Variational Numerical Methods for Solving Nonlinear Diffusion Equations Arising in Image Processing

Angela Handlovičová and Karol Mikula

Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava, Slovakia

and

Fiorella Sgallari

Department of Mathematics, University of Bologna, Piazza di Porta S. Donato 5, 40127 Bologna, Italy

In this paper we give a general, robust, and efficient approach for numerical solutions of partial differential equations (PDEs) arising in image processing and computer vision. The well-established variational computational techniques, namely, finite element, finite volume, and complementary volume methods, are introduced on a common base to solve nonlinear problems in image multiscale analysis. Since they are based on principles like minimization of energy (finite element method) or conservation laws (finite and complemetary volume methods), they have strong physical backgrounds. They allow clear and physically meaningful derivation of difference equations that are local and easy to implement. The variational methods are combined with semi-implicit discretization in scale, which gives favorable stability and efficiency properties of computations. We show here L_∞-stability without any restrictions on scale steps. Our approach leads finally to solving linear systems in every discrete scale level, which can be done efficiently by fast preconditioned iterative solvers. We discuss such computational schemes for the regularized (in the sense of F. Catté *et al., SIAM J. Numer. Anal.* **129**, 1992, 182–193) Perona–Malik anisotropic diffusion equation (P. Perona and J. Malik, *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 1990, 629–639) and for nonlinear degenerate diffusion equation of mean curvature flow type studied by L. Alvarez *et al.* (*SIAM J. Numer. Anal.* **129**, 1992, 845–866).   © 2002 Elsevier Science (USA)

Key Words: image processing; nonlinear diffusion; Perona–Malik equation; mean curvature flow; level set equation; numerical solution; linear semi-implicit scheme; finite element method; finite volume method; complementary volume method.

1. INTRODUCTION

The aim of this paper is to present numerical schemes for solving nonlinear diffusion equations arising in image processing and computer vision. The schemes are based on linear semi-implicit approximations in scale and variational methods in space. For our presentation we have chosen two representative geometry-driven diffusion models: the regularized (in the sense of Catté *et al.* [1]) Perona–Malik anisotropic diffusion [2] Eq. (1) and the nonlinear degenerate diffusion equation of mean curvature flow type (2) suggested by Alvarez *et al.* [3]. Both models represent a similar view to image selective smoothing that preserves the edge positions in the multiscale analysis [4, 5]. We consider the nonlinear partial differential equations

$$u_t - \nabla \cdot (g(|\nabla G_\sigma * u|)\nabla u) = f(u^0 - u), \tag{1}$$

$$u_t - g(|\nabla G_\sigma * u|)|\nabla u|\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right) = 0, \tag{2}$$

where $u(t, x)$ is an unknown function defined in $Q_T \equiv I \times \Omega$. The equations are accompanied by zero Neumann boundary conditions and the initial condition

$$\frac{\partial u}{\partial \nu} = 0 \quad \text{on } I \times \partial\Omega, \tag{3}$$

$$u(0, x) = u^0(x) \quad \text{in } \Omega, \tag{4}$$

where ν is the unit normal to the boundary of Ω. We assume that $\Omega \subset I\!\!R^d$ is a bounded rectangular domain, $I = [0, T]$ is a scaling interval,

$$g: I\!\!R_0^+ \to I\!\!R^+ \text{ is a nonincreasing function, } g(\sqrt{s}) \text{ is smooth,} \\ g(0) = 1, \text{ and we admit } g(s) \to 0 \text{ for } s \to \infty, \tag{5}$$

$$G_\sigma \in C^\infty(I\!\!R^d) \text{ is a smoothing kernel (e.g., Gauss function),} \\ \int_{I\!\!R^d} G_\sigma(x)\,dx = 1, \int_{I\!\!R^d} |\nabla G_\sigma|\,dx \le C_\sigma, \tag{6}$$

$$G_\sigma(x) \to \delta_x \text{ for } \sigma \to 0, \delta_x \text{ is the Dirac measure at point } x, \\ f \text{ is a Lipschitz continuous, nondecreasing function, } f(0) = 0, \tag{7}$$

$$u^0 \in L_\infty(\Omega), \tag{8}$$

and

$$\nabla G_\sigma * u = \int_{I\!\!R^d} \nabla G_\sigma(x - \xi)\tilde{u}(\xi)\,d\xi, \tag{9}$$

where $\tilde{u}$ is an extension of u to $I\!\!R^d$ given by a periodic reflection through the boundary of Ω.

In image processing, equations of the type (1), (2) arise in nonlinear data filtration, edge detection, and image enhancement and restoration [1–3]. The initial condition $u^0(x)$ represents a gray-level intensity function of the processed image. The solution $u(t, x)$ of

either (1) or (2) represents a family of scaled (filtered, smoothed) versions of $u^0(x)$; t is understood as an abstract parameter called scale. In a sense, the processing of u^0 by an evolutionary PDE like (1) or (2) represents an embedding of the initial image into the nonlinear scale space.

Equation (1) represents a slight modification of the well-known Perona–Malik equation [2, 6], called also anisotropic diffusion in the computer vision community. It selectively diffuses an image in the regions where the signal is of a constant mean, in contrast to those regions where the signal changes its tendency. Such a diffusion process is governed by the shape of the diffusion coefficient of (1) given by the function g and by its dependence on ∇u, which is in a sense an edge indicator [2]. In the original Perona–Malik formulation, ∇u stands in the place of $\nabla G_\sigma * u$. However, in such a case, it can behave locally like a backward heat equation, which is an ill-posed problem. One way to overcome this mathematical disadvantage was proposed by Catté *et al.* [1]. They introduced for the value of the diffusion coefficient the convolution with Gaussian kernel G_σ. This slight modification allowed them to prove the existence and uniqueness of a weak solution for the modified equation and to keep the practical advantages of the original formulation. Moreover, the usage of the Gaussian gradient makes the process more stable in the presence of noise. It has also made explicit the regularization included implicitly in numerical realizations of the Perona–Malik equation.

Equation (2) is related to the so-called morphological multiscale analysis, and it respects significantly the image geometrical features (like level lines of the image intensity corresponding to the silhouettes of objects in the image) in the smoothing process given by diffusion. Provided $g(s) \equiv 1$, (2) is called the level set equation proposed by Osher and Sethian [7, 8] for the computation of moving fronts in the dynamics of phase interfaces in thermomechanics. The level set equation moves each level set (namely, level line in two-dimensional (2D) space and level surface in three-dimensional (3D) space) of u with the velocity proportional to its normal mean curvature field. It also fulfills the morphological principle: if u is a solution, then, for any nondecreasing function φ, $\varphi(u)$ is a solution as well. This contrast invariant property has a large significance in the theory of image processing [4]. Such property also means that all level sets move independently of each other, they diffuse only intrinsically (as manifolds, see, e.g., [9–13]) and there is no diffusion across them in the normal direction. This idea was used in [3], where model (2) has been suggested for computational image and shape analysis. From the practical point of view, applying just the level set equation (i.e., $g \equiv 1$) to initial image yields the intrinsic silhouette smoothing [14]. On the other hand, Eq. (2) can be used successively for image selective smoothing with preserving edge positions in a way similar to Eq. (1). The Perona–Malik function $g(s)$ depending on $|\nabla G_\sigma * u|$ is used to strongly slow the motion of edges of silhouettes. The regions between edges are smoothed by the mean curvature flow.

A general view on (1) and (2) without the convolution terms is given in [4]. The nonlinear diffusion term $\nabla \cdot (g(|\nabla u|)\nabla u)$ can be rewritten as

$$\nabla \cdot (g(|\nabla u|)\nabla u) = g(|\nabla u|)u_{\xi\xi} + H'(|\nabla u|)u_{\eta\eta}, \tag{10}$$

where $H(s) = sg(s)$ and ξ, η are the tangential and orthogonal vectors to the level line, respectively. From this form, one can simply see how diffusion works along and across the image silhouettes with different choices of g. There is always a positive, but possibly strongly slowed-down (depending on the shape of g), diffusion along the level lines. Across

the level line there can be a forward diffusion (when $G'(s)$ is positive), zero diffusion (e.g., in the Rudin–Osher–Fatemi model [15] for the total variation denoising, or in the mean curvature flow equation in the level set formulation), or a backward diffusion (in the original Perona–Malik model [2, 6]).

The rest of the paper is organized as follows. In Section 2 we present a semi-implicit approach to scale discretization of Eqs. (1) and (2). It leads to the solving of linear elliptic equations and, at the same time, it gives good stability properties of numerical approximations of these strongly nonlinear problems. In both cases, we present methods for deriving such stability estimates. In Section 3, we present space discretizations of arising elliptic equations by the means of variational techniques. The presented discretizations are based on the integral (weak, variational) formulations of the boundary value problems. Variational methods have strong physical backgrounds since they are based on principles like minimization of energy (finite element method) or conservation laws (finite and complemetary volume methods). They allow a clear and physically meaningful derivation of difference equations that is local and easy to implement. We should note that the semi-implicit approach based on special finite difference approximations was already used in [1] and [3], for the finite element method it was given in [16], and for the finite volume method it was given in [17]. In the discussion of the numerical results in Section 4 we concentrate on the complementary volume method, but we have also found it useful to collect principles of other discretization methods in one paper. In order to achieve strong efficiency of the methods, we also accompany the linear semi-implicit approximations with preconditioned iterative solvers. In Section 5 we give some concluding remarks.

Remark. In the next paragraphs, we will use standard notations for functional spaces $L_p(\Omega)$, $p \geq 1$, $C^k(\bar{\Omega})$ and by V we denote the Sobolev space $W^{1,2}(\Omega)$ of $L_2(\Omega)$ functions with square integrable weak derivates (see, e.g., [18]). The absolute value will be denoted by $|\cdot|$ and the norm will be denoted by $\|\cdot\|$ with a subscript given by the corresponding functional space.

2. SEMI-DISCRETIZATIONS IN SCALE

In this section we study semi-discretizations in scale of the problems given by (1) and (2). We discretize the scaling interval $[0, T]$. Choosing $N \in I\!N$ we obtain the length of uniform discrete scale step $\tau = \frac{T}{N}$. We replace the scale derivative in (1), (2) by backward difference. The nonlinear terms of the equations are treated from the previous scale step while the linear terms are considered on the current scale level—this means semi-implicitness of the method. Let us start with an approximation in scale of Eq. (1) (see also [1, 16, 17]).

Semi-discrete linear scheme for solving Eq. (1). Let $N \in I\!N$, $\tau = \frac{T}{N}$ and $\sigma > 0$ be fixed numbers and u^0 be given by (4). For every $n = 1, \ldots N$, we look for a function u^n, a solution of the equation

$$\frac{u^n - u^{n-1}}{\tau} - \nabla \cdot (g(|\nabla G_\sigma * u^{n-1}|)\nabla u^n) = f(u^0 - u^{n-1}). \tag{11}$$

For this linear scheme we have guaranteed existence of a weak solution in every discrete scale step, and moreover we will prove in the next proposition an important L_∞-stability property for the solutions of subsequent elliptic equations. On one hand, this means that

the scheme is safe from spurious oscilations; on the other hand, since an image can be represented by a family of its level lines, we have guaranteed that no new level lines are created in multiscale analysis (provided $f \equiv 0$) as at least they are controlled by means of a Lipschitz constant of the right-hand side f and by the length T of processing. Any reasonable fully discrete scheme should also satisfy this L_∞-stability property.

PROPOSITION 1.   *There exists a unique weak solution u^n of (11) for every $n = 1, \ldots, N$ and there exists a positive constant C such that*

$$\|u^n\|_{L_\infty(\Omega)} \leq C \|u^0\|_{L_\infty(\Omega)}. \tag{12}$$

The constant C depends on T and the Lipschitz constant L of f only. If $f \equiv 0$, then $C = 1$.

Proof.   A function $u^n \in V$ is a weak solution of (11) if it fullfils the integral identity

$$\int_\Omega \frac{u^n - u^{n-1}}{\tau} v\, dx + \int_\Omega g(|\nabla G_\sigma * u^{n-1}|)\nabla u^n \cdot \nabla v\, dx = \int_\Omega f(u^0 - u^{n-1})v\, dx, \quad \forall v \in V. \tag{13}$$

The properties of convolution and the Lax–Milgram theorem guarantee the existence of the unique solution $u^n \in V$ of (13), which is moreover in $L_\infty(\Omega)$ (see [1, 16, 19]). Let us denote

$$g^{n-1} := g(|\nabla G_\sigma * u^{n-1}|) \geq v > 0. \tag{14}$$

In order to get the uniform L_∞-estimate of the proposition we proceed as follows. Since $u^n \in L_\infty(\Omega)$, we have that the pth power $(u^n)^p \in V$. Using $v = (u^n)^p \tau$ as the test function in (13) for p odd, we obtain

$$\int_\Omega |u^n|^{p+1}\, dx + \tau \int_\Omega g^{n-1} p|u^n|^{p-1}|\nabla u^n|^2\, dx$$

$$\leq \int_\Omega |u^{n-1}||u^n|^p\, dx + L\tau \int_\Omega |u^0||u^n|^p\, dx + L\tau \int_\Omega |u^{n-1}||u^n|^p\, dx.$$

Then due to the positivity of the second term on the left-hand side we have

$$\int_\Omega |u^n|^{p+1}\, dx \leq (1 + L\tau) \int_\Omega |u^{n-1}||u^n|^p\, dx + L\tau \int_\Omega |u^0||u^n|^p\, dx. \tag{15}$$

Next we apply Young's inequality in the form

$$ab \leq \frac{1}{p+1} a^{p+1} + \frac{p}{p+1} b^{\frac{p+1}{p}}$$

to the last term of (15) with $a = \varepsilon|u^0|$, $b = \frac{1}{\varepsilon}|u^n|^p$. This gives us

$$\int_\Omega |u^n|^{p+1}\, dx \leq (1 + L\tau) \int_\Omega |u^{n-1}||u^n|^p\, dx + L\tau \frac{\varepsilon^{p+1}}{p+1} \int_\Omega |u^0|^{p+1}\, dx$$

$$+ L\tau \frac{p}{p+1} \frac{1}{\varepsilon^{\frac{p+1}{p}}} \int_\Omega |u^n|^{p+1}\, dx.$$

Let $\varepsilon = (L\frac{p}{p+1})^{p/p+1}$. Subtracting the last term we get

$$(1 - \tau) \int_\Omega |u^n|^{p+1}\, dx \le (1 + L\tau) \int_\Omega |u^{n-1}||u^n|^p\, dx + L\tau \frac{\varepsilon^{p+1}}{p+1} \int_\Omega |u^0|^{p+1}\, dx.$$

Since for $\tau \le \frac{1}{2}$ we have $\frac{1}{1-\tau} \le (1 + 2\tau)$, we obtain

$$\int_\Omega |u^n|^{p+1}\, dx \le (1 + L\tau)(1 + 2\tau) \int_\Omega |u^{n-1}||u^n|^p\, dx + (1 + 2\tau)L\tau \frac{\varepsilon^{p+1}}{p+1} \int_\Omega |u^0|^{p+1}\, dx.$$

If we use Young's inequality again but with $a = (1 + L\tau)(1 + 2\tau)|u^{n-1}|$, $b = |u^n|^p$ in the first term on the right-hand side, we get

$$\int_\Omega |u^n|^{p+1}\, dx \le \frac{(1 + L\tau)^{p+1}(1 + 2\tau)^{p+1}}{p+1} \int_\Omega |u^{n-1}|^{p+1}\, dx$$

$$+ \frac{p}{p+1} \int_\Omega |u^n|^{p+1}\, dx + (1 + 2\tau)L\tau \frac{\varepsilon^{p+1}}{p+1} \int_\Omega |u^0|^{p+1}\, dx.$$

Multiplying the previous inequality by $p + 1$ and subtracting the second term on the right-hand side we obtain the recurrent relation

$$\int_\Omega |u^n|^{p+1}\, dx \le (1 + L\tau)^{p+1}(1 + 2\tau)^{p+1} \int_\Omega |u^{n-1}|^{p+1}\, dx + (1 + 2\tau)L\tau \varepsilon^{p+1} \int_\Omega |u^0|^{p+1}\, dx.$$

From there we have

$$\int_\Omega |u^n|^{p+1}\, dx \le (1 + L\tau)^{n(p+1)}(1 + 2\tau)^{n(p+1)}(1 + (1 + 2\tau)LT\varepsilon^{p+1}) \int_\Omega |u^0|^{p+1}\, dx.$$

Take the $(p + 1)$-th root in the previous inequality and let $p \to \infty$ (n is fixed and finite). Since $(1 + x)^n \le e^{nx}$, we have

$$\|u^n\|_\infty \le e^{(2+L)T}(1 + L)\|u^0\|_\infty \le C\|u^0\|_\infty,$$

where the constant C depends on T and the Lipschitz constant L of f. Provided $f(s) \equiv 0$, one can review the above arguments to see that $C = 1$ without any restriction on the time step τ. ∎

Despite Eq. (1), Eq. (2) is not written in a divergence form. For partial derivatives of second order, as is usual in variational methods, we would like to use integration by parts or the divergence theorem to get an integral formulation. Thus, first we move the term in front of the divergence to the time derivative and then we write a semi-implicit discretization of (2) in scale.

Semi-discrete linear scheme for solving Eq. (2). Let $N \in \mathbb{N}$, $\tau = \frac{T}{N}$ and $\sigma > 0$ be fixed numbers and u^0 be given by (4). For every $n = 1, \dots N$, we look for a function u^n, a solution of the equation

$$\frac{1}{g(|\nabla G_\sigma * u^{n-1}|)|\nabla u^{n-1}|} \frac{u^n - u^{n-1}}{\tau} - \nabla \cdot \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) = 0. \tag{16}$$

At this place, let us note another possible approach for the scale discretization of (2) based on an interesting approximation of the degenerate diffusion term given in [20]. Following the idea of Walkington one could, instead of (16), use the implicit (nonlinear) semi-discretization

$$\frac{1}{g(|\nabla G_\sigma * u^{n-1}|)|\nabla u^{n-1}|}\frac{u^n - u^{n-1}}{\tau} - 2\nabla \cdot \left(\frac{\nabla u^n}{|\nabla u^n| + |\nabla u^{n-1}|}\right) = 0. \qquad (17)$$

Walkington considered averaging the gradient terms from the previous and current scale steps in the denominator of the divergence term to get an important $W^{1,1}$ estimate, i.e., an estimate on the decay of the total variation of discrete solutions. This estimate is a basic property of the flow by mean curvature and of a solution of the level set equation as well [20] and can be interpreted as a curve-shortening property [9, 11, 21]. Any reasonable numerical approximation should respect this fact. Following [20], one can multiply (17) by $u^n - u^{n-1}$ and integrate it over Ω. Then using integration by parts and zero Neumann boundary conditions one gets

$$\int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}\,dx + 2\int_\Omega \frac{\nabla u^n \cdot (\nabla u^n - \nabla u^{n-1})}{|\nabla u^{n-1}| + |\nabla u^n|}\,dx = 0, \qquad (18)$$

where short notation (14) has been used. Using the relation

$$2a(a - b) = a^2 - b^2 + (a - b)^2, \qquad (19)$$

where a, b are arbitrary real numbers, and by a simple manipulations related to the sum in the denominator, we get

$$\int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}\,dx + \int_\Omega \frac{|\nabla u^n|^2 - |\nabla u^{n-1}|^2}{|\nabla u^{n-1}| + |\nabla u^n|}\,dx + \int_\Omega \frac{|\nabla u^n - \nabla u^{n-1}|^2}{|\nabla u^{n-1}| + |\nabla u^n|}\,dx$$

$$= \int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}\,dx + \int_\Omega \frac{|\nabla u^n - \nabla u^{n-1}|^2}{|\nabla u^{n-1}| + |\nabla u^n|}\,dx$$

$$+ \left(\int_\Omega |\nabla u^n|\,dx - \int_\Omega |\nabla u^{n-1}|\,dx\right) = 0, \qquad (20)$$

which means that

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^{n-1}\|_{L_1(\Omega)} \qquad (21)$$

and by recursion

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^0\|_{L_1(\Omega)}, \quad 1 \le n \le N. \qquad (22)$$

This represents the important stability property of this nonlinear scheme.

However, the previous scheme leads in each discrete scale level (after any spatial discretization) to solving a nonlinear system of equations. This is a rather noneffient approach. In order to have convergence, which is however very slow, one must use fixed point-like nonlinear iterations; faster possibilities like Newton's method have no guarantees for convergence [20] and are rather complicated from the implementation point of view.

In our contribution we get decay of the total variation of solutions in subsequent scale steps (i.e., the basic property (22)) also for the scheme (16). The linear scheme (16) is much more simple and efficient. Since it is linear, it allows the use of the fast preconditioned iterative linear solvers at every scale level (we present some CPU time statistics in Section 4). In order to get (22) for our scheme, we proceed in the following way. We multiply (16) by $u^n - u^{n-1}$, then integrating it over Ω and using (19) we get

$$\int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}\, dx + \frac{1}{2}\int_\Omega \frac{|\nabla u^n|^2 - |\nabla u^{n-1}|^2 + |\nabla u^n - \nabla u^{n-1}|^2}{|\nabla u^{n-1}|}\, dx = 0. \quad (23)$$

Since

$$|\nabla u^n - \nabla u^{n-1}|^2 = (|\nabla u^n| - |\nabla u^{n-1}|)^2 + \left(\frac{\nabla u^n}{|\nabla u^n|} - \frac{\nabla u^{n-1}}{|\nabla u^{n-1}|}\right)^2 |\nabla u^n||\nabla u^{n-1}|, \quad (24)$$

we get

$$\int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}\, dx + \frac{1}{2}\int_\Omega \frac{|\nabla u^n|^2 - |\nabla u^{n-1}|^2 - (|\nabla u^n| - |\nabla u^{n-1}|)^2}{|\nabla u^{n-1}|}\, dx$$
$$+ \int_\Omega \frac{(|\nabla u^n| - |\nabla u^{n-1}|)^2}{|\nabla u^{n-1}|}\, dx + \frac{1}{2}\int_\Omega \left(\frac{\nabla u^n}{|\nabla u^n|} - \frac{\nabla u^{n-1}}{|\nabla u^{n-1}|}\right)^2 |\nabla u^n|\, dx = 0.$$

Due to positivity of other terms we get for the second term

$$\int_\Omega \frac{|\nabla u^n||\nabla u^{n-1}| - |\nabla u^{n-1}|^2}{|\nabla u^{n-1}|}\, dx \le 0, \quad (25)$$

which gives

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^{n-1}\|_{L_1(\Omega)}, \quad (26)$$

and thus

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^0\|_{L_1(\Omega)}, \quad 1 \le n \le N, \quad (27)$$

which is the desired $W^{1,1}$ estimate for the linear semi-implicit scheme (16).

3. SPACE DISCRETIZATIONS

A discrete image is based on the structure of pixels/voxels with rectangular shape (but it is not necessary for the methods presented here) in general. We will use this image structure to create a computational grid for the spatial discretization methods. Concerning a relation of the computational grid to the pixel structure, there will be a difference between the finite element and the complementary volume methods on one side and the finite volume method on other side. The difference is related to a type of approximation of a solution of a partial differential equation assumed in those methods. While in the finite volume method the approximation of the solution is assumed to be piecewise constant [22], in the complementary volume and finite element methods it is assumed to be continuous piecewise linear. Thus, in the finite volume method we will take for the computational grid directly

the pixel structure of the image. The initially given and subsequently computed values of discrete intensity are considered as approximations of the average of the continuous intensity function on the pixels. On the other hand, in the finite element and complementary volume methods, the initially given values of discrete intensity and also the computed ones are considered as approximations of the continuous intensity function in the centers of the pixels. The centers of the pixels then correspond to the nodes of the finite element or the complementary volume triangulation. We can get such triangulation by connecting the centers of the pixels by a new rectangular mesh and then dividing each rectangle into two triangles (or six tetrahedra in 3D). It also means that in these two methods the computational domain Ω is given as the union of all triangles constructed in such a way (Ω thus corresponds to the image domain minus the outer half of each boundary pixel)—see Fig. 1. Let us note that a splitting of each pixel into the two triangles, as depicted in Fig. 1, is not the only possibility. The orientation of the triangles can change locally, e.g., following an edge direction, or it can be given by a refinement procedure in a bisection algorithm [23]. We will assume that the constructed triangulation has no interior angle larger than $\pi/2$. Then, for the complementary volume method we will construct a dual mesh. This dual mesh will again, in a sense, copy the pixel structure of the image. Let us note that in the finite element method we will use just the triangulation, in the complementary volume method we use both triangulation and the dual mesh, and in the finite volume method we use just the pixel structure of the image corresponding to the dual mesh.

Now, we will define some quantities that will be used in the descriptions of fully discrete schemes (see also [20]). Let us assume that we have been given a triangulation $\mathcal{T}_h$ (e.g., given by a previous construction). The dual mesh will consist of cells V_i (also called
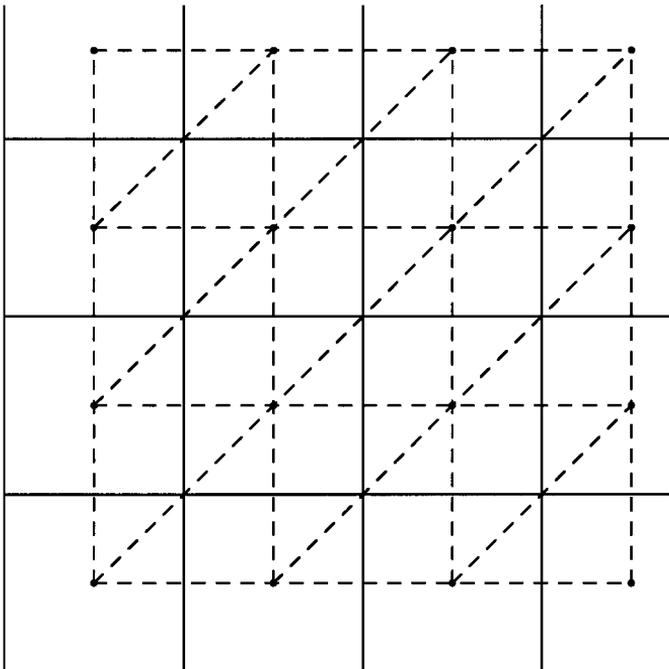
FIG. 1. The image pixels (solid lines) correspond to the finite volume mesh and to the dual mesh for the complementary volume method. Triangulation (dashed lines) corresponds to the finite element and complementary volume methods, with nodes (round points) corresponding to the centers of the pixels.

complementary volumes, control volumes, or co-volumes) associated with the ith node of triangulation $\mathcal{T}_h$, $i = 1, \ldots, M$. The co-volume V_i is bounded by the lines (planes in 3D) that bisect and are perpendicular to the edges emanating from the node. We will denote the edge of $\mathcal{T}_h$ connecting the ith node to the jth by σ_{ij} and its length by h_{ij}. We denote by $\mathcal{E}_{ij}$ the set of simplices having σ_{ij} as an edge, i.e., $\mathcal{E}_{ij} = \{T \in \mathcal{T}_h \mid \sigma_{ij} \subset T\}$. Let e_{ij} denote the co-edge (co-plane) that is the perpendicular bisector of σ_{ij} and x_{ij} be a point of intersection of e_{ij} and σ_{ij}. For each $T \in \mathcal{E}_{ij}$, let c_{ij}^T be the length (area) of the portion of e_{ij} that is in T, i.e., $c_{ij}^T = |e_{ij} \cap T|$. Let $\mathcal{N}_i$ be the set of simplices that have the ith node as a vertex, and for each node of $\mathcal{T}_h$ let C_i denote the set of nodes connected to the ith node by an edge. Given a triangulation $\mathcal{T}_h$, we define the set $V_h \subset V$ of piecewise linear finite elements, i.e., $V_h = V_h(\mathcal{T}_h) := \{v \in C^0(\bar{\Omega}) \mid v_{|T} \in \mathcal{P}_1 \text{ for all } T \in \mathcal{T}_h\}$. For any $v_h \in V_h$ we will use the notation $v_i := v_h(x_i)$, where x_i is the ith node of the triangulation. Let $u_h^0 = I_h(u^0) \in V_h(\mathcal{T}_h)$ be the nodal interpolant of u^0. This will be the initial function for the finite element and complementary volume methods. In the finite volume method we denote by $\bar{u}_i$ the representative value for the cell (the bar indicates that we mean the average value in the cell and not a nodal value). The initial condition for the finite volume scheme is then assumed to be

$$\bar{u}_i^0 = \frac{1}{|V_i|} \int_{V_i} u^0(x)\, dx, \quad i = 1, \ldots, M. \tag{28}$$

Before going to the spatial discretizations we remark on the realization of the convolution included in the evaluation of the Perona–Malik function g either in (11) or in (16). We use two strategies. The first is the following: Using the Gauss function as the smoothing kernel G_σ, one can replace the term $G_\sigma * u^{n-1}$ by solving the linear heat equation for time σ with the initial condition given by u^{n-1}. This linear equation can be solved numerically at the same grid by just one implicit step with length σ. Thus, as a realization of convolution we look for a function u^c that is a solution of the heat equation discretized in time by the backward Euler method with step σ

$$\frac{u^c - u^{n-1}}{\sigma} = \Delta u^c, \tag{29}$$

where Δ denotes the Laplace operator. This strategy is very suitable for the finite element and complementary volume methods, since in this case, the numerical solution u^c of (29) is piecewise linear on triangles, its gradients are constant, and thus we can simply evaluate the Perona–Malik function on every triangle. Moreover, since the coefficients of Eq. (11) are then constant on every triangle, we have also guaranteed L_∞-stability for the fully discrete finite element scheme (see, e.g., [24]). In the finite volume method we can use another approach. Since we consider piecewise constant approximations, we can replace the convolution (integral) by a sum over pixels. For the gradient of the convolution term at any point x we get

$$\nabla G_\sigma * u^{n-1}(x) = \sum_r \bar{u}_r^{n-1} \int_{V_r} \nabla G_\sigma(x - s)\, ds, \tag{30}$$

where we use the convolution derivative property

$$\frac{\partial}{\partial x_i}(G_\sigma * u^{n-1}(x)) = \frac{\partial G_\sigma}{\partial x_i} * u^{n-1}(x)$$

and

$$\frac{\partial G_\sigma}{\partial x_i} * u^{n-1}(x) = \int_{\mathbb{R}^d} \frac{\partial G_\sigma}{\partial x_i}(x-s)u^{n-1}(s)\,ds = \sum_r \bar{u}_r^{n-1} \int_{V_r} \frac{\partial G_\sigma}{\partial z_i}(x-s)\,ds.$$

The sum in (30) is evaluated over control volumes V_r, which are around x. If we choose a compactly supported smoothing kernel with support in a ball $B_{\sigma^*}(0)$ with radius σ^*, e.g., the function

$$G_\sigma(x) = \frac{1}{Z} e^{\frac{|x|^2}{|x|^2 - \sigma^2}},$$

where the constant Z is chosen so that G_σ has the unit mass, the sum is restricted only to the control volumes contained in $B_{\sigma^*}(x)$, the ball centered at x. The coefficients $\int_{V_r} \nabla G_\sigma(x - s)\,ds$ in (30) can be computed in advance using a computer algebra system, e.g., Mathematica [25]. The same situation arises when we use the Gauss function and we consider as the ball $B_\sigma^*(0)$ a "numerical support" of the Gauss function (i.e., we consider a domain in which the values of the Gauss function are above some treshold given related to the computer precision). Then, again, just a finite sum in (30) is evaluated.

3.1.   Finite Element Discretization

To describe the ideas of the finite element space discretization, let us consider Eq. (1) [16, 23]. Let τ, σ be given numbers. Before the discretization we use approach (29) for the realization of the convolution and then we write (11) into a couple of integral identities (weak formulations)

$$\int_\Omega u^n v\,dx + \tau \int_\Omega g(|\nabla u^c|)\nabla u^n \nabla v\,dx = \int_\Omega (u^{n-1} + \tau f(u^0 - u^{n-1}))v\,dx, \quad (31)$$

$$\int_\Omega u^c v\,dx + \sigma \int_\Omega \nabla u^c \nabla v\,dx = \int_\Omega u^{n-1} v\,dx \quad (32)$$

that hold for all $v \in V$ [23]. Then, at each scale level n, we look for a continuous piecewise linear function $u_h^n \in V_h(\mathcal{T}_h)$ satisfying

$$\int_\Omega u_h^n v_h\,dx + \tau \int_\Omega g(|\nabla u_h^c|)\nabla u_h^n \nabla v_h\,dx = \int_\Omega (u_h^{n-1} + \tau f(u^0 - u_h^{n-1}))v_h\,dx \quad (33)$$

for $v_h \in V_h(\mathcal{T}_h)$ with $u_h^c \in V_h(\mathcal{T}_h)$ being the solution of

$$\int_\Omega u_h^c v_h\,dx + \sigma \int_\Omega \nabla u_h^c \nabla v_h\,dx = \int_\Omega u_h^{n-1} v_h\,dx, \quad \forall v_h \in V_h(\mathcal{T}_h). \quad (34)$$

Considering the standard Lagrangian bases functions $\varphi_j \in V_h(\mathcal{T}_h)$, $j = 1, \ldots, M$, given by $\varphi_j(x_i) = \delta_{ji}$ (Kronecker delta) for all nodes x_i, $i = 1, \ldots, M$ of $\mathcal{T}_h$, the functions u_h^n, u_h^c

are given by

$$u_h^n = \sum_{i=1}^{M} u_i^n \varphi_i, \quad u_h^c = \sum_{i=1}^{M} u_i^c \varphi_i.$$

Then using $v_h = \varphi_j$, $j = 1, \ldots, M$ as test function in (33)–(34), we get two Ritz–Galerkin systems of linear equations for the nodal values u_i^n, u_i^c, $i = 1, \ldots, M$ of u_h^n, u_h^c, respectively,

$$\sum_{i=1}^{M} \left(\int_\Omega \varphi_i \varphi_j \, dx + \tau \int_\Omega g(|\nabla u_h^c|) \nabla \varphi_i \nabla \varphi_j \, dx \right) u_i^n = \int_\Omega \left(u_h^{n-1} + \tau f \left(u^0 - u_h^{n-1} \right) \right) \varphi_j \, dx \tag{35}$$

$$\sum_{i=1}^{M} \left(\int_\Omega \varphi_i \varphi_j \, dx + \sigma \int_\Omega \nabla \varphi_i \nabla \varphi_j \, dx \right) u_i^c = \int_\Omega u_h^{n-1} \varphi_j \, dx, \tag{36}$$

for every $j = 1, \ldots, M$. Thus, we need to solve two linear systems with the matrices $\mathbf{M} + \tau \mathbf{A}(g(|\nabla u_h^c|))$, $\mathbf{M} + \sigma \mathbf{A}(1)$, respectively, where $\mathbf{M}_{j,i} = \int_\Omega \varphi_i \varphi_j \, dx$ is the so-called mass matrix and $\mathbf{A}(w)_{j,i} = \int_\Omega w \nabla \varphi_i \nabla \varphi_j \, dx$ is the stiffness matrix, which are symmetric and positive definite. Thus, the discrete solutions can be found efficiently by preconditioned conjugate gradient methods (CPU times for finite element discretization are comparable with those given in Section 4 for the complementary volume method). It is also usual to use the so-called lumped (diagonalized) mass matrix $\mathbf{M}_{j,i} = \sum_{k=1}^{M} \varphi_i(x_k) \varphi_j(x_k) \int_\Omega \varphi_k \, dx$ in systems (35), (36). In order to improve the efficiency of the finite element method, a choice of different (coarsened) triangulations $\mathcal{T}_h^n$ in subsequent scale steps is also possible [23, 26, 27]. The finite element method can also be used on a rectangular grid with bilinear approximating functions [27].

The same ideas as above can be also used for the finite element discretization of Eq. (2). The only difference will be in the facts that the mass matrix $\mathbf{M}(w)$ will depend on $w = 1/(g(|\nabla u_h^c|)|\nabla u_h^{n-1}|)$ and the stiffness matrix $\mathbf{A}(w)$ will depend on $w = 1/|\nabla u_h^{n-1}|$.

3.2. Complementary Volume Discretization

In this subsection we will discretize Eq. (2) or, more precisely, the semi-discrete approximation (16) by means of the complementary volume method. In order to derive the complementary volume spatial discretization ([20]), we integrate (16) over a co-volume V_i

$$\int_{V_i} \frac{u^n - u^{n-1}}{g^{n-1}|\nabla u^{n-1}|\tau} \, dx = \int_{V_i} \nabla \cdot \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) dx. \tag{37}$$

Using the divergence theorem on the right-hand side we get

$$\int_{V_i} \nabla \cdot \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) dx = \int_{\partial V_i} \frac{1}{|\nabla u^{n-1}|} \frac{\partial u^n}{\partial \nu} \, ds = \sum_{j \in C_i} \int_{e_{ij}} \frac{1}{|\nabla u^{n-1}|} \frac{\partial u^n}{\partial \nu} \, ds. \tag{38}$$

If $u_h^n \in V_h(\mathcal{T}_h)$ is a continuous piecewise linear function on the triangulation $\mathcal{T}_h$ and we have denoted $u_i = u_h(x_i)$ as its nodal values, then

$$\sum_{j \in C_i} \int_{e_{ij}} \frac{1}{|\nabla u_h^{n-1}|} \frac{\partial u_h^n}{\partial \nu} \, ds = \sum_{j \in C_i} \left(\sum_{T \in \mathcal{E}_{ij}} \frac{c_{ij}^T}{|\nabla u_T^{n-1}|} \right) \frac{u_j^n - u_i^n}{h_{ij}}, \tag{39}$$

where $|\nabla u_T^{n-1}|$ denotes the constant value of the gradient of u_h^{n-1} in the simplex T. The complementary volume method approximates the left-hand side of (37) by

$$\frac{|V_i|(u_i^n - u_i^{n-1})}{\tau g(|\nabla u_i^c|)|\nabla u_i^{n-1}|},\tag{40}$$

where $|\nabla u_i^{n-1}|$, $|\nabla u_i^c|$ denote an approximation of the gradient in co-volume V_i. For that goal we have chosen the weighted average value of gradients in the co-volume [20]

$$|\nabla u_i| \approx \sum_{T \in \mathcal{N}_i} \frac{|T \cap V_i|}{|V_i|} |\nabla u_T|.\tag{41}$$

If we denote

$$b_i^{n-1} = \frac{|V_i|}{g(|\nabla u_i^c|)|\nabla u_i^{n-1}|},\tag{42}$$

$$a_{ij}^{n-1} = \frac{1}{h_{ij}} \sum_{T \in \mathcal{E}_{ij}} \frac{c_{ij}^T}{|\nabla u_T^{n-1}|},\tag{43}$$

we can write the following:

Linear fully discrete complementary volume scheme for solving Eq. (2). For $n = 1, \ldots, N$ we look for u_i^n, $i = 1, \ldots, M$, satisfying the equation

$$b_i^{n-1}(u_i^n - u_i^{n-1}) + \tau \sum_{j \in C_i} a_{ij}^{n-1}(u_i^n - u_j^n) = 0.\tag{44}$$

Of course, before solving (44), we must put $|\nabla u_i^c|$ into (42). The function $u_h^c \in V_h(\mathcal{T}_h)$ is found by the same idea as given in (37)–(40) applied to (29); i.e., we solve (44) with u_i^n replaced by u_i^c and with $b_i^{n-1} \equiv b_i = |V_i|$, $a_{ij}^{n-1} \equiv a_{ij} = \frac{e_{ij}}{h_{ij}}$. Then $|\nabla u_i^c|$ is computed by (41) and put into the Perona–Malik function g. The system (44) gives a symmetric positive definite M matrix with diagonal dominance.

Using the nonnegativeness of b_i^{n-1}, a_{ij}^{n-1}, we can get the L_∞-stability estimate for the fully discrete scheme (44) in the form

$$\min u_i^0 \le \min u_i^n \le \max u_i^n \le \max u_i^0, \quad 1 \le n \le N,\tag{45}$$

which means that

$$\|u_h^n\|_{L_\infty(\Omega)} \le \|u_h^0\|_{L_\infty(\Omega)}, \quad 1 \le n \le N.\tag{46}$$

To see (45), let us rewrite (44) in the form

$$u_i^n + \frac{\tau}{b_i^{n-1}} \sum_{j \in C_i} a_{ij}^{n-1}(u_i^n - u_j^n) = u_i^{n-1}\tag{47}$$

and let $\max u_h^n = \max(u_1^n, \ldots, u_M^n)$ be achieved in the ith node. Then the whole second term on the left-hand side is nonnegative and thus $u_i^n \le u_i^{n-1}$.

Until now, we have not solved the problem of possible zero gradients in the denominators of the scheme (16) or (44). To prevent such a situation we can use the Evans–Spruck-type regularization [28] and consider

$$|\nabla u|_\varepsilon = \sqrt{\varepsilon + |\nabla u|^2} \tag{48}$$

instead of $|\nabla u|$ everywhere in the schemes (16) and (44), or just when $|\nabla u|$ is vanishing. One can observe that the Evans–Spruck regularization is well suited for image smoothing purposes. For zero gradients such a regularized discretization (44) reduces to a discretization of the linear heat equation. It is quite desirable at points with small changes in intensity. On the other hand, the influence of ε can be neglected, and thus we have a curvature-driven flow of level lines for large gradients. Such a behavior, in a sense, covers implicitly a modification of Eq. (2) given in [3] (see [3], Eq. (8)). All results that we have derived so far for fully discrete or semi-discrete schemes, i.e., L_∞ and $W^{1,1}$ estimates, are valid also for regularization. Moreover, they do not depend on the regularization parameter ε. Thus, we can pass to the limit and understand the solution of either (16) or (44) in a generalized sense. This is much more technical and we avoid this step in this paper. Concerning the computational point of view, we have found preconditioned iterative linear solvers, the efficiency of which is not affected much even when ε is a very small.

Using the integration (37) and the relation (39), one can also derive a complementary volume discretization of Eq. (1), or, more precisely, for the semi-discrete coupling (11), (29). In such a way, we get the system (44) with

$$b_i^{n-1} = |V_i|, \quad a_{ij}^{n-1} = \frac{1}{h_{ij}} \sum_{T \in \mathcal{E}_{ij}} c_{ij}^T g\left(\left|\nabla u_T^c\right|\right), \tag{49}$$

where u_h^c is computed in the same way as above, and we add the right-hand side to (44) given by $\tau f(u_i^0 - u_i^{n-1})|V_i|$. Let us note that such a discretization can be considered as a special mass lumping approximation in the finite element method.

3.3. Finite Volume Discretization

In spirit, the finite volume discretization follows the ideas of the previous subsection. We again integrate the equation (in this subsection we will deal only with Eq. (1)) in every finite volume. In general, the finite volume can be either a simplex of the triangulation itself or a co-volume of the dual mesh [22]. We have chosen the second strategy, and we associate finite volumes with co-volumes because they correspond to the pixel/voxel structure of the image. The main difference, when compared to the complementary volume technique, is that the approximating functions are not in $V_h(\mathcal{T}_h)$, but they are just piecewise constant on finite volumes. Thus, we cannot work directly with gradients or normal derivatives (see (39)) since they are either zeros (inside finite volumes) or infinite (on their boundaries). By means of $\bar{u}_i$ representing an approximate value of the solution inside the finite volume V_i we will only approximate fluxes through the boundary of the finite volume. The value of the diffusion coefficient along e_{ij} will be approximated by its value at the point x_{ij}. Then we get the following:

Linear fully discrete finite volume scheme for solving Eq. (1) (see [17]). For $n = 1, \ldots, N$ we look for $\bar{u}_i^n$, $i = 1, \ldots, M$, satisfying

$$\left(\bar{u}_i^n - \bar{u}_i^{n-1}\right)|V_i| + \tau \sum_{j \in C_i} g_{ij}^{n-1} \frac{|e_{ij}|}{h_{ij}} \left(\bar{u}_i^n - \bar{u}_j^n\right) = \tau f \left(\bar{u}_i^0 - \bar{u}_i^{n-1}\right)|V_i| \tag{50}$$

starting with $\bar{u}_i^0$ (see (28)). The term

$$g_{ij}^{n-1} = g(|\nabla G_\sigma * u^{n-1}(x_{ij})|) \tag{51}$$

is computed by the strategy (30). The system (50) gives a symmetric positive definite M-matrix with a diagonal dominance.

The study of a kind of finite volume scheme analogous to (50) for the-level-set-like Eq. (2) is an open problem.

4. NUMERICAL EXPERIMENTS

This section is devoted to a discussion of some numerically computed examples. It also includes a discussion of the computational efficiency of iterative solvers used in our semi-implicit scheme (44). Since numerical experiments based on the finite element method and finite volume methods in 2D and 3D have been presented elsewhere [14, 16, 17, 23, 26, 27, 29], here we give some 2D examples of the usage of the complementary volume method. In our computations we have chosen $g(s) = 1/(1 + Ks^2)$ with a constant $K > 0$, $f \equiv 0$ while the convolution is realized using (29) with σ less than τ. The space step h in our experiments is always given as $1/n$, where n is the number of pixels in the vertical direction.

EXAMPLE 1.   In Fig. 2 we smooth an initial 321×373 pixel image (ancient coat-of-arms of the Slovak town Kremnica plotted in the left part of the figure) scanned from the book with neither paper nor colors of a good quality. We present the result of the scheme (44)
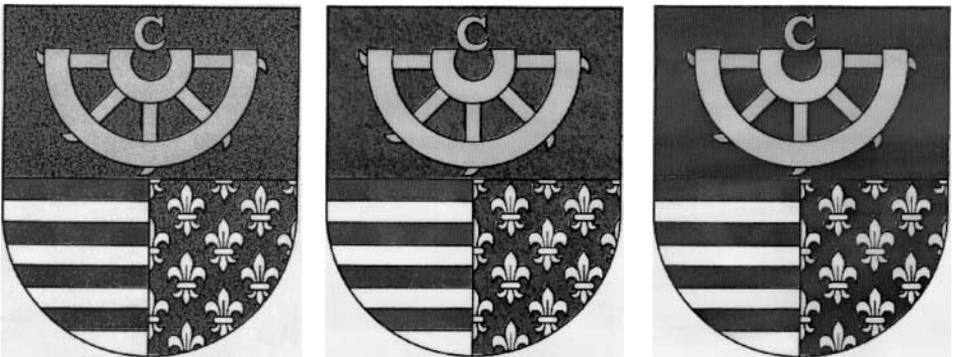
FIG. 2. Initial image (left) and the results of smoothing after 5 (middle) and 10 (right) scale steps (see Example 1).

FIG. 3. Initial image (left) and the results of smoothing after 5 (middle) and 20 (right) scale steps (see Example 2).

after 5 and 10 discrete scale steps, with the parameters $\tau = 0.0025$, $\sigma = 0.0001$, $\varepsilon = 10^{-6}$, $K = 1$.

EXAMPLE 2. In this example we did a multiscale analysis of an old text (417×302 pixels), and in Fig. 3 we present the results after 5 and 20 discrete steps of the scheme (44). The parameters used were $\tau = 0.001$, $\sigma = 0.0001$, $\varepsilon = 10^{-4}$, $K = 2$.

EXAMPLE 3. In this example we performed a multiscale analysis of a medical image (463×397 pixels), and in Fig. 4 we present the result after 10 discrete steps of the scheme (44) for Eq. (1)—see (49). The parameters used were $\tau = 0.001$, $\sigma = 0.0001$, $K = 4$.

EXAMPLE 4. In Fig. 5 we test the behavior of the co-volume algorithm (44) applied to the well-known mean curvature flow problem, i.e., Eq. (2) with $g(s) \equiv 1$. We evolve numerically the initial nonconvex curve given in the top left. In the next images we plot the numerical motion driven by the curvature until a circular shape is obtained and the curve is shrinking into a point. Since we do not know an exact solution, the numerical solutions given by the two conceptually different methods are presented for comparison. We plot the solution given by method (44) by solid lines starting with the so-called signed distance function [30]. We use the space discretization parameter $h = 0.02$, time step $\tau = 0.001$, and $\varepsilon = 10^{-4}$, and we compute 230 time steps. However, practically the same results were obtained with $\tau = 0.01$, computing just 23 time steps. We plot a numerical solution based
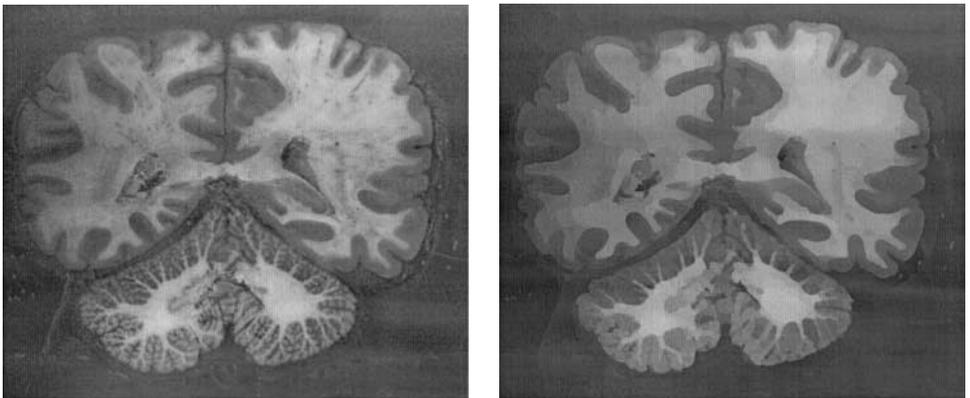
FIG. 4. Initial image (left) and the result after 20 steps of multiscale analysis (right) (see Example 3).
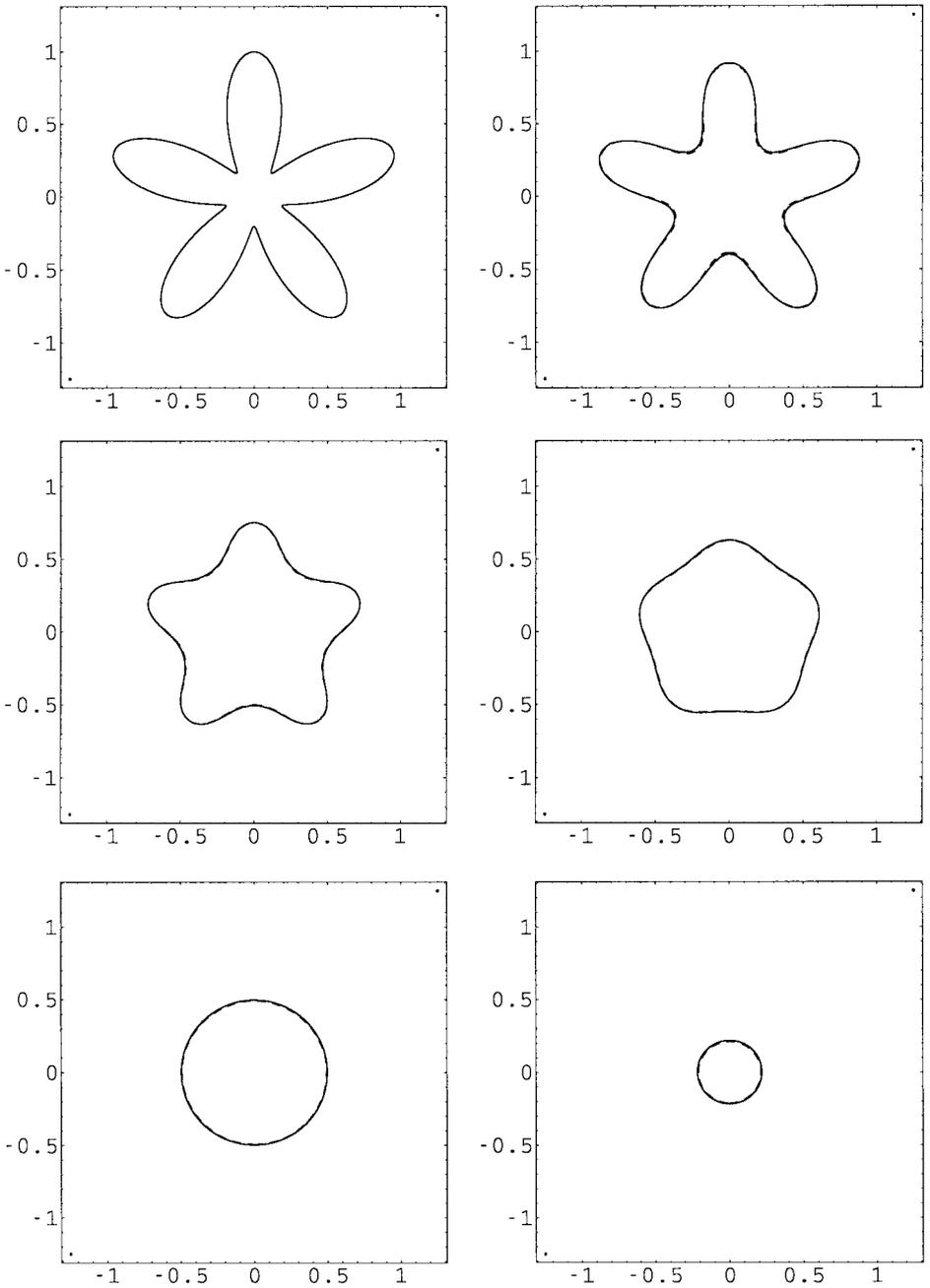
FIG. 5. Evolution of star-shape-like curves by mean curvature flow. Comparison of two different methods (see Example 4).

on the representation of the mean curvature flow by intrinsic heat equations [9, 12, 13] by dashed lines. From the comparison, one can see a precise coincidence of the two methods during the evolution, as well as a correspondence of the numerical extinction times.

Finally, we remark on solving linear systems in a discrete scale step of the semi-implicit method. The huge number of unknowns in each system, the sparsity pattern, and the

<div align="center">

TABLE 1
Comparison of Preconditioned Conjugate Gradient Solvers

</div>

	# Multip/CPU time			
	CG no precon	IC CG	MIC CG	ILUT(7,0.0001) CG
$\varepsilon = 10^{-3}$	223/7.08	35/2.74	17/1.36	11/1.23
$\varepsilon = 10^{-4}$	395/12.41	31/2.43	31/2.48	12/1.34
$\varepsilon = 10^{-5}$	641/20.20	71/4.92	72/5.10	11/1.16
$\varepsilon = 10^{-6}$	1008/31.09	172/14.5	171/13.09	9/0.91

properties of the coefficient matrices suggest using the conjugate gradient (CG) method. With a good preconditioner, the total amount of steps required for the convergence can be reduced dramatically, at the cost of a slight increase in the number of operations per step, resulting in general in much more efficient algorithms.

It is well known that standard incomplete factorization (IC) methods exist for the M-matrices [31] arising in our discretizations, and modified incomplete factorization (MIC) methods exist for weakly diagonally dominant matrices [32, 33]. For that reason, we have used these techniques to precondition the conjugate gradient algorithm to solve our linear systems. Then we used, for comparison, a more memory-consuming preconditioner ILUT (level of fill-in, treshold), allowing fill-in with threshold to compute incomplete LU factorization, even if it ignores any symmetry in the matrix [34].

In Table 1 we report the number of matrix–vector multiplications used to obtain a convergence in the solution of the linear system taken from the first scale step in Example 1, as well as the CPU times in seconds on a Digital Alfa XP1000 workstation. In other examples the behavior was similar. As a stopping criterion we have used $\|r^k\|_2 <=$ tol $\|r^0\|_2$ with tolerance tol $= 0.01$, where $\|\cdot\|_2$ means the discrete L_2 norm and r^k is residual in the kth iteration. For very small $\varepsilon = 10^{-10}$ when even a finer tolerance did not provide satisfactory results, we used another stopping criterion: $\|\Delta x^k\|_2 <=$ tol $\|\Delta x^0\|_2$, where Δx^k is the correction to the solution at the kth iteration. Then we obtain a convergence with an ILUT (7, 0.0001) preconditioner and CG in 15 iterations in 1.42 s. The previous results can be understood when we realize that the estimated condition numbers range from 43599.869, in the case of $\varepsilon = 10^{-3}$, to 137682783.127, in the case of $\varepsilon = 10^{-10}$, and we have 119733 unknowns. These experiments show that a well-known algorithm like the conjugate gradient one with a simple preconditioner allows fast computations. Even though, for instance, the algebraic multigrid could improve the performances, it would require more programming efforts.

5. CONCLUSIONS

We have presented a general, robust, and efficient approach for dealing with numerical solutions of PDEs arising in image processing and computer vision. Using ideas of this paper, the well-established computational methods (finite elements, finite volumes, and complementary volumes), well known in the engineering community, can be applied to nonlinear problems of image multiscale analysis. The methods allow the inclusion of adaptivity. This yields a further increase in efficiency. The advantage of the semi-implicity is also in the use of fast preconditioned linear solvers for discretized equations at every scale level.

Of course, there exist other methods of solution known to have a good behavior (e.g., [3, 7, 35]). Our goal is not to favorize our approach. However, if the finite element or the finite volume packages are at one's disposal, one can get a straightforward solution for the image-processing problems. The implementations of our schemes are quite natural in the framework of numerical methods for partial differential equations. They can also be used for other similar equations used recently in the nonlinear scale space theory (see, e.g., [30, 36–38]).

ACKNOWLEDGMENTS

REFERENCES

1. F. Catté, P. L. Lions, J. M. Morel, and T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, *SIAM J. Numer. Anal.* **29**, 1992, 182–193.

2. P. Perona and J. Malik, Scale space and edge detection using anisotropic diffusion, *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 1990, 629–639.

3. L. Alvarez, P. L. Lions, and J. M. Morel, Image selective smoothing and edge detection by nonlinear diffusion II, *SIAM J. Numer. Anal.* **29**, 1992, 845–866.

4. L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel, Axioms and fundamental equations of image processing, *Arch. Ration. Mech. Anal.* **123**, 1993, 200–257.

5. L. Alvarez and J. M. Morel, Formalization and computational aspects of image analysis, *Acta Numer.* 1994, 1–59.

6. S. Kichenassamy, The Perona–Malik paradox, *SIAM J. Appl. Math.* **57**, 1997, 1328–1342.

7. S. Osher and J. Sethian, Front propagating with curvature dependent speed: Algorithms based on the Hamilton–Jacobi formulation, *J. Comput. Phys.* **79**, 1988, 12–49.

8. J. A. Sethian, Numerical algotithm for propagating interfaces: Hamilton–Jacobi equations and conservation laws, *J. Differential Geom.* **31**, 1990, 131–161.

9. G. Dziuk, Convergence of a semidiscrete scheme for the curve shortening flow, *Math. Models Methods Appl. Sci.* **4**, 1994, 589–606.

10. G. Dziuk, Algorithm for evolutionary surfaces, *Numer. Math.* **58**, 1991, 603–611.

11. G. Sapiro and A. Tannenbaum, On affine plane curve evolution, *J. Funct. Anal.* **119**, 1994, 79–120.

12. K. Mikula and D. Ševčovič, Solution of nonlinearly curvature driven evolution of plane curves, *Appl. Numer. Math.* **31**, 1999, 191–207.

13. K. Mikula and D. Ševčovič, Evolution of plane curves driven by a nonlinear function of curvature and anisotropy, *SIAM J. Appl. Math.* **61**, 2001, 1473–1501.

14. A. Handlovičová, K. Mikula, and A. Sarti, Numerical solution of parabolic equations related to level set formulation of mean curvature flow, *Comput. Visual. Sci.* **1**, 1999, 179–182.

15. L. I. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* **60**, 1992, 259–268.

16. J. Kačur and K. Mikula, Solution of nonlinear diffusion appearing in image smoothing and edge detection, *Appl. Numer. Math.* **17**, 1995, 47–59.

17. K. Mikula and N. Ramarosy, Semi-implicit finite volume scheme for solving nonlinear diffusion equations in image processing, *Numer. Math.* **89**, 2001, 561–590.

18. H. Brezis, *Analyse fonctionelle, theorie et applications*, Masson, Paris, 1992.

19. O. A. Ladyženskaja and N. N. Uralceva, *Linear and Quasilinear Elliptic Equations*, Academic Press, New York, 1968.

20. N. J. Walkington, Algorithms for computing motion by mean curvature, *SIAM J. Numer. Anal.* **33**, 1996, 2215–2238.

21. K. Mikula and J. Kačur, Evolution of convex plane curves describing anisotropic motions of phase interfaces, *SIAM J. Sci. Comp.* **17**, 1996, 1302–1327.

22. R. Eymard, T. Gallouet, and R. Herbin, The finite volume method, in *Handbook for Numerical Analysis* (Ph. Ciarlet and P. L. Lions, Eds.), Vol. 7, pp. 715–1022, Elsevier, Amsterdam, 2000.

23. E. Bänsch and K. Mikula, A coarsening finite element strategy in image selective smoothing, *Comput. Visual. Sci.* **1**, 1997, 53–61.

24. J. Xu and L. Zikatanov, A monotone finite element scheme for convection-diffusion equations, *Math. Comput.* **68**, 1999, 1429–1446.

25. S. Wolfram, *The Mathematica Book*, third ed., Cambridge Univ. Press, Cambridge, UK, 1996.

26. E. Bänsch and K. Mikula, Adaptivity in 3D image processing, Preprint 99-14, Zentrum fur Technomathematik, Universität Bremen, 1999.

27. T. Preusser and M. Rumpf, An Adaptive Finite Element Method for Large Scale Image Processing, in *Proceedings of ScaleSpace'99* (M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, Eds.), pp. 223–234, Springer-Verlag, Berlin, 1999.

28. L. C. Evans and J. Spruck, Motion of level sets by mean curvature I, *J. Differential Geom.* **33**, 1991, 635–681.

29. A. Sarti, K. Mikula, and F. Sgallari, Nonlinear multiscale analysis of 3D echocardiographic sequences, *IEEE Trans. Medical Imaging* **18**, 1999, 453–466.

30. J. A. Sethian, *Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Material Science*, Cambridge Univ. Press, Cambridge, UK, 1999.

31. J. A. Meijerink and H. A. van der Vorst, An iterative solution method for systems of which the coefficient matrix is a symmetric M-matrix, *Math. Comp.* **31**, 1977, 148–162.

32. I. Gustafsson, Modified incomplete Cholesky (MIC) methods, in *Preconditioning Methods, Theory and Applications*, (D. J. Evans, Ed.), pp. 265–293, Gordon and Breach, New York, 1983.

33. A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1999.

34. Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS, Boston, 1996.

35. J. Weickert, B. M. t. H. Romeny, and M. A. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, *IEEE Trans. Image Process.* **7**, 1998, 398–410.

36. J. Weickert, *Anisotropic Diffusion in Computer Vision*, Teubner, Stuttgart, 1998.

37. M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert (Eds.), *Scale-Space Theories in Computer Vision*, Lecture Notes in Computer Science 1682, Springer, Berlin, 1999.

38. V. Caselles, J.-M. Morel, and G. Sapiro (Eds.), *IEEE Trans. Image Process.* **7**, 1998.

ANGELA HANDLOVIČOVÁ finished his studies of mathematics at Comenius University, Bratislava in 1981 and a Ph.D. in numerical analysis at the same university in 1993. She is currently with the Department of Mathematics and Descriptive Geometry, Slovak University of Technology, Bratislava. She works mainly on numerical methods for nonlinear partial differential equations with applications.

KAROL MIKULA was born in Bratislava in 1963. He finished his studies of computer science at Comenius University, Bratislava in 1986 and a Ph.D. in numerical analysis at the same university in 1993. His current position is Head of Division of Informatics and Numerical Analysis at the Department of Mathematics and Descriptive Geometry, Slovak University of Technology, Bratislava. He works mainly on numerical methods for nonlinear partial differential equations with applications in image processing and other applied sciences.

FIORELLA SGALLARI graduated with a degree in mathematics from the University of Bologna, Italy, in 1976. Subsequently, she obtained a research grant issued by the National Research Council (CNR). In 1980 she joined the University of Bologna as a researcher, and now she is Full Professor of Numerical Analysis. Her current research interests are in the field of linear algebra and parallel computation, with particular attention to linear and nonlinear systems of large dimensions and discrete ill-posed problems. She is involved in research projects aimed at the development of algorithms and software tools for the computer-aided design of nonlinear microwave circuits and echocardiographic image processing. She is a member of IEEE, SIAM, and SIMAI.