Fast Marching Method for Generic Shape from Shading

Emmanuel Prados and Stefano Soatto

Dept. of Computer Science, University of California, Los Angeles, CA 90095, USA

Abstract. We develop a fast numerical method to approximate the solutions of a wide class of equations associated to the Shape From Shading problem. Our method, which is based on the control theory and the interfaces propagation, is an extension of the "Fast Marching Method" (FMM) [30,25]. In particular our method extends the FMM to some equations for which the solution is not systematically decreasing along the optimal trajectories. We apply with success our one-pass method to the Shape From Shading equations which are involved by the most relevant and recent modelings [22,21] and which cannot be handled by the most recent extensions of the FMM [26,8].

1 Introduction

The *Shape From Shading* (SFS) problem is to compute the three-dimensional shape of a surface from a single black and white image of that surface. The field was pioneered by Horn who was the first to pose the problem as that of finding the solution of a nonlinear first-order partial differential equation (PDE) called the brightness equation [11]. The first explicit PDE considered in this field is the Eikonal equation

$$|\nabla u| = r(x), \quad \forall x \in \Omega \tag{1}$$

(modeling based on a single far frontal light source and orthographic camera). Ω is an open subset of $\mathbb{R}^2$ which represents the image domain, e.g., the rectangle $]0, X[\times]0, Y[$. $r : \Omega \to \mathbb{R}$ is a non-negative function directly related to the brightness image. From the work of Horn [11], a number of explicit PDEs corresponding to different modelings have been developed and studied [15,24,27,20,6]. By introducing the "generic" SFS equation $H_g(x, \nabla u(x)) = 0, \forall x \in \Omega$ (H_g being described below), Prados and Faugeras [22,21] have recently unified a number of these explicit equations and thus their associated models. The associated "generic" SFS Hamiltonian is defined by

$$H_g(x, p) = \kappa_x \sqrt{|D_x \ R_x p + \mathbf{v}_x|^2 + K_x^2} + \mathbf{w}_x \cdot p + c_x, \tag{2}$$

where $\kappa_x, K_x, \mu_x, \nu_x, c_x, \mathbf{v}_x, \mathbf{w}_x$ depend on the chosen modeling, see [22,17]. For example, the classical Rouy/Tourin Hamiltonian [24]

$$H_{R/T}(x, p) = I(x)\sqrt{1 + |\nabla u|^2} + \mathbf{l} \cdot \nabla u - \gamma \tag{3}$$

(orthographic camera + single far light source) is a particular case of the generic SFS Hamiltonian H_g; $\mathbf{L} = (\mathbf{l}, \gamma)$ represents the direction of the light source; $I(x)$ is the

brightness of the image at pixel x. Here, let us remark that $H_{R/T}(x,0) = I(x) - \gamma$ can be strictly negative.

In addition to the modeling, Prados and Faugeras [17,21] have also unified a number of theoretical results and of SFS numerical methods based on PDEs. Nevertheless they just consider iterative methods. Also, the iterative methods suffer of some optimality weaknesses since they use alternating raster scans similarly as the ones proposed [9,29]. In this paper, we get rid of these weaknesses by proposing a single pass method based on front propagation and Fast Marching techniques.

The single pass methods related to front propagation like the level set method [16] and Fast Marching Methods [25] have already been applied to the SFS problem [12,25,13,31,28] (see [5,25] and references therein for other applications). In particular, Sethian [25] was the first who applies the "Fast Marching Method" for solving the SFS problem. This first work deals only with the simplest version of the SFS problem associated to the Eikonal equation (1) (orthographic camera; single, far and front light source). Recently, Kimmel and Sethian [13] have proposed an adaptation of this initial algorithm in order to deal with far oblique light source. This upgraded method seems very delicate, requires a change of variable and does not seems adaptable to more general modelings. Roughly speaking, most of the authors [13,28,6] use various techniques (e.g. changes of variables, introductions of new unknowns) in order to get back to the Eikonal equation and then to use the initial tools developed by Sethian. As opposed to [28], Yuen et al. [31] propose a real adaptation of the FMMs to the perspective SFS problem. Nevertheless, this work is reduced to frontal light source, also in that case the subjacent cost function is non negative. Contrary to all the previous work, the fast method we propose in this paper allows to solve in one-pass the main PDEs associated to the most recent, realistic and relevant modelings of the SFS literature. In particular, it allows to solve all the SFS equations collected in [22,21] (many of which have cost functions of arbitrary sign). It is completely generic and contrary for example to [13], we do not need any change of variables.

More generally, the most recent work on "Fast Marching Methods" only allows to solve equations of type

$$H(x, \nabla u(x)) = 0 \quad \forall x \in \Omega, \tag{4}$$

such that $H(x,0) = 0$ (with H convex with respect to ∇u). Here, we relax this key assumption which is strongly related to the *causality principle* (see Section 3).

2   Control Formulation of the Problem

In this section we reformulate the problem using tools from control theory. The reader unfamiliar with these tools can refer to the comprehensive book [1]. First, we use the Legendre transform [14] to rewrite equation (4) as: $\forall\, x \in \Omega$,

$$\sup_{a \in A}\{-f(x,a) \cdot \nabla u(x) - l(x,a)\} = 0. \tag{5}$$

The functions l and f are respectively called the *cost function* and the *dynamics*; $a \in A$ is an admissible value of the *control*. For example, we can rewrite the generic SFS Hamiltonian H_g as such a supremum with

$$f(x, a) = - [\, \kappa_x R_x^T D_x R_x.a + \mathbf{w}_x \,],$$
$$l(x, a) = - [\, K_x \kappa_x \sqrt{1 - |a|^2} + \kappa_x (R_x^T \mathbf{v}_x) \cdot a + c_x \,]$$

with A the closed unit ball of $\mathbb{R}^2$; R_x^T being the matrix transpose of R_x; see [17,22] for all details. This kind of equation must be complemented by boundary conditions in order to be well posed. We therefore add the following constraints:

$$u(x) = \varphi(x) \qquad \forall x \in \mathcal{T}, \tag{6}$$
$$u(x) = +\infty \qquad \forall x \in \partial\Omega \setminus \mathcal{T}, \tag{7}$$

where the *target* $\mathcal{T}$ is a nonempty closed subset of $\overline{\Omega}$ and $\varphi : \mathcal{T} \to \mathbb{R}$ defines a boundary condition of Dirichlet type. Also equations (5) and (4) are now considered on $\Omega \setminus \mathcal{T}$ instead of Ω. Let V be the value function

$$V(x) = \inf_{\alpha \in \mathcal{A}} \left\{ \int_0^{\tau_x(\alpha)} l(y_x(\tau, \alpha), \alpha(\tau))d\tau + \varphi(y_x(\tau_x(\alpha), \alpha)) \right\}. \tag{8}$$

$\mathcal{A}$ is the set of the *admissible controls* (a set of the measurable functions of $t \in [0, +\infty[$ with value in A, the latter being a closed bounded subset of $\mathbb{R}^M$). $y_x(., \alpha)$ is a *trajectory* controlled by α starting from x; this is the solution of the dynamical system $y'(\tau) = f(y(\tau), \alpha(\tau))$, $\tau > 0$, $y(0) = x$; $\tau_x(\alpha)$ is the smallest time τ such that $y_x(\tau)$ reaches[1] $\mathcal{T} \cup \partial\Omega$. It is well known [14,1] that V is the unique viscosity solution[2] of equation (5)-(6)-(7) and that V verifies the *dynamical programming principle* [14,1].

Finally, we denote α_x^*, τ_x^* and y_x^* the optimal control, the optimal time and the *optimal trajectory* associated to (8) (i.e. for which the $\inf$ of (8) is reached).

For example, in the particular case of the Eikonal equation (1), the optimal trajectories correspond to the gradient lines. Nevertheless, as shown in [26], this is not true for any equation ($\nabla u(x) \neq y_x^{*\,\prime}(0) = f(x, \alpha_x^*(0))$).

3   Approximation Scheme and Causality

For simplicity, in this paper we deal with a regular mesh. For an extension to the irregular meshes we refer the reader to [17,23]. The reader unfamiliar with the notion of approximation schemes can refer to [2]. Let us just recall that, following [2], an approximation scheme is a functional equation of the form

$$S(\rho, x, u(x), u) = 0 \quad \forall x \in \overline{\Omega},$$

which "approximates" the considered PDE. S is defined on $\mathcal{M} \times \overline{\Omega} \times \mathbb{R} \times B(\overline{\Omega})$ into $\mathbb{R}$, $\mathcal{M} = (\mathbb{R}^+)^N$ and $\rho = (h_1, ..., h_N) \in \mathcal{M}$ defines the size of the mesh that is used in the

---

[1] For notations simplicity, we extend φ on $\mathcal{T} \cup \partial\Omega$, and we define $\varphi(x) = +\infty, \forall x \in \partial\Omega \setminus \mathcal{T}$.

[2] More accurately, we must consider the notion of "Singular Discontinuous Viscosity Solutions" (SDVS) of (5)-(6)-(7) instead of the classical notion of viscosity solutions [17].

corresponding numerical algorithms. $B(D)$ is the space of bounded functions defined on a set D. In order to obtain a consistent scheme, the function S is obtained by approximating ∇u by finite differences and then by substituting ∇u by its approximation in the initial equation. The main property allowing to ensure the convergence of the computed numerical solution toward the viscosity solution [7,14,1,17] is the *monotonicity* of the scheme (i.e. the function $u \mapsto S(\rho, x, t, u)$ is nonincreasing) [2].

In most of the FMMs the gradient ∇u is discretized by

$$\nabla u(x) \simeq \frac{t - u(x + s_i h_i e_i)}{-s_i h_i}$$

Above, t corresponds to $u(x)$; we replace $u(x)$ by t in order to emphasize that this is the *update value*. $(e_1, .., e_n)$ is the canonical basis; $\forall i, \ s_i \in \{\pm 1\}$. With the exception of the recent work [26], the simplex $\{x, x + s_1 h_1 e_1, ..., x + s_n h_n e_n\}$ (in practice the sign vector $(s_1, .., s_n)$) is chosen in such a way that "it contains $-\nabla u$", i.e. such that

$$\forall i = 1..n, \quad s_i \text{ is the opposite of the sign of } [\nabla u]_i. \tag{9}$$

In the Eikonal case, the control formulation of equation (1) is $\sup_{a \in \overline{B}(0,1)} \{a \cdot \nabla u(x) - r(x)\} = 0$ and the optimal control is always $a_x = \frac{\nabla u(x)}{|\nabla u(x)|}$. Thus the scheme

$$S_{eiko}(\rho, x, t, u) = \left| \frac{t - u(x + s_i h_i e_i)}{-s_i h_i} \right| - r(x)$$

$$= \sup_{a \in \overline{B}(0,1)} \left\{ a \cdot \left(\frac{t - u(x + s_i h_i e_i)}{-s_i h_i} \right) - r(x) \right\}, \tag{10}$$

where s_i defined by (9), is clearly monotonic. Nevertheless, this is generally false for any equation of the form (4). In effect, in the general case, the scheme

$$S_c(\rho, x, t, u) = H \left(x, \left(\frac{t - u(x + s_i h_i e_i)}{-s_i h_i} \right) \right)$$

$$= \sup_{a \in A} \left\{ -f(x, a) \cdot \left(\frac{t - u(x + s_i h_i e_i)}{-s_i h_i} \right) - l(x, a) \right\} \tag{11}$$

with s_i defined by (9), is not monotonic anymore as soon as there exists $i \in [1..n]$ s.t. $sign[f(x, a_x)]_i \neq s_i \ (= -sign[\nabla u]_i)$, where a_x is the optimal a of (11) [t being the root of $S_c(\rho, x, t, u) = 0$]. For obtaining a monotonic scheme, we must change the choice of the simplex (i.e. the choice of the s_i). In fact, we must choose the simplex with respect to the dynamics of the optimal control (and not to the gradient). If we define $s_i(x, a) = sign f_i(x, a)$, the scheme[3]

$$S(\rho, x, t, u) = \sup_{a \in A} \left\{ -f(x, a) \cdot \left(\frac{t - u(x + s_i(x, a) h_i e_i)}{-s_i(x, a) h_i} \right) - l(x, a) \right\} \tag{12}$$

is naturally monotonic, and the "good" one. In other words, this scheme is obtained from (4) by using only the backward and forward approximations of the partial derivatives and by choosing the good one *according to the dynamic of the optimal control.*

---

[3] Scheme already suggested by Prados and Faugeras [20,17].

Because of space limitations we omit the details of the implementation of this scheme and refer the reader to [23,17]. Let us just note here that the approximation scheme proposed by Sethian and Vladimirsky is different from the one presented here: [26] is based on the use of more simplexes than those contained in the immediate neighbors of the considered site, unlike the scheme (12). Moreover, our scheme is consistent (see [17]).

Our scheme suggests then that the reconstruction must follow (track) the optimal trajectories, if we want to be able to recover the viscosity solution with a one-pass algorithm. Sethian and Vladimirsky [26] demonstrate this fundamental aspect intuitively and experimentally. As opposed to the first work [30,25] in which the causality was directly related to the gradient lines, in [26] the causality property is based on optimal trajectories. Nevertheless, Sethian and Vladimirsky's causality is based on the fact that the solution is strictly decreasing along the optimal trajectories. This property is verified for the equations considered in [26]

$$\sup_{a \in \mathcal{A}} -\mathsf{f}(x,a)a \cdot \nabla u - 1 = 0, \quad \forall x \in \Omega, \tag{13}$$

since for these equations the cost function is $l(x,a) = 1 > 0$. Nevertheless, for any equation of type (4) such that the cost function l takes some negative values, this monotonicity property does not hold: the solution can arbitrary and alternatively increase and decrease along the optimal trajectories. Also, in the general case the causality property used by Sethian and Vladimirsky [26] does not apply.

Finally, let us emphasize that as it was shown in [17], most of the Shape From Shading equations have generally cost functions of arbitrary sign. This is the case for example for the classical Rouy/Tourin Hamiltonian $H_{R/T}$ (where, $l_{R/T}(x,a) = I(x)\sqrt{1 + |a|^2} - \gamma$) and for the perspective SFS Hamiltonian $H_{P/F}^{pers}$ [20] which fit in the class of Hamiltonians given by equation (4).

In the following, we slightly reinterpret the FMMs; we generalize and weaken the principle of causality. Later on we propose a new practical causality property which extends the one used by [26] to any equation of type (4).

4   Update Ordering for "Single Pass" Method

4.1   Related Work and Basic Ideas

For the moment, let us assume that *we know the optimal trajectories* y_x^*. We can then recover directly the solution of equation (5)-(6)-(7) by going back up these curves:

$$V(x) = \int_0^{\tau_x(\alpha_x^*)} l(y_x^*(\tau), \alpha_x^*(\tau))d\tau \ + \varphi(y_x^*(\tau_x(\alpha_x^*))). \tag{14}$$

Then from the numerical point of view, we can reconstruct the solution by a direct integration along the optimal trajectories. This idea corresponds roughly to the method of *characteristic strips* introduced in the Shape from Shading literature by Horn [10] for solving the Eikonal equation. In this work, the knowledge of the optimal trajectories was implicitly replaced by Neumann conditions. For removing some "resticking"

problems and for improving the stability of the method, we can go back up all the optimal trajectories simultaneously and not separately. More precisely, the idea is then the following. We assume that the solution is known inside a closed curve $\mathcal{C}_t$ ($t \geq 0$) which propagates along the optimal trajectories. Basically, the curves $\mathcal{C}_t$ must verify: for all x in Ω, the optimal trajectory y_x^* always intersects the curve $\mathcal{C}_t$, but only once for each $t \geq 0$. Moreover, if $t_1 < t_2$ then $\Omega_{t_1} \subset \Omega_{t_2}$ (where Ω_t is the open subset such that $\partial \Omega_t = \mathcal{C}_t$). This idea corresponds exactly to the one introduced by Bruckstein [3] for solving the Eikonal SFS problem. To alleviate some instability and topological problems [16], the best way is then to use the "level set" method introduced by Osher and Sethian [16] or Fast Marching techniques [25].

4.2   Curve Propagations and Associated Costs

Now, we assume that *we do not know the optimal trajectories*. This is usually the case in most of the applications, in particular in Shape From Shading. In this case, we must reconstruct simultaneously the solution u and the curves $\mathcal{C}_t$. In this way, we are sure that the updates of the values of the solution propagate in the same way as the optimal trajectories. The idea is then the following: let us assume that we already know $\mathcal{C}_t$ and the values of the solution on Ω_t. Now, if we want to compute the values of the solution u on Ω_{t+dt}, we need to compute explicitly these values, but also we need to compute the new curve $\mathcal{C}_{t+dt}$. In order to *practically handle the curves $\mathcal{C}_t$, we define them as the level sets* (i.e. the isocontours) of a *"cost"* $\mathsf{C}(x)$ defined on Ω into $\mathbb{R}$. For example, in the basic "Fast Marching Method" for the Eikonal equation [30,25], the chosen cost C is equal to u. Also, the curves $\mathcal{C}_t$ correspond to the isocontours of the solutions. In other respects, in a sense the function $\tilde{z}$ introduced in [13] could be consider as such a cost, (nevertheless let us note that $\tilde{z}$ does not depend on x but on $\tilde{x}$ [which himself depends on x and $u(x)$]; see [13]).

In order to reach our goal, let us remind an important (but too many times neglected) difficulty encountered when we deal with equations of type (5)-(6)-(7): the optimal trajectories y_x^* and the solution V significantly depend on the set Ω. Therefore, in the sequel, when it is relevant, we indicate the associated set by specifying it by an index as follow: $\alpha_x^* = \alpha_{x(\Omega)}^*$, $y_x^* = y_{x(\Omega)}^*$, $\tau_x^* = \tau_{x(\Omega)}^*$, $V = V_\Omega$.

Finally, to be completely rigorous, if the optimal trajectories depend on the set on which we work, then the curves $\mathcal{C}_t$ and their associated cost C must also depend on it. Also, we will use the same type of notations as above for the cost C (nevertheless, in the sequel we use the notations $\mathcal{C}_t$ and Ω_t associated with the iso-curves only in reference to Ω and equation (5)-(6)-(7)).

The above remark is mainly motivated by the following idea. First, let us remind that we do not know the cost C_Ω on $\Omega \setminus \mathcal{C}_t$ (since this is equivalent to knowing all the propagating closed curves). Moreover, it seems reasonable to assume that we are not able to directly compute C_Ω even on a very close neighborhood $\mathcal{N}$ of Ω_t ($\mathcal{N} \subset \Omega$). In effect, any point $x \in \mathcal{N} \setminus \overline{\Omega_t}$ (even if it is extremely close to $\mathcal{C}_t$) can have an optimal trajectory $y_{x(\Omega)}^*$ which goes away very far from $\mathcal{C}_t$ before coming back to it. Nevertheless it is reasonable to assume that we are able to compute (an approximation of) $\mathsf{C}_\mathcal{N}$ on a close neighborhood $\mathcal{N}$ of Ω_t (let us remind that $\mathsf{C}_\mathcal{N}$ is the cost associated to the equation $(5_\mathcal{N})$-$(6_\mathcal{N})$-$(7_\mathcal{N})$).

For example in [26], the neighborhood $\mathcal{N}$ (considered above) corresponds with the set of the "*Considered*" points. In the classical methods, the cost C corresponds with the solution of the considered equation. Also, the values $V(x)$ computed for all the "*Considered*" points x are (numerical approximation of) the cost $\mathsf{C}_{\mathcal{N}}(x)$.

Later on, from the knowledge of the cost $\mathsf{C}_{\mathcal{N}}$ on the neighborhood $\mathcal{N}$ of Ω_t, we would like to find $\mathcal{C}_{t+dt}$ such that $\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$. Also, if the costs verify the hypotheses

(H1) $\mathsf{C}_{\mathcal{W}}(x)$ is decreasing along the optimal trajectories $y_x^*(\mathcal{W})$, i.e. : for all x and $\tau > 0$ (small enough) $\mathsf{C}_{\mathcal{W}}(y_x^*(\mathcal{W})(\tau)) \leq \mathsf{C}_{\mathcal{W}}(x)$,

(H2) it is decreasing with respect to $\mathcal{W}$, i.e.:
if $\Omega_1 \subset \Omega_2$, then $\mathsf{C}_{\Omega_1}(x) \geq \mathsf{C}_{\Omega_2}(x)$,

(H3) let $x_1 \in \Omega_1 \subset \Omega_2$. If $y_{x_1}^*(\Omega_2)(.)$ stays in Ω_1 then $\mathsf{C}_{\Omega_2}(x_1) = \mathsf{C}_{\Omega_1}(x_1)$,

then we have the following

Proposition 1. *Let us assume that the hypotheses* (H1)*,* (H2) *and* (H3) *hold. Let* $\mathcal{C}_{t+dt}$ *be such that* $\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$*. Therefore for all* x *in* Ω_{t+dt},

$$\mathsf{C}_{\Omega}(x) = \mathsf{C}_{\mathcal{N}}(x).$$

In other words, $\mathsf{C}_{\mathcal{N}}$ coincide with C_{Ω} on Ω_{t+dt}.

Proof. Let x in Ω_{t+dt}. By the hypothesis (H1), the optimal trajectory $y_x^*(\Omega)$ stays in Ω_{t+dt}. We then apply the hypothesis (H3) with $\Omega_1 = \Omega_{t+dt}$ and $\Omega_2 = \Omega$. We have therefore $\mathsf{C}_{\Omega}(x) = \mathsf{C}_{\Omega_{t+dt}}(x)$. Since we have $\mathcal{C}_{t+dt} \subset \mathcal{N} \subset \Omega$, then by the hypothesis (H2), $\mathsf{C}_{\Omega}(x) = \mathsf{C}_{\Omega_{t+dt}}(x) = \mathsf{C}_{\mathcal{N}}(x)$. □

Now, let us consider the problem from the discret point of view. We assume that the set $\overline{\Omega}$ is "covered" by a grid of points. We divide the set of the grid points into three classes (as for the classical "Fast Marching Methods" [26]): *Accepted* **A**, *Considered* **C**, *Far* **F**. The *Accepted* points are the ones in Ω_t (i.e. inside $\mathcal{C}_t$); we therefore suppose that we already know the values of the solution for all the grid points in **A**. The set of *Considered* points **C** is the set of the adjacent points to the *Accepted* points. The union **C** $\cup$ **A** is the discrete version of the neighborhood $\mathcal{N}$ of $\overline{\Omega}_t$. The set of the *Far* points corresponds to the other points of the grid.

Let us remind that we know the values of the solution of (5)-(6)-(7) on Ω_t and that we want to compute the ones on Ω_{t+dt}. Also, this requires the preliminary computation of the new curve $\mathcal{C}_{t+dt}$. From the practical and discrete point of view, we want to extend Ω_t to Ω_{t+dt} in such a way that $\Omega_{t+dt} \setminus \Omega_t$ contains a single point of the grid. This is therefore equivalent to finding the point x_0 of **C** which has the lowest cost C_{Ω} ($\mathsf{C}_{\Omega}(x_0) = \min_{x \in \mathbf{C}} \mathsf{C}_{\Omega}(x)$) and then we transfer it to the set of *Accepted* points **A** (which is then enlarged). Also, if the costs C verify the hypotheses (H1), (H2) and (H3), then the following proposition allows to find this point x_0 (assuming that we know the costs $\mathsf{C}_{\mathcal{N}}(x)$ for all x in $\mathcal{N}$).

Proposition 2. *Let us assume that the costs* C *verify* (H1)*,* (H2) *and* (H3)*. The point* $x_0 \in \mathbf{C}$ *which has the smallest cost* $\mathsf{C}_{\Omega}(x_0)$ *is the point* x *of* **C** *which has the smallest cost* $\mathsf{C}_{\mathcal{N}}(x)$*; Also,* $\mathsf{C}_{\Omega}(x_0) = \mathsf{C}_{\mathcal{N}}(x_0)$.

Proof. Let x_0 be the point in **C** which has the less cost $\mathsf{C}_{\Omega}(x_0)$. Let $\mathcal{C}_{t+dt}$ be the level set of the cost C_{Ω} associated to the value $\mathsf{C}_{\Omega}(x_0)$. Since $\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$, By Proposition 1 we have $\mathsf{C}_{\Omega}(x_0) = \mathsf{C}_{\mathcal{N}}(x_0)$.

Let x be any point in $\mathbf{C}$. By hypothesis (H2), we have $\mathsf{C}_{\mathcal{N}}(x) \geq \mathsf{C}_{\Omega}(x)$. By definition of x_0, we have $\mathsf{C}_{\Omega}(x) \geq \mathsf{C}_{\Omega}(x_0)$. Therefore, $\mathsf{C}_{\mathcal{N}}(x) \geq \mathsf{C}_{\mathcal{N}}(x_0)$.   □

Thus if (H1), (H2) and (H3) hold, then from the numerical point of view, for extending Ω_t to Ω_{t+dt}, we have just to compute the costs $\mathsf{C}_{\mathcal{N}}(x)$ for all x in the $\mathbf{C}$ and to search the point x_0 in $\mathbf{C}$ which has the smallest cost $\mathsf{C}_{\mathcal{N}}$.

4.3 Proposed Costs

At this stage, let us remind that the method we propose here needs the explicit computations of some approximations of the cost C and of the solution V. Since our final objective is only to compute some approximations of V, the computations of the approximations $\mathsf{C}_{\mathcal{N}}$ can appear useless and expensive, and so decrease the interest of the method. It seems therefore reasonable and quite relevant to search and use some costs $\mathsf{C}_{\mathcal{W}}$ which are directly related to the values of the solution $V_{\mathcal{W}}$, as in the Eikonal case and the classical FMMs.

Example 1. In [26], Sethian and Vladimirsky consider equations of type (13), i.e. some equations (5) with $f(x,a) = \mathsf{f}(x,a)a$ and $l(x,a) = 1$. So the viscosity solution V of (13) (complemented by some adequate boundary conditions) is $V(x) = \tau_x^* + \varphi(y_x^*(\tau_x^*))$. Thus, if the boundary condition φ imposed on $\mathcal{T}$ is a constant function ($\varphi(x) = c \in \mathbb{R}$ for all x in $\mathcal{T}$) then $V(x) = \tau_x^* + c$. In this case, it is therefore judicious and reasonable to choose $\mathsf{C}_{\mathcal{W}}(x) = \tau_{x(\mathcal{W})}^*$. $V_{\mathcal{W}}$ and the $\mathsf{C}_{\mathcal{W}}$ are thus directly related; Consequently the curves $\mathcal{C}_t$ correspond with the isocontours of the solution of (13). Also, the costs $\mathsf{C}_{\mathcal{W}}$ verify trivially the hypotheses (H1), (H2) and (H3).

In the particular case considered by Sethian and Vladimirsky [26] (Example 1), the cost directly related to the values of the solution is trivial. Nevertheless in the general case, in particular for HJB equations with a cost fonction $l(x,a)$ depending on a or with an arbitrary sign, we need to work a little more.

Let us remind that generally the isocontours of the solution cannot play the role of the curves $\mathcal{C}_t$ since the values of the solution u are *not decreasing* along the optimal trajectories (i.e. for all x, the function $t \mapsto u(y_x^*(t))$ is not decreasing). A second basic idea could be to choose systematically the cost $\mathsf{C}(x) = \tau_x^*$ since this cost verifies naturally the hypotheses (H1), (H2) and (H3). Nevertheless, computing this cost can be very difficult, and generally it is really not related with the solution u of the considered equation. In order to define an adequate cost in the general case, let us introduce the following definitions.

Let ψ be a subsolution of (4), i.e.

$$H(x, \psi(x)) \leq 0, \quad \forall x \in \Omega.$$

In [17], Prados and Faugeras describe the subsolutions of the main classical SFS equations. For example, for the classical Rouy/Tourin Hamiltonian $H_{R/T}$ (equation (3)) $\psi(x) = -\frac{1}{\gamma}\mathbf{1} \cdot x$ is subsolution.

Let $\mathcal{Z}(x)$ be the multivalued map [4] on $\overline{\Omega}$ defined as:

$$\mathcal{Z}(x) = \{p \in \mathbb{R}^N : H(x, p) \leq 0\}.$$

Let $\delta : \overline{\Omega} \times \mathbb{R}^N \to \mathbb{R}$ be the support function [4] of the set $\tilde{\mathcal{Z}}(x) = \mathcal{Z}(x) - \nabla\psi(x)$, i.e.:

$$\delta(x,p) = \max\{pq : q \in \tilde{\mathcal{Z}}(x)\}.$$

Finally, for all x_1, x_2 in $\overline{\mathcal{W}}$, let us denote

$$L_{\mathcal{W}}(x_1, x_2) = \inf_{\xi \in \varXi_{x_1,x_2}} \left\{ \int_0^1 \delta(\xi(t), -\dot{\xi}(t))dt \right\} \tag{15}$$

where $\varXi_{x_1,x_2}$ is the set of $\xi(t) \in W^{1,\infty}([0,1], \overline{\mathcal{W}})$ s.t. $\xi(0) = x_1$ and $\xi(1) = x_2$. A complete study and description on these notions can be found in [17,4,19,18]. In particular, in [17], it is proved that

$$V(x) = \psi(x) + \min\{L_{\Omega\backslash\mathcal{T}}(x,z) + \varphi(z) - \psi(z) \mid z \in \mathcal{T}\}. \tag{16}$$

is the unique Singular Discontinuous Viscosity Solution (SDVS) of (5)-(6)-(7), see [17]. Let us emphasize here that $\delta(.,.)$ and so $L_{\mathcal{W}}$ are non-negative. Also $L_{\mathcal{W}}(x,z)$ is a semi-distance.

These results and notations in hand, we can now define an appropriate cost for our "generic" HJB equation:

$$\mathsf{C}_{\mathcal{W}}(x) = L_{\mathcal{W}}(x, z_x^*),$$

where z_x^* is the optimal z of (16). This cost is trivially decreasing along the optimal trajectories and it verifies naturally the assumptions (H1)-(H2)-(H3). Finally, if the boundary condition φ verifies "$\varphi - \psi$ is constant on the target $\mathcal{T}$" then the viscosity solution $V_{\mathcal{W}}$ is directly related to $\mathsf{C}_{\mathcal{W}}$: it coincides with $\mathsf{C}_{\mathcal{W}} + \psi$ (up to the addition of a constant). Also, the isocontour of the cost $\mathsf{C}(x)$, and so $\mathcal{C}_t$, corresponds with the isocontour of the $V - \psi$.

5   Proposed Algorithm

The grid points are divided into the three classes (as in the basic "Fast Marching Method"): *Accepted* **A**, *Considered* **C**, *Far* **F** (see above). ψ is a subsolution of $H(x, \nabla u) = 0$ as defined in Section 4.3. U defined on the whole grid is the approximation of the solution of equation (5)-(6)-(7). $\tilde{U}$ defined on the *Considered* points is the approximation of the solution of equation $(5_{\mathcal{N}})$-$(6_{\mathcal{N}})$-$(7_{\mathcal{N}})$, where $\mathcal{N}$ is the neighborhood associated to **A** $\cup$ **C**.
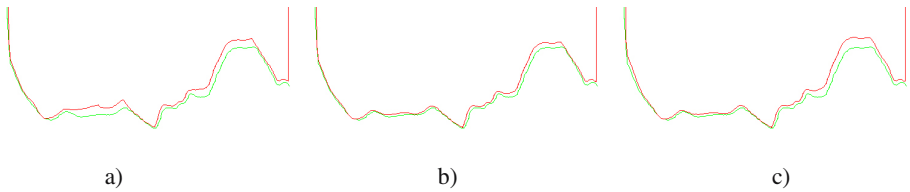
Algorithm 1.
1. *Start with all the grid points in* Far.
2. *Move the grid points on the boundary $\partial\Omega$ and on the target $\mathcal{T}$ to the* Accepted. *For all these points $x \in \mathcal{T}$, put $U(x) = \varphi(x)$, for the other points put $U(x) = +\infty$.*
3. *Move all the grid points $x \in \Omega$ adjacent to the* Accepted *points into* Considered *and for each of these points, evaluate $\tilde{U}(x)$ by using the update scheme (12).*
4. *Find the grid point $x_0 \in$* **C** *with the smallest value $\tilde{U}(x) - \psi(x)$. Move x_0 from* Considered *to* Accepted; *Put $U(x_0) = \tilde{U}(x_0)$.*
5. *Move from* Far *into* Considered, *all the* Far *points which are adjacent to x_0.*
6. *Re-evaluate $\tilde{U}$ by using the scheme (12), for all the* Considered *points adjacent x_0.*
7. *If the set* **C** *is not empty, return to item 4.*

The complete proof of the convergence of the computed numerical solution toward the viscosity solution is postponed to a following paper.

a)   b)   c)   d)
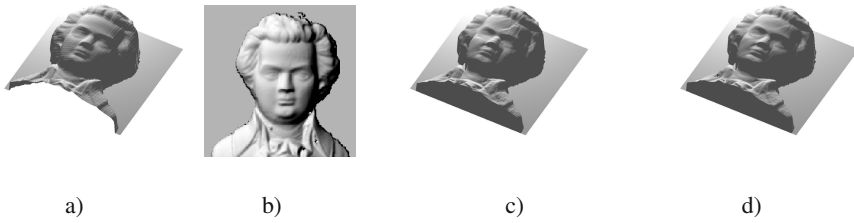
Fig. 1. a) original surface (groundtruth); b) image synthesized from a); c) surface reconstructed by using the classical causality; d) surface reconstructed by our algorithm

a)   b)   c)

Fig. 2. a) vertical section of Fig.1-c); b) vertical section of Fig.1-d); c) vertical section of the surface reconstructed by the iterative algorithm after three complete iterations

6   Numerical Experiments

We have implemented our method for the generic SFS Hamiltonian H_g (2). As a consequence our algorithm applies to any modeling considered in [21,22].

Because of space and since the contributions of this paper concern mainly some numerical improvements, here we illustrate our results only with the classical Rouy/Tourin Hamiltonian $H_{R/T}$ (3). We also only deal with synthetic images; more exactly with the classical example of the Mozarts face [32]. Moreover as the theory suggests [19], in our experiments we assume that we know and we use the values of the solutions at all its local minima.

Mainly, the differences between our new method and the previous FMMs are two fold. 1) We propose a new *causality principle* (the updating order is related to the level sets of $u - \psi$ instead of the ones of u; u being the solution of the considered equation). 2) We propose also to use an approximation scheme (already described in [17]) which is different from the previous ones proposed in the FMM literature. Here, we focus on the improvement due to the change of the causality. Figure 1 shows: a) the original surface; b) the image synthesized from (a) with a single far light source ($\mathbf{l} = (-0.3, -0.3)$; see page 320) and by using an orthographic projection; c) the surface reconstructed by our algorithm [our scheme + our causality]; d) the surface reconstructed by using the classical causality [our scheme + replacing ψ by 0]. Let us note that due to the black shadows, we are not able to completely recover the original surface. Nevertheless, the improvement involves by the change of the causality is quite visible. To better show the differences between the surfaces 1-c) and 1-d), we display vertical sections of them in

Figures 2-a) and 2-b). In Figure 2, the red curves represent the sections of the computed approximations and the green curve is the section of the groundtruth. In other respects, Figure 2-c) displays a vertical section of the surface reconstructed by the corresponding iterative algorithm [22,17] after three complete alternating raster scans similar to those used in [9,29]. In this example, the number of pixels considered is around 20200. With our FMM method, the computation of the solution requires around 40500 updates. With the iterative version, around 76000 updates are required for computing an approximation of the solution with an error of the same order of magnitude. About computational time, our FMM method returns this result after 6 seconds (computer: Intel Celeron 1.5GHz; Let us note that we do not have tried to optimize our code yet). The iterative version requires approximatively the same computational time (7 seconds). Many more numerical results and qualitative and quantitative comparison tests can be found in our forthcoming associated papers [23]. Also, we postpone to [23] the comparisons of the results obtained with Prados and Faugeras's scheme with those obtained with the other ones.

7   Conclusion

In this article, we revisit the classical "Fast Marching Methods" [30,25,26,8] and we extend them to a wide class of HJ equations. In particular, our method can deal with HJB equations with arbitrary signs cost functions when the previous methods just deal with positive cost functions. Also, in the cases where the solution does not decrease along the optimal trajectories, we correct the causality property by using a subsolution. Finally, our method is generic and it applies indifferently to all the SFS equations obtained by the most recent and relevant modelings of this problem [17].

References

1. M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Birkhauser, 1997.
2. G. Barles and P.E. Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, 4:271–283, 1991.
3. A. M. Bruckstein. On shape fron shading. *Computer. Vision Graphics Image Process*, 44:139–154, 1988.
4. F. Camilli and A. Siconolfi. Nonconvex degenerate Hamilton-Jacobi equations. *Mathematische Zeitschrift*, 242:1–21, 2002.
5. L. Cohen. Minimal paths and fast marching methods for image analysis. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Mathematical Models in Computer Vision: The Handbook*, chapter 7. Springer, 2005.
6. F. Courteille, A. Crouzil, J-D. Durou, and P. Gurdjos. Towards shape from shading under realistic photographic conditions. In *Proceedings of ICPR'04*, 2004.
7. M.G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton–Jacobi equations. *Trans. AMS*, 277:1–43, 1983.
8. E. Cristiani and M. Falcone. Fast Marching Semi-Lagrangian methods for the Eikonal equation. In *SIMAI 2004*, pages 20–24, Venezia, Italy, September 2004.
9. P.-E. Danielsson. Euclidean Distance Mapping. *Computer Graphics and Image Processing*, 14(3):227–248, 1980.

10. B.K. Horn. *Robot Vision*. MIT Press, 1986.

11. B.K.P. Horn. Obtaining shape from shading information. In P.H. Winston, editor, *The Psychology of Computer Vision*. McGraw-Hill, New York, 1975.

12. R. Kimmel and A.M. Bruckstein. Tracking level sets by level sets : A method for solving the shape from shading problem. *CVIU*, 62(2):47–58, July 1995.

13. R. Kimmel and J.A. Sethian. Optimal algorithm for shape from shading and path planning. *Journal of Mathematical Imaging and Vision*, 14(2):237–244, May 2001.

14. P.-L. Lions. *Generalized Solutions of Hamilton–Jacobi Equations*. Number 69 in Research Notes in Mathematics. Pitman Advanced Publishing Program, 1982.

15. J. Oliensis and P. Dupuis. Direct method for reconstructing shape from shading. *SPIE*, 1570:116–128, 1991.

16. S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: algorithms based on the Hamilton–Jacobi formulation. *Journal of Comput. Physics*, 79:12–49, 1988.

17. E. Prados. *Application of the theory of the viscosity solutions to the Shape From Shading problem*. PhD thesis, Univ. of Nice-Sophia Antipolis, 2004.

18. E. Prados, F. Camilli, and O. Faugeras. A unifying and rigorous shape from shading method adapted to realistic data and applications. *Accepted to the Journal of Mathematical Imaging and Vision (JMIV)*, 2005.

19. E. Prados, F. Camilli, and O. Faugeras. A viscosity solution method for shape-from-shading without boundary data. *Submitted to ESAIM Mathematical Modelling and Numerical Analysis*, 2005.

20. E. Prados and O. Faugeras. "Perspective Shape from Shading" and viscosity solutions. In *Proceedings of ICCV'03*, volume 2, pages 826–831, October 2003.

21. E. Prados and O. Faugeras. Unifying approaches and removing unrealistic assumptions in Shape from Shading: Mathematics can help. In *Proceedings of ECCV'04*, 2004.

22. E. Prados and O. Faugeras. A generic and provably convergent shape-from-shading method for orthographic and pinhole cameras. *accepted to IJCV*, 2005.

23. E. Prados and S. Soatto. Fast Marching Method for Hamilton-Jacobi-Bellman equations and applications. Technical report, UCLA, to appear in 2005; Journal version in preparation.

24. E. Rouy and A. Tourin. A Viscosity Solutions Approach to Shape-from-Shading. *SIAM Journal of Numerical Analysis*, 29(3):867–884, June 1992.

25. J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.

26. J.A. Sethian and A. Vladimirsky. Ordered upwind methods for static hamilton–jacobi equations: Theory and algorithms. *SIAM Journal on Numerical Analysis*, 41(1):325–363, 2003.

27. A. Tankus, N. Sochen, and Y. Yeshurun. A new perspective [on] Shape-from-Shading. In *Proceedings of ICCV'03*, volume 2, pages 862–869, October 2003.

28. A. Tankus, N. Sochen, and Y. Yeshurun. Perspective Shape-from-Shading by Fast Marching. In *Proceedings of CVPR'04*, 2004.

29. Y.-H. Tsai, L.-T. Cheng, S. Osher, and H.-K. Zhao. Fast sweeping algorithms for a class of hamilton-jacobi equations. *SIAM J. Numer. Anal.*, 41(2):673–694, 2003.

30. J.N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automatic Control*, 40:1528–1538, 1995.

31. S.Y. Yuen, Y.Y. Tsui, Y.W. Leung, and R.M.M. Chen. Fast marching method for shape from shading under perspective projection. In *Proceedings of VIIP'02*, pages 584–589, 2002.

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