x}_{i,j}(\mathbf{a})\right) \right\} + 1 - \exp(-h) & \text{for } (i,j) \in D_{\mathrm{in}}, \\ v(\mathbf{x}_{i,j}) = 0 \quad \text{or} \quad v(\mathbf{x}_{i,j}) = 1 - \exp\left(-g(\mathbf{x}_{i,j})\right) & \text{for } (i,j) \in D_{\mathrm{bd}}, \end{cases} \tag{17}$$

where h is a small parameter, $\mathbf{x}_{i,j}(\mathbf{a}) = \mathbf{x}_{i,j} + h\mathbf{a}/f(\mathbf{x}_{i,j})$ and $v\left(\mathbf{x}_{i,j}(\mathbf{a})\right)$ is computed by linear interpolation on the pixels of the grid. If $\mathbf{V}$ denotes the array containing the N values $v(\mathbf{x}_{i,j})$, for $(i,j) \in D$, then (17) can be reformulated as $\mathbf{V} = H(\mathbf{V})$, $H : \mathbb{R}^N \to \mathbb{R}^N$. In [92,93] it has been proved that the numerical solution to (17) exists and is unique by a fixed point argument on the iteration $\mathbf{V}^{k+1} = H(\mathbf{V}^k)$. Moreover, an *a priori* estimate for the convergence holds true provided function f is Lipschitz continuous (we refer the interested reader to [93] and [45] for a precise result). We note in passing that $0 \leq \mathbf{V} \leq 1$ implies $0 \leq H(\mathbf{V}) \leq 1$ and that $\mathbf{V}_1 \leq \mathbf{V}_2$ implies $H(\mathbf{V}_1) \leq H(\mathbf{V}_2)$. This monotonicity property implies that, starting from a subsolution ($\mathbf{V}^0 \leq H(\mathbf{V}^0)$), the sequence will monotically converge to the fixed point. This property is crucial in speeding up convergence and it also helps to compute the maximal solution (see [43]). Other acceleration techniques for the eikonal equation can be found in [94,95] where the so-called "fast marching method" is described. It should be noted that the presence of singular points makes the vectorfield $\mathbf{a}/f(\mathbf{x}_{i,j})$ unbounded since f vanishes. This is an additional difficulty which is usually solved by truncating f below at the ε level. In this way, $f(\mathbf{x})$ is replaced by $f_\varepsilon(\mathbf{x}) = \max\{f(\mathbf{x}), \varepsilon\}$ in eikonal equation (4). Naturally, the algorithm has to be analyzed with respect to ε. This was first done by Camilli and Siconolfi in [42], who showed that the solution of the perturbed problem actually converges to the solution of the original problem for ε going to 0. Following that result, Camilli and Grüne suggested a scheme which converges to the maximal solution of the eikonal equation in [44]. More recently, Sagona [45] has proved the convergence to the maximal solution of the above algorithm and has established an *a priori* error estimate in the L^∞ norm which takes into account all the perturbation and discretization parameters. The crucial condition for the convergence to the maximal solution is:

$$h \left| \frac{1}{f_\varepsilon} \right|_\infty \leq \delta. \tag{18}$$

If $\mathbf{V}_{\text{in}}$ denotes the array containing the values $v(\mathbf{x}_{i,j})$, for $(i,j) \in D_{\text{in}}$, and $\mathbf{V}_{\text{bd}}$ the array containing the values $v(\mathbf{x}_{i,j})$, for $(i,j) \in D_{\text{bd}}$, then the algorithm corresponding to Falcone and Sagona's method [43], designated in the following by FS, is:

```
fix $h$, $\varepsilon$ and $\zeta$
fix $\mathbf{V}_{\text{bd}}$ as in (17), $k \leftarrow 0$ and $\mathbf{V}^0_{\text{in}} \leftarrow (0, ..., 0)$
repeat
    compute $H(\mathbf{V}^k_{\text{in}})$ as in (17)
    $\mathbf{V}^{k+1}_{\text{in}} \leftarrow H(\mathbf{V}^k_{\text{in}})$
    $k \leftarrow k + 1$
until $|H(\mathbf{V}^k_{\text{in}}) - \mathbf{V}^k_{\text{in}}|_\infty < \zeta$
```

The values of the parameters that are used for our tests are $h = \delta / \left|\frac{1}{f_\varepsilon}\right|_\infty$, which is the biggest admissible value of h for which Eq. (18) holds, $\varepsilon = 0.20$, meaning that greylevel I is truncated at the maximal value 0.98, and $\zeta = 10^{-8}$.

4.2.2 Daniel and Durou's Method

The method of resolution of SFS described in [62] uses functional $\mathcal{F}_5(p, q)$ given in Eq. (15). A straightforward discretization of this functional gives the following "energy":

$$E(\mathbf{G}) = \delta^2 \sum_{(i,j)\in D} \left[r(p_{i,j}, q_{i,j}) - I_{i,j}\right]^2 + \lambda_{\mathrm{i}} \sum_{(i,j)\in \overline{D}} \left[(p_{i,j+1} - p_{i,j}) - (q_{i+1,j} - q_{i,j})\right]^2$$
$$+\lambda_{\mathrm{s}} \sum_{(i,j)\in \overline{D}} \left[(p_{i+1,j} - p_{i,j})^2 + (p_{i,j+1} - p_{i,j})^2 + (q_{i+1,j} - q_{i,j})^2 + (q_{i,j+1} - q_{i,j})^2\right], \tag{19}$$

where $\mathbf{G}$ is the array containing the $2N$ values $(p_{i,j}, q_{i,j})$, for $(i, j) \in D$, which are the unknowns of the discrete problem. It has been proved in [60] that this energy is quasi-independent of δ.

The use of gradient descent combined with "line search" was suggested in [62]: at each step k, a positive value d^k which is a minimizer (at least, a local minimizer) of function $\phi^k(d) = E\left(\mathbf{G}^k - d\ \nabla E(\mathbf{G}^k)\right)$ is found, assuming that $\phi^k(d)$ can be approximated by a parabola (quadratic approximation). The iteration is then defined by $\mathbf{G}^{k+1} = \mathbf{G}^k - d^k\ \nabla E(\mathbf{G}^k)$, and is stopped when either the Euclidean norm $|\nabla E(\mathbf{G}^k)|$ or d^k is less than a threshold.

The algorithm corresponding to Daniel and Durou's method [62], designated in the following by DD, is:

```
fix $\lambda_{\mathrm{i}}$, $\lambda_{\mathrm{s}}$, $\beta$ and $\gamma$
$k \leftarrow 0$ and fix starting vector $\mathbf{G}^0$
repeat
    compute $\nabla E(\mathbf{G}^k)$
    find a local minimizer $d^k$ of $\phi^k(d) = E\left(\mathbf{G}^k - d\ \nabla E(\mathbf{G}^k)\right)$
    $\mathbf{G}^{k+1} \leftarrow \mathbf{G}^k - d^k\ \nabla E(\mathbf{G}^k)$
    $k \leftarrow k + 1$
until $|\nabla E(\mathbf{G}^k)| < \beta\sqrt{2N}$ or $d^k < \gamma\sqrt{2N}$
```

Vector $\mathbf{G}^0$ contains the values $(p_{i,j}, q_{i,j})_{(i,j)\in D}$ of a "starting shape", which in our tests is the Gaussian curve $u(x, y) = 2\,e^{-(x^2+y^2)}$. The values of the parameters that are used for our tests are $\lambda_{\mathrm{i}} = 10$, $\lambda_{\mathrm{s}} = 50$, $\beta = 1$, and $\gamma = 10^{-7}$. For DD, as well as for FS, the optimal configuration will be obtained in the limit. Of course, a stopping criterion has to be chosen. The thresholds on $|\nabla E(\mathbf{G}^k)|$ and on d^k are proportional to $\sqrt{2N}$ because $\mathbf{G}$ is a vector in $\mathbb{R}^{2N}$. Of course, bigger values of the thresholds would reduce the CPU time, but they would also provide worse results.

This algorithm provides an estimate of $\mathbf{G}$, but subsequently, a residual problem, called "integration", consists in computing from this estimate either the N values $(u_{i,j})_{(i,j)\in D}$, if no Dirichlet boundary condition is imposed, or the $N_{\text{in}} = \text{card}(D_{\text{in}})$ values $(u_{i,j})_{(i,j)\in D_{\text{in}}}$ otherwise. Some tests [62] have shown that Wu and Li's method [96] combined with that put forward by Horn and Brooks in [60] give good results. Wu and Li's method computes the height along diagonals and is very rapid. The result is then used as an initial shape for Horn and Brooks' method which is iterative. Of course, this computation is taken into account in the CPU time.

4.2.3 Tsai and Shah's Method

Tsai and Shah make a development of the reflectance function to the first order at each point in the image [87], unlike the other linear methods. The basic idea of the method is to approximate $p_{i,j}$ and $q_{i,j}$ by finite backward differences, which gives from (1) the following system of equations:

$$r\left(\frac{u_{i,j}-u_{i-1,j}}{\delta}, \frac{u_{i,j}-u_{i,j-1}}{\delta}\right) = I_{i,j} \qquad \text{for } (i,j)\in D. \tag{20}$$

Surprisingly, Tsai and Shah take $\delta = 1$ in (20). Indeed, the tests have shown that using the exact value of δ causes numerical instabilities, so we did as Tsai and Shah did. Developping to the first order in $u_{i,j}$, and denoting $u_{i,j}^{\text{cur}}$ the current estimate of height $u_{i,j}$ and $u_{i,j}^{\text{new}}$ a refinement, system (20) becomes:

$$r_{i,j}^{\text{cur}} + (u_{i,j}^{\text{new}} - u_{i,j}^{\text{cur}})\left[(\partial r/\partial p)_{i,j}^{\text{cur}} + (\partial r/\partial q)_{i,j}^{\text{cur}}\right] = I_{i,j} \qquad \text{for } (i,j)\in D, \tag{21}$$

where $f_{i,j}^{\text{cur}}$ means, whatever the function f:

$$f_{i,j}^{\text{cur}} = f(u_{i,j}^{\text{cur}} - u_{i-1,j}^{\text{cur}}, u_{i,j}^{\text{cur}} - u_{i,j-1}^{\text{cur}}). \tag{22}$$

It might seem that system (21) expresses the Newton-Raphson method, in the case of a unidimensional non-linear equation, but this is not true, because $u_{i-1,j}$ and $u_{i,j-1}$ are modified at each step, as well as $u_{i,j}$. Thus, (21) is the following variant of the Newton-Raphson method, for which no general proof of convergence exists:

$$u_{i,j}^{\text{new}} = u_{i,j}^{\text{cur}} - \frac{r_{i,j}^{\text{cur}} - I_{i,j}}{(\partial r/\partial p)_{i,j}^{\text{cur}} + (\partial r/\partial q)_{i,j}^{\text{cur}}} \qquad \text{for } (i,j)\in D. \tag{23}$$

Because there is no guarantee that the denominator in (23) will not vanish, Tsai and Shah put forward the following modification of system (23):

$$u_{i,j}^{\text{new}} = u_{i,j}^{\text{cur}} - K_{i,j}^{\text{cur}} \left(r_{i,j}^{\text{cur}} - I_{i,j} \right) \qquad \text{for } (i,j) \in D, \tag{24}$$

where $K_{i,j}^{\text{cur}}$ has to approximate the inverse of this denominator when it is non-zero, and must be zero otherwise. This leads to the following algorithm [87], called TS in the course of this article:

```
fix $W$ and $k_{\max}$
for $(i,j) \in D$ do
    $u_{i,j}^0 \leftarrow 0$ and $S_{i,j}^0 \leftarrow 1$
for $k \leftarrow 0 \cdots k_{\max}$ do
    for $(i,j) \in D$ do
        $K_{i,j}^k \leftarrow \dfrac{S_{i,j}^k \left[ (\partial r/\partial p)_{i,j}^k + (\partial r/\partial q)_{i,j}^k \right]}{W + S_{i,j}^k \left[ (\partial r/\partial p)_{i,j}^k + (\partial r/\partial q)_{i,j}^k \right]^2}$
        $u_{i,j}^{k+1} \leftarrow u_{i,j}^k - K_{i,j}^k \left( r_{i,j}^k - I_{i,j} \right)$
        $S_{i,j}^{k+1} \leftarrow \left\{ 1 - K_{i,j}^k \left[ (\partial r/\partial p)_{i,j}^k + (\partial r/\partial q)_{i,j}^k \right] \right\} S_{i,j}^k$
```

Following Tsai and Shah, we take $W = 0.01$. The stopping criterion consists in fixing the value of $k_{\max}$, while keeping in mind that an iteration exists for which the configuration is optimal. Our tests have led us to consider that the 5^{th} iteration works well, even if in [87], $k_{\max} = 2$ is recommended as a good choice. Finally, let us mention that, even if the scene is lit from the observer's direction, it is necessary to model r by an oblique source $\boldsymbol{\omega} = (0, -1/\sqrt{2}, 1/\sqrt{2})$. There is no obvious explanation for this last observation.

It might seem that this method does not require any boundary condition. This is not true, since Eq. (20) is unusable for a pixel $(i,j) \in D$ such that $(i-1,j) \notin D$ or $(i,j-1) \notin D$. Tsai and Shah recommend choosing $u_{i-1,j}^k = u_{i,j}^k$ if $(i-1,j) \notin D$, and $u_{i,j-1}^k = u_{i,j}^k$ if $(i,j-1) \notin D$, which is equivalent to imposing $p_{i,j} = 0$ and $q_{i,j} = 0$, respectively, *i.e.*, boundary conditions of the Neumann type. On the other hand, no Dirichlet boundary condition is imposed, which is an advantage, since no such information is usually available on real images. Nevertheless, the counterpart is that system (21) is an ill-posed problem in the sense of Hadamard, even with Neumann boundary conditions: it is easy to verify that a solution is only determined up to a constant. Indeed, the tests have shown a global drift, depending on step k. In order to numerically compare the computed shape with the ground truth, this drift must be corrected in a final step.

5 Methodology

In this section, we describe the panel of images selected for the tests, and the measures by which to numerically compare the performances of the methods.

5.1 Panel of Images Selected for the Tests

The choice of images for the tests is a serious difficulty. Of course, it seems logical to choose images that conform to the basic assumptions of SFS. As noted in [1], this is an easy task for synthetic images, but much more difficult for real images, even if a process by Daniel and Durou [97] creates real images verifying almost all the assumptions of SFS. For our tests, we decided to select three synthetic and four real images.

5.1.1 Synthetic Images

The three synthetic images are computed from known shapes. We compute all the synthetic images on the same domain $[-6.4, 6.4]^2$ of $\mathbb{R}^2$ projected on a regular square mesh of 256×256 pixels, meaning that the size of the mesh is $\delta = 12.8/256 = 0.05$. Then, p and q are evaluated by differentiation and, finally, I is computed through Eq. (3), depending on the value of $\boldsymbol{\omega}$. We first use $\boldsymbol{\omega}^0 = (0, 0, 1)$.

The first shape represents a vase lying on a flat background, called "synthetic vase" (SV in the following). It is defined by:

$$\begin{cases} u_{\mathrm{SV}}(x,y) = \sqrt{P\left(\overline{x}\right)^2 - y^2} & \text{for } (x,y) \in \Omega_{\mathrm{SV}}, \\ u_{\mathrm{SV}}(x,y) = 0 & \text{elsewhere}, \end{cases} \tag{25}$$

where $\overline{x} = x/12.8$, $P\left(\overline{x}\right) = -138.24\,\overline{x}^6 + 92.16\,\overline{x}^5 + 84.48\,\overline{x}^4 - 48.64\,\overline{x}^3 - 17.60\,\overline{x}^2 + 6.40\,\overline{x} + 3.20$ and $\Omega_{\mathrm{SV}} = \{(x,y) \in [-6.4, 6.4] \times \mathbb{R}, P\left(\overline{x}\right)^2 \geq y^2\}$. This shape is shown in Fig. 3-a. The corresponding synthetic image is represented in Fig. 3-b. Finally, the reconstruction domain Ω_{SV} is represented in Fig. 3-c.

The second shape represents a Canadian tent (CT in the following) lying on a flat background. It is defined by:

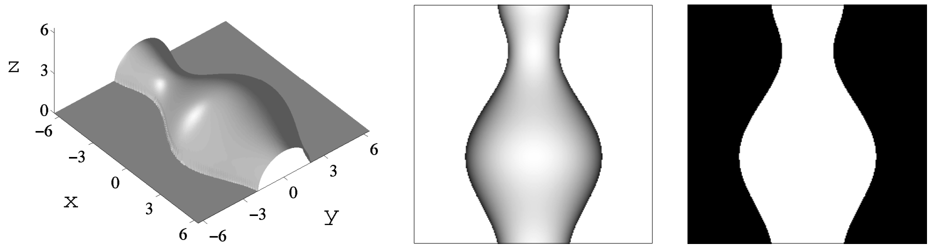
$$\begin{cases} u_{\mathrm{CT}}(x,y) = \min\left\{-2\,|x| + 10.24, -|y| + 5.12\right\} & \text{for } (x,y) \in \Omega_{\mathrm{CT}}, \\ u_{\mathrm{CT}}(x,y) = 0 & \text{elsewhere}, \end{cases} \tag{26}$$

where $\Omega_{\mathrm{CT}} = [-5.12, 5.12]^2$. This shape, the corresponding image and the reconstruction domain Ω_{CT} are represented in Fig. 4.

The third shape represents a digital elevation model (DEM in the following). It is defined by:

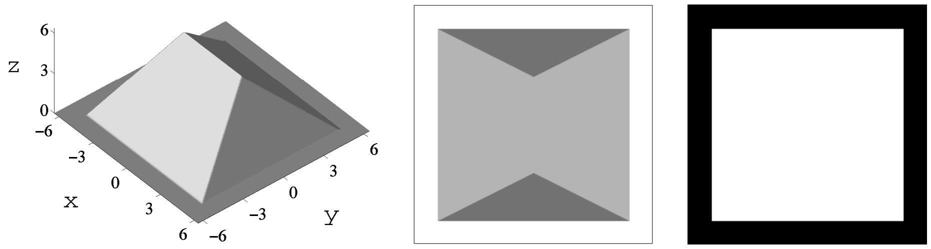
$$\begin{aligned} u_{\mathrm{DEM}}(x,y) = 3\,(1-\overline{\overline{x}})^2 \exp\left(-\overline{\overline{x}}^2 - (\overline{\overline{y}}+1)^2\right) - 10\,(\overline{\overline{x}}/5 - \overline{\overline{x}}^3 - \overline{\overline{y}}^5) \\ \exp\left(-\overline{\overline{x}}^2 - \overline{\overline{y}}^2\right) - 1/3 \exp\left(-(\overline{\overline{x}}+1)^2 - \overline{\overline{y}}^2\right), \end{aligned} \tag{27}$$

where $(\overline{\overline{x}}, \overline{\overline{y}}) = (x,y)/1.6$. For DEM, we use the reconstruction domain Ω_{DEM} which contains all the pixels $\mathbf{x}$ in $[-6.4, 6.4]^2$, except those near the edges such that $I(\mathbf{x}) \geq 254/255$ (by analogy with SV and CT, these points constitute a flat background). This shape, the corresponding image and Ω_{DEM} are represented in Fig. 5.



(a) (b) (c)

Fig. 3. SV: (a) shape, (b) image and (c) Ω_{SV}.

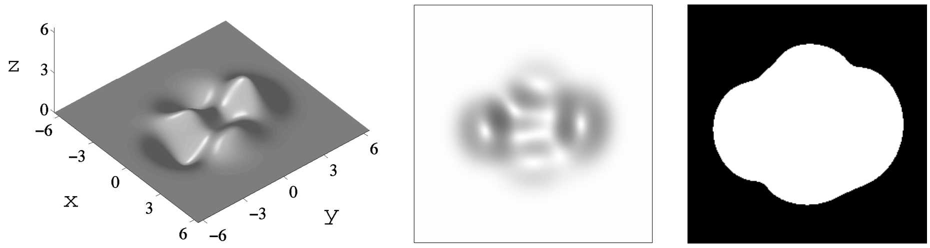


(a) (b) (c)

Fig. 4. CT: (a) shape, (b) image and (c) Ω_{CT}.

5.1.2 *Real Images*

As the four real images selected for the tests are also of size 256 × 256, we interpret the corresponding scenes as if δ were equal to 0.05, that is to say, as if the scenes were of size 12.8 × 12.8.



(a) (b) (c)

Fig. 5. DEM: (a) shape, (b) image and (c) Ω_{DEM}.

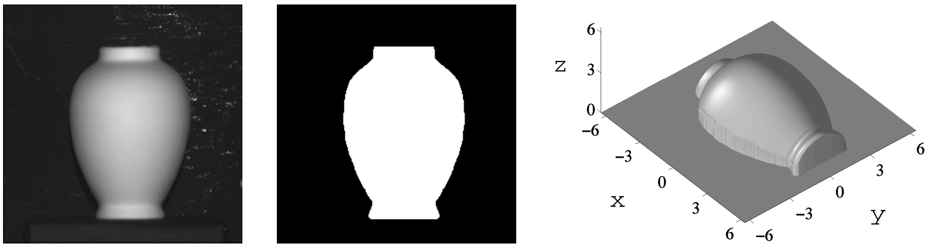
The first real image, represented in Fig. 6-a, is that of a vase, called "real vase" (RV in the following). The reconstruction domain Ω_{RV}, represented in Fig. 6-b, is constituted by the pixels situated on the vase. A problem is then to obtain the corresponding shape. Since the vase is radially symmetrical, this can be done by assuming that the symmetry axis is parallel to the image plane. Using this assumption, we can subsequently detect the occluding boundary and deduce the complete shape by rotation around the symmetry axis (*cf.* Fig. 6-c).

The second real image, represented in Fig. 7-a, is that of the moulding of an elk's head (called "Elk" in the following). This image was obtained through Daniel and Durou's process already mentioned [97]. Through mechanical metrology[2], we could obtain the set of 3D points plotted in Fig. 7-b. A number of these 3D points, whose height is zero, lie on the boundary of the elk's head. This allows us to determine the reconstruction domain Ω_{Elk} which is represented in Fig. 7-c: here again, the pixels outside Ω_{Elk} constitute a flat background. It can be noted that the 3D points of Fig. 7-b are not regularly arranged, but they will allow us to compare the computed shapes with the ground truth. It can also be noted that their density is rather low and, therefore, estimating p and q using finite differences would be senseless in this case.

Finally, two other classical real images used in [1] are represented in Figs. 8-a and 8-b: they are called "Pepper" and "Lena" in the following. No ground truth is available for these images and their common reconstruction domain $\Omega_{\mathrm{Pepper}} = \Omega_{\mathrm{Lena}}$ contains all the pixels.

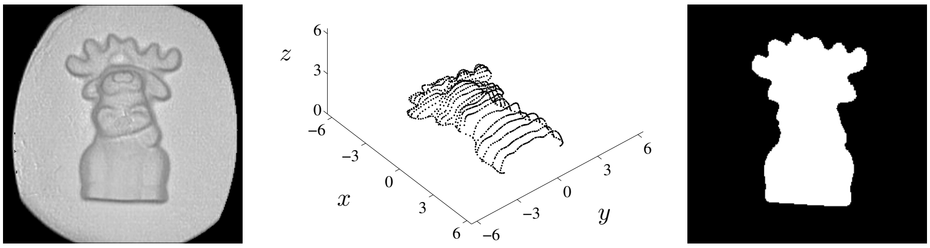
---

[2] We would like to thank Michel Labarrère, metrologist at the *Département de Génie Mécanique*, École Nationale Supérieure d'Ingénieurs de Constructions Aéronautiques, Toulouse, France.



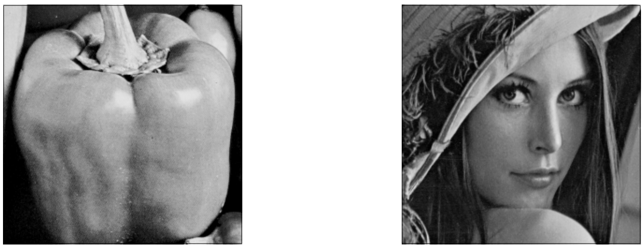
(a) (b) (c)

Fig. 6. RV: (a) image, (b) Ω_{RV} and (c) estimated shape.



(a) (b) (c)

Fig. 7. Elk: (a) image, (b) measured shape and (c) Ω_{Elk}.



(a) (b)

Fig. 8. (a) Pepper and (b) Lena.

5.1.3 Discussion

Synthetic images are particularly useful because it is impossible that there be no solution ("impossible images", see *e.g.* [98,99]). Moreover, the reconstructed and real shapes can be compared. On the other hand, the ground truth is usually not available for real images. From this point of view, our work clearly complements the survey by Zhang *et al.*, since we provide the ground truth for two real images: RV and Elk.

It must be noted that other images than the seven described in the previous section will be used in the tests: in Section 6.8, different sizes of SV are used, implying that δ will take other values than 0.05, in order to test the robustness of the methods with respect to low image resolution; in Section 6.9, three values of $\boldsymbol{\omega}$ other than $\boldsymbol{\omega}^0 = (0, 0, 1)$ are used to compute SV, in order to test the robustness of the methods faced with a wrong estimate of $\boldsymbol{\omega}$.

Finally, let us note that SV is the same image[3] as the one displayed in Fig. 5-a of [1] (Eq. (25) seems quite different from the equation given in [1], just because another coordinate system is used). Thus, four of the images selected for our tests are used in [1]: SV, RV, Pepper and Lena. Amongst the test images used in [1], only "Mozart" is not selected: it is replaced by DEM, which is in fact the function `peaks` of `Matlab`. As for "Mozart", DEM is smooth, has the simple boundary condition $u = 0$, and can be considered as a "hard benchmark" (see *e.g.* [77]). On the other hand, it is a little more convenient to deal with DEM than with Mozart, because the ground truth of DEM is analytically available, unlike that of Mozart. Since we wanted to choose images having different peculiarities, we complemented these five test images with two other ones: CT is an example of a non-smooth surface with $u = 0$ boundary condition (note that I is discontinuous inside Ω_{CT}); Elk is a real image which theoretically verifies all the assumptions of SFS, unlike Pepper and Lena, and whose shape is more complicated than that of RV.

5.2 Evaluation of the Performances of Shape-from-Shading Methods

The most natural criterion for the evaluation of a shape reconstruction is of course to measure the difference between reconstructed shape $\tilde{u}$ and real shape u. An interesting alternative is to compare input image I with estimate image $\tilde{I}$ computed from the reconstructed shape, particularly for Pepper and Lena, whose shapes are not available. Finally, when (p, q) are known for the real shape, it seems interesting to compare them with their estimates $(\tilde{p}, \tilde{q})$ or, even better, to compare normal $\mathbf{n}$ with its estimate $\tilde{\mathbf{n}}$, because these two vectors are normed and their comparison is equivalent to the measure of an angle in $\mathbb{R}^3$. Even for DD, which computes (p, q) before u, we will consider that $\tilde{u}$ is computed first, in order to be consistent with the other methods (a similar observation is made in [1]). As will be described now, $\tilde{\mathbf{n}}$ and $\tilde{I}$ are computed in a way which is the same for all the methods. After that, we will detail the panel of error estimators by which to compare the methods.

---

[3] This image was first provided in [100].

5.2.1 Estimation of the Normals and of the Greylevels

Once surface $\tilde{u}$ has been computed, we usually adopt an estimate $\tilde{\mathbf{n}}$ of the normal which is coherent with the computation of the greylevel. As can be seen in Fig. 9, for each pixel $\mathbf{x}_{i,j}$, we consider the four triangles $T^1_{i,j}$, $T^2_{i,j}$, $T^3_{i,j}$ and $T^4_{i,j}$ having $\mathbf{x}_{i,j}$ in common. For each triangle, the numerical solution $\tilde{u}$ is known at its three vertices, so by linear interpolation we can estimate $\tilde{p}_{i,j}$ and $\tilde{q}_{i,j}$. For example, on $T^1_{i,j} = \{\mathbf{x}_{i,j}, \mathbf{x}_{i-1,j}, \mathbf{x}_{i,j+1}\}$, the following estimates hold:

$$\begin{cases} \tilde{p}^1_{i,j} = \dfrac{\tilde{u}_{i,j} - \tilde{u}_{i-1,j}}{\delta}, \\ \tilde{q}^1_{i,j} = \dfrac{\tilde{u}_{i,j+1} - \tilde{u}_{i,j}}{\delta}, \end{cases} \tag{28}$$

and this gives $\tilde{\mathbf{n}}^1_{i,j}$ using (2). It is possible to obtain three other estimates $\tilde{\mathbf{n}}^2_{i,j}$, $\tilde{\mathbf{n}}^3_{i,j}$ and $\tilde{\mathbf{n}}^4_{i,j}$ of the normal, which correspond to the three other triangles $T^2_{i,j}$, $T^3_{i,j}$ and $T^4_{i,j}$. Four related estimates $\tilde{I}^1_{i,j}$, $\tilde{I}^2_{i,j}$, $\tilde{I}^3_{i,j}$ and $\tilde{I}^4_{i,j}$ of the greylevel are given by the four scalar products between $\boldsymbol{\omega}$ and the four normals $\tilde{\mathbf{n}}^1_{i,j}$, $\tilde{\mathbf{n}}^2_{i,j}$, $\tilde{\mathbf{n}}^3_{i,j}$ and $\tilde{\mathbf{n}}^4_{i,j}$. The greylevel estimate at pixel $\mathbf{x}_{i,j}$ is done in the following way:

$$\tilde{I}_{i,j} = \min\{\tilde{I}^1_{i,j}, \tilde{I}^2_{i,j}, \tilde{I}^3_{i,j}, \tilde{I}^4_{i,j}\}. \tag{29}$$

Subsequently, we chose as estimate $\tilde{\mathbf{n}}_{i,j}$ the normal which corresponds to the lowest greylevel. In [45], the greylevel estimate is defined as the average of $\tilde{I}^1_{i,j}$, $\tilde{I}^2_{i,j}$, $\tilde{I}^3_{i,j}$ and $\tilde{I}^4_{i,j}$: the computed images are very similar to those obtained using (29), but it is more difficult to find an estimate of the normal which is coherent with that of the greylevel.

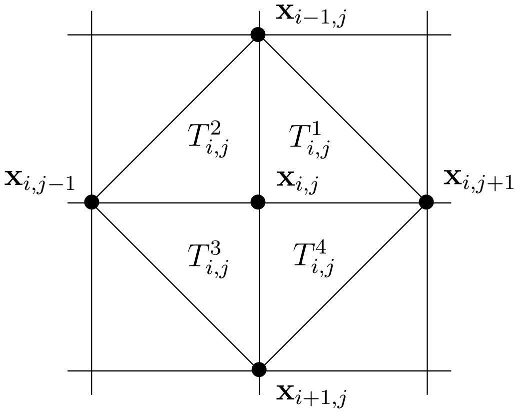


Fig. 9. Triangles used for the estimates of the greylevel and of the normal.

5.2.2 Panel of Error Estimators

We compare the methods in accordance with three error estimators. Weighted L^1 and L^2 errors and L^∞ error are defined as follows, for every known f and computable $\tilde{f}$:

$$\begin{cases} |\Delta f|_1 = \dfrac{1}{nb} \displaystyle\sum_{\mathbf{x}\in\Omega} \left|\tilde{f}(\mathbf{x}) - f(\mathbf{x})\right|, \\ |\Delta f|_2 = \left[\dfrac{1}{nb} \displaystyle\sum_{\mathbf{x}\in\Omega} \left|\tilde{f}(\mathbf{x}) - f(\mathbf{x})\right|^2\right]^{1/2}, \\ |\Delta f|_\infty = \max_{\mathbf{x}\in\Omega} \left\{\left|\tilde{f}(\mathbf{x}) - f(\mathbf{x})\right|\right\}. \end{cases} \quad (30)$$

These estimators are commonly called, respectively: the mean absolute deviation error, the root mean square error and the maximal absolute deviation error. On the one hand, let us make it clear that $|\tilde{u}(\mathbf{x}) - u(\mathbf{x})|$ and $\left|\tilde{I}(\mathbf{x}) - I(\mathbf{x})\right|$ are absolute values, whereas $|\tilde{\mathbf{n}}(\mathbf{x}) - \mathbf{n}(\mathbf{x})|$ is the Euclidean norm of a vector in $\mathbb{R}^3$. On the other hand, nb denotes the number of points $\mathbf{x} \in \Omega$ which are used to compute these three estimators. In most of the cases, f is known at each pixel, thus all the pixels $\mathbf{x}_{i,j} \in \Omega$ are used and $nb = N$. For Elk, u is known on a non-regular grid, so that $nb = 1104$ only, whereas Ω_{Elk} contains 15325 pixels. Finally, u is unknown for Pepper and Lena, and $\mathbf{n}$ is unknown for Elk, Pepper and Lena. Thus, we will measure the accuracy of the reconstructed shapes using nine numerical values for SV, CT, DEM and RV ($|\Delta u|_1$, $|\Delta u|_2$, $|\Delta u|_\infty$, $|\Delta \mathbf{n}|_1$, $|\Delta \mathbf{n}|_2$, $|\Delta \mathbf{n}|_\infty$, $|\Delta I|_1$, $|\Delta I|_2$, $|\Delta I|_\infty$), six numerical values for Elk ($|\Delta u|_1$, $|\Delta u|_2$, $|\Delta u|_\infty$, $|\Delta I|_1$, $|\Delta I|_2$, $|\Delta I|_\infty$) and three numerical values for Pepper and Lena ($|\Delta I|_1$, $|\Delta I|_2$, $|\Delta I|_\infty$). Note that the three errors on u are not bounded, while the three errors on $\mathbf{n}$ are bounded by 2 and the three errors on I are bounded by 1. In comparison, the survey made in [1] provided only three numerical values for the synthetic images (the mean absolute deviation error on u and on (p, q), and the standard deviation on u), and no value for the real images, because the greylevels were not estimated from the computed shapes. Finally, let us mention that a histogram of the percentage depth error is used as error estimator in [7].

6 Tests

This section is devoted to the tests and to their analysis. For each test, the CPU time of the reconstruction is displayed in the caption in seconds [4]. Beside each computed shape, the corresponding estimated image is shown. Moreover, for each test, the numerical estimates of the results are displayed in a table which contains the errors $|\Delta u|_1$, $|\Delta u|_2$ and $|\Delta u|_\infty$ on the computed shape, the errors $|\Delta \mathbf{n}|_1$, $|\Delta \mathbf{n}|_2$ and $|\Delta \mathbf{n}|_\infty$ on the estimated normals, and the errors $|\Delta I|_1$, $|\Delta I|_2$ and $|\Delta I|_\infty$ on the estimated greylevels. For Elk, Pepper and Lena, only some of these errors are available. Furthermore, we test the robustness of the algorithms by implementing them on reduced images. Finally, we discuss their robustness with respect to a perturbation in the direction of the light source.

---

[4] All the tests are done on a P4 2.4 GHz.

6.1 Choice of a Boundary Condition

Three different ways of imposing constraints on the boundary can be used: no Dirichlet boundary condition (BC_0); the homogeneous Dirichlet boundary condition (6) *i.e.*, $u = 0$ on $\partial\Omega$ (BC_1); the second Dirichlet boundary condition (7) *i.e.*, $u = g$ on $\partial\Omega$ (BC_2). On the one hand, FS can be tested only with BC_1 and BC_2. On the other hand, TS is tested only with BC_0: our attempts to impose BC_1 or BC_2 onto TS failed, since the computed shapes were of worse quality. Finally, DD can be tested with any these three types of boundary conditions.

The tests which are reported in this section do not use all the possible Dirichlet boundary conditions. In all cases, TS is tested with BC_0 only. If $g = 0$ *i.e.*, for CT, DEM and Elk, then FS and DD are tested with BC_1 only. If $g \neq 0$ and is unknown *i.e.*, for Pepper and Lena, then FS is tested with BC_1 and DD with BC_0. If $g \neq 0$ but is known *i.e.*, for SV and RV, then FS is tested either with BC_1 and with BC_2, and DD is tested either with BC_0 and with BC_2. Every time FS or DD are tested with BC_2, they are marked with a * (FS* or DD*), meaning that such knowledge is rarely available.

It must also be noted that the shapes which are reconstructed without boundary condition (BC_0) are modified *a posteriori*: a global shift is applied to the computed shape inside Ω, which is determined in such a way that $|\Delta u|_2$ is minimized, when the ground truth is known. Fortunately, this shift does not affect the errors on the estimated normals and on the estimated greylevels.

6.2 Test 1: SV

The starting point is the synthetic image of a vase shown in Fig. 3-b. Two boundary conditions are considered for FS: BC_1 (*cf.* Fig. 10-left) and BC_2 (*cf.* Fig. 13-left). Since g_{SV} is a continuous function vanishing on the left and on the right hand sides of the vase, but not at the top and at the bottom, BC_1 is clearly wrong. The qualitative effect of boundary conditions is clearly visible in the shapes obtained by this method. FS does not introduce any smoothing, thus the small kinks on the boundary (which is approximated by square cells) produce several lines of discontinuity for $\tilde{\mathbf{n}}$ inside Ω_{SV}. Moreover, the wrong boundary condition $u = 0$ produces a wrong shape due to the concave/convex ambiguity in the model (*cf.* Fig. 10-left). Two situations are also considered for DD: BC_0 (*cf.* Fig. 11-left) and BC_2 (*cf.* Fig. 14-left). Both solutions are smooth, due to the regularization term which is present in the functional, and do not seem to be affected by the concave/convex ambiguity. Finally, TS does not need any Dirichlet boundary condition. The computed shape with TS (*cf.*

Fig. 12-left) is qualitatively better than that obtained by FS with BC_1, thanks to discontinuity at the top and at the bottom of the vase *i.e.*, where BC_1 is clearly wrong. Let us recall that the computed shapes obtained by TS and DD are shifted *a posteriori* inside Ω_{SV}, since a global drift occurs during the computation. This phenomenon is mentioned neither in [87] nor in [1] for TS, but is necessarily compensated for in both these papers, since otherwise our results would differ from theirs.

The reconstruction of the image on the basis of the shape uses the methodology described in Section 5.2.1. The estimation of I is quite satisfactory for all the methods. Table 1 confirms this statement, since the errors on I have the same order of magnitude. Conversely, Table 1 shows that FS* and DD* are the most accurate in reconstructing u. Table 1 also shows the errors in the estimation of the normals: it can be seen that all the methods have the same order of magnitude. Finally, TS is much faster (by a factor of 50) with respect to DD and FS.

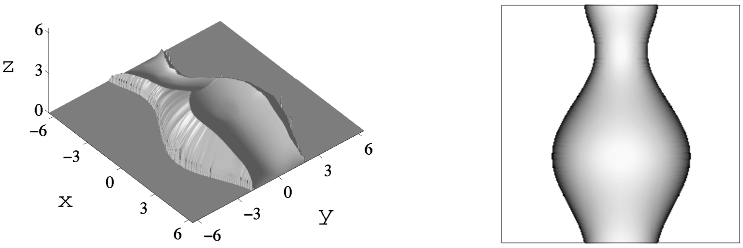


Fig. 10. FS on SV: 2.14 s.

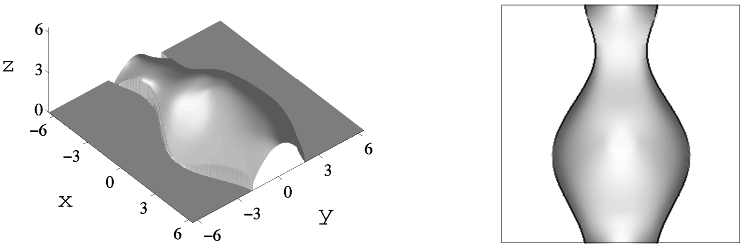


Fig. 11. DD on SV: 3.30 s.

6.3 Test 2: CT

The second test we discuss is related to a synthetic image representing a Canadian tent with two different slopes (*cf.* Fig. 4-b). The important difference in comparison with the first test consists in the sharp edges of the surface

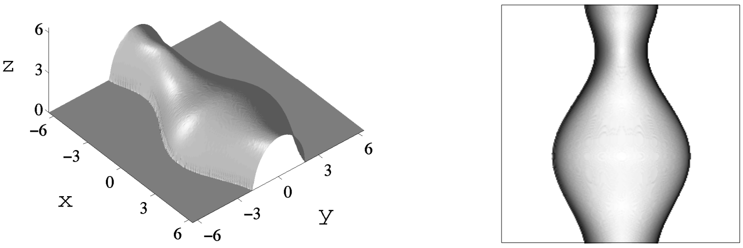


Fig. 12. TS on SV: 0.03 s.

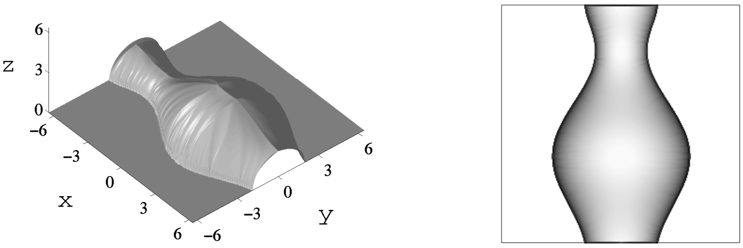


Fig. 13. FS* on SV: 1.08 s.

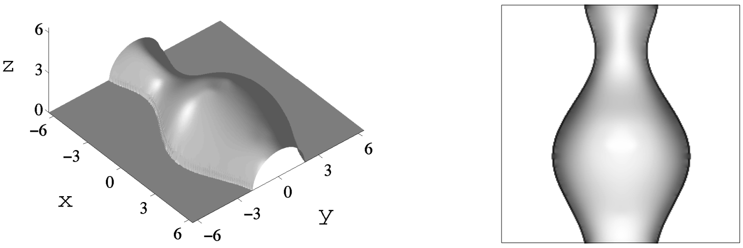


Fig. 14. DD* on SV: 2.63 s.

Table 1
Errors on SV ($\boldsymbol{\omega} = \boldsymbol{\omega}^0$).

	$\lvert\Delta u\rvert_1$	$\lvert\Delta u\rvert_2$	$\lvert\Delta u\rvert_\infty$	$\lvert\Delta \mathbf{n}\rvert_1$	$\lvert\Delta \mathbf{n}\rvert_2$	$\lvert\Delta \mathbf{n}\rvert_\infty$	$\lvert\Delta I\rvert_1$	$\lvert\Delta I\rvert_2$	$\lvert\Delta I\rvert_\infty$
FS	0.80	1.00	1.93	0.49	0.63	1.95	0.01	0.01	0.17
DD	0.29	0.38	1.61	0.21	0.28	1.99	0.05	0.08	0.62
TS	0.53	0.62	1.63	0.24	0.30	1.08	0.07	0.10	0.37
FS*	0.23	0.25	0.48	0.14	0.23	1.35	0.01	0.06	0.78
DD*	0.19	0.23	0.43	0.11	0.14	0.58	0.03	0.04	0.26

and the corresponding discontinuities in the input image. As can be seen in Figs. 15-left, 16-left and 17-left, the best result is obtained by FS, due to its capacity to follow the jumps in the gradient of u. DD provides a very smooth surface with no kinks and TS computes a quite flat surface. The qualitative behaviour of the three methods is numerically confirmed in Table 2. On the other hand, TS CPU time is much shorter than the two others.

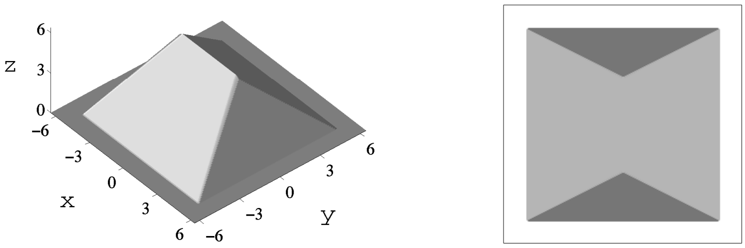


Fig. 15. FS on CT: 1.62 s.

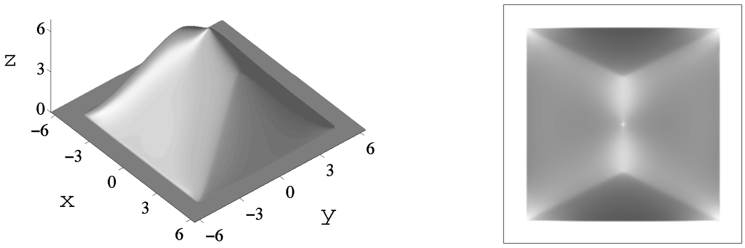


Fig. 16. DD on CT: 3.77 s.

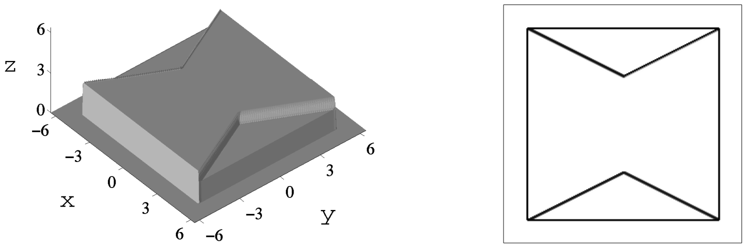


Fig. 17. TS on CT: 0.04 s.

6.4 Test 3: RV

Consider the real image of a vase given in Fig. 6-a. As in Test 1 (SV), we consider two boundary conditions for FS and DD. It can be seen from Figs. 18-left, 19-left, 20-left, 21-left and 22-left that the results show behaviour in the

Table 2
Errors on CT.

	$\lvert\Delta u\rvert_1$	$\lvert\Delta u\rvert_2$	$\lvert\Delta u\rvert_\infty$	$\lvert\Delta \mathbf{n}\rvert_1$	$\lvert\Delta \mathbf{n}\rvert_2$	$\lvert\Delta \mathbf{n}\rvert_\infty$	$\lvert\Delta I\rvert_1$	$\lvert\Delta I\rvert_2$	$\lvert\Delta I\rvert_\infty$
FS	0.03	0.04	0.20	0.03	0.11	1.41	0.01	0.01	0.08
DD	0.57	0.74	1.89	0.26	0.32	1.29	0.08	0.10	0.51
TS	1.19	1.40	3.49	0.83	0.84	1.71	0.36	0.38	0.69

three methods which is similar to Test 1, except FS* (*cf.* Fig. 21-left) which causes a small inversion at the top of the vase, due to the concave/convex ambiguity. Since all errors are of the same order of magnitude as those in Test 1 (compare Table 3 with Table 1), and considering that RV is a noisy version of SV, this shows that the three methods are stable in the presence of noise in the image. However, TS is qualitatively affected by the noise, since the computed shape and the estimated image displayed in Fig. 20 look grainy. It can be noted that DD* (*cf.* Fig. 22-left) provides a surface which seems to be the most accurate reconstruction, and comparison of the estimated images indicates a better performance of FS: Table 3 confirms both these statements. Also in this case, CPU times show that TS is much faster than DD and FS.

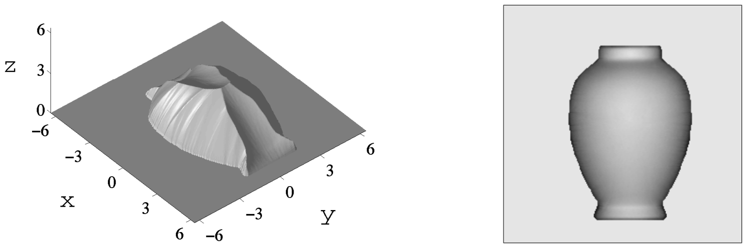


Fig. 18. FS on RV: 0.75 s.

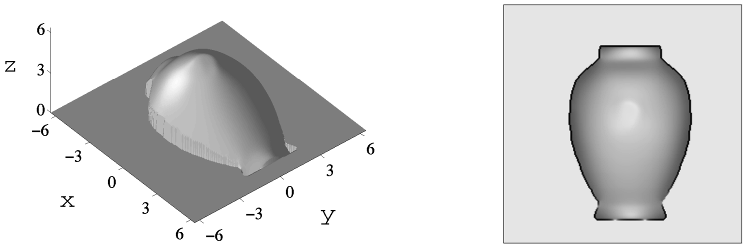


Fig. 19. DD on RV: 2.53 s.

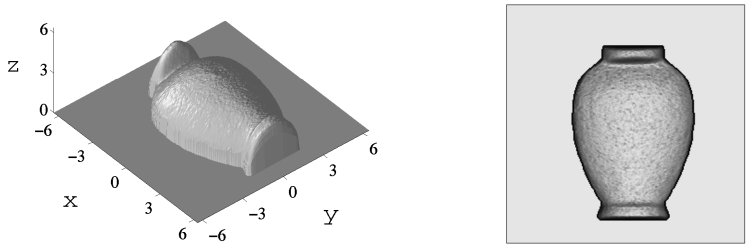


Fig. 20. TS on RV: 0.02 s.

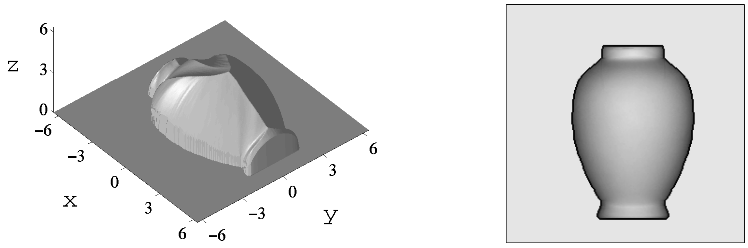


Fig. 21. FS* on RV: 0.78 s.

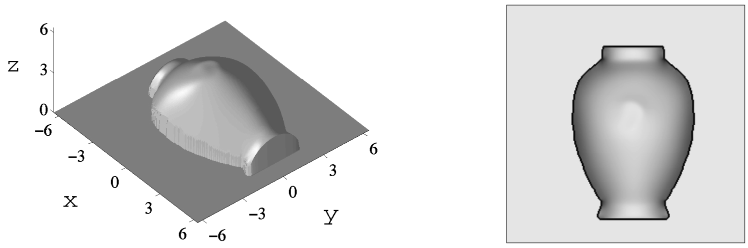


Fig. 22. DD* on RV: 1.80 s.

Table 3
Errors on RV.

	$\lvert\Delta u\rvert_1$	$\lvert\Delta u\rvert_2$	$\lvert\Delta u\rvert_\infty$	$\lvert\Delta \mathbf{n}\rvert_1$	$\lvert\Delta \mathbf{n}\rvert_2$	$\lvert\Delta \mathbf{n}\rvert_\infty$	$\lvert\Delta I\rvert_1$	$\lvert\Delta I\rvert_2$	$\lvert\Delta I\rvert_\infty$
FS	0.78	0.84	1.83	0.37	0.46	1.67	0.01	0.01	0.08
DD	0.39	0.52	2.00	0.24	0.33	1.98	0.03	0.06	0.51
TS	0.30	0.41	1.61	0.32	0.45	1.87	0.10	0.15	0.72
FS*	0.24	0.31	0.86	0.26	0.32	1.74	0.01	0.04	0.52
DD*	0.17	0.20	0.45	0.15	0.20	1.16	0.05	0.07	0.52

6.5 Test 4: DEM

This test uses the image of a DEM (*cf.* Fig. 5-b). It is first noted that all the reconstructed shapes are qualitatively wrong (*cf.* Figs. 23-left, 24-left and 25-left): FS computes the maximal solution, which does not coincide with the real surface; DD computes one of the admissible minimum energy configurations; TS shape excessively ondulates, since each singular point is interpreted as a local maximum. In any case, we can observe in Table 4 that the order of magnitude of the errors on the shapes and on the normals is the same for the three methods. As can be seen in Figs. 23-right, 24-right and 25-right, the estimated image is qualitatively much better using FS than the two other methods. Table 4 numerically confirms this statement.

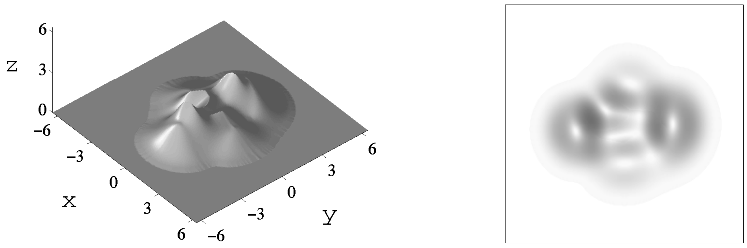


Fig. 23. FS on DEM: 0.73 s.

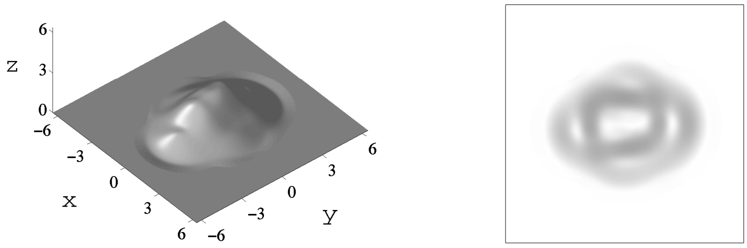


Fig. 24. DD on DEM: 1.69 s.

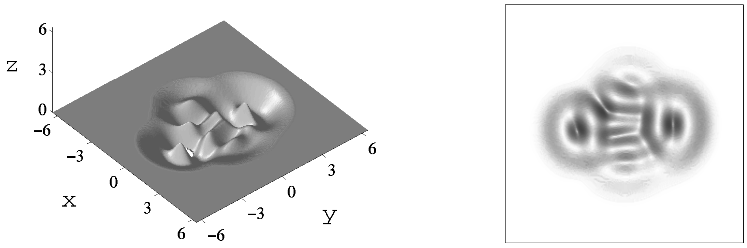


Fig. 25. TS on DEM: 0.03 s.

Table 4
Errors on DEM.

	$\lvert\Delta u\rvert_1$	$\lvert\Delta u\rvert_2$	$\lvert\Delta u\rvert_\infty$	$\lvert\Delta \mathbf{n}\rvert_1$	$\lvert\Delta \mathbf{n}\rvert_2$	$\lvert\Delta \mathbf{n}\rvert_\infty$	$\lvert\Delta I\rvert_1$	$\lvert\Delta I\rvert_2$	$\lvert\Delta I\rvert_\infty$
FS	0.52	0.87	3.14	0.49	0.68	1.79	0.01	0.01	0.06
DD	0.59	0.88	2.51	0.52	0.65	1.59	0.06	0.08	0.31
TS	0.64	0.94	2.93	0.67	0.81	1.77	0.12	0.17	0.67

6.6 Test 5: Elk

This test deals with a real greylevel image representing the moulding of an elk's head (*cf.* Fig. 7-a) which was used in [62] as a test problem. In this test, FS and TS seem to produce more realistic shapes than DD (*cf.* Figs. 26-left, 27-left and 28-left), but this is not numerically confirmed. In Table 5, it can be noted that FS produces slightly better results on I compared to DD and TS. This is confirmed from the computed images shown in Figs. 26-right, 27-right and 28-right. Once again, TS shows a faster CPU time, and DD is slower than FS.

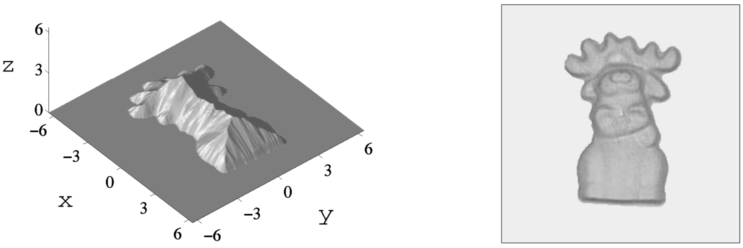


Fig. 26. FS on Elk: 0.42 s.

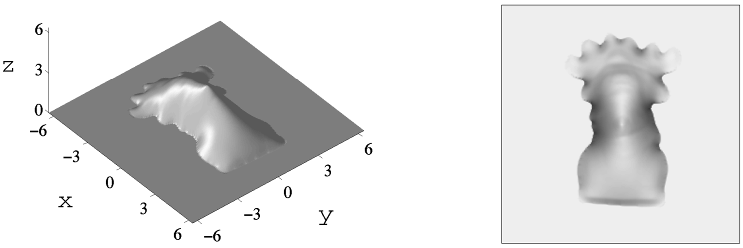


Fig. 27. DD on Elk: 1.25 s.

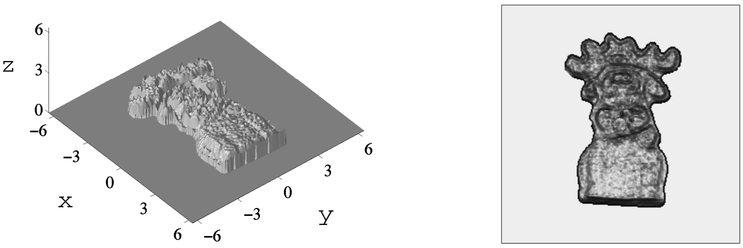


Fig. 28. TS on Elk: 0.01 *s*.

Table 5
Errors on Elk.

	$\|\Delta u\|_1$	$\|\Delta u\|_2$	$\|\Delta u\|_\infty$	$\|\Delta I\|_1$	$\|\Delta I\|_2$	$\|\Delta I\|_\infty$
FS	0.21	0.30	1.00	0.02	0.03	0.18
DD	0.32	0.43	1.44	0.10	0.14	0.53
TS	0.35	0.42	1.20	0.28	0.33	0.89

6.7 Tests 6 and 7: Pepper and Lena

These two tests deal with images which clearly do not conform to the basic assumptions of SFS, but are considered therefore as classical benchmarks. For both these images, we have no additional information for the ground truth, so only computed images will be numerically compared. Whereas the computed shapes seem to be very far from the real ones, it appears that the estimated images are not so bad, at least for FS, even if a saturation phenomenon can be observed on both sides of Fig. 29-right. The images provided by DD (*cf.* Figs. 30-right and 33-right) are blurred, due to the smoothness term, whereas those provided by TS (*cf.* Figs. 31-right and 34-right) are noisy and excessively emphasize the contours. This ranking is numerically confirmed in Table 6.

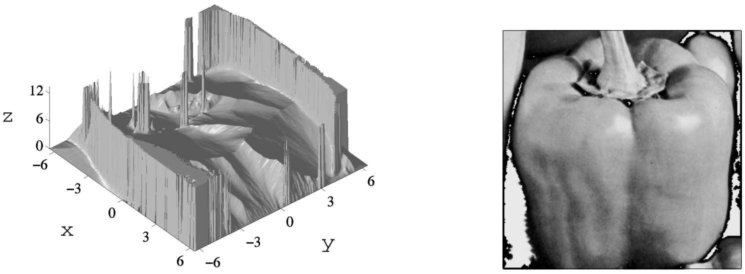


Fig. 29. FS on Pepper: 5.98 *s*.

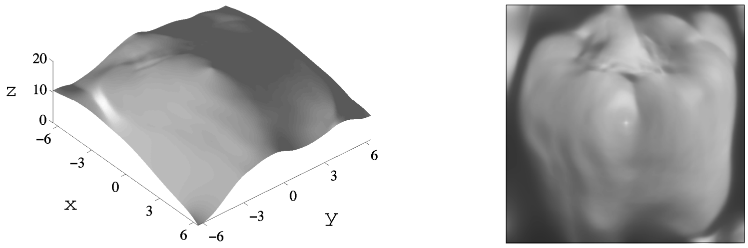


Fig. 30. DD on Pepper: 7.62 s.

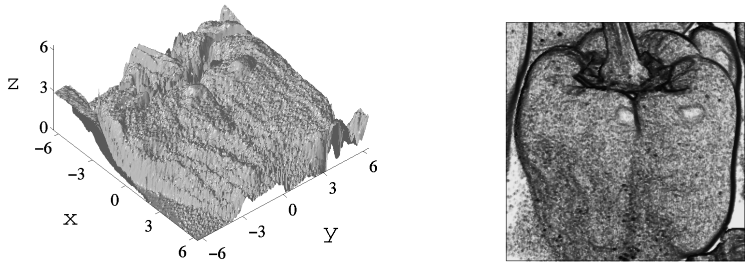


Fig. 31. TS on Pepper: 0.05 s.

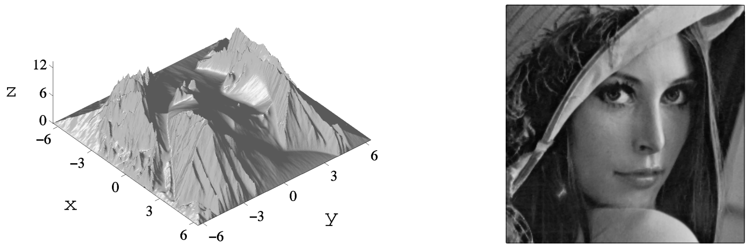


Fig. 32. FS on Lena: 6.15 s.

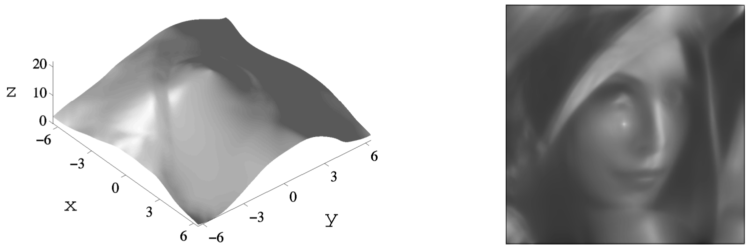


Fig. 33. DD on Lena: 7.93 s.

6.8 Test 8: Reduced Images of SV

Next, we test the robustness of the algorithms by implementing them on reduced square images of SV, whose size varies between 16 and 256. Some of the

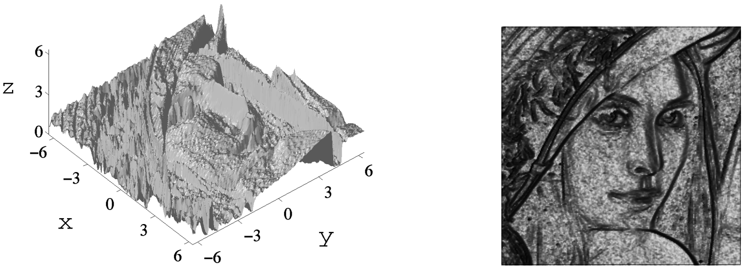


Fig. 34. TS on Lena: 0.05 s.

Table 6
Errors on Pepper (columns on the left) and on Lena (columns on the right).

	$\vert\Delta I\vert_1$	$\vert\Delta I\vert_2$	$\vert\Delta I\vert_\infty$	$\vert\Delta I\vert_1$	$\vert\Delta I\vert_2$	$\vert\Delta I\vert_\infty$
FS	0.08	0.24	1.00	0.04	0.07	0.77
DD	0.10	0.14	0.70	0.12	0.16	0.74
TS	0.27	0.36	1.00	0.25	0.31	0.95

numerical results are shown in Figs 35, 36 and 37: $|\Delta u|_1$, $|\Delta \mathbf{n}|_1$ and $|\Delta I|_1$ are plotted as a function of the image size. It can be observed that the accuracy of the methods on the full-size images is on the whole confirmed on the low resolution images. With very few exceptions ($|\Delta u|_1$ for DD and DD*; $|\Delta \mathbf{n}|_1$ for FS), all curves are decreasing. Surprisingly, some of the curves relating to FS and FS* oscillate.

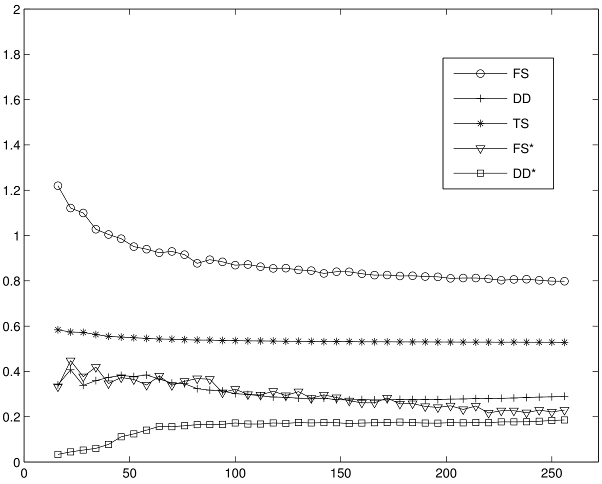


Fig. 35. Tests on reduced images of SV: $|\Delta u|_1$ in function of the image size.

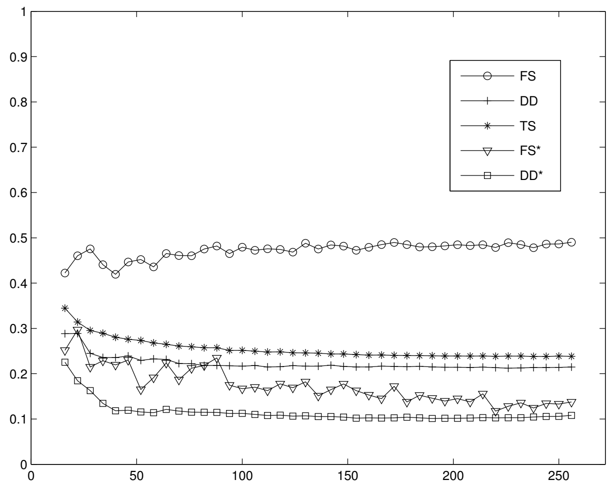


Fig. 36. Tests on reduced images of SV: $|\Delta \mathbf{n}|_1$ in function of the image size.

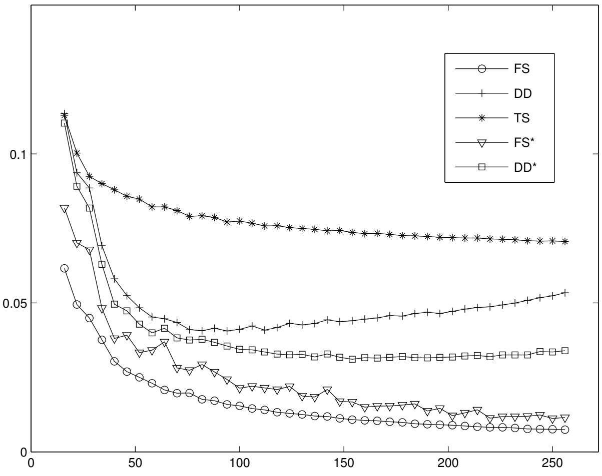


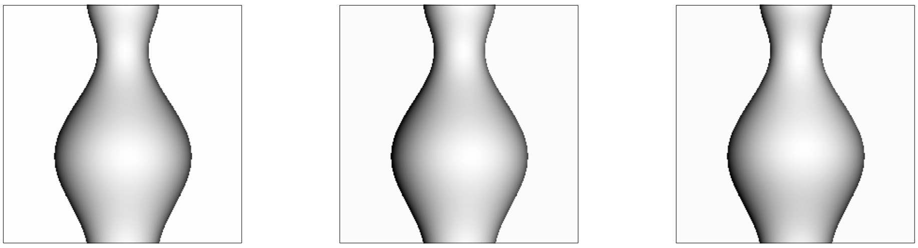
Fig. 37. Tests on reduced images of SV: $|\Delta I|_1$ in function of the image size.

6.9 Test 9: Error on the Lighting Direction

Finally, we also investigate the stability of the three methods with respect to perturbations in the light source direction. We consider three images of the synthetic vase (cf. Fig. 38) that are obtained using three different non-frontal light source directions, *viz.*:

$$\begin{cases} \boldsymbol{\omega}^1 = (0, 0.087, 0.996), \\ \boldsymbol{\omega}^2 = (0, 0.174, 0.985), \\ \boldsymbol{\omega}^3 = (-0.123, 0.123, 0.985). \end{cases} \tag{31}$$

Note that the shadows in the background are not taken into account, but this is not a problem since the background is outside Ω_{SV}. These images are



(a) (b) (c)

Fig. 38. SV: images with (a) $\boldsymbol{\omega} = \boldsymbol{\omega}^1$, (b) $\boldsymbol{\omega} = \boldsymbol{\omega}^2$ and (c) $\boldsymbol{\omega} = \boldsymbol{\omega}^3$.

processed as if the light source were frontal (except with TS, for the reason which has already been mentioned in section 4.2.3).

Since the numerical results show that the behaviour of the three methods is similar for the three light source directions, we limit ourselves to commenting on the results relating to $\boldsymbol{\omega}^1$ (cf. Table 7) as a representative example. All errors have the same order of magnitude when $\boldsymbol{\omega} = \boldsymbol{\omega}^1$ as in the case of frontal lighting (compare Table 7 with Table 1). Nevertheless, it must be noted that the error increase due to this perturbation on $\boldsymbol{\omega}$ is (proportionally) higher when the reconstruction is more accurate: TS is less affected than FS and DD; FS is less affected than FS*.

Table 7
Errors on SV ($\boldsymbol{\omega} = \boldsymbol{\omega}^1$).

	$\lvert\Delta u\rvert_1$	$\lvert\Delta u\rvert_2$	$\lvert\Delta u\rvert_\infty$	$\lvert\Delta \mathbf{n}\rvert_1$	$\lvert\Delta \mathbf{n}\rvert_2$	$\lvert\Delta \mathbf{n}\rvert_\infty$	$\lvert\Delta I\rvert_1$	$\lvert\Delta I\rvert_2$	$\lvert\Delta I\rvert_\infty$
FS	0.88	1.20	13.47	0.53	0.68	1.99	0.01	0.01	0.19
DD	0.33	0.45	1.99	0.23	0.31	2.00	0.05	0.08	0.68
TS	0.55	0.65	1.66	0.26	0.33	1.97	0.07	0.10	0.49
FS*	0.39	0.47	1.09	0.28	0.40	1.49	0.01	0.07	0.90
DD*	0.22	0.28	0.62	0.15	0.18	0.67	0.05	0.06	0.32

6.10 Summary

We finish this section by summing up our observations.

Convergence. Convergence properties are often crucial in the choice of a method. Let us note that FS and DD always converge to a solution of the SFS problem, whereas convergence is not guaranteed by TS. In fact, increasing the number of iterations in TS generally does not improve the solution in terms of any of our error indicators. It should also be noted that DD will converge in general to a local minimum. The search for an overall minimum would require much more computational effort as it would require switching to stochastic algorithms. The FS method always converges to the maximal solution which, in general, will not coincide with the real surface. One can try to obtain other solutions by reflection, once the maximal solution is computed, and/or by adding information such as the height at singular points.

Boundary Conditions. In real images, it is sometimes possible to have an *a priori* knowledge on the surface, such as the height on the boundary. It is interesting to note that this information can easily be included in FS and DD. Of course, a wrong *a priori* knowledge on the height on the boundary of the reconstruction domain affects the accuracy of the reconstruction.

Accuracy. DD produces accurate results for smooth surfaces, whereas its accuracy decreases when it is applied to non-differentiable surfaces. In the same way, TS gives its best results on smooth surfaces although its performance decreases for non-differentiable surfaces and for smooth surfaces that exhibit several maxima and minima (as in Test 4). Both these methods have good accuracy in the reconstruction of the normals. Finally, FS seems to be well adapted to a variety of different situations which include smooth and non-smooth surfaces. Its accuracy on the reconstruction of the surface is good, provided the surface corresponds to the maximal solution. A weak point of FS seems to be the absence of smoothing in the surfaces, which in some cases produces larger errors in the reconstruction of the normals. Moreover, FS always requires a Dirichlet boundary condition to be imposed, even if it is wrong.

Stability. The three methods appear to be stable in the presence of a perturbation in the light source direction. Moreover, their behaviour is not affected by a change in the image resolution. Only TS seems to be affected by the presence of noise in the image (in [1], a Gaussian smoothing is applied to the results of TS).

CPU Time. TS is the fastest method in all the tests since the algorithm is stopped after 5 iterations, where the other methods go on until convergence. For the tests reported in this section, FS requires between 446 and 1879 iterations, and DD requires between 93 and 274 iterations, but these numbers strongly depend on the accuracy which is requested.

7 Conclusion

We hope that these comparisons of three methods of resolution, which are prototypes in their classes, will help researchers and practitioners to make their choices when dealing with SFS problems. The analysis contained in this paper could give rise to several interesting developments and improvements, such as the extension of the SFS model to more realistic situations, including for example non-frontal lighting or perspective projection.

Of course, we also plan to increase little by little the number of methods of resolution and of benchmarks available on our web site. The portable platform contains at the moment about 2600 lines of C-ANSI code, which is quite a lot of work, but implementing a method of resolution, just using the description given in a paper, can be much trickier than it might appear. Therefore, we hope that many other researchers will make available their own codes and images.

References

[1] R. Zhang, P.-S. Tsai, J. E. Cryer, M. Shah, Shape from Shading: A Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (8) (1999) 690–706.

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

[3] B. K. P. Horn, Height and Gradient from Shading, International Journal of Computer Vision 5 (1) (1990) 37–75.

[4] R. Kimmel, K. Siddiqi, B. B. Kimia, A. M. Bruckstein, Shape from Shading: Level Set Propagation and Viscosity Solutions, International Journal of Computer Vision 16 (2) (1995) 107–133.

[5] R. Kozera, An Overview of the Shape from Shading Problem, Machine Graphics and Vision 7 (1–2) (1998) 291–312.

[6] R. Klette, R. Kozera, K. Schlüns, Reflectance-Based Shape Recovery, in: B. Jähne, H. Haussecker, P. Geissler (Eds.), Signal Processing and Pattern Recognition, Vol. 2 of Handbook of Computer Vision and Applications, Academic Press, 1999, Ch. 19, pp. 531–590.

[7] D. Samaras, D. N. Metaxas, Incorporating Illumination Constraints in Deformable Models for Shape from Shading and Light Direction Estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2) (2003) 247–264.

[8] W. Chojnacki, M. J. Brooks, D. Gibbins, Revisiting Pentland's Estimator of Light Source Direction, Journal of the Optical Society of America - Part A: Optics, Image Science, and Vision 11 (1) (1994) 118–124.

[9] M. S. Drew, Direct Solution of Orientation-from-Color Problem Using a Modification of Pentland's Light Source Direction Estimator, Computer Vision and Image Understanding 64 (2) (1996) 286–299.

[10] P. N. Belhumeur, D. J. Kriegman, A. L. Yuille, The Bas-Relief Ambiguity, International Journal of Computer Vision 35 (1) (1999) 33–44.

[11] M. Oren, S. K. Nayar, Generalization of the Lambertian Model and Implications for Machine Vision, International Journal of Computer Vision 14 (3) (1995) 227–251.

[12] S. M. Bakshi, Y.-H. Yang, Shape From Shading for Non-Lambertian Surfaces, in: Proceedings of the IEEE International Conference on Image Processing (volume II), Austin, Texas, USA, 1994, pp. 130–134.

[13] K. M. Lee, C.-C. J. Kuo, Shape from Shading with a Generalized Reflectance Map Model, Computer Vision and Image Understanding 67 (2) (1997) 143–160.

[14] H. Ragheb, E. R. Hancock, A probabilistic framework for specular shape-from-shading, Pattern Recognition 36 (2) (2003) 407–427.

[15] A. J. Stewart, M. S. Langer, Toward Accurate Recovery of Shape from Shading Under Diffuse Lighting, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (9) (1997) 1020–1025.

[16] Y. Wang, D. Samaras, Estimation of Multiple Illuminants from a Single Image of Arbitrary Known Geometry, in: Proceedings of the 7th European Conference on Computer Vision (volume III), Vol. 2352 of Lecture Notes in Computer Science, Copenhagen, Denmark, 2002, pp. 272–288.

[17] S. K. Nayar, K. Ikeuchi, T. Kanade, Shape from Interreflections, International Journal of Computer Vision 6 (3) (1991) 173–195.

[18] D. A. Forsyth, A. Zisserman, Reflections on Shading, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (7) (1991) 671–679.

[19] M. A. Penna, A Shape from Shading Analysis for a Single Perspective Image of a Polyhedron, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (6) (1989) 545–554.

[20] K. M. Lee, C.-C. J. Kuo, Shape from Shading with Perspective Projection, Computer Vision, Graphics, and Image Processing: Image Understanding 59 (2) (1994) 202–212.

[21] J. K. Hasegawa, C. L. Tozzi, Shape from Shading with Perspective Projection and Camera Calibration, Computers and Graphics 20 (3) (1996) 351–364.

[22] T. Okatani, K. Deguchi, Shape Reconstruction from an Endoscope Image by Shape from Shading Technique for a Point Light Source at the Projection Center, Computer Vision and Image Understanding 66 (2) (1997) 119–131.

[23] D. Samaras, D. N. Metaxas, Coupled Lighting Direction and Shape Estimation from Single Images, in: Proceedings of the 7th IEEE International Conference on Computer Vision (volume II), Kerkyra, Greece, 1999, pp. 868–874.

[24] E. Prados, O. Faugeras, "Perspective Shape from Shading" and Viscosity Solutions, in: Proceedings of the 9th IEEE International Conference on Computer Vision (volume II), Nice, France, 2003, pp. 826–831.

[25] A. Tankus, N. Sochen, Y. Yeshurun, A New Perspective [on] Shape-from-Shading, in: Proceedings of the 9th IEEE International Conference on Computer Vision (volume II), Nice, France, 2003, pp. 862–869.

[26] F. Courteille, A. Crouzil, J.-D. Durou, P. Gurdjos, Towards shape from shading under realistic photographic conditions, in: Proceedings of the 17th International Conference on Pattern Recognition (volume II), Cambridge, UK, 2004, pp. 277–280.

[27] E. Prados, O. Faugeras, F. Camilli, Shape from Shading: a well-posed problem?, Rapport de Recherche 5297, Institut National de Recherche en Informatique et en Automatique, Sophia Antipolis, France (Aug. 2004).

[28] J. Van Diggelen, A Photometric Investigation of the Slopes and Heights of the Ranges of Hills in the Maria of the Moon, Bulletin of the Astronomical Institute of the Netherlands 11 (423) (1951) 283–290.

[29] T. Rindfleisch, Photometric Method for Lunar Topography, Photometric Engineering 32 (2) (1966) 262–277.

[30] B. K. P. Horn, Obtaining Shape from Shading Information, in: P. H. Winston (Ed.), The Psychology of Computer Vision, McGraw-Hill, 1975, Ch. 4, pp. 115–155.

[31] J. Oliensis, Uniqueness in Shape from Shading, International Journal of Computer Vision 6 (2) (1991) 75–104.

[32] J. Oliensis, Shape from Shading as a Partially Well-Constrained Problem, Computer Vision, Graphics, and Image Processing: Image Understanding 54 (2) (1991) 163–183.

[33] M. J. Brooks, W. Chojnacki, R. Kozera, Impossible and Ambiguous Shading Patterns, International Journal of Computer Vision 7 (2) (1992) 119–126.

[34] R. Kozera, Uniqueness in Shape from Shading Revisited, Journal of Mathematical Imaging and Vision 7 (2) (1997) 123–138.

[35] E. Rouy, A. Tourin, A Viscosity Solutions Approach to Shape-from-shading, SIAM Journal on Numerical Analysis 29 (3) (1992) 867–884.

[36] G. Barles, Solutions de viscosité des équations de Hamilton-Jacobi, Springer-Verlag, 1994, (in french).

[37] P.-L. Lions, E. Rouy, A. Tourin, Shape-from-Shading, viscosity solutions and edges, Numerische Mathematik 64 (3) (1993) 323–353.

[38] E. Prados, O. Faugeras, E. Rouy, Shape from Shading and Viscosity Solutions, in: Proceedings of the 7th European Conference on Computer Vision (volume II), Vol. 2351 of Lecture Notes in Computer Science, Copenhagen, Denmark, 2002, pp. 790–804.

[39] J. Oliensis, P. Dupuis, A Global Algorithm for Shape from Shading, in: Proceedings of the 4th IEEE International Conference on Computer Vision, Berlin, Germany, 1993, pp. 692–701.

[40] H. Ishii, M. Ramaswamy, Uniqueness Results for a Class of Hamilton-Jacobi Equations with Singular Coefficients, Communications in Partial Differential Equations 20 (1995) 2187–2213.

[41] F. Camilli, M. Falcone, An Approximation Scheme for the Maximal Solution of the Shape-from-Shading Model, in: Proceedings of the IEEE International Conference on Image Processing (volume I), Lausanne, Switzerland, 1996, pp. 49–52.

[42] F. Camilli, A. Siconolfi, Maximal Subsolutions of a Class of Degenerate Hamilton-Jacobi Problems, Indiana University Mathematics Journal 48 (3) (1999) 1111–1131.

[43] M. Falcone, M. Sagona, An algorithm for the global solution of the Shape-from-Shading model, in: Proceedings of the 9^{th} International Conference on Image Analysis and Processing (volume I), Vol. 1310 of Lecture Notes in Computer Science, Florence, Italy, 1997, pp. 596–603.

[44] F. Camilli, L. Grüne, Numerical Approximation of the Maximal Solutions for a Class of Degenerate Hamilton-Jacobi Equations, SIAM Journal on Numerical Analysis 38 (5) (2000) 1540–1560.

[45] M. Sagona, Numerical methods for degenerate Eikonal type equations and applications, Tesi di dottorato, Dipartimento di Matematica dell'Università di Napoli "Federico II", Napoli, Italy (Nov. 2001).

[46] A. Tankus, N. Sochen, Y. Yeshurun, Perspective Shape-from-Shading by Fast Marching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (volume I), Washington, D.C., USA, 2004, pp. 43–49.

[47] A. Tankus, N. Sochen, Y. Yeshurun, Shape-from-Shading Under Perspective Projection, International Journal of Computer Vision 63 (1) (2005) 21–43.

[48] R. Kimmel, J. A. Sethian, Optimal Algorithm for Shape from Shading and Path Planning, Journal of Mathematical Imaging and Vision 14 (3) (2001) 237–244.

[49] E. Prados, O. Faugeras, A generic and provably convergent Shape-From-Shading Method for Orthographic and Pinhole Cameras, International Journal of Computer Vision 65 (1–2) (2005) 97–125.

[50] R. Dovgard, R. Basri, Statistical Symmetric Shape from Shading for 3D Structure Recovery of Faces, in: Proceedings of the 8^{th} European Conference on Computer Vision (volume II), Vol. 3022 of Lecture Notes in Computer Science, Prague, Czech Republic, 2004, pp. 99–113.

[51] J. Kain, D. N. Ostrov, Numerical Shape-from-Shading for Discontinuous Photographic Images, International Journal of Computer Vision 44 (3) (2001) 163–173.

[52] E. Prados, O. Faugeras, A mathematical and algorithmic study of the Lambertian SFS problem for orthographic and pinhole cameras, Rapport

de Recherche 5005, Institut National de Recherche en Informatique et en Automatique, Sophia Antipolis, France (Nov. 2003).

[53] M. Falcone, M. Sagona, A. Seghini, A global algorithm for the Shape-from-Shading model with "black shadows", in: F. Brezzi, A. Buffa, S. Corsaro, A. Murli (Eds.), Numerical Mathematics and Advanced Applications - ENUMATH 2001, Springer-Verlag, 2003, pp. 503–512.

[54] A. M. Bruckstein, On Shape from Shading, Computer Vision, Graphics, and Image Processing 44 (2) (1988) 139–154.

[55] R. Kimmel, A. M. Bruckstein, Global Shape from Shading, Computer Vision and Image Understanding 62 (3) (1995) 360–369.

[56] R. Kimmel, A. M. Bruckstein, Tracking Level Sets by Level Sets: A Method for Solving the Shape from Shading Problem, Computer Vision and Image Understanding 62 (1) (1995) 47–58.

[57] A. R. Bruss, The Eikonal Equation: Some Results Applicable to Computer Vision, Journal of Mathematical Physics 23 (5) (1982) 890–896.

[58] J.-D. Durou, D. Piau, Ambiguous Shape from Shading with Critical Points, Journal of Mathematical Imaging and Vision 12 (2) (2000) 99–108.

[59] E. Prados, F. Camilli, O. Faugeras, A viscosity solution method for Shape-From-Shading without image boundary data, ESAIM: Mathematical Modelling and Numerical Analysis 40 (2) (2006) 393–412.

[60] B. K. P. Horn, M. J. Brooks, The Variational Approach to Shape From Shading, Computer Vision, Graphics, and Image Processing 33 (2) (1986) 174–208.

[61] Y. G. Leclerc, A. F. Bobick, The Direct Computation of Height from Shading, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, USA, 1991, pp. 552–558.

[62] P. Daniel, J.-D. Durou, From Deterministic to Stochastic Methods for Shape From Shading, in: Proceedings of the 4th Asian Conference on Computer Vision, Taipei, Taiwan, 2000, pp. 187–192.

[63] T. M. Strat, A Numerical Method for Shape-From-Shading From a Single Image, Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA (1979).

[64] R. T. Frankot, R. Chellappa, A Method for Enforcing Integrability in Shape from Shading Algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence 10 (4) (1988) 439–451.

[65] R. Szeliski, Fast Shape from Shading, Computer Vision, Graphics, and Image Processing: Image Understanding 53 (2) (1991) 129–153.

[66] K. Ikeuchi, B. K. P. Horn, Numerical Shape from Shading and Occluding Boundaries, Artificial Intelligence 17 (1–3) (1981) 141–184.

[67] D. Lee, A Provably Convergent Algorithm for Shape from Shading, in: Proceedings of the DARPA Image Understanding Workshop, Miami Beach, Florida, USA, 1985, pp. 489–496.

[68] M. J. Brooks, B. K. P. Horn, Shape and Source from Shading, in: Proceedings of the 9$^{\text{th}}$ International Joint Conference on Artificial Intelligence (volume II), Los Angeles, California, USA, 1985, pp. 932–936.

[69] P. L. Worthington, E. R. Hancock, New Constraints on Data-Closeness and Needle Map Consistency for Shape-from-Shading, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (12) (1999) 1250–1267.

[70] T. C. Pong, R. M. Haralick, L. G. Shapiro, Shape from Shading Using the Facet Model, Pattern Recognition 22 (6) (1989) 683–695.

[71] K. M. Lee, C.-C. J. Kuo, Shape from Shading with a Linear Triangular Element Surface Model, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (8) (1993) 815–822.

[72] D. Samaras, D. N. Metaxas, Incorporating Illumination Constraints in Deformable Models, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, California, USA, 1998, pp. 322–329.

[73] F. Courteille, J.-D. Durou, G. Morin, A Global Solution to the SFS Problem Using B-spline Surface and Simulated Annealing, in: Proceedings of the 18$^{\text{th}}$ International Conference on Pattern Recognition (volume II), Hong Kong, 2006, pp. 332–335.

[74] Q. Zheng, R. Chellappa, Estimation of Illuminant Direction, Albedo, and Shape from Shading, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (7) (1991) 680–702.

[75] P. L. Worthington, E. R. Hancock, Needle map recovery using robust regularizers, Image and Vision Computing 17 (8) (1999) 545–557.

[76] J.-D. Durou, H. Maître, On Convergence in the Methods of Strat and of Smith for Shape from Shading, International Journal of Computer Vision 17 (3) (1996) 273–289.

[77] A. Crouzil, X. Descombes, J.-D. Durou, A Multiresolution Approach for Shape from Shading Coupling Deterministic and Stochastic Optimization, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (11) (2003) 1416–1421.

[78] H. Saito, N. Tsunashima, Estimation of 3-D Parametric Models from Shading Image Using Genetic Algorithms, in: Proceedings of the 12$^{\text{th}}$ International Conference on Pattern Recognition (volume I), Jerusalem, Israel, 1994, pp. 668–670.

[79] A. P. Pentland, Local Shading Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (2) (1984) 170–187.

[80] C.-H. Lee, A. Rosenfeld, Improved Methods of Estimating Shape from Shading Using the Light Source Coordinate System, Artificial Intelligence 26 (2) (1985) 125–143.

[81] H. Hayakawa, S. Nishida, Y. Wada, M. Kawato, A Computational Model for Shape Estimation by Integration of Shading and Edge Information, Neural Networks 7 (8) (1994) 1193–1209.

[82] R. L. Wildey, Radarclinometry for the Venus Radar Mapper, Photogrammetric Engineering and Remote Sensing 52 (1) (1986) 41–50.

[83] A. P. Pentland, Linear Shape From Shading, International Journal of Computer Vision 4 (2) (1990) 153–162.

[84] R. Kozera, R. Klette, Finite Difference Based Algorithms for Linear Shape from Shading, Machine Graphics and Vision 6 (2) (1997) 157–201.

[85] G. Ulich, Provably Convergent Methods for the Linear and Nonlinear Shape from Shading Problem, Journal of Mathematical Imaging and Vision 9 (1) (1998) 69–82.

[86] R. Kozera, R. Klette, Evaluation of Numerical Solution Schemes for Differential Equations, in: R. Klette, H. S. Stiehl, M. A. Viergever, K. L. Vincken (Eds.), Performance Characterization in Computer Vision, Vol. 17 of Computational Imaging and Vision, Kluwer Academic Publishers, 1999, pp. 153–166.

[87] P.-S. Tsai, M. Shah, Shape from Shading Using Linear Approximation, Image and Vision Computing 12 (8) (1994) 487–498.

[88] M. Bichsel, A. P. Pentland, A Simple Algorithm for Shape from Shading, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Champaign, Illinois, USA, 1992, pp. 459–465.

[89] A. Crouzil, X. Descombes, J.-D. Durou, A Multiresolution Approach for Shape from Shading Coupling Deterministic and Stochastic Optimization, Rapport de Recherche 2003-19-R, Institut de Recherche en Informatique de Toulouse, Toulouse, France (Sep. 2003).

[90] A. P. Pentland, Shape Information From Shading: A Theory About Human Perception, in: Proceedings of the 2nd IEEE International Conference on Computer Vision, Tampa, Florida, USA, 1988, pp. 404–413.

[91] M. Bardi, M. Falcone, An Approximation Scheme for the Minimum Time Function, SIAM Journal of Control and Optimization 28 (4) (1990) 950–965.

[92] M. Bardi, M. Falcone, Discrete Approximation of the Minimal Time Function for Systems with Regular Optimal Trajectories, in: A. Bensoussan, P.-L. Lions (Eds.), Analysis and Optimization of Systems, Vol. 144 of Lecture Notes in Control and Information Science, Springer-Verlag, 1990, pp. 103–112.

[93] M. Falcone, T. Giorgi, P. Loreti, Level Sets of Viscosity Solutions: Some Applications to Fronts and Rendez-Vous Problems, SIAM Journal on Applied Mathematics 54 (5) (1994) 1335–1354.

[94] J. A. Sethian, A. Vladimirsky, Fast Methods for the Eikonal and Related Hamilton-Jacobi Equations on Unstructured Meshes, Proceedings of the National Academy of Sciences of the United States of America 97 (11) (2000) 5699–5703.

[95] E. Cristiani, M. Falcone, Fast Semi-Lagrangian Schemes for the Eikonal Equation and Applications, SIAM Journal on Numerical Analysis 45 (5) (2007) 1979–2011.

[96] Z. Wu, L. Li, A Line-Integration Based Method for Depth Recovery from Surface Normals, Computer Vision, Graphics, and Image Processing 43 (1) (1988) 53–66.

[97] P. Daniel, J.-D. Durou, Creation of Real Images which are Valid for the Assumptions Made in Shape From Shading, in: Proceedings of the 10th International Conference on Image Analysis and Processing, Venice, Italy, 1999, pp. 418–423.

[98] R. Kozera, A Note on Existence and Uniqueness in Shape from Shading, in: Proceedings of the 4th IEEE International Conference on Computer Vision, Berlin, Germany, 1993, pp. 507–511.

[99] B. K. P. Horn, R. Szeliski, A. L. Yuille, Impossible Shaded Images, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (2) (1993) 166–170.

[100] U. M. Ascher, P. M. Carter, A Multigrid Method for Shape from Shading, SIAM Journal on Numerical Analysis 30 (1) (1993) 102–115.