Efficient Algorithm for Level Set Method Preserving Distance Function

Virginia Estellers, Dominique Zosso, *Member, IEEE*, Rongjie Lai, Stanley Osher, Jean-Philippe Thiran, *Senior Member, IEEE*, and Xavier Bresson

***Abstract*— The level set method is a popular technique for tracking moving interfaces in several disciplines, including computer vision and fluid dynamics. However, despite its high flexibility, the original level set method is limited by two important numerical issues. First, the level set method does not implicitly preserve the level set function as a distance function, which is necessary to estimate accurately geometric features, s.a. the curvature or the contour normal. Second, the level set algorithm is slow because the time step is limited by the standard Courant–Friedrichs–Lewy (CFL) condition, which is also essential to the numerical stability of the iterative scheme. Recent advances with graph cut methods and continuous convex relaxation methods provide powerful alternatives to the level set method for image processing problems because they are fast, accurate, and guaranteed to find the global minimizer independently to the initialization. These recent techniques use binary functions to represent the contour rather than distance functions, which are usually considered for the level set method. However, the binary function cannot provide the distance information, which can be essential for some applications, s.a. the surface reconstruction problem from scattered points and the cortex segmentation problem in medical imaging. In this paper, we propose a fast algorithm to preserve distance functions in level set methods. Our algorithm is inspired by recent efficient ℓ^1 optimization techniques, which will provide an efficient and easy to implement algorithm. It is interesting to note that our algorithm is not limited by the CFL condition and it naturally preserves the level set function as a distance function during the evolution, which avoids the classical re-distancing problem in level set methods. We apply the proposed algorithm to carry out image segmentation, where our methods prove to be 5–6 times faster than standard distance preserving level set techniques. We also present two applications where preserving a distance function is essential. Nonetheless, our method stays generic and can be applied to any level set methods that require the distance information.**

***Index Terms*— Image segmentation, level set, numerical scheme, signed distance function, splitting, surface reconstruction.**

Manuscript received September 10, 2011; revised May 9, 2012; accepted May 13, 2012. Date of publication June 5, 2012; date of current version November 14, 2012. The work of V. Estellers was supported by the Swiss National Science Foundation under Grant 200021-130152. The work of D. Zosso was supported by the National Competence Center in Biomedical Imaging. The work of R. Lai was supported by the Zumberge Individual Award from the USC's J. H. Zumberge Faculty Research and Innovation Fund. The work of S. Osher was supported by the USC/U.S. Army under Grant 071413, and under the Office of Naval Research (ONR) Grant N00014-11-1-0719 and Grant N00014-08-1-1119. The work of X. Bresson was supported by the Hong Kong Research Grants Council under Grant GRF110311. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xin Li.

V. Estellers, D. Zosso, and J.-P. Thiran are with the Signal Processing Laboratory, École Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: virginia.estellers@epfl.ch; dominique.zosso@epfl.ch; jp.thiran@epfl.ch).

R. Lai is with the Department of Mathematics, University of Southern California, Los Angeles, CA 90089 USA (e-mail: rongjiel@usc.edu).

S. Osher is with the Department of Mathematics, University of California, Los Angeles, CA 90095-1555 USA (e-mail: sjo@math.ucla.edu).

X. Bresson is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: xbresson@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2202674

I. Introduction

In THE last twenty years, the level set method (LSM) of Osher and Sethian [1] has become a popular numerical technique for tracking moving interfaces in computational geometry, fluid mechanics, computer graphics, computer vision and material sciences. The main reasons of its success are the high flexibility of this method to adapt to different problems, the ability to deal with changes of topology (contour breaking and merging) without any extra functions and the guarantee of the existence of solutions in the class of viscosity partial differential equations (PDEs). Moreover, extensive numerical algorithms based on Hamilton-Jacobi equations have been developed, accurately handling shocks and providing stable numerical schemas.

The key idea of the LSM is to implicitly represent a contour or interface as the zero level set of a higher dimensional function, called the level set function (LSF), and formulate the evolution of the contour through the evolution of the level set function. For closed contours, signed distance functions (SDFs) were originally adopted to represent level set functions because they directly provide stability and accuracy to the LSM [2], [3]. In the last years, however, new approaches have proposed to use binary functions, rather than distance functions, to represent the contour. This change of representation allows to use fast and convex optimization techniques with graph cut [4], [5] and convex relaxation methods [6]–[8], provide powerful techniques for the LSM but cannot preserve the LSF as a distance function. Distance functions are essential in several applications and justify the necessity of developping a fast and accurate algorithm for distance preserving level set methods. For instance, in medical image SDFs are used in the segmentation of cortical surfaces presented in Section V; they also provide an efficient estimation of surface normals for the surface reconstruction task from Section II and are used in the creation of several special effects in computer graphics.

A major issue of distance preserving level set methods is the limited speed of the existing algorithms, which are based on Hamilton-Jacobi equations and upwind schemes [1], [3].

Two main issues handicap these iterative methods. First, the speed of the algorithms is limited by the CFL condition [9], which is a necessary condition to guarantee the stability of PDE-based iterative schemes. Secondly, the level set energy is hard to optimize because of the ℓ^1-based total variation (TV) term, which is not differentiable and is usually regularized. However, the regularization significantly slows down the minimization process and does not provide an exact solution to the problem. In our work, we will overcome these speed limitations using recent efficient techniques in ℓ^1 optimization.

Another major issue of the LSM, pointed out by Gomes and Faugeras in [10], is a contradiction between the theory and the implementation when LSF are represented by SDF. Indeed, the LSM does not intrinsically preserve the SDF during the contour propagation because SDFs are not solutions of the Hamilton-Jacobi equations associated to the LSM. Additional techniques are then necessary to preserve the LSF as an SDF during contour evolution, but no fully satisfying methods have been proposed so far. The two most common approaches that have been suggested to fix this problem are either re-distancing regularly the LSF as an SDF (this procedure is called re-distancing or re-initialization), or constraining the LSF to remain an SDF during the contour evolution.

Re-distancing is the most common approach. It consists in stopping the evolution of the LSF periodically and re-initializing it as an SDF while preserving the zero level set. This approach introduces the questions of when and how to re-initialize the LSF. It is hard to say when the re-distancing must be applied as there is a trade-off between speed (each re-distancing task takes time) and accuracy (the LSM will develop irregularities during the evolution without re-distancing). Similarly, the re-initialization can be performed with different LSM techniques, using PDEs [11] or the fast marching algorithm [12], [13]. The main issue with the re-distancing approach is the difficulty to preserve exactly the location of the zero level set during the re-distancing process, which might require interpolation strategies to avoid shifting the moving interface to undesired positions.

The second approach aims at constraining the LSF to stay an SDF during the contour evolution, avoiding the previous re-distancing procedure altogether. Gomes and Faugeras introduced in [10] a new level set formulation to restrict the LSF to an SDF. The new formulation consists of three coupled PDEs, which makes the analysis of the existence of a solution and the numerical implementation more difficult than the standard LSM. More recently, Li *et al*. [14], [15] proposed to add a penalty term in the level set energy to constrain the LSF to be close to an SDF. They also developed a new algorithm, simpler and more efficient than the conventional LSM method. However, the time step of the algorithm is still restricted by the CFL condition and the SDF property is only encouraged but not enforced. We will see that our method overcomes all of these limitations and provides an efficient way to constrain exactly the LSF to be an SDF.

Both the previous approaches and the method that we propose can be optimized with narrow-band implementations [16], [17]. In [18] for instance, the authors improve the time step restriction by an implicit PDE iterative scheme, where the CFL restriction is avoided by means of an additive operator splitting (AOS) technique [19], but redistancing is still required at each iteration, a limitation overcome by our algorithm.

The level set method was first introduced in computer vision to carry out image segmentation [20]–[22] and then extended to other tasks as stereo reconstruction [23], object tracking [24], and object recognition [25]. In this paper, we focus on image segmentation and surface reconstruction, but the proposed method can be easily extended to other problems. In image segmentation, the level set method re-formulates the parametric active contour into a non-parametric energy minimization problem (i.e. independent of the contour discretization). The active contour is embedded as the zero level set of the LSF, which moves according to the active contour evolution equation and drives its zero level set to the edges of the desired object. The level set formulation easily includes edge-, region- and shape-based terms in the image segmentation criteria, proving the high flexibility of the LSM to adapt to different tasks and justifying its extensive use in the field. Similar segmentation models have been applied to the problem of reconstructing a surface form unorganized data points [26]. In that case, the LSM allows us to obtain an implicit representation of the surface by an SDF, avoiding complex 3D parametrization methods which require a prior knowledge of the topology of the surface, and providing an easy and reliable way to directly estimate surface normals and curvature from the level set function.

Here, we propose a fast and accurate algorithm for distance preserving level set methods. We constrain the LSF to remain an SDF via a constrained minimization problem. The constrained minimization problem is hard to solve directly, so we propose to split the original hard problem into sub-optimization problems which are easier to solve, and combine them together using an augmented Lagrangian approach. This idea is borrowed from the split-Bregman method [27] (and more generally from the alternating direction methods of multipliers [28]–[32]), which is an efficient ℓ^1 minimization method originally developed to solve the compressed sensing problem. The proposed algorithm holds several important properties. Firstly, it is fast because it is not limited by the CFL condition, leading to a method 5 to 6 times faster than most distance preserving LSM algorithms for the image segmentation problem. This improvement is due to the splitting strategy and the reformulation of a constraint minimization, and allows us to deal with the non-differentiability of the TV norm and go beyond the CFL time step restriction. Secondly, our algorithm preserves the LSM as an SDF, avoiding the classical re-distancing problem and providing desirable properties for some applications. For example, this makes an important difference in surface reconstruction, where surface normals can be fast and reliably estimated during the surface evolution instead of being required as input data s.a. [33], [34], and in medical image segmentation, where the distance information can be exploited to include the topology restrictions associated to the cortical surface into the problem. Finally, our algorithm is easy to implement because the iterative scheme is based on standard minimization problems.

The rest of the paper is organized as follows. In Section II we review very briefly the level set method applied to image segmentation and surface reconstruction. We introduce our algorithm to solve efficiently the level set method in Sections III and IV. Section V presents the results and Section VI draws the conclusions.

II. Level Set Method for Image Segmentation and Surface Reconstruction

Level set method can be applied to perform image segmentation using the active contour method. In this context, the segmentation problem is defined as the following energy minimization problem with respect to a contour C:

$$\min_{C\subset\Omega} \int_C w_b(s)ds + \int_{C_{in}} w_r^{in}(x)dx + \int_{C_{out}} w_r^{out}(x)dx, \quad (1)$$

where Ω is the image domain, s is the arc-length parametrization and the first term weights the length of C by an edge detector function w_b, such as in [20], [21]. In the other terms, C_{in}, C_{out} designate the regions inside and outside the contour C and w_r^{in}, w_r^{out} are region-based terms, such as the ones proposed in [22]. We adopt this minimization model for image segmentation, as it represents a large class of active contour models published in the literature.

Actually, similar models have also been used to reconstruct smooth surfaces from unorganized data points. Given a set of (noisy) points $\{x_i\}_{1\le i\le N}$ lying close to the unknown surface, a smooth estimate of the surface S can be reconstructed by minimizing the area energy weighted by the distance to the set of points $\{x_i\}_{1\le i\le N}$, i.e. $w_b(x) = d_{\{x_i\}}(x)$ [26], [35]. Inspired by [34], [36], we propose a model which also includes a region-based term w_r^{in} to improve the performance with sparse data sets. We adapt the model proposed by Lempitsky and Boykov [34], which was designed to align the normals of the reconstructed surface to the pre-computed orientations $\{n_i\}_{1\le i\le N}$ of the data points. Based on this surface information, a semi-dense vector field of the surface normal $\hat{n}$ can be estimated everywhere in space and the unknown surface can thus be reconstructed by iteratively maximizing the alignment of the reconstructed surface normal $\mathcal{N}$ to $\hat{n}$. In fact, that method can be traced back to the inclusion of equivalent alignment terms in image segmentation for low contrast images [37]–[39]. Making use of the divergence theorem, the function w_r^{in} can be identified as follows:

$$\arg\max_{C\subset\Omega} \int_C \hat{n}\cdot\mathcal{N}ds = \arg\min_{C\subset\Omega} \int_{C_{in}} -\operatorname{div}\hat{n}dx, \quad (2)$$

which corresponds to defining $w_r^{in} = -\operatorname{div}\hat{n}$. The flux of the semi-dense normal field at the point x is estimated with the formula:

$$\operatorname{div}\hat{n}(x) = \sum_{1\le i\le N} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{|x-x_i|^2}{2\sigma^2}} \langle x-x_i, n_i\rangle, \quad (3)$$

where $x - x_i$ denotes the vector centred at x_i and pointing at x and n_i is the surface normal estimated at x_i. The proposed distance preserving LSM can estimate the surface normal during the reconstruction process: $n_i = \nabla\phi(\boldsymbol{x}_i)$, where ϕ is the SDF. To sum up, the proposed functions in (1) for the surface reconstruction problem are: $w_b(x) = d_{\{x_i\}}(x)$, $w_r^{in} = -\operatorname{div}\hat{n}$, and $w_r^{out} = 0$. Unlike the method of Lempitsky and Boykov, the proposed surface reconstruction algorithm does not need to know *a priori* the normal of the surface at the data points.

The level set method can be applied to solve (1) in general (and specifically the image segmentation problem and the surface reconstruction problem), by re-writing the minimization in terms of a level set function $\phi : \Omega \to \mathbb{R}$ as follows:

$$\min_{\phi} \int_\Omega w_b(x)|\nabla H(\phi)| + w_r(x)H(\phi) \quad \text{s.t.} \quad |\nabla\phi| = 1, \quad (4)$$

where the curve in $\mathbb{R}^2$ (or surface in $\mathbb{R}^3$) C is represented by the zero level set of ϕ, H is the Heaviside function, and the constraint $|\nabla\phi| = 1$ guarantees the level set function to be a signed distance function [11].

III. Efficient Algorithm for Level Set Method Preserving Signed Distance Function

In this section, we introduce an efficient algorithm to solve the level set minimization problem (4). The main idea is to split the original hard problem (4) into sub-optimization problems which are well-known and easy to solve, and combine them together using an augmented Lagrangian. This idea is borrowed from the split-Bregman method [27], which is an efficient ℓ^1 optimization method recently introduced in image processing to solve the compressed sensing problem.

Let us consider the following constrained minimization problem, which is equivalent to the original LSM problem (4):

$$\min_{\phi,\varphi,u,\boldsymbol{q},\boldsymbol{p}} \int_\Omega w_b(x)|\boldsymbol{q}| + w_r(x)u \quad \text{s.t.} \quad \begin{cases} u=H(\varphi) \\ \varphi=\phi \\ \boldsymbol{q}=\nabla u \\ \boldsymbol{p}=\nabla\phi \\ |\boldsymbol{p}|=1 \end{cases} \quad (5)$$

where we introduce functions $\varphi(x), u(x) \in \mathbb{R}$, $\boldsymbol{q}(x), \boldsymbol{p}(x) \in \mathbb{R}^n$ (boldface letters are used to denote vector functions and $n = 2$ for 2D images, and $n = 3$ for 3D images). The proposed splitting approach makes the original problem (4) easier to solve because (5) can better handle the non-differentiability and non-linearity of (4). Indeed, it is known from [27], [40] that the minimization of $|\nabla\phi|$ can be carried out efficiently by decoupling the ℓ^1-norm $|.|$ and the gradient operator ∇. The term $|\nabla\phi|$ is thus replaced by $|\boldsymbol{p}|$ and $\boldsymbol{p} = \nabla\phi$, the variable $\varphi = \phi$ is introduced to handle the non-linear term $u = H(\phi)$ and the term $|\nabla H(\phi)|$ can be changed into $|\boldsymbol{q}|$, $\boldsymbol{q} = \nabla u$.

Next, we want to reformulate this constrained minimization problem as an unconstrained optimization task. This can be done with an augmented Lagrangian approach [29], which translates the constraints into pairs of Lagrangian multiplier and penalty terms. Let us define the augmented Lagrangian energy associated to (5):

$$\begin{aligned}\mathcal{L}(\phi,\varphi,u,\boldsymbol{q},\boldsymbol{p},\Lambda) = \int_\Omega\ & w_b|\boldsymbol{q}| + w_r u + \lambda_1(\varphi-\phi) \\ & + \frac{r_1}{2}(\varphi-\phi)^2 + \lambda_2(u-H(\varphi)) + \frac{r_2}{2}(u-H(\varphi))^2 \\ & + \boldsymbol{\lambda_3}\cdot(\boldsymbol{q}-\nabla u) + \frac{r_3}{2}|\boldsymbol{q}-\nabla u|^2 + \boldsymbol{\lambda_4}\cdot(\boldsymbol{p}-\nabla\phi) \\ & + \frac{r_4}{2}|\boldsymbol{p}-\nabla\phi|^2 \quad \text{s.t.} \quad |\boldsymbol{p}| = 1 \end{aligned} \quad (6)$$

Algorithm 1 Augmented Lagrangian Method for Distance Preserving Level Set Methods

1: Initialize $\phi, \varphi, u, \boldsymbol{q}, \boldsymbol{p}, \Lambda$
2: Find a minimizer of $\mathcal{L}$ with respect to variables $(\phi, \varphi, u, \boldsymbol{q}, \boldsymbol{p})$ with fixed Lagrange multipliers Λ^k:

$$(\phi^k, \varphi^k, u^k, \boldsymbol{q}^k, \boldsymbol{p}^k) = \underset{\substack{\phi,\varphi,u,\boldsymbol{q},\boldsymbol{p} \\ |\boldsymbol{p}|=1}}{\arg\min}\ \mathcal{L}\left(\phi, \varphi, u, \boldsymbol{q}, \boldsymbol{p}, \Lambda^{k-1}\right). \quad (7)$$

3: Update Lagrange multipliers

$$\lambda_1^k = \lambda_1^{k-1} + r_1(\varphi^k - \phi^k) \quad (8)$$

$$\lambda_2^k = \lambda_2^{k-1} + r_2\left(u^k - H\left(\varphi^k\right)\right) \quad (9)$$

$$\boldsymbol{\lambda_3}^k = \boldsymbol{\lambda_3}^{k-1} + r_3(\boldsymbol{q}^k - \nabla u^k) \quad (10)$$

$$\boldsymbol{\lambda_4}^k = \boldsymbol{\lambda_4}^{k-1} + r_4(\boldsymbol{p}^k - \nabla \phi^k). \quad (11)$$

4: Stop the iterative process when $\|\phi^k - \phi^{k-1}\|_2 < \epsilon$.

where $\Lambda = (\lambda_1, \lambda_2, \boldsymbol{\lambda_3}, \boldsymbol{\lambda_4})$ are the Lagrangian functions, i.e. $\lambda_1(x), \lambda_2(x) \in \mathbb{R}$, $\boldsymbol{\lambda_3}(x)$, $\boldsymbol{\lambda_4}(x) \in \mathbb{R}^n$, and $r_1, \ldots, r_4$ are positive constants. The constraint minimization problem (5) reduces to finding the saddle-point of the augmented Lagrangian energy $\mathcal{L}$ [29]. The solution to the saddle point problem (6) can be approximated by the iterative Algorithm 1.

We initialize ϕ^0, φ^0 with the signed distance function of the initial contour, $u^0 = H(\phi_0)$, $\boldsymbol{q}^0 = \nabla u^0$, $\boldsymbol{p}^0 = \nabla \phi^0$ and the Lagrange multipliers $\Lambda^0 = 0$. At each iteration, an alternating minimization method is used to find an approximate minimizer of $\mathcal{L}(\phi, \varphi, u, \boldsymbol{q}, \boldsymbol{p}, \Lambda^{k-1})$ considering the Lagrange multipliers Λ^{k-1} fixed. Then the Lagrange multipliers are updated with the residual associated to each constraint and the process is repeated until the change of the level set function ϕ falls below a certain threshold ϵ, which happens at convergence.

In general, it is difficult to find the exact minimizer of the minimization problem (7) because the energy (6) is not convex (there is no guarantee of convergence). However, experiments show that a good approximation can be found by the alternating direction method of multipliers [29]. An approximate solution is thus computed by iteratively alternating the minimization of $\mathcal{L}(\phi, \varphi, u, \boldsymbol{q}, \boldsymbol{p}, \Lambda^{k-1})$ with respect to each variable while considering the others fixed. This leads to Algorithm 2.

The next step is to determine the solutions of the five sub-minimization problems (12), (13), (14), (15) and (16), which can actually be computed efficiently.

IV. Sub-Minimization Problems

In this section, we simplify notation by omitting the super-index and the tilde symbol in the sub-minimization problems (12)–(16).

A. Sub-Minimization Problems with respect to ϕ and u

The sub-minimization problems (12) and (14) can be written as follows:

$$\min_{\phi} \int_\Omega \frac{r_1}{2}\left(\phi - \left(\varphi + \frac{\lambda_1}{r_1}\right)\right)^2 + \frac{r_4}{2}\left|\nabla\phi - \left(\boldsymbol{p} + \frac{\boldsymbol{\lambda_4}}{r_4}\right)\right|^2 \quad (17)$$

Algorithm 2 Alternate Minimizations for an Approximate Solution of (7)

1: Initialize $\tilde{\phi}^0 = \phi^{k-1}, \tilde{\varphi}^0 = \varphi^{k-1}, \tilde{u}^0 = u^{k-1}, \tilde{\boldsymbol{q}}^0 = \boldsymbol{q}^{k-1}, \tilde{\boldsymbol{p}}^0 = \boldsymbol{p}^{k-1}$.
2: For $l = 1, \ldots, L$ and and fixed Lagrange multipliers Λ^k, solve the following sub-problems alternatively:

$$\tilde{\phi}^l = \underset{\phi}{\arg\min}\, \mathcal{L}(\phi, \tilde{\varphi}^{l-1}, \tilde{u}^{l-1}, \tilde{\boldsymbol{q}}^{l-1}, \tilde{\boldsymbol{p}}^{l-1}, \Lambda^{l-1}) \quad (12)$$

$$\tilde{\varphi}^l = \underset{\varphi}{\arg\min}\, \mathcal{L}(\tilde{\phi}^l, \varphi, \tilde{u}^{l-1}, \tilde{\boldsymbol{q}}^{l-1}, \tilde{\boldsymbol{p}}^{l-1}, \Lambda^{l-1}) \quad (13)$$

$$\tilde{u}^l = \underset{u}{\arg\min}\, \mathcal{L}(\tilde{\phi}^l, \tilde{\varphi}^l, u, \tilde{\boldsymbol{q}}^{l-1}, \tilde{\boldsymbol{p}}^{l-1}, \Lambda^{l-1}) \quad (14)$$

$$\tilde{\boldsymbol{q}}^l = \underset{\boldsymbol{q}}{\arg\min}\, \mathcal{L}(\tilde{\phi}^l, \tilde{\varphi}^l, \tilde{u}^l, \boldsymbol{q}, \tilde{\boldsymbol{p}}^{l-1}, \Lambda^{l-1}) \quad (15)$$

$$\tilde{\boldsymbol{p}}^l = \underset{\boldsymbol{p}}{\arg\min}\, \mathcal{L}(\tilde{\phi}^l, \tilde{\varphi}^l, \tilde{u}^l, \tilde{\boldsymbol{q}}^l, \boldsymbol{p}, \Lambda^{l-1}) \ \text{ s.t. } |\boldsymbol{p}| = 1. \quad (16)$$

3: Set $(\phi^k, \varphi^k, u^k, \boldsymbol{q}^k, \boldsymbol{p}^k) = (\tilde{\phi}^L, \tilde{\varphi}^L, \tilde{u}^L, \tilde{\boldsymbol{q}}^L, \tilde{\boldsymbol{p}}^L)$.

$$\min_u \int_\Omega w_r u + \frac{r_2}{2}\left(u - \left(H(\varphi) - \frac{\lambda_2}{r_2}\right)\right)^2 + \frac{r_3}{2}\left|\nabla u - \left(\boldsymbol{q} + \frac{\boldsymbol{\lambda_3}}{r_3}\right)\right|^2. \quad (18)$$

The Euler-Lagrange equation of (17) is:

$$(-r_4\Delta + r_1)\phi = -r_4 \operatorname{div} \boldsymbol{p} - \operatorname{div} \boldsymbol{\lambda_4} + r_1\left(\varphi + \frac{\lambda_1}{r_1}\right) \quad (19)$$

which can be solved efficiently by the fast Fourier transform (FFT) as proposed in [40], [41]. Let us denote by $f(i, j)$ a scalar 2D function discretized at the pixel location (i, j) in the discrete image domain $\Omega = [1, N_x] \times [1, N_y]$. We also define the identity operator $\mathcal{I}f(i, j) = f(i, j)$ and shifting operators:

$$\mathcal{S}_x^{\pm} f(i, j) = f(i \pm 1, j)\, \mathcal{S}_y^{\pm} f(i, j) = f(i, j \pm 1), \quad (20)$$

and write the discretization of (19) as:

$$\left[-r_4\left(\mathcal{S}_x^- + \mathcal{S}_x^+ - 4\mathcal{I} + \mathcal{S}_y^- + \mathcal{S}_y^+\right) + r_1\right]\phi(i, j) = h(i, j), \quad (21)$$

whose right hand side is discretized as

$$h(i, j) = -r_4(\mathcal{I} - \mathcal{S}_x^-)p_x(i, j) - r_4(\mathcal{I} - \mathcal{S}_y^-)p_y(i, j) + r_1\varphi(i, j) \quad (22)$$

$$-(\mathcal{I} - \mathcal{S}_x^-)\lambda_{4x}(i, j) - (\mathcal{I} - \mathcal{S}_y^-)\lambda_{4y}(i, j) + \lambda_1(i, j). \quad (23)$$

Then we apply the discrete Fourier transform $\mathcal{F}$, taking into account that the shifting operators in frequency domain correspond to:

$$\mathcal{F}\mathcal{S}_1^{\pm} f\left(y_i, y_j\right) = e^{\pm \imath z_i}\mathcal{F}f\left(y_i, y_j\right) \qquad z_i = \frac{2\pi}{N} y_i \quad (24)$$

$$\mathcal{F}\mathcal{S}_2^{\pm} f\left(y_i, y_j\right) = e^{\pm \imath z_j}\mathcal{F}f\left(y_i, y_j\right) \qquad z_j = \frac{2\pi}{N} y_j. \quad (25)$$

where (y_i, y_j) are the discrete coordinates in the frequency domain. Assuming periodic boundary conditions, expression (21) in Fourier domain now equals:

$$\underbrace{\left[-2r_4\left(\cos(z_i) + \cos\left(z_j\right) - 2\right) + r_1\right]}_{\mathcal{G}}\mathcal{F}\phi\left(y_i, y_j\right) = \mathcal{F}h\left(y_i, y_j\right) \quad (26)$$

and provides us with a closed-form solution $\phi^\star$ of (17):

$$\phi^\star = \mathcal{F}^{-1}(\ \mathcal{F}h/\mathcal{G}\ ) \tag{27}$$

which we can efficiently compute by FFT. It is now straightforward to apply the same procedure to compute the minimizer $u^\star$ of (18), whose Euler-Lagrange equation is:

$$(-r_3\Delta + r_2)u = -w_r - r_3 \operatorname{div} \boldsymbol{q} + \operatorname{div} \boldsymbol{\lambda_3} + r_2\Big(H(\varphi) - \frac{\lambda_2}{r_2}\Big). \tag{28}$$

These minimizations can also be efficiently performed assuming Neumann or Dirichlet boundary conditions, making use in that case of the discrete cosine or sine transforms.

B. Sub-Minimization Problem with respect to φ

The sub-minimization problem (13) can be written as follows:

$$\min_{\varphi} \int_\Omega \frac{r_1}{2}\Big(\varphi - \Big(\phi - \frac{\lambda_1}{r_1}\Big)\Big)^2 + \frac{r_2}{2}\Big(H(\varphi) - \Big(u + \frac{\lambda_2}{r_2}\Big)\Big)^2. \tag{29}$$

Let us call $z = \phi - \frac{\lambda_1}{r_1}$ and $v = u + \frac{\lambda_2}{r_2}$ and observe that the minimization is decoupled for each pixel. Let us define the function we want to minimize as

$$F(\varphi) = \frac{r_1}{2}(\varphi - z)^2 + \frac{r_2}{2}(H_\varepsilon(\varphi) - v)^2. \tag{30}$$

Observe that for practical implementations, the minimization problem (30) involves a smooth approximation H_ε of the Heaviside function. We propose two steps to find quickly a minimizer of (30).

Step 1: Find a solution φ^0 of (30) for $\epsilon = 0$ (i.e. for the distributional/non-smooth Heaviside function). A closed-form solution exists for this problem and can be computed as follows. The first term of (30) is minimized for $\varphi^0 = z$. As the distributional Heaviside function can take only values 0 or 1, the second term of (30) is minimized for $\varphi^0 < 0$ when $v < \frac{1}{2}$ and $\varphi^0 \geq 0$ when $v \geq \frac{1}{2}$. This means that both terms are minimized for $\varphi^0 = z$ if $v < \frac{1}{2}$ and $z < 0$ or $v \geq \frac{1}{2}$ and $z \geq 0$. Otherwise we must choose to minimize the greater of these terms and set $\varphi^0 = 0$ if $E(0) < E(z)$ and $\varphi^0 = z$ otherwise.

Step 2: Find a solution φ^0 of (30) for $\epsilon > 0$ using the standard Newton's method with φ^0 as initialization. The iterative Newton's method for finding the minimizer of (30) is as follows:

$$\varphi^{m+1} = \varphi^m - \frac{F'(\varphi^m)}{F''(\varphi^m)}, \tag{31}$$

with $\varphi^{m=0} = \varphi_0$, in our case it gives

$$\varphi^{m+1} = \varphi^m - \frac{r_1(\varphi^m - z) + r_2(H_\varepsilon(\varphi^m) - v)\,\delta_\varepsilon(\varphi^m)}{r_1 + r_2(H_\varepsilon(\varphi^m) - v)\,\delta'_\varepsilon(\varphi^m) + r_2\delta^2_\varepsilon(\varphi^m)},$$

where δ_ε and δ'_ε are smooth approximations of the Dirac function and its derivative. Finally, we have observed that two iterations are enough to find a good approximation of the minimizer of (29), leading to a technique almost as fast as a closed-form solution.

C. Sub-Minimization Problem with respect to $\boldsymbol{q}$

The sub-minimization problem (15) can be written as follows:

$$\min_{\boldsymbol{q}} \int_\Omega w_b|\boldsymbol{q}| + \frac{r_3}{2}\Big|\boldsymbol{q} - \Big(\nabla u - \frac{\boldsymbol{\lambda_3}}{r_3}\Big)\Big|^2. \tag{32}$$

Let us call $z = \nabla u - \frac{\boldsymbol{\lambda_3}}{r_3}$ and observe that the minimization (32) is decoupled for each pixel. The solution $\boldsymbol{q}^\star$ of (32) is given by soft-thresholding with the shrinkage operator [42]:

$$\boldsymbol{q}^\star = \max\Big\{|z| - \frac{w_b}{r_3}, 0\Big\}\frac{z}{|z|}. \tag{33}$$

D. Sub-Minimization Problem with respect to $\boldsymbol{p}$

The sub-minimization problem (16) can be written as follows:

$$\min_{\boldsymbol{p}} \int_\Omega \frac{r_4}{2}\Big|\boldsymbol{p} - \Big(\nabla\phi - \frac{\boldsymbol{\lambda_4}}{r_4}\Big)\Big|^2 \quad \text{s.t.} \quad |\boldsymbol{p}| = 1. \tag{34}$$

Let us call $z = \nabla\phi - \frac{\boldsymbol{\lambda_4}}{r_4}$ and observe again that the minimization (34) is decoupled for each pixel and is equivalent to the minimization of the following function on the plane $F(\boldsymbol{p}) = \frac{r_4}{2}|\boldsymbol{p} - z|^2 = \frac{r_4}{2}\left(|\boldsymbol{p}|^2 - |z|^2 - 2\boldsymbol{p}\cdot z\right)$.

We can introduce the constraint in $|\boldsymbol{p}| = 1$ in F by writing it $F(\boldsymbol{p}) = \frac{r_4}{2}\left(1 - |z|^2 - 2\boldsymbol{p}\cdot z\right)$. It is obvious that the minimizer of F is of the form $\boldsymbol{p}^\star = \frac{z}{|z|}$, i.e. when vectors z and $\boldsymbol{p}$ have the same orientation, the scalar product $\boldsymbol{p}\cdot z$ reaches a maximum and the constraint $|\boldsymbol{p}| = 1$ is verified. The minimizer $\boldsymbol{p}^\star$ of (34) is then given by:

$$\boldsymbol{p}^\star = \frac{r_4}{|r_4\nabla\phi - \lambda_4|}\nabla\phi - \frac{1}{|r_4\nabla\phi - \lambda_4|}\lambda_4. \tag{35}$$

V. Experiments and Discussion

Sections V-A and V-B present the results in image segmentation and surface reconstruction obtained with the proposed method. Section V-C compares our algorithm with existing distance preserving LSMs. The code for our algorithm is available online at matlab exchange file and *http://lts5srv2.epfl.ch/~estellers/*.

A. Image Segmentation and Surface Reconstruction

We apply Algorithm 2 for image segmentation by defining the edge and region terms proposed in [20], [22], which have been extended to handle both gray level and color images. Figure 1 shows the results obtained for different images from the Berkeley[1], Weizmann[2] and GrabCut[3] databases. The method behaves as expected, providing the same results as redistancing or penalty methods in terms of final segmentation, but with a considerable speed-up in time.

We also use the proposed model to successfully reconstruct several surfaces from the Stanford dataset[4]. We initialize the method with a sphere containing all the data points and

[1] http://www.eecs.berkeley.edu/vision.

[2] http://www.wisdom.weizmann.ac.il/~vision.

[3] http://research.microsoft.com/en-us/um/cambridge/projects/visionimagevideoediting/segmentation/grabcut.htm.

[4] http://graphics.stanford.edu/data/3Dscanrep/.

Fig. 1. Proposed level set-based segmentation method applied to natural images. For each result, we show the segmentation result and the level set function. We plot the initial zero level set in blue, the final contour in pink, and the $\pm 1, \pm 2$ level sets of the final function in black.

recompute the region term $w_r^{in} = -\operatorname{div} \hat{\boldsymbol{n}}$ every 5 iterations as described in Section II. Unlike [34], [36], we do not need to estimate *a priori* the surface normal at the data points, as the surface normal at the points is directly estimated from the current value of the LSF. Finally, the process is sped up with a standard multi-resolution approach, see Figure 2. The final reconstructed surfaces are shown in figures 3 and 4.

B. Cortex Segmentation With Coupled Surfaces

Cortex segmentation [43] is a problem that requires the distance information to be solved successfully. Based on [10], we can develop a segmentation algorithm for the cortical grey matter with two active contours coupled by their relative distance, which constrains the thickness of the cerebral cortex. Graph cut methods and convex relaxation methods cannot be directly applied to solve this problem because they use the binary function for representing the contour and binary functions do not hold the distance information.

The cerebral cortex is the layer of the brain bounded by the outer and inner cortical surfaces, that is, the outer interface between cerebral spinal fluid (CSF) and grey matter and the inner interface between grey and white matter. Locating this cortical surface is a first step in many brain imaging processes and measuring its thickness is a common procedure in the diagnosis of many neurological diseases. We will see that the use of SDF in the segmentation of cortical surface allows us to include information about the cortical structure into the segmentation problem and, at the same time, provides an estimate of the cortical thickness.

In order to extract the cortical layer we need to extract its two bounding surfaces C^1 and C^2. In theory, therefore, we could simply detect each of its bounding surfaces or segment the regions defined by the white matter and the exterior parts of the brain by independent minimization of the following functionals associated to C^1 and C^2

$$\min_{C^1} \mathcal{E}\left(w_b, w_{r1}, C^1\right) = \int_{C^1} w_b(s)ds + \int_{C^1_{in}} w_{r1}(x)dx, \quad (36)$$

$$\min_{C^2} \mathcal{E}\left(w_b, w_{r2}, C^2\right) = \int_{C^1} w_b(s)ds + \int_{C^2_{in}} w_{r2}(x)dx, \quad (37)$$

where w_b is a boundary indicator and w_{r1}, w_{r2} are region

(a) (b) (c)

Fig. 2. Reconstructed bunny from the Stanford dataset at different resolutions. Linear interpolation of the SDF obtained at lower resolutions was used as initial LSF for higher resolutions. (a) Grid $30 \times 30 \times 30$, time 2.4 s. (b) Grid $60 \times 60 \times 60$, time 10.6 s. (c) Grid $120 \times 120 \times 120$, time 50.5 s.

descriptors for the white matter and the exterior areas of the brain. In practice, however, the boundaries between grey and white matter are not clear, MRI images suffer from intensity nonuniformity and the segmentations obtained with local region descriptors and boundary detectors do not correctly locate the cortical layer.

To overcome these limitations, it has been proposed to incorporate a constraint on the cortical structure to obtain a better segmentation. For simplicity, we modify the model proposed in [44] to suit our variational formulation. An equivalent method is obtained taking a Bayesian approach on [45], and similarly other models for cortex segmentation [46] could be adopted. We use a coupled surface model, where a functional is minimized when C^1 captures the CSF-grey matter interface, C^2 the grey-white matter boundary and the distance between them is close to the expected cortical thickness d (about 3mm). The problem is written in terms of LSM, making use of the SDF ϕ_1 and ϕ_2 to define the bounding surfaces C^1 and C^1. The functional to minimize is then given by

$$\min_{\phi_1,\phi_2} \mathcal{E}\left(w_b, w_{r1}, \phi_1\right)+\mathcal{E}\left(w_b, w_{r2}, \phi_2\right)+\frac{c}{2}\int_\Omega \left(\phi_1-\phi_2-d\right)^2 \tag{38}$$

subject to the constraints $|\nabla\phi_1| = 1$, $|\nabla\phi_2| = 1$. The term $(\phi_1 - \phi_2 - d)^2$ penalizes segmentations where the distance between the bounding surfaces differs from the expected cortical thickness d. Indeed, when ϕ_1 and ϕ_2 are the SDF defined by C^1 and C^2, the distance between the surfaces can be measured on the whole domain by $\phi_1 - \phi_2$ and, consequently, the term $(\phi_1 - \phi_2 - d)^2$ drives the segmentation to solutions where the distance between the surfaces is consistent with the cortical structure.

The minimization technique presented in Section IV can be directly applied to this problem. The same splitting variables and Lagrange multipliers are now defined and solved for each level set function and only the alternate minimization w.r.t ϕ_1 and ϕ_2 are modified. The minimization problems w.r.t ϕ_1 (the analogous applies to ϕ_2) is now

$$\min_{\phi_1} \int_\Omega \frac{r_1}{2}\left(\phi_1 - \left(\varphi_1 + \frac{\lambda_{1,1}}{r_1}\right)\right)^2 + \frac{c}{2}\left(\phi_1 - \phi_2 - d\right)^2 + \frac{r_4}{2}\left|\nabla\phi_1 - \left(\boldsymbol{p}_1 + \frac{\boldsymbol{\lambda_{1,4}}}{r_4}\right)\right|^2, \tag{39}$$

and its associated Euler-Lagrange equation

$$\begin{aligned}(-r_4\Delta + r_1 + c)\phi_1 = {} & -r_4 \operatorname{div} \boldsymbol{p} - \operatorname{div} \boldsymbol{\lambda_4} + r_1\left(\varphi + \frac{\lambda_1}{r_1}\right) \\ & + c\left(\phi_2 + d\right)\end{aligned} \tag{40}$$

can be solved efficiently by the fast Fourier transform as we have already explained in Section IV.

We have applied this technique in an illustrative experiment to segment the cortical layer in different slices of MRI images. Results are shown in Figure 5(b). We observe that coupling the level set functions and constraining the expected cortical thickness in the segmentation procedure produces a segmentation result (Fig. 5) close to the ground truth (Fig. 5)(d). The segmentation obtained without coupling (Fig. 5(c)) is not able to find the fine structures.

C. Comparison to Other Distance Preserving Level Set Methods

In this section, we compare our proposed algorithm to the other techniques designed to preserve the SDF in the LSM. To that purpose, we first remind the necessity of preserving the LSF as an SDF in the proposed segmentation task with a simple example. We initialize the LSF as an SDF and evolve it with the PDE techniques associated to the LSM for the image segmentation problem (4). The level set function becomes too steep around its zero level set after a few iterations, and stops its evolution because the geometric features (i.e. curvature and normal) are not correctly estimated, see Figure 6. Two common approaches have been introduced to overcome this problem, either by re-distancing the LSF or by maintaining the SDF during the contour evolution (as proposed in our method).

We will compare three different approaches in the case of image segmentation: our method, the standard re-distancing approach and the methods of Li *et al.* [14] and [15]. The re-distancing process is carried out with the fast marching method [47], while the methods of Li *et al.* are defined by introducing a penalty term $\mathcal{P}$ in the energy to encourage the LSF to be close to an SDF as follows:

$$\int_\Omega w_b|\nabla H(\phi)| + w_r H(\phi) + \frac{\mu}{2}\mathcal{P}\left(|\nabla\phi|\right), \tag{41}$$

where $\mathcal{P}(x)$ is a potential function with a minimum at $x = 1$. In [14] the authors define the potential $\mathcal{P}_1(x) = (x-1)^2$

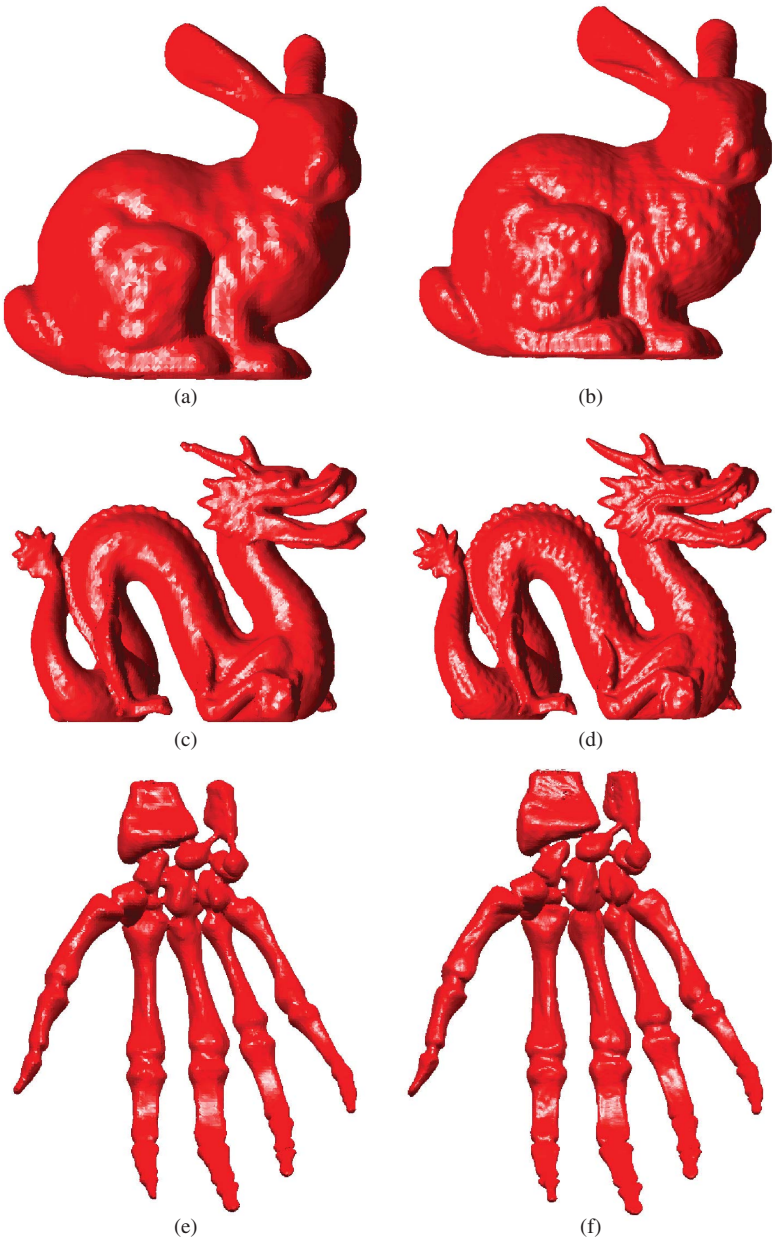


Fig. 3. Reconstructed surfaces from scattered data points at different resolutions. (a) Grid $120 \times 120 \times 120$, time 50 s. (b) Grid $240 \times 240 \times 240$, time 391 s. (c) Grid $200 \times 160 \times 120$, time 151 s. (d) Grid $490 \times 320 \times 240$, time 1076 s. (e) Grid $200 \times 160 \times 120$, time 173 s. (f) Grid $490 \times 320 \times 240$, time 1096 s.

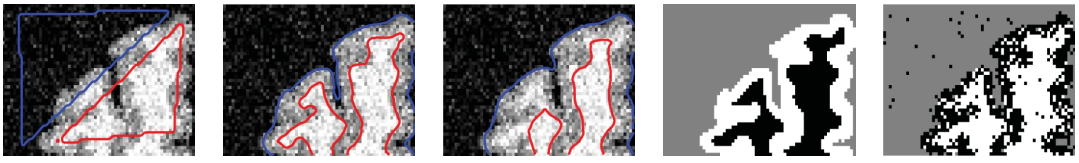
penalizing deviations from SDF on the whole domain and encouraging the SDF property $|\nabla\phi| = 1$. Later, the authors modify this potential to encourage the SDF property close to the zero level set and flat LSF elsewhere, i.e. $\mathcal{P}_2(x)$ is defined by a double-well potential with two minima at $x = 1$ and $x = 0$. We refer the reader to [14], [15] for more details.

Our algorithm is presented on Figures 7(a)–7(e). Figures 7(f)–7(j) show the re-distancing method [47]. Although the final segmentation and the final LSF provide the desired results, the periodic re-initialization process produces a non-smooth minimization of the level set energy (observe the jumps in the energy plots that appear each time we reinitialize the LSF as an SDF). Besides, we remind that we do not know in general when to re-initialize the LSF as an SDF. In our experiments, we applied the re-initialization every 5 iterations. We also remind that the re-distancing process is not guaranteed to exactly preserve the location of the zero level set representing the moving interface, which results in a non-smooth minimization of the segmentation energy.

Next, we will consider the method of Li *et al.* [14], which is more related to our method. Li *et al.*introduced the nice idea to constrain the LSF to be close to an SDF via a penalty term. However, the penalty term does not constrain exactly the LSF to be an SDF. Besides, the value of the penalty constant μ is a trade-off between speed and accuracy. A small penalty value μ in (41) provides a fast algorithm but does not preserve faithfully an SDF, leading to an LSF with small instabilities, while a large penalty value μ provides a better SDF but slows down significantly the minimization process as the number of iterations to reach the convergence state

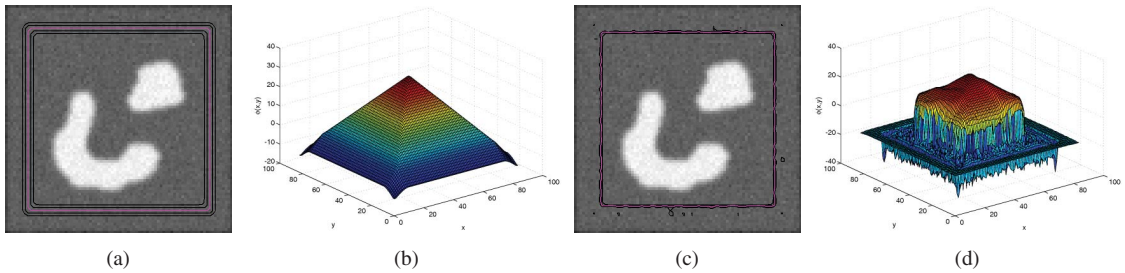
(a) (b) (c) (d)

Fig. 4. Reconstructed surfaces from scattered data points at different resolutions. (a) Grid $120 \times 240 \times 120$, time 173 s. (b) Grid $240 \times 480 \times 240$, time 940 s. (c) Grid $60 \times 60 \times 120$, time 46 s. (d) Grid $120 \times 120 \times 240$, time 315 s.



(a) (b) (c) (d) (e)

Fig. 5. Segmentation of grey–white matter interface on MRI images of human cortex. The segmentation obtained with the proposed method V-B is clearly closer to the ground truth V-B than the results obtained when no coupling terms is considered and the segmentation is performed independently for the inner and outer cortical surfaces V-B. The segmentation V-B obtained with three phases of the k-mean algorithm fails because region descriptors alone cannot segment grey-matter. (a) Initialization. (b) Coupled level set functions. (c) Noncoupled level set functions. (d) Ground truth. (e) k-mean algorithm.



(a) (b) (c) (d)

Fig. 6. Level set function ($\phi = 0$ in pink and $\phi = \pm 1, \pm 2$ in dark) develops irregularities during the propagation and stops after a few iterations if the LSF is not constrained to be (or at least close to) an SDF. (a) Initial contour defined by level set function. (b) Initial level set function. (c) Final contour defined by level set function. (d) Final level set function.

increases considerably. Figures 7(k)–7(o) present the results with a small value μ and Figures 7(p)–7(t) show the results with a large value μ. We observe that for small values of μ the method converges faster, but the LSF differs from an SDF and presents peaks and valleys close to the zero-level set (small island-like contours in black in Figures 7(k)–7(o)) which might

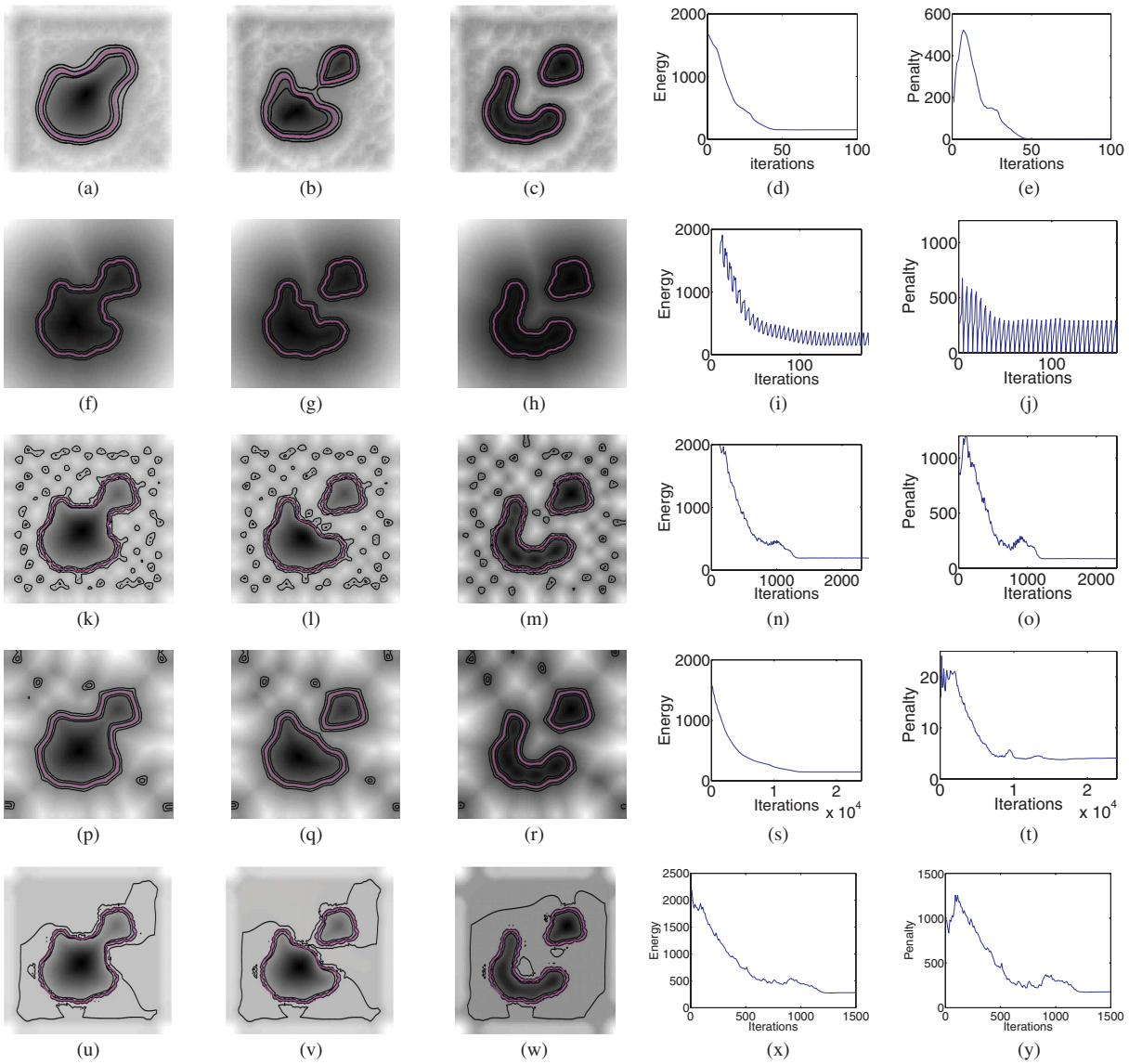


Fig. 7. Comparison of the evolution of the level set function ϕ. The first row corresponds to our algorithm, the second row to the re-distancing procedure [47], the fifth row to Li *et al*.algorithm [14] with a small penalty and large timestep, the fourth row to a large penalty and small timestep and the fifth row to Li *et al*. algorithm [15] with a large timestep. The first three columns represent the evolution of the LSF ϕ at two intermediate and at the final iteration. The zero level set $\phi = 0$ is shown in magenta, and the iso level sets $\phi = \pm 1, \pm 2$ are shown in black. While only the 0 isocontours actually define the segmentation, the position of the $\pm 1, \pm 2$ isocontours shows the quality of the SDF approximation and points out to possible instabilities in the LSF. The fourth column plots the energy $\int_\Omega w_b|\nabla H(\phi)| + w_r H(\phi) + (1/2)(|\nabla\phi| - 1)^2$ and the last column shows the evolution of the penalty energy $\int(|\nabla\phi| - 1)^2$ (closeness measure between LSF and SDF) with respect to iterations. Our method provides the best tradeoff between speed and preservation of the distance function. (a) 15 iterations, 0.12 s. (b) 19 iterations, 0.14 s. (c) 30 iterations, 0.21s. (d) Energy evolution with respect to iterations. (e) Penalty energy with respect to iterations. (f) 65 iterations, 1.04s. (g) 85 iterations, 1.35 s. (h) 120 iterations, 1.84 s. (i) Energy evolution with respect to iterations. (j) Penalty energy with respect to iterations. (k) 750 iterations, 0.85 s. (l) 950 iterations, 1.35 s. (m) 1500 iterations, 1.84 s. (n) Energy evolution with respect to iterations. (o) Penalty energy with respect to iterations. (p) 8, 000 iterations, 8.98 s. (q) 10 200 iterations, 11.37 s. (r) 20 000 iterations, 22.02 s. (s) Energy evolution with respect to iterations. (t) Penalty energy with respect to iterations. (u) 500 iterations, 1.27s. (v) 750 iterations, 1.89 s. (w) 1500 iterations, 2.60 s. (x) Energy evolution with respect to iterations. (y) Penalty energy with respect to iterations.

lead to instabilities and eventually affect the location of the zero level contours (in pink). For that reason the authours improve this method in [15] by the use of a double-well potential $\mathcal{P}_2$, which encourages the LSF to be an SDF close to the zero level set and flat elsewhere. The resulting method, Figures 7(u)–7(y), avoid the instabilities associated to [14] when a small penalty is used, but it does not really provide a SDF. In terms of speed [15] it is as fast as [14] with a small penalty and without potential instabilities, but it is still limited by the CFL conditions and, consequently, is slower than the method that we propose.

Our proposed method overcomes the limitations of Li *et al.*, as our formulation constrains the LSF to be an SDF, and the proposed algorithm is fast because there is no need to assign a large penalty constant to the penalty term. Indeed, Figures 7(o), 7(t) and 7(e) show that our method keeps more faithfully the LSF as an SDF because the penalty energy is lower, by at least one order of magnitude (the minimum of the penalty energy is around 1 with our method, around 5 with Li *et al.*'s method with a large μ, and around 100 with Li *et al.*'s method with a small μ). Our method is almost as accurate as redistancing, when it comes to preserving the SDF,

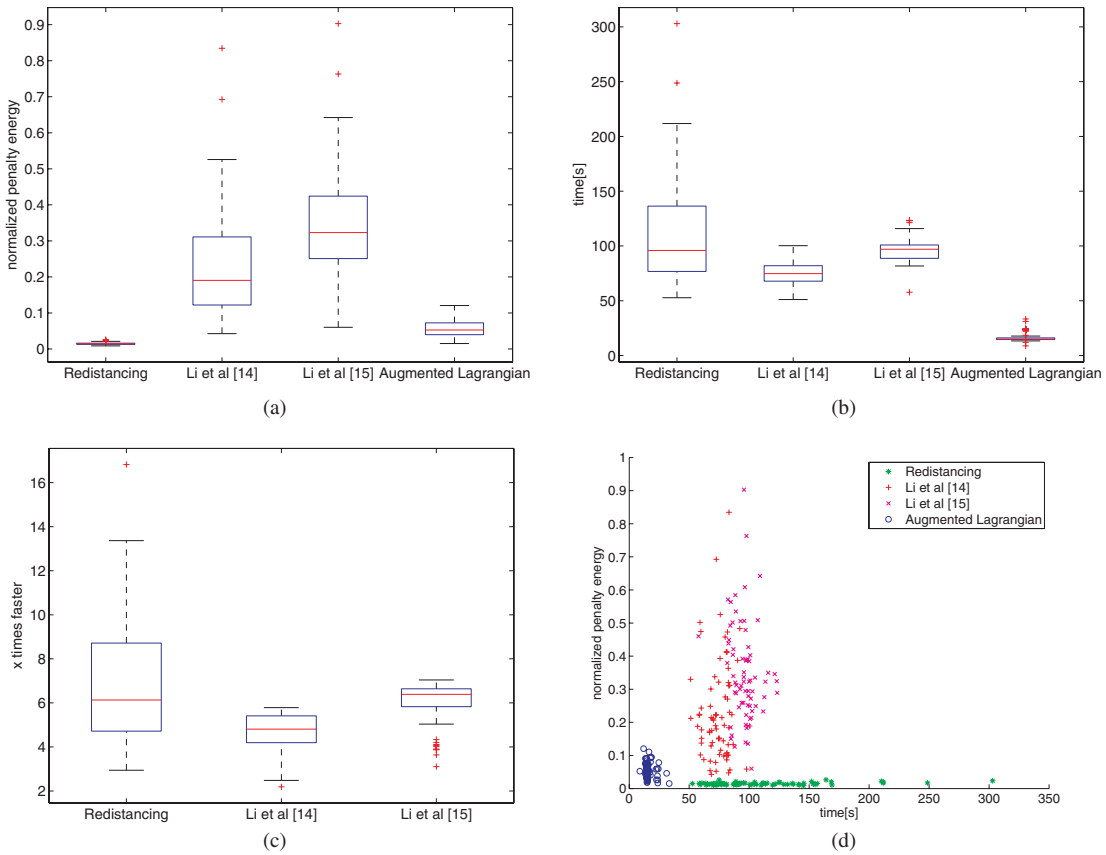


Fig. 8. Comparison of the quality of segmentation and speed for the different methods on a dataset of 72 images. Quality of the LSF is measured in terms of the penalty term at convergence $(1/|\Omega|)\int(|\nabla\phi|-1)^2$ when the obtained contours are equivalent. Our method preserves the SDF almost as well as redistancing, clearly better than Li *et al.*method and is faster than any of them. (a) LSF quality for the different methods. (b) Segmentation time for the different methods. (c) Speed improvement with our method. (d) Quality against speed for the three LSM.

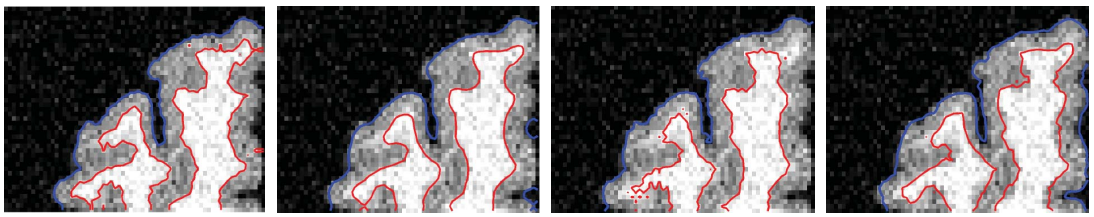


Fig. 9. Segmentation of grey–white matter interface on MRI images of human cortex. The segmentation model (38) is solved with different approaches: redistancing in Fig. 9, with Li *et al.* methods [14] in Fig. 9, and [15] in Fig. 9 and with the proposed algorithm in Fig. 9. (a) Redistancing: 5.3% misclassification in 17.1 s. (b) Li *et al.* [14]: 5.8% misclassification in 4.6 s. (c) Li *et al.* [15]: 7.0% misclassification in 4.5 s. (d) Our method: 5.8% misclassification in 3.8 s.

and overall faster than redistancing and Li's method, as shown in the next experiment with more detail. These advantages are provided by the augmented Lagrangian approach, which can preserve accurately the constraint while keeping a good minimization speed.

To quantify the improvement obtained with our method with respect to the other distance preserving LSM techniques, we have used the previous algorithms to segment 72 images from the Berkeley, Weizmann and GrabCut databases. We compare on Figure 8 the different algorithms in terms of the quality of preserving the distance function and speed. Figure 8 presents the values of the penalty terms at convergence for the 72 segmented images, showing that our method preserves the SDF almost as well as the redistancing method and clearly better than Li *et al.*'s methods (with a relatively small penalty μ, i.e. large time steps). Indeed, the penalty values for our method and the redistancing method are similar, while being an order of magnitude smaller than Li *et al.*'s method. We also compare the time for each method to converge, which is assumed when $\frac{||\phi-\phi^*||_2}{||\phi^*||_2} < \epsilon$. We observe in Figure 8 and Figure 8 that our algorithm is on average 5 to 6 times faster than the redistancing's or Li's methods for all images. Finally, we present on Figure 8(d) a scatter plot of the penalty obtained at convergence against the time required to converge, which shows that our method presents a good trade-off between accuracy of SDF preservation and speed.

Finally, in order to compare the different methods in terms of accuracy of segmentation, we have tested the surface coupling of section V-B with the different methods for a synthetic image. In Figure 9 we show the resulting segmentations with the different methods and measure their performance in terms of grey-white matter tissue classification. While redistancing obtains the best performance (5.3% of errors), it is the slower option (17.1s). The two methods of Li *et al*.'s take 4.5s and 4.6s to converge, but [14] (5.8% of errors) performs better than [15] (7.0% of errors) because it encourages an SDF everywhere and is able to consider the topology of the brain over the whole image domain. Finally, our method provides the best trade-off between speed and segmentation accuracy as it takes 3.8s to converge and misclassifies 5.8% of the samples.

VI. Conclusion

We have introduced an efficient algorithm for distance preserving level set methods which overcomes the main numerical limitations of the original level set method, i.e. the speed and the preservation of the distance function. Although other fast optimization techniques have been developed for the level set method [5]–[8], [18], they cannot preserve the level set function as a distance function, which is essential in some applications like surface reconstruction, medical image segmentation or segmentation with higher order geometric features [48], [49]. Finally, observe that the proposed algorithm can be sped up significantly using a narrow-band approach and a parallelized version of the algorithm on graphics processing units (GPUs).

Acknowledgment

The authors would like to thank M. B. Cuadra for kindly providing the MRI images, and Li and his coauthors for making their code available.

References

[1] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.

[2] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. New York: Springer-Verlag, 2003.

[3] S. Osher and N. Paragios, *Geometric Level Set Methods in Imaging, Vision, and Graphics*. New York: Springer-Verlag, 2003.

[4] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[5] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

[6] T. Chan, S. Esedoglu, and M. Nikolova, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM J. Appl. Math.*, vol. 66, no. 5, pp. 1632–1648, 2006.

[7] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher, "Fast global minimization of the active contour/snake models," *J. Math. Imag. Vis.*, vol. 28, no. 2, pp. 151–167, 2007.

[8] T. Goldstein, X. Bresson, and S. Osher, "Geometric applications of the split bregman method: Segmentation and surface reconstruction," *J. Sci. Comput.*, vol. 45, nos. 1–3, pp. 272–293, 2009.

[9] R. Courant, K. Friedrichs, and H. Lewy, "On the partial difference equations of mathematical physics," *IBM J.*, vol. 11, no. 2, pp. 215–234, Mar. 1967.

[10] J. Gomes and O. Faugeras, "Reconciling distance functions and level sets," in *Scale-Space Theories in Computer Vision*. New York: Springer-Verlag, 1999, pp. 70–81.

[11] M. Sussman, P. Smereka, and S. Osher, "A level set approach for computing solutions to incompressible two-phase flow," *J. Comput. Phys.*, vol. 114, no. 1, pp. 146–159, 1994.

[12] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proc. Nat. Acad. Sci.*, vol. 93, no. 4, pp. 1591–1595, Feb. 1996.

[13] J. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Trans. Autom. Control*, vol. 40, no. 9, pp. 1528–1538, Sep. 1995.

[14] C. Li, C. Xu, C. Gui, and M. Fox, "Level set evolution without re-initialization: A new variational formulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 430–436.

[15] C. Li, C. Xu, C. Gui, and M. Fox, "Distance regularized level set evolution and its application to image segmentation," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3243–3254, Dec. 2010.

[16] D. L. Chopp, "Computing minimal surfaces via level set curvature flow," *J. Comput. Phys.*, vol. 106, no. 1, pp. 77–91, 1993.

[17] L. Chunming, X. Chenyang, K. Konwar, and M. Fox, "Fast distance preserving level set evolution for medical image segmentation," in *Proc. Int. Conf. Control, Autom., Robot. Vis.*, Dec. 2006, pp. 1–7.

[18] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Fast geodesic active contours," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1467–1475, Oct. 2002.

[19] J. Weickert, B. Romeny, and M. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 398–410, Mar. 1998.

[20] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comput. Vis.*, vol. 22, no. 1, pp. 61–79, 1997.

[21] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Gradient flows and geometric active contour models," in *Proc. Int. Conf. Comput. Vis.*, 1995, pp. 810–815.

[22] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[23] O. Faugeras and R. Keriven, "Level set methods and the stereo problem," in *Scale-Space Theory in Computer Vision*. New York: Springer-Verlag, 1997, pp. 272–283.

[24] N. Paragios and R. Deriche, "Geodesic active contours for supervised texture segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun. 1999, pp. 1–6.

[25] M. Leventon, W. Grimson, and O. Faugeras, "Statistical shape influence in geodesic active contours," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2000, pp. 316–323.

[26] H. Zhao, S. Osher, B. Merriman, and M. Kang, "Implicit and nonparametric shape reconstruction from unorganized data using a variational level set method," *Comput. Vis. Image Understand.*, vol. 80, no. 3, pp. 295–314, 2000.

[27] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 323–343, 2009.

[28] R. Glowinski and A. Marroco, "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires. Revue française d'automatique, informatique, recherche opérationnelle," *Anal. Numér.*, vol. 9, no. 2, pp. 41–76, 1975.

[29] R. Glowinski and P. L. Tallec, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*. Philadelphia, PA: SIAM, 1989.

[30] P.-L. Lions and B. Mercier, "Splitting algorithms for the sum of two nonlinear operators," *SIAM J. Numer. Anal.*, vol. 16, no. 6, pp. 964–979, 1979.

[31] J. Bioucas-Dias and M. Figueiredo, "A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2992–3004, Dec. 2007.

[32] S. Setzer, "Operator splittings, Bregman methods and frame shrinkage in image processing," *Int. J. Comput. Vis.*, vol. 92, no. 3, pp. 265–280, 2011.

[33] P. Savadjiev, F. Ferrie, and K. Siddiqi, "Surface recovery from 3D point data using a combined parametric and geometric flow approach," in *Proc. Energy Minimizat. Methods Comput. Vis. Pattern Recognit.*, 2003, pp. 325–340.

[34] V. Lempitsky and Y. Boykov, "Global optimization for shape fitting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[35] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *Comput. Graph.*, vol. 26, no. 2, pp. 71–78, 1992.

[36] J. Ye, X. Bresson, T. Goldstein, and S. Osher, "A fast variational method for surface reconstruction from sets of scattered points," Dept. Comput. Appl. Math., UCLA, Los Angeles, Tech. Rep. 10-01, 2010.
[37] R. Kimmel and A. Bruckstein, "Regularized Laplacian zero crossings as optimal edge integrators," *Int. J. Comput. Vis.*, vol. 53, no. 3, pp. 225–243, 2003.
[38] A. Vasilevskiy, "Flux maximizing geometric flows," *Pattern Anal. Mach.*, vol. 24, no. 12, pp. 1565–1578, Dec. 2002.
[39] M. Jacob, T. Blu, and M. Unser, "Efficient energies and algorithms for parametric snakes," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1231–1244, Sep. 2004.
[40] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, 2008.
[41] X.-C. Tai, J. Hahn, and G. Chung, "A fast algorithm for Euler's elastica model using augmented Lagrangian method," Dept. Comput. Appl. Math., UCLA, Los Angeles, Tech. Rep. 10-47, 2010.
[42] D. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 2002.
[43] X. Han, C. Xu, and J. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 755–768, Jun. 2003.
[44] X. Zeng, L. Staib, R. Schultz, and J. Duncan, "Segmentation and measurement of the cortex from 3-D MR images using coupled-surfaces propagation," *IEEE Trans. Med. Imag.*, vol. 18, no. 10, pp. 927–937, Oct. 1999.
[45] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Cortex segmentation: A fast variational geometric approach," *IEEE Trans. Med. Imag.*, vol. 21, no. 12, pp. 1544–1551, Dec. 2002.
[46] D. MacDonald, N. Kabani, D. Avis, and A. Evans, "Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI," *Neuroimage*, vol. 12, no. 3, pp. 340–356, 2000.
[47] D. Adalsteinsson and J. Sethian, "The fast construction of extension velocities in level set methods," *J. Comput. Phys.*, vol. 148, no. 1, pp. 2–22, 1999.
[48] L. Lorigo, W. Grimson, O. Faugeras, R. Keriven, R. Kikinis, A. Nabavi, and C. Westin, "Codimension-two geodesic active contours for the segmentation of tubular structures," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2000, pp. 444–451.
[49] M. Rochery, I. Jermyn, and J. Zerubia, "Higher order active contours," *Int. J. Comput. Vis.*, vol. 69, no. 1, pp. 27–42, 2006.

Virginia Estellers received the B.S. and M.S. degrees in mathematics and electrical engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2008. She is currently pursuing the Ph.D. degree with the École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

She has been a Research and Teaching Assistant with the Signal Processing Laboratory, EPFL, since September 2008. Her current research interests include numerical partial differential equations and variational methods for image processing.

Dominique Zosso (S'06–M'11) was born in Bern, Switzerland, in 1983. He received the M.Sc. degree in electrical and electronics engineering and the Ph.D. degree from the École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, in 2006 and 2011, respectively.

He was a Researcher with the Structural Bioinformatics Group, Swiss Institute of Bioinformatics and Biozentrum, University of Basel, Basel, Switzerland. From 2007 to 2012, he was a Research and Teaching Assistant with the Signal Processing Laboratory, EPFL. He is currently a Post-Doctoral Scholar with the Department of Mathematics, University of California, Los Angeles, with L. Vese and A. Bertozzi. His current research interests include partial differential equations and variational models for inverse problems in image processing and computer vision, and efficient algorithms to solve them.

Rongjie Lai received the B.S. degree in mathematics from the University of Science and Technology of China, Hefei, China, in 2003, the M.S. degree in mathematics from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2006, and the Ph.D. degree in applied mathematics from the University of California, Los Angeles, in 2010.

He is currently an NTT Assistant Professor with the University of Southern California, Los Angeles. His current research interests include computational differential geometry and calculus of variation and numerical partial differential equation methods for image and surface processing.

Stanley Osher received the Ph.D. degree from the Courant Institute of Mathematical Sciences, New York University, New York, in 1966.

He is a Professor of mathematics, computer science, and electrical engineering with the University of California, Los Angeles. He is an Associate Director of the National Science Foundation-funded Institute for Pure and Applied Mathematics, Los Angeles, CA. He is one of the top 25 most highly cited researchers in mathematics and computer sciences. He is a co-founder of three successful companies, each based largely on his own (joint) research. His current research interests include information science, specifically, image processing, compressed sensing, and machine learning.

Dr. Osher is a member of the National Academy of Sciences and the American Academy of Arts and Sciences. He is a recipient of numerous academic honors.

Jean-Philippe Thiran (SM'93) was born in Namur, Belgium, in August 1970. He received the Bachelor's and Master's degrees in electrical engineering and the Ph.D. degree from the Universitè Catholique de Louvain, Louvain-la-Neuve, Belgium, in 1993 and 1997, respectively.

He joined the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in February 1998, as a Senior Lecturer. From 2004 to 2011, he was an Assistant Professor (tenure track), responsible for the Image Analysis Group (LTS5). Since 2011, he has been an Associate Professor of signal processing with EPFL. He is involved in a 20% Associate Professor position with the Department of Radiology, University Hospital Center and University of Lausanne, Lausanne. From 2001 to 2005, he was the Co-Editor-in-Chief of the *Signal Processing International Journal* (Elsevier Science). He has authored or co-authored nine book chapters, 95 journal papers, and more than 150 peer-reviewed papers published in proceedings of international conferences. He holds five international patents. His current research interests include image segmentation, analysis, and multimodal signal processing, with application to medical imaging, remote sensing, face image analysis, and human–computer interaction.

Dr. Thiran is currently an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He was the General Chairman of the European Signal Processing Conference in 2008 and will be a Technical Co-Chair of the IEEE International Conference on Image Processing in 2015. He is a member of the Image, Video, and Multidimensional Signal Processing Technical Committee from 2009 to 2014 of the IEEE Signal Processing Society.

Xavier Bresson received the B.S. degree in theoretical physics from the University of Marseille, Marseille, France, in 1998, the M.Sc. degree in electrical engineering from École Supérieure d'Électricité, Paris, France, the M.Sc. degree in signal processing from the University of Paris XI, Orsay, France, in 2000, and the Ph.D. degree in computer vision from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 2005.

He was a Post-Doctoral Scholar with the Department of Mathematics, University of California, Los Angeles, from 2006 to 2010. In 2010, he joined the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, as an Assistant Professor. He has authored or co-authored more than 39 papers in international journals and conferences. His current research interests include continuous relaxation methods for imaging science and machine learning.