

Chapter 9

A Brief Survey on Semi-Lagrangian Schemes for Image Processing

Elisabetta Carlini, Maurizio Falcone, and Adriano Festa

Abstract In this survey we present some semi-Lagrangian schemes for the approximation of weak solutions of first and second order differential problems related to image processing and computer vision. The general framework is given by the theory of viscosity solutions and, in some cases, of calculus of variations. The schemes proposed here have interesting stability properties for evolutive problems since they allow for large time steps, can deal with degenerate problems and are more accurate if compared to standard finite difference/element methods of the same order. Several examples on classical problems will illustrate these properties.

9.1 Introduction

Nonlinear Partial Differential Equations (PDEs) appear in many classical image processing problems and there is a need for accurate and efficient numerical methods to solve them. Usually we have to deal with several difficulties, including the fact that in most cases these are degenerate problems with non smooth solutions. Classical methods (Finite Elements, Finite Differences) must be modified and often stabilized to solve this kind of problems (the interested reader will find in the books by Strikwerda [46] and Brenner and Scott [6] a comprehensive introduction to these classical techniques and well as some recent results). The goal of this paper is to present semi-Lagrangian (SL in the sequel) approximation techniques and illustrate their properties through the solution of some classical image processing problems.

SL methods were introduced long time ago for linear advection problems. As far as we know the first example is the Courant-Isaacson and Rees scheme [19] which dates back to the 1960s. However, their application to nonlinear problems is more

E. Carlini (✉) · M. Falcone · A. Festa
Dipartimento di Matematica “G. Castelnuovo”, Sapienza – Università di Roma,
P. Aldo Moro, 2 – 00185 Roma, Italy
e-mail: carlini@mat.uniroma1.it; falcone@mat.uniroma1.it; festa@mat.uniroma1.it

recent as well as the analysis of their mathematical properties in terms of consistency, stability, convergence and convergence rates. A comprehensive introduction to this class of schemes can be found in the book by Falcone and Ferretti [32].

Let us just recall here some of the motivations which make semi-Lagrangian schemes an interesting alternative to more classical finite differences and finite elements techniques. The first and perhaps more interesting point is that, due to the fact that they mimic the method of characteristics, they allow for large time steps in evolutive problems. This property is particularly useful when one looks for the limiting behavior for t going to infinity in order to recover the solution of the corresponding stationary problem. The second important property is that they introduce only a small numerical viscosity so that their smoothing effect on the solution is very limited. The third property to be mentioned is that they have a natural interpretation in terms of the representation formula for some nonlinear problems. One example is the solution of the eikonal equation where the Hopf-Lax formula applies (see the next section for details). Roughly speaking they can be interpreted as a discrete Hopf-Lax formula and this means that their accuracy can be increased to high-order. Finally, they can work on structured and unstructured grids. Although the latter are not very popular among the image processing community they are a natural choice in other fields of applications like fluid dynamics, meteorology, geophysics. The details regarding the analytical properties of the schemes will not be illustrated here since this survey is mainly intended to present the schemes and their potential via the solution of some test problems.

In the next section we will briefly describe how the schemes are derived for some first and second order problems which come from image processing. We will consider some simplified models in order to introduce the main ideas of SL approximation and to explain how the methods are constructed. The following sections are devoted to some classical problems in image processing: Shape from Shading, nonlinear filtering, segmentation and optical flow. For each of them we will introduce the model which we are going to approximate, present the corresponding SL scheme and illustrate some numerical experiments.

9.2 An Introduction to Semi-Lagrangian Schemes for Nonlinear PDEs

We will consider three classical examples to show the main features and properties of the SL approximation: Shape from Shading (SfS in the sequel), segmentation via the level set (LS) method and nonlinear filtering via Mean Curvature Motion (MCM). It is worth to note that they all lead to nonlinear (and possibly degenerate) partial differential models.

Let us start with the *Shape from Shading* problem. This is an inverse problem where we want to compute the surface corresponding to a single image. A simple model for this problem can be derived (see the next section for more details)

assuming that the light source is at infinity in the direction ω , the surface is Lambertian, and that we can neglect perspective deformations. We consider then the equation which appears in most of the papers and corresponds to frontal light source at infinity, i.e. $\omega = (0, 0, 1)$ and

$$|\nabla u(x)| = f(x) \quad \text{for } x \in \Omega, \tag{9.1}$$

where $I : \Omega \subset \mathbb{R}^2 \rightarrow [0, 1]$ represents the brightness function of the image and

$$f(x) = \sqrt{\frac{1}{I(x)^2} - 1}. \tag{9.2}$$

This is an *eikonal equation*, a first order stationary nonlinear pde. In order to solve (9.1) we must provide some boundary conditions on $\partial\Omega$ and/or add some informations to select a unique solution. In fact, the equation just depends on the gradient so that if u is a solution also $u + \text{constant}$ is a solution and we need to fix the height of the surface at least at the boundary to select a solution. For an image containing an “occluding boundary”, it is usual to consider this boundary as $\partial\Omega$ and the domain of reconstruction will be the region Ω enclosed by the occluding boundary.

Another classical problem is the *segmentation problem*, where we want to detect the boundaries of objects represented in a picture. A very popular method for segmentation is based on the level set method, this application is often called “active contour” since the segmentation is obtained following the evolution of a simple curve (a circle for example) in its normal direction. Starting from a circle inside an object we follow its evolution until the curve touches the borders of the object. It is important to know that the LS method has had a great success for the analysis of front propagation problems for its capability to handle many different physical phenomena within the same theoretical framework. Moreover, LS methods allow to develop the analysis of interface propagation also after the onset of singularities (as for example, when there is a topological change in the front). See the books by Sethian [45] and Osher and Fedkiw [42].

In the LS methods the unknown is a “representation” function $u : \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}$ of the interface, the position of the interface Γ_t at time t is given by the 0-level set of $u(., t)$, i.e.

$$\Gamma_t := \{x : u(x, t) = 0\}.$$

The model equation corresponding to the LS method (see e.g.[43] for details) is

$$\begin{cases} u_t + c(x, t)|\nabla u| = 0 & (x, t) \in \mathbb{R}^2 \times [0, T] \\ u(x) = u_0(x) & x \in \mathbb{R}^2 \end{cases} \tag{9.3}$$

where u_0 must be a *representation function for the initial front* $\partial\Omega_0$, where Ω_0 is a given open bounded subset of \mathbb{R}^2 , this means

$$\begin{cases} u_0(x) > 0 & \text{for } x \in \mathbb{R}^2 \setminus \overline{\Omega_0} \\ u_0(x) = 0 & \text{for } x \in \partial\Omega_0 := \Gamma_0 \\ u_0(x) < 0 & \text{for } x \in \Omega_0. \end{cases} \quad (9.4)$$

Note that usually the scalar velocity $c : \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}$ is given and it represents the velocity of the front in its normal direction $\eta(x, t) := \frac{\nabla u(x, t)}{|\nabla u(x, t)|}$. However, the same approach applies to other scalar velocities. For example one can use it to describe isotropic and anisotropic front propagation, Mean Curvature Motion (MCM) and other situations when the velocity depends on some geometrical properties of the front. In summary, this corresponds to the following choices:

$$c(x, t) \quad \text{isotropic growth with time varying velocity} \quad (9.5)$$

$$c(x, t, \eta) \quad \text{anisotropic growth, i.e. dependent on the normal direction} \quad (9.6)$$

$$c(x, t, k) \quad \text{Mean Curvature Motion} \quad (9.7)$$

where $k = k(x, t)$ is the mean curvature to the front at time t . There are even models where the velocity is obtained by convolution (nonlocal velocity).

Since the borders of objects inside a given image I are characterized by the fact that there is a rapid change in the values of I , in the segmentation problem one can take the gradient of I as an indicator of the edges. If $|\nabla I|$ is large this means that we are at an edge. For this reason, a popular choice of the velocity when we apply LS methods to the segmentation problem is

$$c(x) := (1 + |\nabla I(x)|^p)^{-1} \text{ where } p \geq 1.$$

With this choice, the velocity is 1 inside the region where the gray level is constant and rapidly decreases to 0 when x is close to an edge (which corresponds to a jump in the I derivative).

Another problem we will consider is *nonlinear filtering*. A classical model to treat this problem is to consider (9.3) with the velocity given by Mean Curvature $c(x, t) = k(x, t)$, see [34]. In this case one gets

$$\begin{cases} u_t(x, t) = \operatorname{div} \left(\frac{\nabla u(x, t)}{|\nabla u(x, t)|} \right) |\nabla u(x, t)| & (x, t) \in \mathbb{R}^2 \times [0, T] \\ u(x, 0) = I_0(x) \end{cases} \quad (9.8)$$

where I_0 is a starting noisy image. This equation, in the level set formalism, represents the Mean Curvature Motion of curves, since each level set of u moves in normal direction with a speed proportional to their mean curvature. The goal is to recover a new image reducing the noise. Note that the evolutive equation is not well defined at all points where the gradient vanishes so it is a singular parabolic equation. This implies that classical finite difference and finite element methods must be adapted in order to deal with the singularities (see [43] for a finite difference scheme and [15, 41] for two convergence results).

Let us now turn our attention to the construction of the semi-Lagrangian schemes for the above model equations. For readers' convenience, we will start from the linear advection equation:

$$u_t + b(x, t) \cdot \nabla u = f(x, t) \quad (x, t) \in \mathbb{R}^2 \times (0, T). \tag{9.9}$$

Here, $b : \mathbb{R}^2 \times (0, T) \rightarrow \mathbb{R}^2$ is a vectorfield (the *drift*) and $f : \mathbb{R}^2 \times (0, T) \rightarrow \mathbb{R}$ is the *source* term. We look for the unique solution $u : \mathbb{R}^2 \times (0, T) \rightarrow \mathbb{R}$ of (9.9) satisfying the initial condition

$$u(x, 0) = u_0(x) \quad x \in \mathbb{R}^2. \tag{9.10}$$

A simple case corresponds to $f(x, t) \equiv 0$ and $b(x, t) \equiv c$ (constant). Then, the solution u is given by the representation formula

$$u(x, t) = u_0(x - ct) \quad (x, t) \in \mathbb{R}^2 \times [0, T) \tag{9.11}$$

which can be easily obtained by the *method of characteristics*. The representation formula (9.11) contains an important information: the solution at the point x at time t only depends on the initial value at the point $x - ct$ which is called *the foot of the characteristic* passing through (x, t) (more details on the method of characteristics can be found on many classical books presenting the basic theory of partial differential equations, e.g. [28]). A typical choice is to build a space grid with constant step Δx

$$\mathcal{G}_{\Delta x} := \{x_j : x_j = (j_1 \Delta x, j_2 \Delta x) \text{ for } j_1 \in \mathbb{Z}, j_2 \in \mathbb{Z}\}. \tag{9.12}$$

This grid is called *lattice*. In order to build an approximation scheme for (9.9) and more in general for evolutive problems, we need to consider a space-time grid

$$\mathcal{G}_{\Delta x, \Delta t} := \{(x_j, t_n) : x_j = (j_1 \Delta x, j_2 \Delta x), t_n = n \Delta t, \text{ for } j_1 \in \mathbb{Z}, j_2 \in \mathbb{Z}, n \in \mathbb{N}\} \tag{9.13}$$

where in the simplest case Δt is a constant time step. Naturally for numerical purposes the number of nodes of our grids has to be finite and we will consider the intersections of our grids with the computational domain Ω .

The basic idea behind all finite difference approximations is to replace every derivative by an incremental ratio. Thus, one obtains a finite dimensional problem whose unknown are the values of the numerical solution at all the nodes of the lattice, so that the value u_j^n associated to the node (x_j, t_n) should be regarded as an approximation of $u(x_j, t_n)$. For the time derivative it is natural to choose the forward incremental ratio

$$u_t(x, t) \approx \frac{u(x, t + \Delta t) - u(x, t)}{\Delta t}. \tag{9.14}$$

For finite difference, it is well known that dealing with advection problems we need to consider the incremental ratio *in the up-wind direction*, so that we must take the first node on the left (right) if the velocity is positive (negative). This choice guarantees the stability of the approximation scheme as well as a good fit with the physics behind the model. When $b(x_j, t_n)$ is positive, this corresponds to the following *up-wind* scheme for the left hand side of the advection equation

$$u_t(x_j, t_n) + b(x_j, t_n) \cdot \nabla u(x_j, t_n) \approx \frac{u(x_j, t_n + \Delta t) - u(x_j, t_n)}{\Delta t} + \\ -b_1(x_j, t_n) \frac{u(x_{j_1-1, j_2}, t_n) - u(x_{j_1, j_2}, t_n)}{\Delta x} - b_2(x_j, t_n) \frac{u(x_{j_1, j_2-1}, t_n) - u(x_{j_1, j_2}, t_n)}{\Delta x}. \quad (9.15)$$

Adopting the standard notation $u_j^n = u(x_j, t_n)$, we write the scheme in the standard explicit time marching form

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{\Delta x} b(x_j, t_n) \cdot (u_{j_1-1, j_2}^n - u_j^n, u_{j_1, j_2-1}^n - u_j^n). \quad (9.16)$$

Semi-Lagrangian schemes are based on a different way to construct the approximation of (9.9), in particular this affects the discretization of the advection term $b(x, t) \cdot \nabla u$. Since this is a directional derivative, we can write

$$b(x_j, t_n) \cdot \nabla u(x_j, t_n) \approx -\frac{u^n(x_j - \delta b(x_j, t_n)) - u_j^n}{\delta}, \quad (9.17)$$

where δ is a “small” positive parameter, and u^n denotes an extension of the numerical solution (at time t_n) to be computed outside of the grid. Coupling the forward finite difference in time with this approximation we get

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{u^n(x_j - \delta b(x_j, t_n)) - u_j^n}{\delta} = 0, \quad (9.18)$$

and finally, choosing $\delta = \Delta t$, we obtain the scheme

$$u_j^{n+1} = u^n(x_j - \Delta t b(x_j, t_n)). \quad (9.19)$$

This is the semi-Lagrangian scheme for (9.9) for $f(x, t) = 0$. Note that, for $b(x, t) \equiv c$ (9.18) can be easily interpreted as a discrete version of the representation formula (9.11). Even in this simple case, the value at the point $x_j - \Delta t c$ does not belong to the grid $\mathcal{G}_{\Delta x}$ and must be obtained by interpolation on the values at the grid nodes. We will denote by $\Pi[w]$ the interpolation operator which reconstructs locally the values of a function w only known on $\mathcal{G}_{\Delta x}$. There are many possible choices for Π , e.g. linear, bilinear or cubic interpolation (see also [32] for more general choices). A very popular choice is the piecewise linear interpolation which produces a monotone first order scheme.

Now let turn our attention to the nonlinear case and consider the eikonal equation. First note that

$$|\nabla u(x)| = \max_{a \in B(0,1)} a \cdot \nabla u(x). \quad (9.20)$$

where $B(0, 1)$ denotes the unit ball centered at the origin. In fact, the maximum in (9.20) is attained at $a^* = \nabla u(x)/|\nabla u(x)|$ and this shows that the right-hand side coincides with $|\nabla u(x)|$. Using now (9.20) to rewrite Eq. (9.3), we get

$$u_t + \max_{a \in B(0,1)} [c(x, t)a \cdot \nabla u(x)] = 0. \quad (9.21)$$

Since the term inside the square brackets is linear, we can apply the semi-Lagrangian scheme for the advection equation obtaining the following scheme

$$u_j^{n+1} = \min_{a \in B(0,1)} [u^n(x_j - \Delta t c(x_j, t_n)a)]. \quad (9.22)$$

Note that again u^n is computed at a point which in general is not a node of the grid, this will require a local reconstruction via the interpolation operator Π . We can write (9.22) in a more explicit way as

$$u_j^{n+1} = \min_{a \in B(0,1)} [\Pi[u^n](x_j - \Delta t c(x_j, t_n)a)]. \quad (9.23)$$

The meaning of the right-hand side is: to recover the correct solution of the nonlinear stationary equation one needs to solve a family of advection problems corresponding to a vector field pointing in every direction and then has to take the minimum value among all possible ones. This step clearly requires a constrained optimization method to be solved. In practice, for low order accuracy it suffices to compare a finite number of values corresponding to a uniform discretization of the unit ball.

It is important to note that the SL scheme is *up-wind* by construction and this explains its strong stability properties which allow to use large time steps with respect to other methods.

The derivation of an efficient discretization scheme for the MCM equation is more complicated. One can think to simply substitute $c_j^n = k_j^n$ where $k_j^n = k(x_j, t_n)$ is the curvature at the point x_j at time t_n . This can be done using a discrete version of the curvature for the level set of the solution as in [43]. However, via a stochastic interpretation of the equation, one can derive a semi-Lagrangian approximation where the role of characteristics is taken by the solution of an appropriate stochastic differential equation associated to the equation. More information on this approach for the MCM can be found in [14] and are too technical to be presented here. However, we will present the resulting SL scheme in Sect. 9.4.1.

In conclusion, the construction of SL-schemes is based on three essential building blocks: an integration backward along the characteristics (or the generalized characteristics in the second order problem), an interpolation to get the values at the foot of the characteristics and an optimization method to recover the

minimum among all the values. The first step relies on standard techniques for ordinary differential equation, the interpolation can be obtained either by standard polynomial techniques or via more sophisticated techniques, for example ENO (Essentially Non Oscillatory) or WENO (Weighted ENO) methods as in [13]. Finally, for the optimization step, one can simply discretize the unit ball or use a minimization method without derivatives such as Brent algorithm [7]. An extensive presentation of those building blocks and an analysis of SL schemes can be found in [32].

9.3 Shape from Shading

As we said, we want to reconstruct a surface $u(x)$ from a single image.

This means that given an image I we are looking for the surface $u(x)$ corresponding to the object represented in it. The irradiance function I is the datum in the model since it is measured at each pixel $x := (x_1, x_2)$ of the image, for example in terms of a grey level (from 0 to 255). To construct a continuous model, we will assume that I takes real values in the interval $[0, 1]$. The height function u which is the unknown of the problem has to be reconstructed on the “reconstruction domain” Ω . The characterization of the surface via a partial differential equation relies on several assumptions. Assume that there is a unique light source at infinity whose direction is indicated by the unit vector $\omega = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$. Also assume for simplicity that ω is given. For a Lambertian surface of uniform albedo equal to 1, the reflectance function giving the brightness at a point on the surface is $R(\eta(x)) := \omega \cdot \eta(x)$, where $\eta(x)$ is the unit normal to the surface at $(x_1, x_2, u(x))$. This equation is called the *irradiance equation*. Recalling that the normal is given by $\eta(x) = (-u_{x_1}, -u_{x_2}, 1)$ we obtain the following equation

$$I(x) = \frac{(\omega_1, \omega_2) \cdot \nabla u(x) - \omega_3}{\sqrt{1 + |\nabla u(x)|^2}} \quad \text{for } x \in \Omega, \quad (9.24)$$

which is a first order nonlinear partial differential equation of Hamilton-Jacobi type. We have seen that in the vertical light source case (i.e. $\omega = (0, 0, 1)$) this reduces to the eikonal equation (9.1).

As we said, this equation must be complemented with boundary conditions. A natural choice is to consider homogeneous Dirichlet type boundary conditions corresponding to flat background i.e., setting

$$u(x) = 0 \quad \text{for } x \in \partial\Omega. \quad (9.25)$$

However, one can also choose a more general Dirichlet boundary condition

$$u(x) = g(x) \quad \text{for } x \in \partial\Omega. \quad (9.26)$$

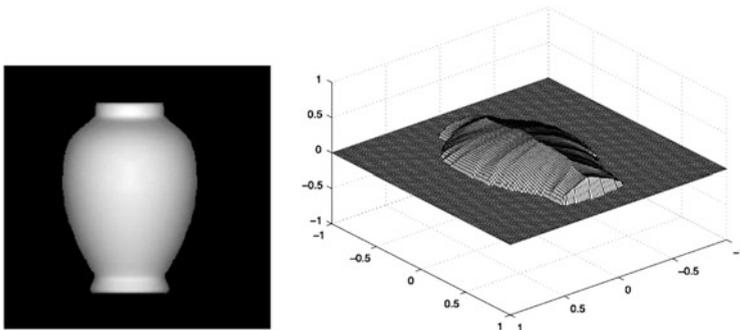


Fig. 9.1 The image and its corresponding surface (approximation)

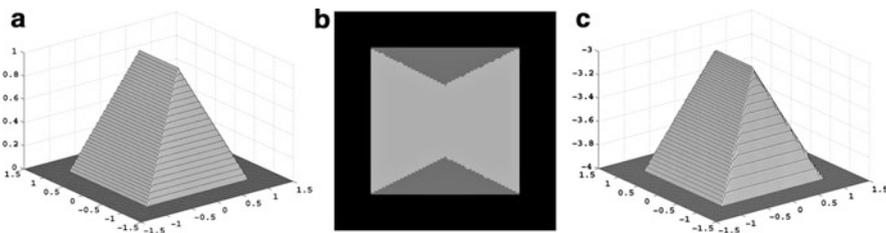


Fig. 9.2 (a) The original surface, (b) its brightness function in the plane, (c) the approximate surface

The solution of the above Dirichlet problems (9.24) and (9.25) or (9.24) and (9.26) will give the surface corresponding to grey level $I(x)$ measured in Ω . To set $u(x) = g(x)$ at least on a part of the boundary we must know something about the surface. This is the case for surfaces obtained by revolution around an axis (as for our vase in Fig. 9.1).

In order to illustrate one of the features of the SL scheme, let us consider a virtual image corresponding to the surface

$$u(x_1, x_2) = \begin{cases} 2(1 - |x_1|) & x_1 \in [-1, 1], \quad x_2 \in \left[-1, -\frac{1}{2}|x_1| - \frac{1}{2}\right] \\ 2(1 - |x_2|) & x_1 \in [-1, 1], \quad x_2 \in \left[\frac{1}{2}|x_1| + \frac{1}{2}, 1\right] \\ 1 - |x_1| & \text{otherwise.} \end{cases}$$

The surface looks like a “ridge tent” (see Fig. 9.2), so it is non regular but the boundary conditions are very simple: 0 on each side of the rectangle. We apply our SL scheme for the eikonal equation (9.22). Looking at the right picture of Fig. 9.2, we can see that the algorithm is accurate around the kinks and that there are no spurious oscillations where the gradient is discontinuous. Similar results have been

obtained for other virtual and real images. A detailed presentation of these results as well as a comparison with other numerical methods can be found in the survey [27].

Let us also mention that the same problem can be solved by applying optimization methods based on the variational approach. The interested reader can find in Horn and Brooks' book [35] several results and references relating to the variational approach. In this class of methods two basic ingredients must be chosen: the functional which has to be optimized (in fact, minimized) and the minimization method. The first difficulty encountered in the SfS problem is the choice of unknowns. The natural unknown is of course height u as we have done before. However, this is not the only choice since u appears in the image irradiance equation only through its first derivatives $p = \partial u / \partial x_1$ and $q = \partial u / \partial x_2$, which are two non-independent functions. In fact, for $u \in C^2$ we know that

$$\partial p / \partial x_2 = \partial q / \partial x_1. \quad (9.27)$$

The only technical difficulty with these unknowns is that p or q becomes infinite at each point x belonging to an occluding boundary. This is not a real trouble if no point x in the reconstruction domain Ω is such that $I(x) = 0$. As equation (9.27) is a hard constraint on p and q , the most natural functional associated with the irradiance equation and (9.27) is

$$\mathcal{F}_1(p, q, \mu) = \int_{\Omega} [r(p(x), q(x)) - I(x)]^2 dx + \int_{\Omega} \mu(x) \left[\frac{\partial p}{\partial x_2}(x) - \frac{\partial q}{\partial x_1}(x) \right] dx \quad (9.28)$$

where μ is a Lagrange multiplier and the function r is such that $r(p(x), q(x)) = R(n(x))$. Note that \mathcal{F}_1 is defined on the unknowns p , q and μ . However Horn and Brooks have shown that the three Euler equations associated with \mathcal{F}_1 can be reduced, for $u \in C^2$, to the Euler equation associated with the following functional:

$$\mathcal{F}_2(p, q) = \int_{\Omega} [r(p, q) - I(x)]^2 dx. \quad (9.29)$$

The choice of p and q as unknown is due to the fact that the algorithms dealing directly with u are very slow, this choice implies that at the end we will need also another procedure to get back to u . Typically this is done integrating along paths.

As we said, in the previous model we have neglected the *perspective deformations*. New models have been studied to deal with this problem, which is usually called in the literature Perspective Shape from Shading (PSfS in short) problem. Let us define our model problem adopting the same notation used in [20] (a different model with an attenuation term has been studied in [44]). The point (X_0, Y_0) is the principal point of the image, i.e. the intersection between the optical axis of the camera and the perspective plane Π (the film). The parameters d and d' are respectively the distance of the optical lens from the perspective plane and the distance of the optical lens from the (flat) background, l and $l' = \frac{d'}{d}l$ are

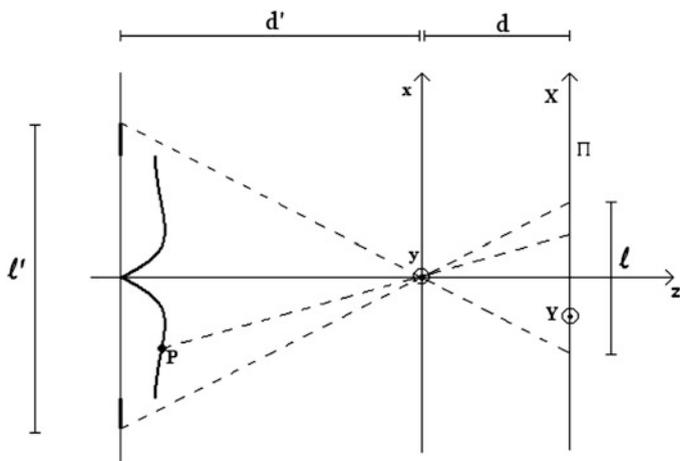


Fig. 9.3 Diagram of the optical lens and of the perspective transformation

respectively the length of a segment in the perspective plane (i.e. in the image) and the length of the real segment corresponding to it (see Fig. 9.3 for more details). The representation of the surface defined by points with coordinates (x, y, z) in terms of the (X, Y) coordinates of the points in the perspective plane is given by three parametric equations

$$x = r(X, Y), \quad y = s(X, Y), \quad z = t(X, Y) \tag{9.30}$$

where (see [20])

$$\begin{cases} r(X, Y) = \frac{X - X_0}{d} t(X, Y) \\ s(X, Y) = \frac{Y - Y_0}{d} t(X, Y). \end{cases} \tag{9.31}$$

The problem is then reduced to compute the third component t . Writing the classical irradiance equation, taking into account the new variables for the perspective deformation and introducing the new variable

$$\bar{t}(X, Y) := t(X, Y) + (X - X_0, Y - Y_0) \cdot \nabla t(X, Y), \tag{9.32}$$

we can get the following eikonal type equation

$$\left[\frac{d}{\bar{t}(X, Y)} \right]^2 |\nabla t(X, Y)|^2 = \frac{I_{max}^2}{I'(X, Y)^2} - 1 \quad \text{in } \Omega \tag{9.33}$$

where

$$I'(X, Y) := \frac{I(X, Y)}{\cos^4 \alpha(X, Y)}, \quad (9.34)$$

I_{max} is a constant depending on the parameters of the problem (typically is the maximum value of I measured on the image) and the α is the angle between the optical axis and the the ray starting at a point of the scene and passing through the optical center. Note that α varies with the point. Moreover, the coefficient

$$\cos^4(\alpha(X, Y)) = \frac{d^4}{((X - X_0)^2 + (Y - Y_0)^2 + d^2)^2}, \quad (9.35)$$

is used to decrease the brightness of the points corresponding to a large α . Note that Ω is the internal region bounded by the silhouette of the object ($\partial\Omega$ will denote its boundary) which is embedded in a rectangular domain Q , so that the set $Q \setminus \Omega$ is the background.

Defining

$$f(X, Y) := \frac{1}{d^2} \left(\frac{I_{max}^2}{I'(X, Y)^2} - 1 \right) \quad (9.36)$$

we can write (9.33) in short as

$$|\nabla t(X, Y)| = \sqrt{f(X, Y)} \left| \bar{t}(X, Y) \right|. \quad (9.37)$$

We want to write (9.37) in a fixed point form and construct an approximation scheme for this equation. To this end it is important to note that \bar{t} has a sign. In fact, the normal to the original surface in the point P is given by

$$\hat{n}(P) = N(P)/|N(P)| \quad (9.38)$$

where

$$N(P) \equiv \left(\frac{\partial t}{\partial X}(X, Y), \frac{\partial t}{\partial Y}(X, Y), -\bar{t}(X, Y) \right) \quad (9.39)$$

and since $-\bar{t}$ must be positive (according to the orientation of the z axis in Fig. 9.3), \bar{t} must be negative. This implies that (9.37) is in fact

$$|\nabla t(X, Y)| + \sqrt{f(X, Y)}(t(X, Y) + (X - X_0, Y - Y_0) \cdot \nabla t(X, Y)) = 0$$

which can be written in short as

$$H((X, Y), t, \nabla t) = 0, \quad \text{in } \Omega \quad (9.40)$$

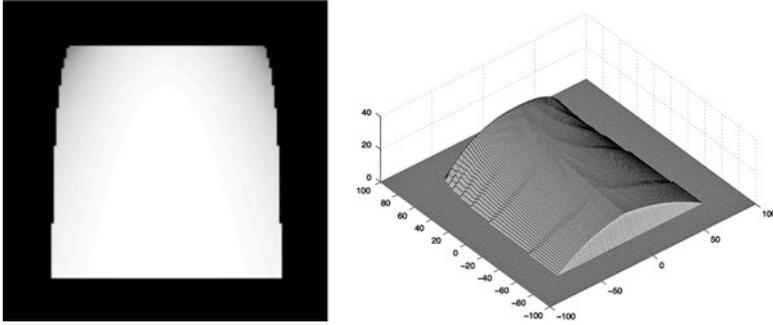


Fig. 9.4 A real photograph of one page, 128×128 pixels (*left*) and reconstructed surface with Dirichlet boundary conditions (*right*)

where the Hamiltonian H represents the left-hand side of (9.40). It should be noted that, with respect to the classical SfS problem without perspective, we have an additional term which is linear in ∇t .

Let us consider equation (9.40) complemented with the Dirichlet boundary condition

$$t = g(X, Y) \quad \text{on } \partial\Omega, \quad \text{where } -d' \leq g \leq 0 \tag{9.41}$$

The arguments presented in Sect. 9.2 for the approximation of the directional derivative lead to the standard semi-Lagrangian scheme for (9.40) which we write in a fixed point form as

$$t(X, Y) = F[t](X, Y), \quad \text{in } \Omega \tag{9.42}$$

where the operator F is defined as

$$F[t](X, Y) := \frac{1}{1 + \delta} \inf_{a \in B(0,1)} \left\{ t \left((X, Y) + \delta \left(\frac{-a}{\sqrt{f}} - (X, Y) \right) \right) \right\}, \quad \text{on } \Omega. \tag{9.43}$$

and $B(0, 1)$ is the unit ball in \mathbb{R}^2 . Introducing a space discretization, and defining $x_j = (X_{j_1}, Y_{j_2})$, we can write for every node $x_j \in \Omega \cap \mathcal{G}_{\Delta x}$ Eq. (9.42)

$$t(x_j) = F[t](x_j) := \frac{1}{1 + \delta} \inf_{a \in B(0,1)} \left\{ \Pi[t] \left(x_j + \delta \left(\frac{-a}{\sqrt{f(x_j)}} - x_j \right) \right) \right\} \tag{9.44}$$

which gives the fully discrete semi-Lagrangian scheme for the PSfS problem. Naturally, this equation must be complemented with the boundary conditions (9.41).

In the following test we start from a real photograph where the effect of perspective is visible. The surface is a sheet of paper with the shape of a roof tile. For this image the parameter values are: $l = 6.91$ mm, $d = 5.8$ mm, $l' = 200$ mm, $d' = \frac{l'}{l}d = 167.87$ mm, $\Delta x = 0.05$ mm. We note that we performed a light correction in the preprocessing step, so we can assume $I_{max} = 1$ during computation. Figure 9.4 shows the photograph (128×128 pixels) and the surface reconstructed

using Dirichlet boundary conditions ($t = 0$ on the left and right sides of the boundary and $t = g$ on the top and the bottom). We can see that the solution is quite good considering the fact that light source (flash camera) is not far from the object and that direction of light source is not perfectly vertical as the mathematical model would have required.

9.4 Nonlinear Filtering via MCM

The aim of this section is to present two classical methods based on Mean Curvature Motion for image restoration. These models can be written in the following general form:

$$\begin{cases} u_t(x, t) = \operatorname{div} \left(\frac{\nabla u(x, t)}{|\nabla u(x, t)|} \right)^\beta |\nabla u(x, t)| & (x, t) \in \Omega \times (0, T] \\ u(x, 0) = I_0(x) & x \in \Omega \\ \frac{\partial u(x, t)}{\partial \eta} = 0 & (x, t) \in \partial\Omega \times (0, T] \end{cases} \quad (9.45)$$

where $\eta(x)$ is the normal direction at the point x . We will consider the case $\beta = 1$ corresponding to classical Mean Curvature Motion (MCM) and the case $\beta = 1/3$ corresponding to Affine Morphological Scale Space (AMSS). Here $\{u(x, t)\}_{t \geq 0}$ represents a family of successive restored versions of the initial noisy image I_0 . The variable t is called the scale variable and, as t increases, we expect $u(x, t)$ to be a smoothed image, in which the noise has been removed. The model is able to preserve significant features, like edges and corners, which are typically lost when using linear PDE-methods based on the heat equation where the smoothing effect is isotropic. In curvature-related diffusion, the smoothing effect is anisotropic, i.e. the diffusion depends on the image and is not constant in all directions. The model with $\beta = 1$ corresponds to “pure” anisotropic diffusion and it was introduced for image processing by Alvarez et al. in [1]. The term $\operatorname{div} \left(\frac{\nabla u(x, t)}{|\nabla u(x, t)|} \right) |\nabla u(x, t)|$ represents a degenerate diffusion term, which diffuses in the direction orthogonal to the gradient ∇u and does not diffuse in the direction of ∇u . In fact, denoting by $\sigma(\nabla u) = \left(\frac{\nabla u}{|\nabla u|} \right)^\perp$ the unit vector orthogonal to ∇u , a formulation of the previous equation with respect to $\sigma(\nabla u)$ would be:

$$u_t = \sigma(\nabla u)^\top D^2 u \sigma(\nabla u),$$

where $D^2 u$ is the Hessian matrix with respect to x . This equation admits a unique continuous viscosity solution, if $u(x, 0)$ is bounded uniformly continuous, see [18, 29].

In the paper [2] by Alvarez et al., a connection between scale space analysis and PDEs is proved, the authors also show that the MCM operator satisfies mostly

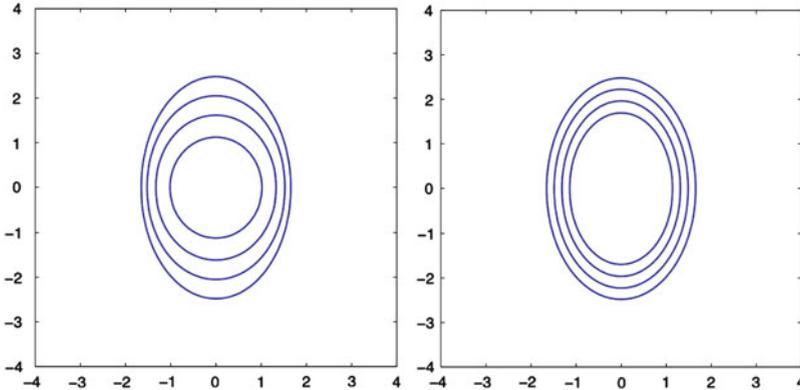


Fig. 9.5 Shrinking of an ellipse by MCM (left) and AMSS (right)

all the relevant properties in multi-scale analysis as monotonicity, invariance under grey scale change, translation invariance. Moreover, they prove that the only operator satisfying all these properties together with affine invariance is the model represented by (9.45) with $\beta = 1/3$. Affine invariance requires the solution of the equation to be invariant with respect any affine transformation: let $\{T_t\}_{t \geq 0}$ represents a family of operators which applied to the original image yields a sequence of images $u(x, t) = T_t(u)$, solution of (9.45), then the affine invariance is stated as follows:

$$T_t(u \circ \phi) = T_{t \cdot |J_\phi|} u \circ \phi$$

for any affine map ϕ , where J_ϕ represents the Jacobian of ϕ , which is an invertible matrix. Such property guarantees that shapes are better maintained during smoothing, a classical example is the evolution of an ellipse. It is well known that under MCM any convex curve will propagate to a circle and then collapse to point, instead by the AMSS model the ellipse will collapse to a point preserving the shape, we show the evolution of an ellipse by MCM and AMSS in Fig. 9.5. This property makes the AMSS model particularly attractive for image analysis. In next section, we will show a semi-Lagrangian approximation of both equations with an application to image denoising.

9.4.1 SL Approximation for the Nonlinear Filtering Problem via MCM

We present a semi-Lagrangian (SL) scheme to approximate equation (9.45) for MCM, case $\beta = 1$, and AMSS, case $\beta = 1/3$. For the first equation, we use the scheme described in [14], where a fully explicit scheme is obtained. We denote by u_j^n the numerical approximation of $u(x, t)$ at the node $(x_j, t_n) \in \mathcal{G}_{\Delta x, \Delta t}$, by



Fig. 9.6 Original, noisy ($\nu = 12$) and filtered image by MCM ($C = 0.02$, $s = 1$, $\Delta x = 1$, $\Delta t = 0.25$, $N = 5$)

D_j^n the central finite difference approximation of the space gradient, by $\mathcal{D}(j) = \{(j_1 + 1, j_2), (j_1 - 1, j_2), (j_1, j_2 + 1), (j_1, j_2 - 1)\}$ and we define $\sigma_j^n = \sigma(D_j^n)$.

Now, we can write the SL scheme for MCM:

$$u_j^{n+1} \equiv \begin{cases} \frac{1}{2} \left(\Pi[u^n](x_j + \sigma_j^n \sqrt{\Delta t}) + \Pi[u^n](x_j - \sigma_j^n \sqrt{\Delta t}) \right) & \text{if } |D_j^n| > C \Delta x^s \\ \frac{1}{4} \sum_{i \in \mathcal{D}(j)} u_i^n & \text{if } |D_j^n| \leq C \Delta x^s. \end{cases} \quad (9.46)$$

Here, C and s are positive constant. Let us notice that the vector $\sigma_j^n = \sigma(D_j^n)$ is defined only where the discrete gradient D_j^n is not vanishing. When the gradient is zero the MCM is consistent with the heat equation, see [14, 22]. Then, in the case $|D_j^n| \leq C \Delta x^s$, the numerical solution is obtained as an average of the neighboring values, so that on these points the scheme results to be consistent with the following heat equation

$$u_t = \frac{\Delta x^2}{4 \Delta t} \Delta u.$$

We show an application of the scheme to filter noisy image. The noisy image is obtained adding Gaussian white noise of mean 0 and variance $(\nu/255)^2$ to a given clean image, I_{ex} (note that the noise parameter has been denoted by ν to avoid confusions). In all the tests we have chosen a cubic interpolation, and in order to deal with non-smooth data we have regularized the gradient using $G * \nabla u$, where G is a smoothing kernel. Moreover we have used the following approximation $(G * \nabla u)(x_j, t_n) \simeq \frac{1}{4} \sum_{i \in \mathcal{D}(j)} D_i^n$. The errors are measured by a relative L_1 discrete norm $\|\cdot\|_{1,r}$, defined as follows:

$$\|I_{ex}(\cdot) - u^N\|_{1,r} := \frac{\sum_{x_j \in \mathcal{G}_{\Delta x}} |I_{ex}(x_j) - u_j^N|}{\sum_{x_j \in \mathcal{G}_{\Delta x}} |I_{ex}(x_j)|}.$$

In Figs. 9.6 and 9.7 we show the clean image, the noisy image and the restored image.



Fig. 9.7 Original, noisy ($\nu = 25$) and filtered image by MCM ($C = 0.02$, $s = 1$, $\Delta x = 1$, $\Delta t = 0.25$, $N = 5$)

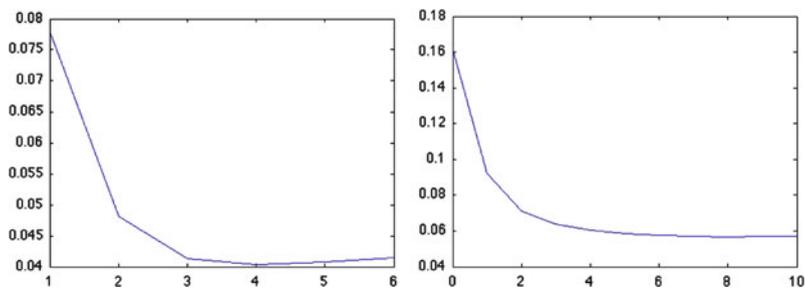


Fig. 9.8 L_1 errors (y -axis) and time iteration (x -axis) for MCM model applied to a noisy images with $\nu = 12$ (left) and $\nu = 25$ (right)

In Fig. 9.8 we show the relative L_1 discrete errors evolution with respect to t . We see that in few iterations we reached the minimization of the L_1 errors.

To describe the SL scheme for the AMSS model, we first need to remind that the equation (9.45) in this case can be rewritten as following:

$$u_t = (\hat{\sigma}(\nabla u)^\top D^2 u \hat{\sigma}(\nabla u))^{1/3},$$

where $\hat{\sigma}(\nabla u) = (\nabla u)^\perp$, see for instance [34]. Then we introduce an extra parameter ρ , we define $\hat{\sigma}_j^n := \hat{\sigma}(D_j^n)$ and we take a directional second finite difference:

$$\hat{\sigma}(\nabla u)^\top D^2 u \hat{\sigma}(\nabla u)(x_j, t_n) \simeq \frac{u(x_j + \rho \hat{\sigma}_j^n, t_n) + u(x_j - \rho \hat{\sigma}_j^n, t_n) - 2u_j^n}{\rho^2}. \quad (9.47)$$

Finally, the SL scheme is written as

$$u_j^{n+1} \equiv \begin{cases} w_j^{n+1} & \text{if } |D_j^n| > C \Delta x \\ \frac{1}{4} \sum_{i \in \mathcal{D}(j)} w_i^{n+1} & \text{if } |D_j^n| \leq C \Delta x \end{cases}$$



Fig. 9.9 Original, noisy ($v = 12$) and filtered image by AMSS ($C = 0.02, s = 1, \Delta x = 1, \Delta t = 0.2, \rho = (\Delta t)^{1/6}, N = 10$)



Fig. 9.10 Original, noisy ($v = 25$) and filtered image by AMSS ($C = 0.02, s = 1, \Delta x = 1, \Delta t = 0.25, \rho = (\Delta t)^{1/6}, N = 10$)

where

$$w_i^{n+1} \equiv \begin{cases} u_i^n + \Delta t \left(\frac{\Pi[u^n](x_i + \rho\hat{\sigma}_i^n) + \Pi[u^n](x_i - \rho\hat{\sigma}_i^n) - 2u_i^n}{2\rho^2} \right)^{1/3} & \text{if } |D_i^n| > C\Delta x \\ u_i^n & \text{if } |D_i^n| \leq C\Delta x. \end{cases}$$

Such scheme as been proposed in [12], where a convergence result for its monotonic version is proved.

Let us notice that in this case the vector $\hat{\sigma}$ is always defined, anyhow we need to consider differently the case $\hat{\sigma} = 0$. In fact, if we would use the approximation (9.47) on the points where $\hat{\sigma} = 0$, no diffusion would happen. On these points, we extend the solution by continuity: we take an average of the numerical solution, computed at the previous step when $\hat{\sigma}$ is small, computed using (9.47) when $\hat{\sigma}$ is big.

As for the previous model, we have chosen a cubic interpolation and regularized the gradient.

In Fig. 9.9 and in Fig. 9.10, we show respectively the clean image, the noisy image and the restored image. In Fig. 9.11 we show the relative L_1 discrete errors evolution with respect to t . We see that in few iterations the minimum error is reached.

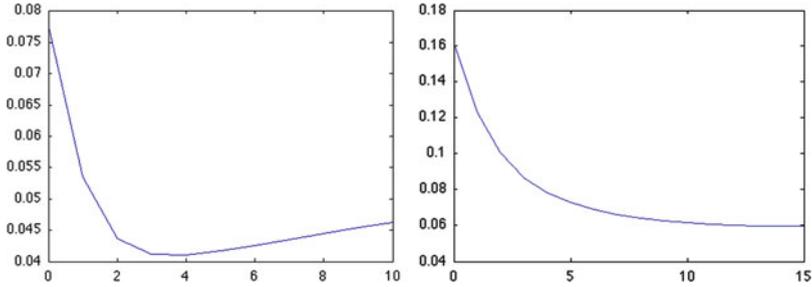


Fig. 9.11 Errors for MCM model for noisy images (left $\nu = 1$, right $\nu = 25$)

9.5 Segmentation via the LS Method

The segmentation problem in computer vision, as formulated by Mumford and Shah [40], can be defined in the following way: given an observed image $I_0 : \Omega \rightarrow [0, 1]$, find a decomposition of disjoint open sets $\Omega = \bigcup_i \overline{\Omega}_i$ and $C = \bigcup_i \partial\Omega_i$, such that I_0 varies smoothly within each Ω_i and rapidly or discontinuously across the boundaries of Ω_i . We denote by $|C|$ the length of curves which belong to C .

A classical way to solve this problem is solving the following minimization problem:

$$\inf_{I,C} F^{MS}(I, C) \tag{9.48}$$

where

$$F^{MS}(I, C) = \lambda \int_{\Omega} (I - I_0)^2 dx + \nu \int_{\Omega \setminus C} |\nabla I|^2 dx + \mu |C| \tag{9.49}$$

and μ, ν are fixed parameters of weight for the different terms of the functional. For a minimum (I^*, C^*) , I^* is an optimal piecewise smooth approximation of the initial image I_0 , and C^* approximates the edges of I_0 . I^* will be smooth on $\Omega \setminus C$ and will have its discontinuities on C^* .

Theoretical results of existence of minimizers for this problem can be found in Mumford and Shah [40], Morel and Solimini [38] and De Giorgi et al. [26].

A reduced case can be obtained by restricting the segmentation image I to piecewise constant functions, i.e. for $c_i = \text{mean}(I_0)$ in Ω_i then $I \equiv c_i$ inside each Ω_i . In this event, for the special case $i = \{0, 1\}$ (that is the classical case of segmenting one object from the background image) the problem (9.48) becomes minimizing

$$E^{MS}(c_0, c_1, C) = \lambda \int_{\Omega_0} (I_0 - c_0)^2 dx + \lambda \int_{\Omega_1} (I_0 - c_1)^2 dx + \mu |C|. \tag{9.50}$$

We want now to use LS methods to solve this problem. LS methods are a very useful tool for computing evolving contours since they accommodate topological changes and allow to compute on a simple structured grid. For LS methods the curve C is defined as the zero-level set of a sufficiently regular function ϕ called level set function, i.e. $\phi(x) < 0$ for $x \in \Omega_0$ and $\phi(x) > 0$ for $x \in \Omega_1$.

In this formulation, denoting by $H(\phi)$ the usual Heaviside function, (9.50) becomes (see [17])

$$\begin{aligned} E^{VC}(c_0, c_1, \phi) &= \lambda \int_{\Omega} (I_0 - c_0)^2 H(\phi) dx + \lambda \int_{\Omega} (I_0 - c_1)^2 (1 - H(\phi)) dx \\ &\quad + \mu \int_{\Omega} |\nabla H(\phi)| dx. \end{aligned} \quad (9.51)$$

Considering H_{ϵ} and δ_{ϵ} two C^1 regular approximations of the Heaviside function H and of the delta function so that $H'_{\epsilon} = \delta_{\epsilon}$, we can write the Euler-Lagrange equation corresponding to the energy E^{VC} obtaining

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= \delta_{\epsilon}(\phi) \left[\mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \lambda (I_0 - c_0)^2 + \lambda (I_0 - c_1)^2 \right] \\ \text{where } c_0 &= \frac{\int_{\Omega} I(x) H(\phi) dx}{\int_{\Omega} H(\phi) dx}, \quad c_1 = \frac{\int_{\Omega} I(x) (1 - H(\phi)) dx}{\int_{\Omega} (1 - H(\phi)) dx}. \end{aligned} \quad (9.52)$$

This is a time dependent equation converging to a steady solution which is a minimizer of the functional (9.50).

A standard rescaling can be made, as in Zhao et al. [49], by replacing $\delta(\phi)$ by $|\nabla \phi|$. This time rescaling does not affect the steady state solution, but helps removing stiffness near the zero level sets of ϕ . Finally, we get the following nonlinear evolution equation:

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| \left[\mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \lambda (I_0 - c_0)^2 + \lambda (I_0 - c_1)^2 \right] \quad (9.53)$$

which is solved for t tending to $+\infty$, yielding the solution of the segmentation problem.

9.5.1 SL Scheme for Segmentation via the LS Method

In this section, we describe a SL approximation for Eq. (9.53). This equation has a first-order and a second-order term, which we approximate using SL schemes introduced in previous sections. We call $c(x)$ the speed of propagation of the first-order term,

$$c(x) = -\lambda (I_0 - c_0)^2 + \lambda (I_0 - c_1)^2, \quad (9.54)$$

and using (9.20) we obtain

$$\frac{\partial \phi}{\partial t} = \mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi| + c(x) \max_{a \in B(0,1)} \{a \cdot \nabla \phi\}. \quad (9.55)$$

As in every SL approximation, we want to build first an approximation of the directional derivative. This implies that the velocity $c(x)$ will appear directly inside the argument of the unknown function ϕ . For this particular problem, we have to take care of the change of sign of the velocity since will produce a sign change also in our scheme. We can rewrite (9.55) as

$$\frac{\partial \phi}{\partial t} = \begin{cases} \mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi| + \max_{a \in B(0,1)} \{c(x)a \cdot \nabla \phi\}, & \text{where } c(x) \geq 0 \\ \mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi| - \max_{a \in B(0,1)} \{-c(x)a \cdot \nabla \phi\}, & \text{where } c(x) < 0. \end{cases} \quad (9.56)$$

Next, we define $\phi_j^n = \phi(x_j, t_n)$ where $(x_j, t_n) \in \mathcal{G}_{\Delta x, \Delta t}$.

Using, the schemes (9.46) for the second order term, and (9.22) for the first order term, and explicit Euler finite differences for the temporal derivative, Eq. (9.56) becomes

$$\begin{aligned} \frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \mu & \frac{\left\{ \frac{1}{2} \Pi[\phi^n](x_j + \sigma_j^n \sqrt{\Delta t}) + \frac{1}{2} \Pi[\phi^n](x_j - \sigma_j^n \sqrt{\Delta t}) \right\} - \phi_j^n}{\Delta t} \\ & - \operatorname{sign}(c(x_j)) \min_{a \in B(0,1)} \frac{\Pi[\phi^n](x_j + ac(x_j)\Delta t) - \phi_j^n}{-\Delta t}, \end{aligned} \quad (9.57)$$

where we have used the fact that $-\max(-p) = \min(p)$ and $\min_{a \in B(0,1)}(a \cdot f) = \min_{a \in B(0,1)}(-a \cdot f)$ to simplify the notations.

Then, we get the fully discrete SL scheme:

$$\begin{aligned} \phi_j^{n+1} = \phi_j^n + \mu & \left\{ \frac{1}{2} \Pi[\phi^n](x_j + \sigma_j^n \sqrt{\Delta t}) + \frac{1}{2} \Pi[\phi^n](x_j - \sigma_j^n \sqrt{\Delta t}) - \phi_j^n \right\} \\ & + \operatorname{sign}(c(x_j)) \left\{ \min_{a \in B(0,1)} \{ \Pi[\phi^n](x_j + ac(x_j)\Delta t) \} - \phi_j^n \right\}, \end{aligned} \quad (9.58)$$

and coherently with (9.46) in the case of discrete space gradient $|D_j^n| \leq C\Delta x^s$.

We used this scheme to segment a variety of images, the results are shown in Figs. 9.12–9.14.

The most interesting feature of this approach is that the method is very rapid. While in classical finite difference schemes the information runs just one pixel for iteration, in this case we take information from a larger set. In particular to compute the value on the grid point x_j we use all the values on the grid nodes that are

Fig. 9.12 Segmentation:
 $\mu = 0.1, \lambda = 10^4$,
 two iterations

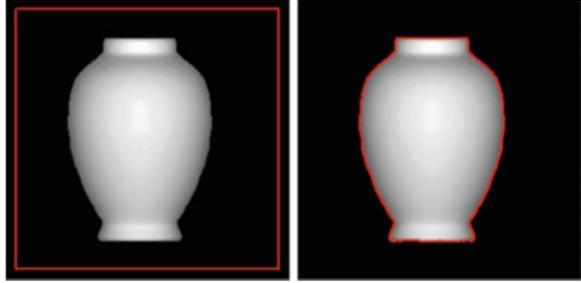


Fig. 9.13 Segmentation:
 $\mu = 1, \lambda = 0.3 * 10^4$,
 two iterations

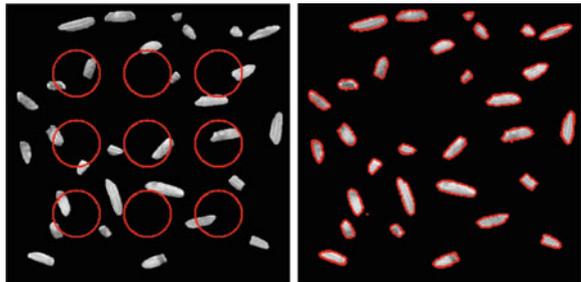
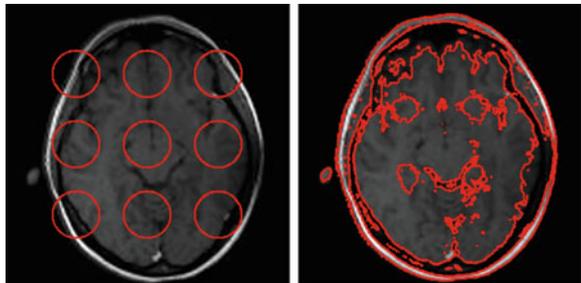


Fig. 9.14 Segmentation:
 $\mu = 1, \lambda = 0.3 * 10^4$,
 two iterations



contained in a ball centered in x_j with radius $c(x_j)$. This is the reason of the rapid convergence of the scheme. In our tests we found the correct segmentation of the image always after 3 or less iterations. However, this is an experimental observation, a detailed treatment of SL scheme performances on this problem is still missing.

9.6 The Motion Segmentation Problem

Optical flow is the motion field in an image sequence; this problem belongs to the more general problem of motion estimation. Motion estimation and segmentation are strongly related topics that can increase the performances from each other. In particular segmentation can help us to solve the classical ambiguity near motion boundaries.

Both topics are traditional in computer vision. Various approaches to optical flow computation have been suggested in [5], especially variational techniques based on modifications of the method of Horn and Schunck [36] have yielded very convincing results.

Also in segmentation, variational techniques perform well. As we showed in the previous section, segmentation can be obtained minimizing the functional (9.48). In recent years the functional optimization is computed using LS methods [43].

In [3] and in [10], the authors propose to compute optical flow coupling estimators for optical flow and segmentation problem in a Mumford and Shah-like functional. Due to the presence of unspecified discontinuities in the integration domain, the minimization of this kind of functionals is difficult and we are going to solve it using level set techniques.

We will use the studied semi-Lagrangian schemes for the Mean Curvature Motion in this situation, to compute the front propagation of a level set function. Later, we compare the results for this application, in terms of accuracy and speed. The variational model which we use, is based on the optical flow functional used in various works (for a short summary the dedicated section in [9]) and the segmentation model presented in [16].

Given two images $I_0(x), I_1(x) : \Omega \rightarrow \mathbb{R}$, we have to find, at each point $x \in \Omega$ the optical flow vector $w(x) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ that describes the shift of the pixel at x in the image I_0 to the new location $x + w$ in the image I_1 . Let us assume the following classical constraints: $I_0(x + w) = I_1(x)$ this is *grey level constancy*. It means that a pixel does not change its brightness from an image to the next. The second important principle is *gradient constancy*, an assumption that makes the model robust with respect to global additive illumination changes in the scene. So, our term of consistency with the data is

$$f_{of}(w) = k|I_0(x + w) - I_1(x)|^2 + \gamma|\nabla I_0(x + w) - \nabla I_1(x)|^2. \quad (9.59)$$

The minimization of an optical flow functional with only the image constancy assumptions is ill-posed, to overcome this problem, a smoothness term regularizing the flow field is considered. The typical smoothness term as in [36] is

$$s_{of}(w) = |\nabla w|^2. \quad (9.60)$$

Using this term, nevertheless, we introduce regularization of the field and the known ambiguities along the discontinuities. For this reason, we place the smoothness term in a Mumford-Shah-like functional that provide to add regularization only in the regions of regularity of the solution.

Finally, calling Γ the discontinuity set of the optical flow, and $|\Gamma|$ the length of the curve, we want to minimize the following functional

$$\begin{aligned} E(w, \Gamma) = & \int_{\Omega} k|I_0(x + w) - I_1(x)|^2 + \gamma|\nabla I_0(x + w) - \nabla I_1(x)|^2 dx \\ & + \mu \int_{\Omega \setminus \Gamma} |\nabla w|^2 dx + \nu |\Gamma|. \end{aligned} \quad (9.61)$$

This energy functional follows the principle of *motion competition* proposed for example in [23] and in [3], that was inspired by the work on segmentation by Vese and Chan [48]. This energy drives the interface of discontinuity and, simultaneously, makes an estimation of the optical flow in the different regions of smoothness.

In the optimization problem related to (9.61), if at the same time the optical flow, the number of regions separated by Γ and the position of the curve Γ are all unknown, then minimization can be hard. For this reason, in the next section we will consider the following simpler case. The motion field has an interface of discontinuity that divides the domain Ω in only two regions; inside these regions the motion field is constant. This choice is made for simplicity, and because, in this section, our aim is to present an easily understandable case. However, more general situations can be considered, e.g. in [10].

9.6.1 SL Scheme for the Motion Segmentation Problem

We want to deal with the functional (9.61) in a similar way as in segmentation, so we define a level set function ϕ such that the interface of discontinuity of the motion field Γ is the 0-level set of ϕ . Then, using the Euler-Lagrange equation and considering a time dependent equation converging to a steady solution which is a minimizer of the energy functional (9.61), we get the following evolutive equation:

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| \left[\nu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + [f_{of}(w^+) - f_{of}(w^-)] - \mu [s_{of}(w^+) - s_{of}(w^-)] \right] \quad (9.62)$$

where, w^+ and w^- are the constant values of w inside and outside Γ : we discretize this equation with the same technique as in (9.58). We obtain the following numerical scheme:

$$\begin{aligned} \phi_j^{n+1} = & \phi_j^n + \nu \left\{ \frac{1}{2} \Pi[\phi^n](x_j + \sigma_j^n \sqrt{\Delta t}) + \frac{1}{2} \Pi[\phi^n](x_j - \sigma_j^n \sqrt{\Delta t}) - \phi_j^n \right\} \\ & + \operatorname{sign}(c(x_j)) \left\{ \min_{a \in B(0,1)} \{ \Pi[\phi^n](x_j + ac(x_j)\Delta t) \} - \phi_j^n \right\}, \end{aligned} \quad (9.63)$$

where the first-order velocity term $c(x_j)$ is

$$c(x_j) = \left[f_{of}(w_j^+) - f_{of}(w_j^-) \right] - \mu \left[s_{of}(w_j^+) - s_{of}(w_j^-) \right]. \quad (9.64)$$

We used this scheme for two simple tests obtaining the results shown in Fig. 9.15 and 9.16. The first is a synthetic sequence where a ball is moving on a background, the second one is a small box taken from a more complicated real sequence (available on the site [47]). In these tests we have small displacements so we do not need a coarse-to-fine framework to capture larger ones.

Fig. 9.15 Motion segmentation: $\mu = 0.4$, $k = 0.5 * 10^5$, $\nu = 2$, $\gamma = 0.2$, convergence in five iterations

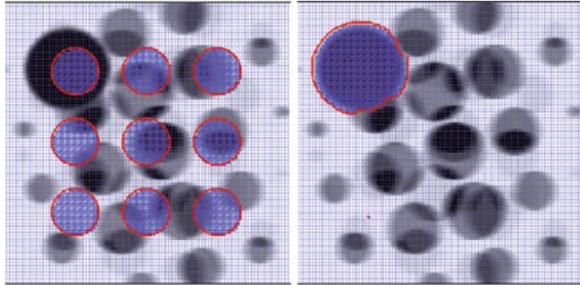
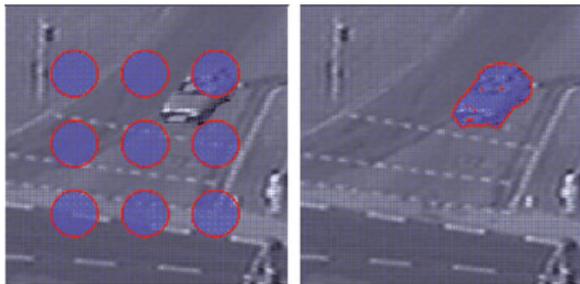


Fig. 9.16 Motion segmentation: $\mu = 0.4$, $k = 0.5 * 10^5$, $\nu = 2$, $\gamma = 0.2$, convergence in four iterations



9.7 Conclusions

As we have seen in the previous sections, the approximation schemes based on the semi-Lagrangian approximation of first and second order partial differential equations can be successfully applied to the solution of classical image processing problems. The main features of SL schemes are the following:

- They mimic the method of characteristics and give even in their simplest version a more accurate approximation with respect to first order finite difference schemes;
- High-order approximation schemes can be obtained coupling high-order methods for integrating back along the characteristics (e.g. Heun, Runge-Kutta) with high-order interpolation techniques (e.g. polynomial, essentially non-oscillatory (ENO), weighted ENO);
- They can be applied to structured and unstructured grids, so they are well suited also to problems where there is a need to use triangular grids as, for example, in computer vision;
- SL approximation allows to obtain accurate schemes even for second order nonlinear degenerate equations. Such approximations are fully explicit (i.e. with moderate computing cost) and allow for large time steps. It is important to note that the last property is crucial when we look to the asymptotic behavior for t tending to $+\infty$. Since this implies a time integration over a large interval, a large time step guarantees a fast convergence to the steady state;

- SL schemes have shown to be rather robust when applied to non smooth data, as noisy images. Moreover, in the time marching evolution there is less need for a re-initialization in order to keep the level sets and acceleration techniques (such as Fast Marching methods, [24, 45]) can be implemented.

A detailed analysis of these methods for first order model problems can be found in the book [32] and some of the results for second order problems can be found in [12, 14]. We conclude observing that there are still many open problems in the theory of SL schemes, e.g. in the construction and analysis of high-order approximations and acceleration methods (fast sweeping, fast marching). We are confident that these methods will be also useful for new challenging applications in image processing.

References

1. Alvarez, L., Lions, P.L., Morel, J.M.: Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.* **29**, 845–866 (1992)
2. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations of image processing. *Arch. Ration. Mech.* **123**, 199–257 (1993)
3. Amiaz, T., Kiryati, N.: Piecewise-smooth dense optical flow via level sets. *Int. J. Comput. Vis.* **68**(2), 111–124 (2006)
4. Barles, G.: *Solutions de Viscosité des Equations d’Hamilton–Jacobi*. Springer, New York (1998)
5. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. *Int. J. Comput. Vis.* **12**, 75–104 (1996)
6. Brenner S.C., Scott, L.R.: *The Mathematical Theory of Finite Element Methods*, 3rd edn. Springer, New York (2008)
7. Brent, R.: *Algorithms for Minimization Without Derivatives*. Pentice–Hall, Englewood Cliffs (1973)
8. Breuss, M., Cristiani, E., Durou, J.D., Falcone, M., Vogel, O.: Numerical algorithms for perspective shape from shading. *Kybernetika* **46**, 207–225 (2010)
9. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High Accuracy Optical Flow Estimation Based on a Theory for Warping. *Lecture Notes in Computer Science*, pp. 25–36. Springer, Berlin (2004)
10. Brox, T., Bruhn, A., Weickert, J.: Variational segmentation with level sets. In: *Computer Vision–ECCV, Graz*, pp. 471–483 (2006)
11. Camilli, F., Falcone, M.: An approximation scheme for the maximal solution of the shape-from-shading model. In: *Proceedings ICIP 96*, vol. I, pp. 49–52. IEEE, Piscataway (1996)
12. Carlini, E., Ferretti, R.: A semi-Lagrangian approximation for the AMSS model of image processing. submitted to *Applied Numerical Mathematics* (in press)
13. Carlini, E., Ferretti, R., Russo, G.: A weighted essentially non oscillatory, large time-step scheme for Hamilton Jacobi equations. *SIAM J. Sci. Comput.* **27**(3), 1071–1091 (2005)
14. Carlini, E., Falcone, M., Ferretti, R.: Convergence of a large time-step scheme for mean curvature motion. *Interface Free Bound.* **12**, 409–441 (2010)
15. Catté, F., Dibos, F., Koepfler, G.: A morphological scheme for mean curvature motion and applications to anisotropic diffusion and motion of level sets. *SIAM J. Numer. Anal.* **32**, 1895–1909 (1995)
16. Chan, T., Vese, L.: Active contours without edges. *IEEE Trans. Image Process.* **10**(2), 266–277 (2001)

17. Chan, T., Sandberg, B., Vese, L.: Active contours without edges for vector-valued images. *J. Vis. Commun. Image R.* **11**(2), 130–141 (2000)
18. Chen, Y.G., Giga, Y., Goto, S.: Uniqueness and existence of viscosity solutions of generalized mean curvature flow equation. *J. Diff. Geom.* **33**, 749–786 (1991)
19. Courant, R., Isaacson, E., Rees, M.: On the solution of nonlinear hyperbolic differential equations by finite differences. *Commun. Pure Appl. Math.* **5**, 243–255 (1952)
20. Courteille, F., Crouzil, A., Durou, J.D., Gurdjos, P.: Towards shape from shading under realistic photographic conditions, *Int. C. Patt. Recog.- ICPR 2004*, Cambridge, vol. 2, pp. 277–280 (2004)
21. Crandall, M.G., Lions, P.L.: Two approximations of solutions of Hamilton–Jacobi equations. *Math. Comput.* **43**, 1–19 (1984)
22. Crandall, M.G., Ishii, H., Lions, P.L.: User’s guide to viscosity solutions of second order partial differential equations. *Bull. Am. Math. Soc.* **27**, 1–67 (1992)
23. Cremers, D., Soatto, S.: Motion competition: a variational framework for piecewise parametric motion segmentation. *Int. J. Comput. Vis.* **63**, 249–265 (2005)
24. Cristiani, E., Falcone, M.: Fast semi-Lagrangian schemes for the eikonal equation and applications. *SIAM J. Numer. Anal.* **45**(5), 1979–2011 (2007)
25. Cristiani, E., Falcone, M., Seghini, A.: Numerical solution of the shape-from-shading problem. In: *Proceedings of Science POS (CSTNA2005) 008*, 1–17, Electronic Journal site <http://pos.sissa.it/>
26. De Giorgi, E.: New functionals in calculus of variations, nonsmooth optimization and related topics. In: *Proceedings of the Fourth Course of the International School of Mathematics*, Erice (1988)
27. Durou, J.D., Falcone, M., Sagona, M.: Numerical methods for shape from shading: a new survey with benchmarks. *Comput. Vis. Image Underst. Elsevier* **109**, 22–43 (2008)
28. Evans, L.C.: *Partial Differential Equations*. AMS, Providence (2010)
29. Evans, L.C., Spruck, J.: Motion of level sets by mean curvature. *Int. J. Diff. Geom.* **33**, 635–681 (1991)
30. Falcone, M.: The minimum time problem and its applications to front propagation. In: *Motion by Mean Curvature and Related Topics*. De Gruyter Verlag, Berlino (1994)
31. Falcone, M., Ferretti, R.: Convergence analysis for a class of high-order semi-Lagrangian advection schemes. *SIAM J. Numer. Anal.* **35**(3), 909–940 (1998)
32. Falcone, M., Ferretti, R.: *Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations*. SIAM, in preparation
33. Falcone, M., Sagona, M., Seghini, A.: A global algorithm for the Shape-from-Shading problem with black shadows. *Numerical Mathematics and Advanced Applications ENUMATH 2001*, pp. 503–512. Springer, Milano (2003)
34. Guichard, F., Morel, J.M.: *Image Analysis and P.D.E.s*. IPAM GBM Tutorial, March 27–April 6 (2001)
35. Horn, B.K.P., Brooks M.J. (eds.): *Shape from Shading*. MIT, Cambridge (1989)
36. Horn, B.K.P., Schunck, B.: Determining optical flow. *Artif. Intell.* **17**, 185–203 (1981)
37. Lions, P.L., Rouy, E., Tourin, A.: A viscosity solution approach to shape from shading. *Numer. Math.* **64**, 323–353 (1993)
38. Morel, J.M., Solimini, S.: Segmentation of images by variational methods: a constructive approach. *Rev. Mat. Univ. Compl. de Madr.* **1**, 169–182 (1988)
39. Mumford, D., Shah, J.: Boundary detection by minimizing functional. In: *Proceedings of the CVPR*, pp. 22–26. IEEE, Silver Spring (1985)
40. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* **XLII**, 577–685 (1989)
41. Oberman, A.M.: A convergent monotone difference scheme for motion of level sets by mean curvature. *Numer. Math.* **99**, 365–379 (2004)
42. Osher, S.J., Fedkiw, R.P.: *Level Set Methods and Dynamic Implicit Surfaces*. Applied Mathematical Sciences, vol. 153. Springer, New York (2003)

43. Osher, S.J., Sethian, J.A.: Front propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulation. *J. Comput. Phys.* **79**, 12–49 (1988)
44. Prados, E., Faugeras, O.: Perspective Shape-from-Shading and viscosity solutions. In: Proceedings of ICCV'03, pp. 826–831. IEEE, Los Alamitos (2003)
45. Sethian, J.A.: *Level Set Methods and Fast Marching Methods Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, Cambridge (1999)
46. Strikwerda, J.C.: *Finite Difference Schemes and Partial Differential Equations*, 2nd edn. SIAM, Philadelphia (2004)
47. Universität Kalsruhe. http://i21www.ira.uka.de/image_sequences/
48. Vese, L., Chan, T.: A multiphase level set framework for image segmentation using the Mumford and Shah model. *Int. J. Comput. Vis.* **50**, 271–293 (2002)
49. Zhao, H., Chan, T., Merriman, B., Osher, S.J.: A variational level set approach to multiphase motion. *J. Comput. Phys.* **127**, 179–195 (1996)