MODELING SEGMENTATION VIA GEOMETRIC DEFORMABLE REGULARIZERS, PDE AND LEVEL SETS IN STILL AND MOTION IMAGERY: A REVISIT

JASJIT SURI, DEE WU and LAURA REDEN

MR Clinical Research Division, Philips Medical Systems, Inc.,
Cleveland, OH 44143, USA

JIANBO GAO

KLA-Tencor, Milpitas, CA 95035, USA

SAMEER SINGH

PANN Research, Department of Computer Science,
University of Exeter, Exeter, EX4 4PT, UK

SWAMY LAXMINARAYAN

Department of Biomedical Engineering/Department of Medical Informatics,
NJIT, Newark, NJ 07102, USA

Partial Differential Equations (PDEs) have dominated image processing research recently. The three main reasons for their success are: first, their ability to transform a segmentation modeling problem into a partial differential equation framework and their ability to embed and integrate different regularizers into these models; second, their ability to solve PDEs in the level set framework using finite difference methods; and third, their easy extension to a higher dimensional space.

This paper is an attempt to survey and understand the power of PDEs to incorporate into geometric deformable models for segmentation of objects in 2D and 3D in *still* and *motion* imagery. The paper first presents PDEs and their solutions applied to image diffusion. The main concentration of this paper is to demonstrate the usage of regularizers in PDEs and level set framework to achieve the image segmentation in *still* and *motion* imagery. Lastly, we cover miscellaneous applications such as: mathematical morphology, computation of missing boundaries for shape recovery and low pass filtering, all under the PDE framework. The paper concludes with the merits and the demerits of PDEs and level set-based framework for segmentation modeling. The paper presents a variety of examples covering both synthetic and real world images.

Keywords: Partial Differential Equations (PDEs); level sets; deformable models; medical imaging; filtering; low level vision; regularization; segmentation; motion estimation; topology; shapes.

1. Introduction

Partial Differential Equations (PDEs)[a] have recently dominated the fields of computer vision, image processing and applied mathematics for the following reasons: their ability to transform a segmentation modeling problem into a PDE framework; their ability to embed and integrate regularizers into these models; their ability to solve PDEs using finite difference methods (FDM); their ability to link between PDEs and the level set framework for implementing finite difference methods; their ability to extend the PDE framework from 2D to 3D or even higher dimensions; their ability to control the degree of PDE in the image processing domain; their ability to provide solutions in a fast, stable and closed form; and lastly, their ability to interactively handle image segmentation in the PDE framework.

Application of PDE has recently become more prominent in the biomedical and non-biomedical imaging fields (see Suri *et al.*,[1−7] Haker,[8] Chambolle[9] and Morel *et al.*[10]) for shape recovery and the recently published book by Sapiro.[11] This is because the role of shape recovery has always been a critical component in 2D and 3D medical and non-medical imagery. This assists largely in medical therapy and object detection/tracking in industrial applications, respectively (see the recent book by Suri *et al.*[4] and the references therein and also see Weickert *et al.*[12,13] and the references therein). Shape recovery of medical organs in medical images is more difficult compared to other imaging fields. This is primarily due to the large shape variability, structure complexity, different kinds of artifacts and restrictive body scanning methods. With PDE-based segmentation techniques, it has been possible to integrate low level vision techniques to make the segmentation system robust, reliable, fast, closed-form and accurate. This paper revisits the application of PDE in the field of computer vision and image processing and demonstrates the ability to model segmentation in PDE and the level set framework. Before discussing PDE techniques in detail, we will first discuss the different kinds of PDE applications.

Figure 1 shows the classification tree of non-coupled and coupled PDE applications. Even though the applications are very large in number, we have narrowed them down for the CVGIP (Computer Vision, Graphics and Image Processing) domain only. Although the tree shows applications such as image smoothing/filtering, image segmentation, optic flow, mathematical morphology, image matching and coupled PDEs, the main focus of this paper is on modeling segmentation using non-coupled and coupled PDEs in the level set framework by fusing geometric regularizers (also known as geometric deformable models). Thus, under the class of geometric segmentation techniques PDEs have become almost an integral part of deformable modeling (see the segmentation classification paper by Suri *et al.*[5] and all the references therein. In Ref. 5 the three major techniques of segmentation were discussed: region-based, boundary-based and the fusion of region and boundary-based. Boundary-based techniques were further classified into parametric

---

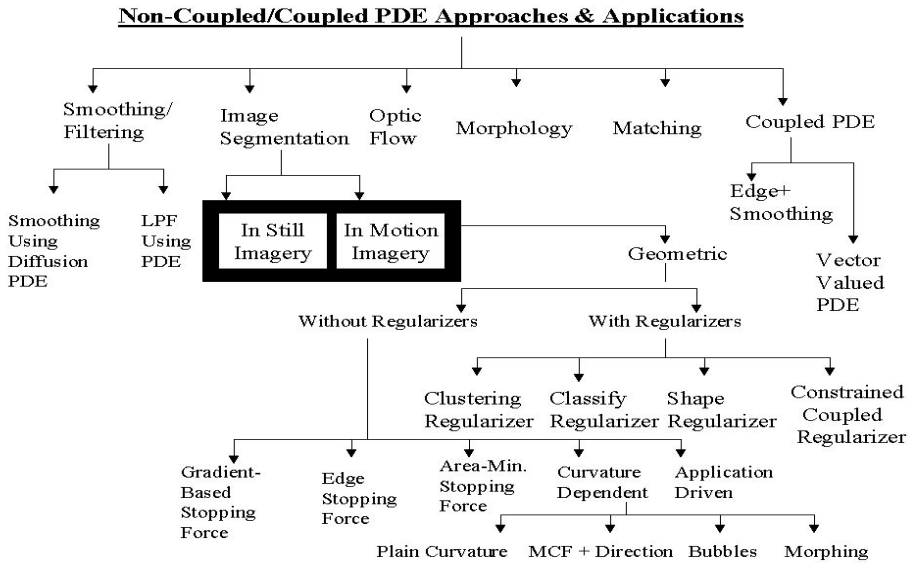[a]Whenever PDE is seen in this paper, it means that it is a PDE-based method.

Fig. 1. Classification of PDE treee of non-coupled and coupled PDE applications applied to Computer Vision, Image Processing and Computer Graphics.

and geometric. Similarly, the fusion of region and boundary-based techniques were also further classified into parametric and geometric). The deformation class of segmentation is so naturally handled in the level set framework that PDE and the level set framework go side-by-side in achieving the objectives of segmentation. PDE-based techniques have recently replaced finite element models (FEM) and finite difference methods (FDM), only because FEM and FDM are expensive in time and tedious to use in the design phases. PDE-based techniques are also less sensitive to complex structures and are extremely accurate. Having discussed the advantages of PDE in segmentation modeling and the broad classification tree for PDE applications, we will now classify the geometric deformable models and see their relationship to PDEs and level sets along with conclusions and a discussion on where this may lead to in the future.

Referring back to Fig. 1, geometric deformable models (GDMs) are classified broadly into two classes: first, GDM without regularizers and second, GDM with regularizers. The *first core class* of segmentation based on the PDE/level set framework is GDM without regularizers. These are techniques where the propagation force, i.e., the force which navigates the propagation front inwards and outwards, does not utilize the region-based strategy for its computation. These forces are constant and do not change. Sometimes they are also called "level set stoppers." Earlier research called these "leakage prevention" techniques because they tried to prevent any bleeding of boundaries during propagation. These are further classified into five different kinds, depending upon the design of the stopping force: (1) gradient-based stopping force; (2) edge-based stopping force; (3) area-minimization-based

stopping force; (4) curvature-dependent stopping force; and (5) application-driven level sets. The curvature-dependent class has four sub-classes: (1) plain curvature-based; (2) mean curvature flow (MCF) with directionality-based; (3) bubbles; and (4) morphing. Plain curvature based techniques are those which are driven solely by the curvature that is computed using differential geometry. Mean curvature flow with directionality-based techniques are those which use the combination of Euclidean curvature and direction together to achieve the deformation process. Such techniques are good for tiny, occluded and twisted objects like blood vessels. Bubbles are a set of seeds, or fourth order shocks, which grow, shrink, merge, split, disappear and deform under the influence of image information such as edges and gradients to segment objects in images and volumes. Morphing techniques are those which undergo shape deformation from one initial shape to the target shape, driven by the combination of signed distance at coordinate transformation and the gradient of the signed distance transform functions. This transformation captures the similarity between user-defined shape and target shape.

The *second core class* of PDE-based segmentation techniques uses regularizers or level sets that derive the propagation force using statistical means such as region-based strategy. This is further classified into four types, depending upon the design of propagation force. They are: (1) fuzzy clustering-based; (2) classification based on Bayesian statistics; (3) shape-based; and (4) constrained coupled level sets where the propagation force is derived from Bayesian strategies.

This paper will survey and discuss the role of regularizers in PDEs and the level set framework. We will take a few sample representation techniques from Suri *et al.*[6] where regularizers are designed and used in the PDE and the level set framework, but details on GDM with/without regularizers can be seen explictly in the paper by Suri *et al.*[6] Thus, the fundamental differences between this paper and Ref. 6 are: (1) this paper focuses on PDEs and level sets for segmentation modeling, while Ref. 6 focused on different kinds of level set methods. (2) This paper shows applications of PDE in the area of CVGIP for *still* and *motion* imagery, while Ref. 6 showed applications of level sets for static 2D and 3D medical imagery only. (3) This paper covers a wide variety of PDE applications such as image smoothing, coupled PDE, low pass filtering and miscellaneous applications such as mathematical morphology, missing shape recovery from partial information. In contrast, Ref. 6 focused on the "optimization techniques" based on the fast marching method, narrow banding and adaptive level sets. In other words, Ref. 6 was a detailed version on the growth of level sets and was a good tutorial for researchers who wished to design medical image segmentation techniques in 2D and 3D based just on level sets. Having discussed the previous work in the area of PDE and level sets, we will next discuss the goals of this paper.

The goals of this paper are the following: (1) to understand the role of PDE in different image processing applications, particularly in segmentation for *still* and *motion* imagery; (2) to understand the relationship between PDE, level sets and regularizers; (3) to understand how one can derive the image segmentation process by fusing the regional-based PDE information with boundary-based PDE models,

which is the crux of this paper; (4) to discuss the state-of-the-art research published in the area of PDE applications in relation to image processing, computer graphics and numerical algorithms; and (5) to present state-of-the-art ready references for readers interested in further exploring into the field of image segmentation using PDE and summarizing the state-of-the-art work done by major research groups such as: Osher and Sethian (UCLA, University of California, Los Angeles, CA, USA), Faugeras and Deriche (INRIA, Institut National de Recherche en Informatique et Automatique, Sophia-Antipolis, France), Kimmel (Technion, Israel Institute of Technology, Haifa, Israel), Sapiro and Tannenbaum (UM, University of Minnesota, Minneapolis, MN, USA), Malladi (LBL, Lawrence Berkeley Labs., Berkeley, CA, USA), Paragios (Siemens, Siemens Corporate Research, Siemens Medical Systems, Inc., Iselin, NJ, USA), Suri (Marconi, Marconi Medical Systems, Inc., Cleveland, OH, USA), Vemuri (UF, University of Florida, Gainesville, FL, USA) and Zhang (UW, University of Wisconsin, Milwaukee, WI, USA). Also note that this paper does not discuss: (1) PDE-based approaches to vector-valued, i.e., color images or hyper-stack images or multi-band images; (2) coupled PDE approaches to image processing. These topics are beyond the scope of this paper.

The remaining sections of this paper are as follows. Section 2 presents the fundamentals on level sets, curve evolution and the Eikonal Equation. Image smoothing and the anisotropic diffusion method based on PDE are covered in Sec. 3. Segmentation in *still* imagery via PDE in the level set framework is covered in Sec. 4. Segmentation in *motion* imagery using PDE and the level set framework is discussed in Sec. 5. Miscellaneous applications of PDEs in mathematical morphology, surface smoothing and missing shape recovery are covered in Sec. 6. Finally, this paper concludes in Sec. 7 by discussing the advantages and the disadvantages of segmentation modeling via geometric deformable models (GDM), PDE and level sets along with conclusions and future work.

2. Level Set Concepts: Curve Evolution and Eikonal Equation

The level set framework has provided one of the beds which implements the PDE. The concept of level sets was introduced by Osher and Sethian,[48] which bubbled out from the Ph.D. Thesis of Sethian.[49] The diversity of applications of level sets has reached into several fields of engineering. Although this paper will not go in depth on level sets, it will cover the fundamental equation of the level sets. Before discussing the level set equation, we present a list of authors who have covered the level sets in the following fields: (1) *geometry*: (see Angenent *et al.*,[50] Chopp[51,52] and Sethian[53]), (2) *grid generation*: (see Sethian[54]), (3) *fluid mechanics* (see Mulder *et al.*,[55] Sethian,[81] Sussman *et al.*[56]), (4) *combustion*: (see Rhee *et al.*[57]), (5) *solidification*: (see Sethian *et al.*[82]), (6) *device fabrication*: (see Adalsteinsson *et al.*[58]), (7) *deformation modeling*: (see Whitaker *et al.*[59−61]), (8) *object tracking/image sequence analysis in images*: (see the recent work by Paragios *et al.*[62,63] and Kornprobst *et al.*[64]), (9) *stereo vision*: (see the recent work by Faugeras and his coworkers at INRIA[65]), (10) *mathematical morphology*: (see Sapiro *et al.*,[66] Arehart *et al.*,[67]

Catte *et al.*[68] and Sochen *et al.*[69]), (11) *color image segmentation*: (see Sapiro[83]) and (12) *2D and 3D medical image processing*: (see the works by Malladi *et al.*,[70−74] Gray-Matter/White-Matter (GM/WM) boundary estimation by Gomes *et al.*,[79] GM/WM boundary estimation with fuzzy models by Suri,[1,84] GM/WM thickness estimation by Zeng *et al.*,[85] leakage prevention in fast level sets using fuzzy models by Suri,[2] also a survey article on brain segmentation by Suri *et al.*[5] and a recent article for cell segmentation by Sarti *et al.*[86]). For a detailed review of some of the above mentioned applications, the reader must see Sethian.[87,88] Although both of these publications cover a good collection of the level set applications, with the advancement of image processing technology, these publications are behind the latest trends. Recently, Suri *et al.*[6] wrote another extensive survey on level sets and their applications. Having stated the applications of level sets, next the fundamental equation of curve evolution will be covered.

2.1. *The fundamental equation of curve evolution*

Since this paper uses level sets as the framework, this sub-section first presents the derivation of the fundamental equation of level sets, known as "curve evolution." Let $\Gamma(t)$ be the closed interface or front propagating along its normal direction (see Fig. 2). This closed interface $\Gamma(t)$: $[0, \infty) \to \mathcal{R}^N$, is either to be a curve in 2D space or a surface in 3D space. The main idea is to represent the front $\Gamma(t)$ as the *zero level set* of a higher dimensional function ϕ.
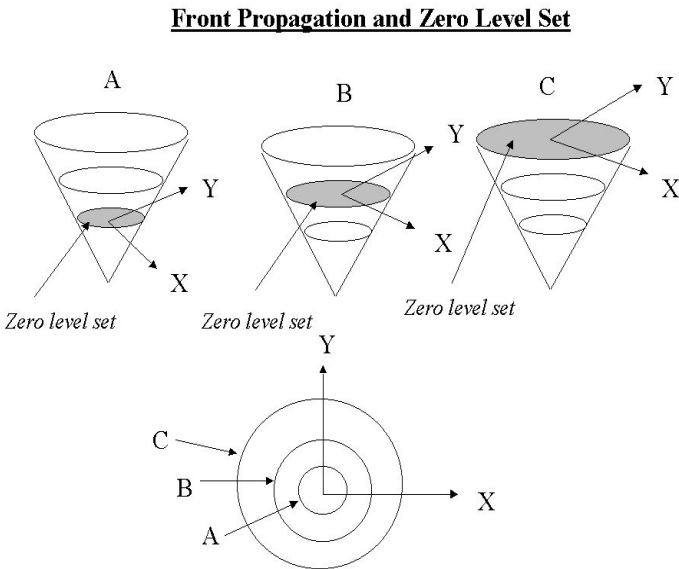
Front Propagation and Zero Level Set

Fig. 2.   Front propoagation of the *zero level set*. Filled circles in figures (A), (B) and (C) show the position of the *zero level curve* as the front propagates. These three projected circled are seen in the bottom drawing.

Let $\phi(x, t = 0)$, where $x \in \mathcal{R}^N$ is defined by $\phi(x, t = 0) = d$, where d is the signed distance from position x to $\Gamma(0)$ and the plus (minus) sign is chosen if the point x is outside (inside) the initial front $\Gamma(0)$. Thus, an initial function is: $\phi(x, t = 0) = \mathcal{R}^N \to \mathcal{R}$ with the property: $\Gamma(t = 0) = (x | \phi(x, t = 0) = 0)$. The goal now is to produce an equation for the evolving function $\phi(x, t)$ so that ϕ always remains zero on the propagating interface. Let $x(t)$, $t \in [0, \infty)$ be the path of a point on the propagation front (see Fig. 2), i.e. $x(t = 0)$ is a point on the initial front $\Gamma(t = 0)$ and $x_t = V(x(t))$ with the vector x_t normal to the front at $x(t)$. Since the evolving function ϕ is always zero on the propagating front, thus $\phi(x(t), t) = 0$. By the chain rule:

$$\phi_t + \sum_{i=1}^N \phi_{x_i} x_{i_t} = 0 \,, \tag{1}$$

where x_i is the ith component of x. Since

$$\sum_{i=1}^N \phi_{x_i} x_{i_t} = (\phi_{x_1}, \phi_{x_2}, \phi_{x_3}, \ldots, \phi_{x_N}) \cdot (x_{1_t}, x_{2_t}, x_{3_t}, \ldots, x_{N_t})$$

$$= V(x(t)) |\nabla \phi| \,, \tag{2}$$

thus, using Eqs. (1) and (2), the final curve evolution equation is given as:

$$\frac{\partial \phi}{\partial t} = V(\kappa) |\nabla \phi| \,, \tag{3}$$

where ϕ is the level set function and $V(\kappa)$ is the speed with which the front (or *zero level curve*) propagates. This fundamental equation describes the time evolution of the level set function (ϕ) in such a way that the *zero level curve* of this evolving function is always identified with the propagating interface. The term "level set function" will interchangeably be used with the term "flow field" or simply "field" during the course of this paper. The above equation is also called an Eulerian representation of evolution due to the work of Osher and Sethian.[48] Equation (3) for the 2D and 3D cases is generalized as: $\partial \phi / \partial t = V_\kappa(x, y) |\nabla \phi|$ and $\partial \phi / \partial t = V_\kappa(x, y, z) |\nabla \phi|$, respectively, where $V_\kappa(x, y)$ and $V_\kappa(x, y, z)$ are curvature dependent speed functions in 2D and 3D, respectively.

Three Analogies of the Curve Evolution Equation: First, the above mentioned equations can be compared with the Euclidean geometric heat equation (see Grayson[80]), given as: $\partial \mathcal{C} / \partial t = \kappa \mathcal{N}$, where κ is the curvature and $\mathcal{N}$ is the inward unit normal and $\mathcal{C}$ is the curve coordinates. Second, Eq. (3) is also called the curvature motion equation, since the rate of change of the length of the curve is a function of $\partial \mathcal{C} / \partial t$. Third, the above equations can be written in terms of differential geometry using divergence as: $\partial \phi / \partial t = \nabla \cdot (\nabla \phi / |\nabla \phi|) |\nabla \phi|$, where geometrical properties such as normal curvature ($\mathcal{N}$) and mean curvature ($\mathcal{H}$) are given as: $\mathcal{N} = \nabla \phi / |\nabla \phi|$ and $\mathcal{H} = \nabla \cdot (\nabla \phi / |\nabla \phi|)$. Note the above equation remains the fundamental form but recently, Faugeras and his coworkers from INRIA (see Gomes

et al.[79]) modified Eq. (3) into the "preserving distance function" as:

$$\frac{\partial \phi}{\partial t} = V(\mathbf{x})(\mathbf{x} - \phi \nabla \phi) \,, \tag{4}$$

where $\mathbf{x}$ is the vector of x and y coordinates, ϕ is the signed distance function. The main characteristic of this equation is that ϕ and V are orthogonal to each other (see details by Gomes *et al.*[79]).

2.1.1. *The Eikonal Equation and its mathematical solution*

In this sub-section, we present the mathematical solution for solving the level set function with unity speed. Such a method is needed to compute the "signed distance transform" when the raw contour crossed the background grid. Consider a case of a "front" moving with a velocity $V = V(x, y)$, such that V is greater than zero. Using Osher–Sethian's[48] level set equation, consider a monotonically advancing front which represents in the form: $\phi_t = V(x,y)\|\nabla\phi\|$, where ϕ_t is the rate of change of the level set and $\nabla\phi$ is the gradient of ϕ. Let $T(x, y)$ be the time at which the front crosses the grid point (x, y). In this time, the surface $T(x, y)$ satisfies the equation: $\|\nabla T\| \cdot V = 1$. By approximation, the solution to the Eikonal Equation is:

$$[\max(\max(D^{-x}T, 0), -\min(D^{+x}T, 0))]^2$$

$$+ [\max(\max(D^{-y}T, 0), -\min(D^{+y}T, 0))]^2 = \frac{1}{V^2_{xy}} \,, \tag{5}$$

where V^2_{xy} is the square of the speed at location (x, y) and $D^{-x}T$, $D^{+x}T$, $D^{-y}T$ and $D^{+y}T$ are the backward and forward differences in time, given as:

$$D^{+x}T = \frac{T(x+1, y) - T(x, y)}{2} \quad \text{and} \quad D^{-x}T = \frac{T(x, y) - T(x-1, y)}{2} \,,$$

$$D^{+y}T = \frac{T(x, y+1) - T(x, y)}{2} \quad \text{and} \quad D^{-y}T = \frac{T(x, y) - T(x, y-1)}{2} \,. \tag{6}$$

There are efficient schemes for solving the Eikonal Eq. (3). For details, see Sethian,[89] Cao *et al.*[90] and Chen *et al.*[91] Having discussed the fundamentals of level sets, curve evolution and the Eikonal Equation, we will now discuss the PDE and level set application for image denoising.

3. Diffusion Imaging: Image Smoothing and Restoration Via PDE

The presence of noise in images is unavoidable. This could be introduced by the image formation process in MR, CT, X-ray or PET images, or image recording or even an image transmission process. Several methods such as morphological smoothing, linear, non-linear, geometric have been presented in noise removal and smoothing, but this paper focuses on "noise removal" or "noise diffusion" using PDE. This has been used for quite some time but recently, robust techniques for image smoothing have been developed (see Perona *et al.*,[19,20] Gerig *et al.*,[21] Alvarez *et al.*,[22,23]

Kimia *et al.*,[26,27] Sapiro *et al.*,[29] Caselles *et al.*,[34] Weickert,[35] Black *et al.*,[36] Arridge *et al.*,[37] Bajla *et al.*,[38] Olver *et al.*,[39] Scherzer *et al.*,[40] Romeny *et al.*[41,43] and Nielsen *et al.*[42]). This section covers these articles in the following way. The fundamental diffusion equation is given in Sec. 3.1. Section 3.2 presents multi-channel anisotropic diffusion imaging. Tensor non-linear anisotropic diffusion is discussed in Sec. 3.3. Anisotropic diffusion based on PDE and the Tukey/Huber weight function is discussed in Sec. 3.4. Image denoising using the curve evolution approach is presented in Sec. 3.5. Image denoising and histogram modification using PDE is presented in Sec. 3.6. Finally, the section concludes with non-linear image denoising in Sec. 3.7.

3.1. *Perona−Malik Anisotropic image Diffusion via PDE* (*Perona*)

One of the first papers on diffusion was from Perona and Malik,[19] called Perona–Malik Anisotropic Diffusion (PMAD) (also called edge-based diffusion). PMADs idea was based on one of the earlier papers by Witkin.[18] Perona *et al.*[19] gave the fundamental PDE-based diffusion equation for image smoothing as:

$$I_t = \mathrm{div}(\mathcal{D}^t(x,y)\nabla I)\,, \tag{7}$$

where "div" was the divergence (also defined as $I_t = \nabla \cdot (\mathcal{D}^t(x,y)\nabla I)$) operator, I_t was the rate of change of image I, $\mathcal{D}^t(x,y)$ was the diffusion constant at location (x,y) at time t and ∇I was the gradient of the image I. Applying the divergence operator, the PDE diffusion equation was re-writtten as:

$$I_t = \mathcal{D}^t(x,y)\Delta I + \nabla \mathcal{D}^t(x,y) \cdot \nabla I\,, \tag{8}$$

where ΔI was the Laplacian operator, $\nabla \mathcal{D}^t(x,y)$ was the gradient of the diffusion constant at location (x,y) for time t. The diffusion constant was the key factor in the smoothing process. Perona *et al.* gave two expressions for the diffusion constants: $\mathcal{D}^t(x,y) = \exp[-(\|\nabla I^t\|/K)^2]$ and $\mathcal{D}^t(x,y) = 1/[1+(\|\nabla I^t\|/K)^2]$, where $\|\nabla I^t\|$ was the absolute value of the gradient of the image I at time t and K was a constant which was either manipulated manually for some fixed value or computed using a "noise estimator" as described by Canny.[24] Using the finite difference method, Eq. (8) was discretized to:

$$
\begin{aligned}
I^{t+1}(x,y) = I^t(x,y) + \lambda\{&\mathcal{D}^t_{\mathrm{e}}(x,y)[I(x+1,y) - I(x,y)] \\
+ &\mathcal{D}^t_{\mathrm{w}}(x,y)[I(x-1,y) - I(x,y)] + \mathcal{D}^t_{\mathrm{n}}(x,y)[I(x,y-1) - I(x,y)] \\
+ &\mathcal{D}^t_{\mathrm{s}}(x,y)[(I(x,y+1) - I(x,y)]\}\,,
\end{aligned}
\tag{9}
$$

where $\mathcal{D}^t_{\mathrm{n}}(x,y)$, $\mathcal{D}^t_{\mathrm{s}}(x,y)$, $\mathcal{D}^t_{\mathrm{e}}(x,y)$ and $\mathcal{D}^t_{\mathrm{w}}(x,y)$ were the finite difference diffusion constants for time t in four different directions (north, south, east and west) given the central location (x,y). The values of these constants were chosen either exponentially or as a ratio as discussed above. To see the performance of the PMAD,
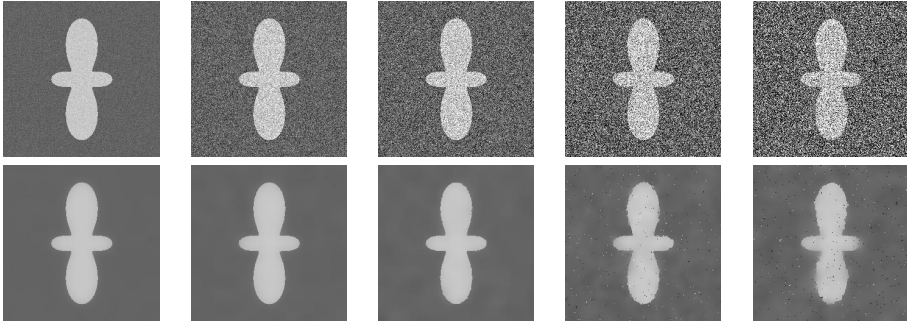
Fig. 3.   Two petal flower with Gaussian noise added and PMAD applied to it. Top row (left to right): Noisy images with 20, 10, 5, 3 and 1 db. Bottom row (left to right): PMAD-PDE diffusion results from the above noisy images. The diffusion constant used was: $D^t(x,y) = 1/[1 + (\|\nabla I^t\|/K)^2]$, where K was chosen as ten, the time step as one and the total number of iterations were 200.
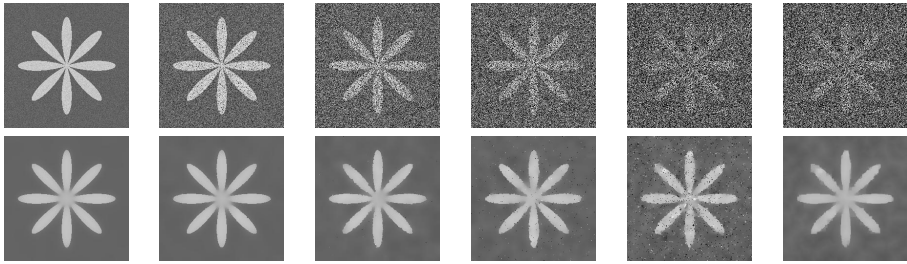

Fig. 4.   This flower image has a larger set of petals compared to Fig. 3 and convolutions. Top row (left to right): Noisy images with 20, 10, 5, 3, 1 and 1 db. Bottom row (left to right): PMAD-PDE diffusion results from the above noisy images. The diffusion constant used was: $\mathcal{D}^t(x,y) = 1/[1 + (\|\nabla I^t\|/K)^2]$, where K was chosen as ten, the time step as one and the total iterations were 200. Note, the last image (right most) in the bottom row is the result due to Pollak *et al.*'s IDM.[25] The number of iterations used was 500.

we took three sets of examples. In case one, we took a simple two petal flower image, then added the Gaussian noise and finally applied the PMAD over it. The results of the input/ouput operation can be seen in Fig. 3. We took a more complex image of a flower image with eight petals, added the same Gaussian noise and applied the PMAD over it. The results can be seen in Fig. 4. In the same figure, we compare the PMAD with Pollak *et al.*'s inverse diffusion method (IDM). Pollak *et al.*[25] proposed a diffusion method which was different from PMAD in two respects: first, discontinuing the inverse flow function and second, merging the regions during diffusion. The first feature made the diffusion stable. Every local maximum was decreased and every local minimum was increased. The second feature made the algorithm fast and unique. In the third example, we applied the PMAD and Pollak *et al.*'s IDM over noisy functional MRI data of the brain. The results can be seen in Fig. 5. Recently, Perona[20] also defined the angular or orientational diffusion
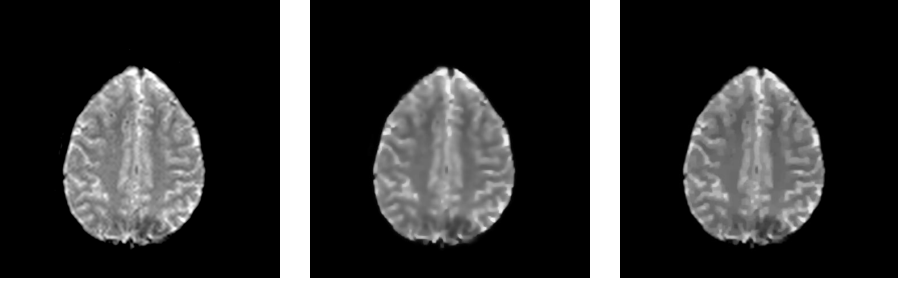
Fig. 5. Noisy fMRI images with PMAD and Pollak *et al.*'s IDM applied to them. Left: fMARI noisy brain axial slice. Middle: PMAD-PDE diffusion results. The diffusion constant used was: $\mathcal{D}^t(x,y) = 1/[1 + (\|\nabla I^t\|/K)^2]$, where K was chosen as ten, the time step as 0.1 and the total iterations were 200. Right: Pollak *et al.*'s IDM with time step 0.1 and number of iterations 500. As seen in this figure, Pollak *et al.*'s method does the best in removing the noise and highlighting the gray matter/white matter interface (for details on the importance of WM/GM interface, see Suri *et al.*[5]

based on the magnetic concept of attracting objects. This research is out of the scope of this paper.

Pros and cons of PMAD using PDE. The main advantage of the PMAD was its relatively low execution time and a good starting point on scale-space and anisotropic diffusion for image denoising. The following were the weaknesses of this method. First, the PMAD method brought blurring at small discontinuities and has a property of sharpening edges (see Gerig *et al.*[21]). Second, the PMAD method did not incorporate convergence criteria (see Gerig *et al.*[21]). Third, the method was not robust at handling large amounts of noise. The method did not preserve the discontinuities between regions. Fourth, the method needed to adjust tuning constants such as K and lastly, the method did not take into account inhomogeneity in data sampling.

3.2. *Multi-Channel Anisotropic image Diffusion via PDE (Gerig)*

Recently, Gerig *et al.*[21] developed the non-linear diffusion system for smoothing or noise reduction in MR brain images. This method was called multi-channel since the processing involved three different kinds of scans: T_1-, T_2- and PD-weighted MR data sets. It was named "coupled" since the diffusion coefficient was coupled between two different MR data sets. Keeping PMAD's diffusion in mind, the multi-channel anisotropic image diffusion was given as:

$$\begin{pmatrix} \dfrac{\partial I_1(x,y,t)}{\partial t} \\[2ex] \dfrac{\partial I_2(x,y,t)}{\partial t} \end{pmatrix} = \begin{pmatrix} \mathrm{div}(D_c(x,y,t)\nabla I_1^t) \\[1ex] \mathrm{div}(D_c(x,y,t)\nabla I_2^t) \end{pmatrix}, \tag{10}$$

where the coupled diffusion coefficient D_c was computed from multi-channel data sets I_1 and I_2. $\partial I_1/\partial t$ and $\partial I_2/\partial t$ were the rate of change of multi-channel images. The coupled diffusion was given as: $\mathcal{D}_c^t(x,y,t) = \exp[-(\|(\nabla I_1^t)^2 + (\nabla I_2^t)^2\|/K)^2]$ and $\mathcal{D}_c^t(x,y,t) = 1/[1+(\|(\nabla I_1^t)^2 + (\nabla I_2^t)^2\|/K)^2]$, where $\|(\nabla I_1^t)^2 + (\nabla I_2^t)^2\|$ was the absolute value of the gradient of the multi-channel images I_1 and I_2 and K was a constant as used by PMAD. Note, if discontinuities were detected in both channels, then the combined diffusion coefficient was larger than any single component and the significance of local estimations was increased. On the other hand, if a discontinuity was detected only in one of the channels, the combined coefficient responded to the discontinuity and halted the diffusion.

Pros and cons of Multi-Channel Anisotropic Diffusion. This technique had two major advantages. First, it showed efficient noise reduction in homogeneous regions and also preserved the object contours, boundaries between different tissues and small structures such as vessels. Second, filtered images appeared clearer and boundaries were better defined, leading to an improved differentiation of adjacent regions of similar intensity characteristics. The following were the major weaknesses of this technique. First, the paper did not show how the PDE flow behaved and how the convergence would get affected if the "coupled PDE diffusion" was computed. Second, the number of iterations in the smoothing process was selected by visual comparison. Thus, the convergence to steady-state and stopping criteria were fuzzy. Third, there was no discussion on the computation time for 3D filtering for non-isotropic volumes (a volume whose three dimensions are not same); and lastly, selection of the parameter K was not automatic.

3.3. *Tensor Non-linear Anisotropic Diffusion via PDE (Weickert)*

To combat the problems of PMAD, Weickert[35] proposed a truly anisotropic diffusion, called Tensor Non-linear Anisotropic Diffusion (TNAD) and was mathematically given as:

$$\frac{\partial I(x,y,t)}{\partial t} = \text{div}[\mathcal{D}(\nabla I)\nabla I]\,, \tag{11}$$

where $\mathcal{D}$ was the diffusion tensor having eigenvectors e_1 and e_2 and defined in a way such that: $\underline{e_1 \| \nabla I}$ and $\underline{e_2 \perp \nabla I}$. Weickert suggested choosing the eigenvalues λ_1 and λ_2 such that: $\lambda_1 = g(\|\nabla I\|)$ and $\lambda_2 = 1$. The sample results of this technique can be seen in Fig. 6 (see Weickert *et al.*[121]). Details on this method can be seen in the book by Weickert.[35] Other authors who worked in non-linear anisotropic diffusion were: Schnörr[14,15] and Catte *et al.*[16,17]

Pros and cons of Tensor Non-linear Anisotropic Diffusion. Besides being robust, the method was applied to a variety of images. The method did not take into consideration, however, the inhomogeneity in data sampling.
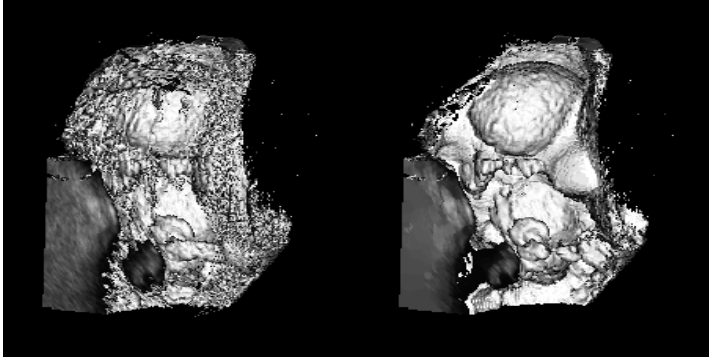
Fig. 6.   Tensor image diffusion on a 3D ultrasound image. Left: Rendering of a 3D ultrasound data set of a ten-week old human fetus. Right: Rendering after denoising by Tensor Non-linear Anisotropic Diffusion (TNAD) filtering (see Weickert *et al.*[121]). Courtesy of Professor Weickert, Computer Vision, Graphics and Pattern Recognition Group, Department of Mathematics and Computer Science, University of Mannheim, Mannheim, Germany.

3.4. *Anisotropic diffusion using the Tukey/Huber weight function (Black)*

Black *et al.*[36] recently showed a comparative study between Perona–Malik Anisotropic Diffusion (PMAD) based on a combination of PDE and Tukey's bi-weight estimator, known as Black *et al.*'s Robust Anisotropic Diffusion (BRAD). In this image smoothing process (estimating piecewise constant), the goal was to minimize I that satisfied:

$$\min_I \sum_{s \in I} \sum_{p \in \eta_s} \rho(I_p - I_s, \sigma) \,, \tag{12}$$

where s was the pixel location, p took one of the four neighbours and η_s was the set of four neighbours of s. ρ was the error norm function (called a robust estimator, see Meer *et al.*[44]) and σ was the scale parameter. The relationship between ρ and PMAD was expressed as: If PMAD was given by the function $g(x)$, where $g(x) = 1/1 + (x/K)^2$, then $g(x) = \rho'/x$, the derivative of the error norm function. Black *et al.* used Tukey's Biweight and the Huber min-max function for error norm. This Tukey function was given as: $\rho(x, \sigma) = (x^2/\sigma^2 - x^4/\sigma^4 + x^6/3\sigma^6)$, if $|x| \leq \sigma$ and $\rho(x, \sigma) = 1/3$, if $|x| > \sigma$. The Huber's min-max for error norm was $\rho(x, \sigma) = (x^2/2\sigma + \sigma/2)$, if $|x| \leq \sigma$ or $|x|$, if $|x| > \sigma$. Note that σ was computed as: $\sigma = 1.4 \times \text{median} \left(\|\nabla I - \text{med}_I(\|\nabla I\|)\| \right)$. Black *et al.* used gradient descent for solving the image smoothing minimization problem. Thus the discrete solution of BRAD was:

$$I_s^{t+1} = I_s^t + \frac{\lambda}{\eta_s} \sum_{p \in \eta_s} \rho'(I_p - I_s^t, \sigma) \,, \tag{13}$$

where ρ' was the derivative of error norm, σ. The whole idea of bringing the robust statistics was to remove outliers and preserve shape boundaries. Thus BRAD = PMAD + Huber/Tukey's Robust estimator. Figure 7 compares Perona–Malik's

Fig. 7.   Comparison of the Perona–Malik function and Tukey function after 100 and 500 iterations. Results of Perona–Malik PDE and Black *et al.*'s Tukey method. Left most: PMAD with number of iterations set to 100. Left middle: Tukey function with number of iterations set to 100. Right middle: PMAD with number of iterations set to 500 iterations. Right most: Tukey function with number of iterations set to 500. (Courtesy of Professor Sapiro, Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN. Source: Black *et al.*[36]

Anisotropic Diffusion (PMAD) with Black *et al.*'s Tukey function (BRAD). This figure shows the results for 100 and 500 iterations. As you can see, the results of Tukey are far superior compared to PMAD. Also at infinity, the PMAD will make the image go flat and the Tukey will not.[b] Readers interested in the application of Huber's weight function in medical application can see the detailed work by Suri, Haralick and Sheehan.[45] Here, the goal was to remove the outlier longitudinal axes of the Left Ventricle (LV) for ruled surface estimation to model the movement of the LV of the heart.

Pros and cons of anisotropic diffusion based on robust statistics. The following were the main advantages of the BRAD method. In comparision to PMAD, BRAD was more robust. The method was also stable for a large number of iterations and the image did not get flat when t was infinity. The major disadvantage of the system was that it did not show how to compute σ for the design of Huber's weight function. Secondly, the system did not discuss the timing issues.

3.5. *Image denoising using PDE and curve evolution (Sarti)*

Recently, Sarti *et al.*[86] presented an image denoising method based on the curve evolution concept. Before we discuss what Sarti *et al.* did, we will present the fundamental equation presented by Kichenassamy *et al.*[75] and Yezzi *et al.*[76] They presented the curve evolution model by introducing an extra stopping term. This was expressed mathematically as:

$$\frac{\partial \phi}{\partial t} = c(\mathbf{x})(\kappa + V_0)|\nabla \phi| + \underbrace{(\nabla c \cdot \nabla \phi)} \,. \tag{14}$$

Note that $(\nabla c \cdot \nabla \phi)$ denoted the projection of an attractive force vector on the normal to the curve. This force was realized as the gradient of a potential field c. This potential field c for the 2D and 3D case was given as: $c(x, y) = -|\nabla(G_\sigma \otimes I(x, y))|$

---

[b]Personal communication with Professor Sapiro.

and $c(x, y, z) = -|\nabla(G_\sigma \otimes I(x, y, z))|$, respectively, where ∇ was the gradient operator, $G_\sigma \otimes I(x, y)$ and $G_\sigma \otimes I(x, y, z)$ were the Gaussian convolution operations, respectively. Note that Eq. (14) is similar to Eq. (7) given by Malladi *et al.* in Ref. 78. Malladi *et al.* called the equation as an additional constraint on the surface motion ϕ_t. Rewriting Eq. (7) of Malladi *et al.*[78] as:

$$\phi_t + c(\mathbf{x})(\epsilon\kappa + V_0)|\nabla\phi| - \beta(\nabla c \cdot \nabla\phi) = 0 \,, \tag{15}$$

where β was the edge strength constant, V_0 was a constant ($= 1$ as used by Malladi *et al.*), κ was the curvature dependent speed, ϵ was the constant term controlling the curvature dependent speed and $(\nabla c \cdot \nabla\phi)$ was the same as defined above. Having presented the level set equation in terms of speed functions and constants, Sarti *et al.*'s method changed the above equation by removing the constant propagation force V_0 and simply solved the remaining equation for image denoising. The smoothing worked in the following way. If $|\nabla(G * I_0(\mathbf{x}))|$ was large, the flow was slow and the exact location of the edge was retained. If $|\nabla(G * I_0(\mathbf{x}))|$ was small, then the flow tended to be fast, thereby increasing the smoothing process. The filtering model was reduced to mean curvature flow when $c(\mathbf{x})$ was an equation to unity. Thus, the data consistency term $c(\mathbf{x})$ served as an edge indicator. The convolution operation simply eliminated the influence of spurious noise. Sarti *et al.* thus solved Eq. (14) iteratively and progressively smoothed it over time. The edge indicator function c became smoother and smoother over time and depended less and less on the spurious noise. Note that the minimal size of the detail was related to the size of the Gaussian kernel, which acted like a scale parameter. The variance of $G_\sigma(\mathbf{x})$ was taken as $1/[C_\sigma \exp -(|\mathbf{x}|^2/4\sigma)]$ which corresponded to the dimension of the smallest structure that had to be preserved. The sharpening of the edge information was due to the hyperbolic term $(\nabla c \cdot \nabla\phi)$.

Pros and cons of image denoising using PDE and curve evolution. The main merits of the system were its simple extension from the basic curve evolution method and the ability to sharpen the image edges due to its hyperbolic term $(\nabla c \cdot \nabla\phi)$. The drawbacks of the system were the following. First, there was not much discussion on the scale-space parameter, the variance of $G_\sigma(\mathbf{x})$, which was one of the critical pieces in the image denoising. Second, there was not much discussion on the stopping method for the PDE. Since constant force played a critical role as will be seen ahead as a "regularizer force" in segmentation modeling, we feel that constant force could have been used more efficiently rather than plain removal in the image denoising process.

3.6. *Image denoising and histogram modification using PDE (Sapiro)*

A description on diffusion would be incomplete if references from Sapiro were missed. Recently, Sapiro *et al.*[28−30] developed the ensemble of two different algorithms in one PDE. This method used histogram modification (or histogram

equalization) and image denoising in one PDE. This was called edge preservation anisotropic diffusion. The idea was to smooth the image only in the direction parallel to the edges, achieving this via curvature flows. This was done using the following flow equation:

$$\frac{\partial \phi}{\partial t} = \frac{1}{1 + \|\nabla(G \otimes \phi)\|} \left(\phi_x^2 \phi_{yy} - 2\phi_x \phi_y \phi_{xy} + \phi_y^2 \phi_{xx}\right)^{1/3} \|\nabla \phi\|, \qquad (16)$$

where ϕ was the level set function which evolved according to the affine heat flow for planar shape smoothing with the velocity given by $1/1 + \|\nabla(G \otimes \phi)\|$, where $\otimes$ was the convolution operator. The histogram equalization flow was given as:

$$\frac{\partial \phi}{\partial t} = (N^2 - H[\phi(x, y, t)]) - \mathcal{A}[(v, w) : \phi(v, w, t) \geq \phi(x, y, t)], \qquad (17)$$

where image $\phi(x, y, t) : [0, N]^2 \to [0, M]$ and $\mathcal{A}$ represented the area (or number of pixels). Combining Eqs. (16) and (17) yielded a joint PDE as:

$$\frac{\partial \phi}{\partial t} = \alpha \frac{1}{1 + \|\nabla(G \otimes \phi)\|} \left(\phi_x^2 \phi_{yy} - 2\phi_x \phi_y \phi_{xy} + \phi_y^2 \phi_{xx}\right)^{1/3} \|\nabla \phi\|$$

$$+ (N^2 - H[\phi(x, y, t)]) - \mathcal{A}[(v, w) : \phi(v, w, t) \geq \phi(x, y, t)]. \qquad (18)$$

Here, H was defined as: If the density function of image $I(x, y)$ is $h(n)$, where $n = 0, \ldots, M$, then $H(n) = \sum_{i=0}^{n}(h(n))$, like a "cumulated" density function. Here, α was the weighting factor. There has been some research which relates anisotropic diffusion, curve evolution and segmentation. Interested researchers can see Sapiro[32] and Shah.[33]

Pros and cons of image denoising and histogram modification via PDE.
The paper took advantage of how different components were diffused together. The demerit of the system was that it was only able to smooth the edges in the direction of the edges. Besides that, there was no discussion about how to select the parameter α.

3.7. *Image denoising using non-linear PDEs (Rudin)*

Rudin *et al.*[132] recently developed a denoising algorithm which could be interpreted as a first step of moving each level set of the image normal to itself with velocity equal to the curvature of the level set divided by the magnitude of the gradient of the image. The second step in this algorithm was to project the image back onto the constrain set. This problem was posed as a constrained minimization problem where the goal was to minimize over I the following: minimize $\int_\Omega \sqrt{(\partial u/\partial x)^2 + (\partial u/\partial y)^2} \, dxdy$. This was subjected to the linear and non-linear constraints given as: $\int_\Omega u \, dxdy = \int_\Omega u_0 \, dxdy$ and $\int_\Omega (u - u_0)^2/2 \, dxdy = \sigma^2$. This equation was changed into a Euler–Lagrange formulation and then into a PDE framework as:

$$u_t = \frac{\partial}{\partial x}\left(\frac{u_x}{\sqrt{u_x + u_y}}\right) + \frac{\partial}{\partial y}\left(\frac{u_y}{\sqrt{u_x + u_y}}\right) - \lambda(u - u_0), \qquad (19)$$

where u_t was the rate of change of u with respect to t, u_x and u_y were the partial derivatives $\partial u/\partial x$ and $\partial u/\partial y$, respectively. This was valid for t greater than zero for every x, $y \in \Omega$. The value of λ was computed as:

$$\lambda = \frac{-1}{2\sigma} \int \left[\sqrt{u_x^2 + u_y^2} - \left(\frac{(u_0)_x u_x}{\sqrt{u_x + u_y}} + \frac{(u_0)_y u_y}{\sqrt{u_x + u_y}} \right) dxdy \right]. \tag{20}$$

Here, we briefly present the numerical solution to Eqs. (19) and (20).

$$u_{ij}^{n+1} = u_{ij}^n + \frac{\Delta t}{h} \left\{ \mathcal{D}_x^- \left[\frac{\mathcal{D}_x^+ u_{ij}^n}{((\mathcal{D}_x^+ u_{ij}^n)^2 + (m(\mathcal{D}_y^+, \mathcal{D}_y^-))^2)^{1/2}} \right] \right.$$

$$\left. + \mathcal{D}_y^- \left[\frac{\mathcal{D}_y^+ u_{ij}^n}{((\mathcal{D}_y^+ u_{ij}^n)^2 + (m(\mathcal{D}_x^+, \mathcal{D}_x^-))^2)^{1/2}} \right] \right\} + \Delta t \lambda^n (u_{ij}^n - u_0(ih, jh)),$$

$$\tag{21}$$

where $x_i = ih$, $y_j = jh$, $i, j = 1, \ldots, N$ with $Nh = 1$, and h is the step size. $t_n = n\Delta t$, $n = 0, 1, \ldots$, $u_{ij}^n = u(x_i, y_j, t_n)$ and $u_{ij}^0 = u_0(ih, jh) + \sigma\phi(ih, jh)$. $\mathcal{D}_x^\pm = \pm(u_{i\pm 1,j} - u_{ij})$; $\mathcal{D}_y^\pm = \pm(u_{i\pm 1,j} - u_{ij})$; $m(a, b) = \min \mod(a, b) = [(\text{sgn}(a) + \text{sgn}(b))/2] \min(|a|, |b|)$ and

$$\lambda^n = -\frac{h}{2\sigma^2} \left\{ \sigma_{i,j} \left[\sqrt{(\mathcal{D}_x^+ u_{ij}^n)^2 + (\mathcal{D}_y^+ u_{ij}^n)^2} - \frac{(\mathcal{D}_x^+ u_{ij}^0)(\mathcal{D}_x^+ u_{ij}^n)}{\sqrt{(\mathcal{D}_x^+ u_{ij}^n)^2 + (\mathcal{D}_y^+ u_{ij}^n)^2}} \right. \right.$$

$$\left. \left. - \frac{(\mathcal{D}_x^+ u_{ij}^0)(\mathcal{D}_x^+ k u_{ij}^n)}{\sqrt{(\mathcal{D}_x^+ u_{ij}^n)^2 + (\mathcal{D}_y^+ u_{ij}^n)^2}} \right] \right\}, \tag{22}$$

where $\mathcal{D}_x^\pm$ and $\mathcal{D}_y^\pm$ were the same as defined above. Note that the step size restriction was imposed for stability, i.e. $\Delta t/h^2 \leq c$, where c was a constant.

Pros and cons of non-linear PDEs for image denoising. The following were the major advantages of the non-linear image denoising method. First, the method introduced the constraint while solving the PDEs. As a result, the regularizer term λ improved the smoothing effect. Second, such a method was applied in techniques for sub-pixel accuracy. Third, this algorithm converged to the steady-state at $t \to \infty$, which was the denoised image. The following were the weaknesses of this method. First, the paper did not discuss the stability issues such as the ratio $\Delta t/h^2$ and secondly, the method introduced the λ term which needed extra computations.

Having discussed the different image diffusion techniques based on PDE, and their pros and cons, interested readers can go into more detail on behavioral analysis on anisotropic diffusion in image processing (see You *et al.*[31]) Also, see knowledge-based tensor anisotropic diffusion for cardiac MRI by Sanchez-Ortiz *et al.*[46] A detailed review on PDE-based diffusion and a comparison between different smoothing techniques using PDE, scale-space mathematical morphology and inverse diffusion

techniques is given by Suri *et al.*[47] Next, we will discuss image segmentation in still images via PDE and the level set framework.

4. Segmentation in Still Imagery Via PDE/Level Set Framework

Dominance of PDE in the level set framework for image/volume segmentation has been tremendous. The following institutions are the major players: (1) UCLA, University of California, Los Angeles, CA, USA, (2) INRIA, Institut National de Recherche en Informatique et Automatique, Sophia-Antipolis, France, (3) IRISA, Institut de recherche en informatique et systemes aleatories, Rennes Cedex, France, (4) Technion, Israel Institute of Technology, Haifa, Israel, (5) MIT, Massachusetts Institute of Technology, Cambridge, MA, USA, (6) MGH, Massachusetts General Hospital, Harvard Medical School, Boston, MA, USA, (7) JHU, Johns Hopkins University, Baltimore, MD, USA, (8) UT, University of Toronto, Toronto, Canada, (9) LBNL, Lawrence Berkeley National Labs., Berkeley, CA, USA, (10) MNI, Montreal Neurological Institute, McGill University, Montreal, Canada, (11) Yale, Image Processing and Analysis Group, Yale University, New Haven, CT, USA, (12) MMS, Marconi Medical Systems, Inc., Cleveland, OH, USA, (13) UF, University of Florida, Gainesville, FL, USA, (14) UM, University of Minnesota, Minneapolis, MN, USA and (15) UW, University of Wisconsin, Milwaukee, WI, USA. Since it is difficult to cover all level sets based techniques for segmentation, an attempt was made by Suri *et al.*[6] to design a level set survey paper which covered the state-of-the-art segmentation methods for medical imaging. This section will focus on the core PDE application for segmentation of still imagery. This section presents five systems which follow the generic segmentation modeling method. The layout of this section is as follows. The PDE-based example of fusing the fuzzy pixel classification in the level set framework is presented in Sec. 4.1. Application of Bayes' method in the PDE/level set framework for segmentation is discussed in Sec. 4.2. Vasculature segmentation in the PDE/level set framework is discussed in the Sec. 4.3. Section 4.4 presents the PDE/level set application using inverse variational criterion. Finally, we conclude by presenting the Bayesian-based regularizer for segmentation in Sec. 4.5.

4.1. *Embedding of the fuzzy model as a bi-directional regional regularizer for PDE design in the level set framework (Suri/Philips Medical Systems, Inc.)*

This section shows how the fuzzy regularizers are embedded in the curve evolution or level sets and model the segmentation process in the level set framework using the PDE. The level set equation will then be derived by embedding the region statistics into the parametric classical energy model. Part of this section has been taken from Chapter 8 of the book by Suri *et al.*[4] Part of that derivation will be discussed here with a few of the implementation details. To start with, the standard

dynamic classical energy model as given by Kass *et al.*[92] was:

$$\gamma \frac{\partial \mathbf{X}}{\partial t} = \frac{\partial}{\partial s}\left(\alpha \frac{\partial \mathbf{X}}{\partial s}\right) - \frac{\partial^2}{\partial s^2}\left(\beta \frac{\partial^2 \mathbf{X}}{\partial s^2}\right) + F_{\text{ext}}(\mathbf{X})\,, \tag{23}$$

where $\mathbf{X}$ was the parametric contour, and α, β were the elastic constants. $F_{\text{ext}}(\mathbf{X})$ was the external force and γ was the damping coefficient. Since the second derivative term did not affect significantly the performance of the active geometric snakes (see Caselles *et al.*[34]), that term was dropped and replaced with a new pressure force term, given as: $F_p(\mathbf{X}) = w_p(\mathbf{X})\mathcal{N}(\mathbf{X})$. This yielded the new parametric active contour, given as:

$$\gamma \frac{\partial \mathbf{X}}{\partial t} = \frac{\partial}{\partial s}\left(\alpha \frac{\partial \mathbf{X}}{\partial s}\right) + \underbrace{w_p(\mathbf{X})\mathcal{N}(\mathbf{X})} + F_{\text{ext}}(\mathbf{X})\,. \tag{24}$$

Note that to make Eq. (24) invariant to changes in the parameterization of $\mathbf{X}$, Suri *et al.* considered s as a parameter of arc-length and w_p was the spatial-varying function. Readjusting the terms by defining $\epsilon = \alpha/\gamma$, $\partial/\partial s(\partial \mathbf{X}/\partial s)$ to be the curvature term κ, $V_p = w_p(\mathbf{X})/\gamma \mathcal{N}(\mathbf{X})$, and $V_{\text{ext}} = F_{\text{ext}}/\gamma$, Eq. (24) was rewritten as:

$$\frac{\partial \mathbf{X}}{\partial t} = (\epsilon \kappa + V_p + V_{\text{ext}} \cdot \mathcal{N})\mathcal{N}\,. \tag{25}$$

From the definition of curve evolution first given by Sethian, this resulted in:

$$\frac{\partial \phi}{\partial t} = V(\kappa)\mathcal{N}\,. \tag{26}$$

Comparing Eqs. (25) and (26), and using the definition of normal as: $\mathcal{N} = -\nabla \phi/|\nabla \phi|$ and considering only the normal components of the internal and external forces: $\partial/\partial s(\alpha \partial \mathbf{X}/\partial s) \cdot \mathcal{N} = (\alpha \kappa)$, the level set function (ϕ) was obtained as:

$$\frac{\partial \phi}{\partial t} = (\epsilon \kappa + V_p)|\nabla \phi| - V_{\text{ext}} \cdot \nabla \phi\,, \tag{27}$$

where V_p was a regional term. An example of such a term was defined as $w_R/\gamma R$, where w_R was the weighting factor and R was the region indicator term which was a function of the fuzzy membership and ranged between zero and one. The results of running the above algorithm can be seen in Fig. 8. The raw contour deforms towards the topology of the WM/CSF interface (for details, see Suri[1]). Other authors who performed the embedding of regional statistics into the boundary estimation framework were Pavlidis *et al.*,[93] Zhu *et al.*[94] and Suri.[95]

Pros and cons of geometric boundary fused with clustering. The above system is one of the latest state-of-the-art systems which had the following advantages. First, in the above system, the fuzzy clustering acted as a regularizer for navigating the geometric geodesic snake. Since it was a pixel-classification procedure, one could easily estimate boundaries for different structures in the brain MRI data such as WM, GM, ventricles, etc. Second, the key characteristic of this system was that it was based on region, so the local noise or edge did not distract the
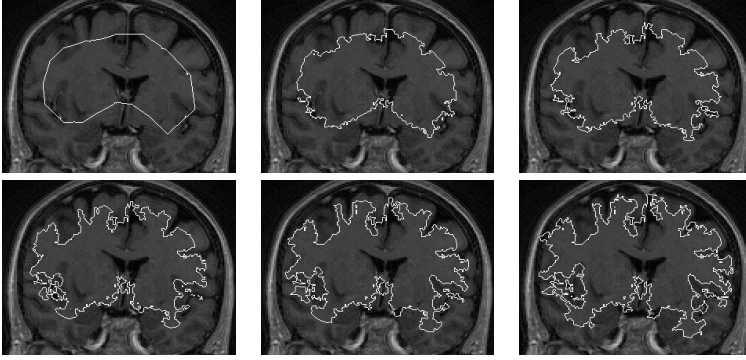
Fig. 8. The above images (left to right, top row to bottom row) shown the growth of the *zero level curve* to segment the white matter in human brain MR images. The narrow band width was 25 pixels on either side of the *zero level curve*, with the land mines at five pixels wide. Note, the deformation process had the ability to move in and out during the growth process. (Courtesy of Dr. Jasjit Suri, Marconi Medical Systems, Inc.)

growth process. Third, the technique was non-local and thus the local noise could not distract the final placement of the contour or the diffusion growth process. Fourth, the technique was very suitable for medical organ segmentation since it could handle any of the cavities, concavities, convolutedness, splitting or merging. Fifth, the technique was extendable to multi-phase, which meant that if there were multiple level set functions, then they automatically merged and split during the course of the segmentation process. The major weakness of such a system was in the computational expense of the fuzzy membership computation, however it is compensated by the fast level sets implementations (see Suri[1]). Besides that, although the system has a minimum number of coefficients compared to parametric snakes, it was not totally independent of image characteristics.

4.2. *Embedding of the Bayesian model as a regional regularizer for PDE design in the level set framework (Paragios/INRIA)*

Paragios *et al.*[99] very recently added a Bayesian model as a regional term to redesign the PDE to improve Zhu *et al.*'s method.[94] Paragios *et al.*[99] implemented the fusion of region-based and boundary-based PDE in the level set framework. This meant incorporating the region statistics for PDE along with the boundary information of the PDE and solving the fused segmentation model in the level set framework. This will next be explained here. The classical deformable model for boundary estimation was originally given by Kass *et al.*[92] as:

$$E(C) = \underbrace{\alpha \int |C'(s)|^2 \, ds + \beta \int |C''(s)|^2 \, ds}_{\text{Internal-Energy-Term}} + \lambda \underbrace{\int |\nabla I(C(s))| \, ds}_{\text{External-Energy-Term}} , \qquad (28)$$

where $E(C)$ was the classical energy of the snake or parametric contour. α, β were the elastic constants for smoothing and λ was the external energy constant. Note,

the sum $\alpha \int |C'(s)|^2 \, ds + \beta \int |C''(s)|^2 ds$ was the smoothing energy term which was a function of the first and second partial derivative of the parametric curve along s. The term $\lambda \int |\nabla I(C(s))| ds$ was the external energy term as a function of image gradient ∇I. Following the derivation as proposed by Caselles *et al.*,[34] the above was changed to the curve evolution form as: $\partial C / \partial t(t) = g(I)\kappa \mathcal{N} - (\nabla g \cdot \mathcal{N})\mathcal{N}$, where $\partial C / \partial t(t)$ was the rate of change of curve evolution, $g(I)$ was the edge detector, κ was the curvature and $\mathcal{N}$ was the surface outward unit normal. Using the reasoning as discussed by Caselles *et al.*,[34] the above equation was given in terms of energy as:

$$E(u) = \int g(X)|\nabla u| \, dX \, . \tag{29}$$

Note, this was still in the boundary mode and the region term was still not integrated. In terms of the derivative operator, this can be given as:

$$E(\partial R) = \int g(\partial R_X(x))|\partial R_X(x)| \, dx \, . \tag{30}$$

If the image was composed of, say, two classes, $\mathcal{A}$ and $\mathcal{B}$, then using Zhu *et al.*'s[94] method of introducing the regional force with the boundary:

$$E(\partial R) = - \underbrace{\sum_{x \in \mathcal{A},\mathcal{B}} \alpha \iint \log(p_r, x(I(x,y)) \, dxdy}_{\text{Region-Term}}$$

$$+ \underbrace{\sum_{x \in \mathcal{A},\mathcal{B}} (1 - \alpha) \iint g(p_b, x(I(\partial R(c_x))))|\partial R_x(c_x)| \, dc_x}_{\text{Boundary-Term}} , \tag{31}$$

where α was the smoothing constant, p_r and p_b were the regional and boundary probabilities. The Eulerian representation of the boundary model as given by Caselles *et al.*[34] was:

$$\frac{\partial \phi}{\partial t} = \left(g(I)\kappa + \nabla g(I) \cdot \frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi| \, , \tag{32}$$

where $\nabla \phi$ was the gradient of the level set function. Thus, the level set framework for the region term fused with boundary was given as:

$$\frac{\partial \phi_i}{\partial t}(u) = \underbrace{\alpha[r_i(u) - r_{o_{i(u)}}]|\nabla \phi_i(u)|}_{\text{Regional-Force}}$$

$$+ \underbrace{(1 - \alpha) \left[b_i(u)\kappa_i(u) + \nabla b_i(u) \cdot \frac{\nabla \phi_i(u)}{|\nabla \phi_i(u)|} \right] |\nabla \phi_i(u)|}_{\text{Boundary-Force}} , \tag{33}$$

where r_i/r_o was a function that "captured" the inside/outside region features of the different classes and was given as: $\mathcal{R} \times \mathcal{R} \rightarrow \mathcal{R}$, $i \in [1, N]$, where N were the

total number of regions. Similarly, b_i was a function that "captured" the boundary features of the different classes and had the range: $\mathcal{R} \times \mathcal{R} \to \mathcal{R}$, $i \in [1, N]$. $\kappa_i(u)$ was the curvature information of the boundary. $\nabla b_i(u)$ is the gradient at the boundary and $\nabla \phi_i(u)/|\nabla \phi_i(u)|$ was the outward/inward normal of the propagating contour. Note that the above contour had two parts: regional force and the boundary force, and i varied from one to N. Paragios *et al.* modified Eq. (33) by introducing the penalty function for the pixels that either did not belong to a region or which belonged to multiple regions. The interested reader can explore Paragios *et al.*[99] on this.

Pros and cons of Paragios *et al.*'s technique. Since the system took the region-based statistics into the geometric deformable model, it had an advantage of being robust to noise. Second, the algorithm had the flexibility of putting multiple initial contours (multi-phase curve propagation) just like the bubble technique by Tek *et al.*[100] Third, the algorithm was validated by taking three applications: texture segmentation, motion segmentation and tracking. The following were the major weaknesses of Paragios *et al.*'s technique. First, the methodology was able to segment texture well, but the results did not show that the segmented boundary was able to penetrate the convoluted shapes. For example in Fig. 9 (left), we see that the final boundary was not able to go deep between the legs of the zebra. This problem was addressed by Xu *et al.*[101] and solved using gradient vector flow. Another solution would be to use a fuzzy regularizer rather than a Bayesian regularizer in this framework. This has been shown by Suri recently.[1] Second, like all region-based methods, this method was also computationally expensive.

4.3. *Vasculature segmentation using PDE (Lorigo/MIT)*

Recently, Lorigo *et al.*[96] presented an algorithm for brain vessel reconstruction based on curve evolution in 3D, also known as "co-dimension two" in geodesic active

Fig. 9.   Segmentation results of Paragios *et al.*'s method. Left: zebra image showing the segmented contour around the zebra. Right: CT image of the chest showning the segmented results. (Courtesy of Nikos Paragios, Siemens Corporate Research, Princeton, NJ.)

Fig. 10. Segmentation results using Lorigo *et al.*'s technique.[96,97] Left: Axial results. Middle: Coronal results. Right: Sagittal results. (Courtesy of Professor Grimson, MIT, Cambridge, MA.)

contours. This method used two components: (i) mean curvature flow (MCF) and (ii) the directionality of vessels. The mean curvature flow component was used to derive the Eulerian representation of the level set equation. If ϕ was the signed distance transform (SDT) and $\lambda(\nabla\phi(\mathbf{x},t), \nabla^2\phi(\mathbf{x},t))$ were the eigenvalues of the projection operator: $\mathcal{P}_{\nabla\phi}\nabla^2\phi P_{\nabla\phi}$, where $\mathcal{P} = I - qq^T/|q|^2$ and q was a non-zero vector and T represented the transpose. Using these eigenvalues, the Eulerian representation of the curve evolution was given by Lorigo *et al.* as: $\partial\phi/\partial t = \lambda(\nabla\phi(\mathbf{x},t), \nabla^2\phi(\mathbf{x},t))$. The second component was the normal of these vessels projected onto the plane and was given as the product of $\nabla\phi$ with the projection vector d. This projection vector was computed using the Hessian H of the intensity image, I and was given as: $g'/g(H\nabla I/|\nabla I|)$, where g was the edge detector operator. Adding these two components, the complete level set equation was:

$$\frac{\partial\phi}{\partial t} = \underbrace{\lambda(\nabla\phi(\mathbf{x},t), \nabla^2\phi(\mathbf{x},t))}_{\text{mean-curvature-force}} + \underbrace{D \times S \times \frac{g'}{g}\left(H\frac{\nabla I}{|\nabla I|}\right)}_{\text{angular-balloon-force}} \tag{34}$$

where D was the directionality term which was the dot product of $\nabla\phi$ and ∇I which was the angle between these two vectors. S was the scale term. Note that the second term was like an angular balloon force which navigated the deformation process. The results of running the above algorithm can be seen in Fig. 10 which shows the results in axial, coronal and sagittal directions, respectively.

Pros and cons of Lorigo *et al.*'s technique. The major advantages of this technique were the following. First, the method successfully demonstrated the segmentation of vessels of the brain. Second, the method used the directional component in the level set framework, which was necessary for segmenting twisted, convoluted and occluded vessels. Third, the technique was used to compute vessel radii, a clinically useful measurement. The following were the weaknesses of Lorigo *et al.*'s work. First, not much discussion was available on the computation of the scale factor S. Second, the method did not show the analytical model, since the output of the system showed relatively thinner vessels compared to maximum intensity

projection (MIP). The MIP is a popular algorithm and an example can be seen by Suri *et al.*[98] Third, there was no comparison made between segmented results and the ground truth, hence, the system needed to be validated.

4.4. *Segmentation using inverse variational criterion (Barlaud/CNRS)*

Curve evolution as an inverse problem was first attempted by Santosa[103] and then by Barlaud and co-workers from CNRS (Center National Dela Recherche Scientifique, Cedex, France, see Amadieu *et al.*[102]). Here, the segmentation method was developed using variational criterion of an inverse problem. The equation given by Barlaud and his co-workers was:

$$\frac{\partial \phi}{\partial t} + \alpha(x,t)\|\nabla\phi\| = 0, \tag{35}$$

where $\partial\phi/\partial t$ was the rate of change of the level set function ϕ, $\alpha(x,t)$ was the rate at which the curve evolved, and $\|\nabla\phi\|$ was the gradient of ϕ. They applied this equation on images which consisted of binary objects represented as I_{in} and the image background as I_{out}, thus I was a *piecewise* constant function. If the image model was given as $g(x,y) = A(I(x,y)) + \eta$, where A was the operator over image I, η was the Gaussian perturbation, then the speed term $\alpha(x,t)$ was modeled as a function of discontinuities between regions and was expressed as the difference of the square of the background residual and foreground residual. This was mathematically expressed as: $\alpha(x,t) = [(AI_{\text{out}} - g(x,y))^2 - (AI_{\text{in}} - g(x,y))^2]$. They also added the length penalty ϵ to the curvature term κ. Thus, the complete PDE equation was:

$$\frac{\partial \phi}{\partial t} + \left[\underbrace{(A(I_{\text{out}}(x,y)) - g(x,y))^2 - (A(I_{\text{in}}(x,y)) - g(x,y))^2}_{\text{region}} + \underbrace{\epsilon\kappa}_{\text{curvature}} \right] \|\nabla\phi\|$$
$$= 0. \tag{36}$$

Note here that the key factors in the PDE equation were the terms $I_{\text{in}}(x,y)$ and $I_{\text{out}}(x,y)$, which acted as the constants. Since the assumption was that $I_{\text{in}}(x,y)$ and $I_{\text{out}}(x,y)$ values were known from the binary objects and $g(x,y)$ was the observational pixel intensity, one could compute the discontinuity as the difference between $(A(I_{\text{out}}(x,y)) - g(x,y))^2$ and $(A(I_{\text{in}}(x,y)) - g(x,y))^2$. If the difference was zero, then the pixel was at the boundary.

Pros and cons of Barlaud *et al.*'s method. The technique was successfully presented to show the usage of regional terms in PDE-framework which detected the boundary of the objects. The following were the major weaknesses of this technique. First, the paper did not present the application of the technique on gray scale images. One of the major drawbacks of this technique was the *a priori* information

needed for the terms $I_{in}(x, y)$ and $I_{out}(x, y)$. Second, the model was too simple for complex gray scale medical imagery.

4.5. *3D regional geometric surface: fusion of the level set with Bayesian-based pixel classification regularizer (Barillot/IRISA)*

Barillot and his co-workers (see Baillard *et al.* [105−107]) recently designed the brain segmentation system based on the fusion of region into boundary/surface estimation. This algorithm was quite similar in approach to Suri *et al.*'s method discussed previously in Sec. 4.1. This algorithm was another instance where the propagation force V_0 in the fundamental level set segmentation equation $\partial\phi/\partial t = c(\mathbf{x})(\kappa + V_0)|\nabla\phi|$ was changed into a regional force. There were, in all, three changes made to this equation by Barillot and co-workers. The first change was in the propagation force V_0, the second was in the data consistency term or stopping term $c(\mathbf{x})$ and the third was on the step size Δt. These equations and their interpretation will be briefly discussed next.

4.5.1. *Design of the propagation force based on the probability distribution*

The key idea was to utilize the probability density function inside and outside the structure to be segmented. The pixel/voxel in the neighbourhood of the segmenting structure was responsible for creating a pull/push force on the propagating front. This was expressed in the form of the probability density function to be estimated inside the structure, $p_i(u)$, the probability density function to be estimated outside the structure, $p_e(u)$ and the prior probability for a voxel to be inside the structure. Note here, u was the intensity value of a voxel at location (x, y, z). Using the above concept, this bi-directional propagation force was estimated as:

$$V_0 = \text{sgn}\{\alpha_i p_i(u) - (1 - \alpha_i)p_e(u)\}, \tag{37}$$

where $\text{sgn}(x)$ was one if $x \geq 0$ and was -1 if $x < 0$. The second modification was to the data consistency term (also called stopping term) that changed from the gradient term into the extended gradient term. This term was changed from $c(\mathbf{x}) = 1/(1 + |\nabla[G_\sigma(\mathbf{x}) * I\mathbf{x})]|)$ to a term which was based on the transitional probability of going from inside to outside the object to be segmented. This was mathematically given as: $c(\mathbf{x}) = g[p_T(x|I, C)]$, where $g[x]$ was $1 - 4x^3$ if $x < 0.5$ and was $4(1-x)^3$, if $x \geq 0.5$. The term p_T was computed based on three parameters: α_i, $p_i(u)$ and $p_e(u)$, and mathematically estimated if the probability of a pixel/voxel class C belonged to a set inside and outside the object. If the class C was inside the region, then p_T was given as $(1 - \alpha_i)p_e(I(x))/[\alpha_i p_i(I(x)) + (1 - \alpha_i)p_e(I(x))]$, while it was $(\alpha_i)p_i(I(x))/[\alpha_i p_i(I(x)) + (1 - \alpha_i)p_e(I(x))]$ if C was outside, derived from the simple Bayesian rule.

Pros and cons of Baillard/Barillot's technique. The major advantages of this technique were: (1) The paper provided an excellent example of the fusion of region-based information into the boundary/surface; (2) The results are very impressive; however, it would have been valuable to see an enlarged version of the results; (3) The algorithm was adaptive since the data consistency term ($c(\mathbf{x})$) and the step size (Δt) were adaptively estimated in every iteration of the front propagation. This provided a good trade-off between convergence speed and stability; (4) This method used stochastic-EM (SEM) instead of expectation-minimization (EM), which is a more robust and accurate method for estimation of probability density function parameters; (5) The method had successfully been successfully applied to various brain structures and to various imaging modalities such as ultrasound; (6) The algorithm hardly needed any tuning parameters and thus, it was very efficient. Both methods (Suri *et al.*'s and Baillard *et al.*'s) were designed to control the propagation force using region-based analysis. Suri *et al.*'s method used regional-force computed using pixel-classification based on fuzzy clustering, while Baillard *et al.*'s method used pixel-classification based on Bayesian-statistics.

5. Segmentation in Motion Images Via PDE/Level Set Framework

Segmentation and tracking in motion imagery are very critical in several applications of computer vision. Here, we will discuss two different techniques of segmenting objects in an image sequence based on PDE and the level sets. In Sec. 5.1, we discuss a technique for segmenting objects in the motion sequence based on global motion compensation and robust frame differencing utilizing the PDE and level sets. Section 5.2 presents object segmentation in the motion sequence using region competition based on PDE and level set framework.

5.1. *Motion segmentation using frame difference via PDE (Zhang/UW)*

Recently, Zhang and his coworkers (see Zhang *et al.*[114,115] and Gao[113]) proposed an object segmentation in motion imagery that consisted of: (1) fast global motion compensation, (2) robust frame differencing, and (3) level set based curve evolution. The algorithm segments the objects in the temporal sequence with a moving background using the fast global motion estimation. The robust frame differencing method was adapted in order to detect motion in a sequence by calculating the 3D structural tensor in the spatial-temporal domain. The curve evolution based on level sets was used to segment the different moving objects.

5.1.1. *Eigenvalue-based-PDE formation for segmentation in motion imagery*

Zhang *et al.*'s[115] fast method of global motion estimation and compensation was in the spirit of Zhang *et al.*'s[110] method. Zhang *et al.*'s method[115] was fast as it used only hundreds of pixels rather than the whole frame for estimating the motion

parameters for an affine model. In this section, we present the mathematical formalism for eigenvalue computation given the image sequence. Motion compensation is beyond the scope of this paper (for details on motion compensation, see Zhang *et al.*[115]).

The robust motion segmentation method based on frame differencing and level set based curve evolution was the key to success in Zhang *et al.*'s technique (see Zhang *et al.*[115]). The main idea in the frame differencing technique was to calculate the 3D structural tensor in the spatial-temporal domain (see Jahne *et al.*[111]). For an image sequence $f(\mathbf{x}, t)$, where $\mathbf{x} = \{x_1, x_2\}$, the 3D structural tensor (a 3×3 matrix) was computed as the convolution of the product of gradient of f and its transpose by the Gaussian function h. This was mathematically given as:

$$\mathbf{R}(\mathbf{x}, t) = \underbrace{\int h(\mathbf{x} - \mathbf{x}', t - t') \underbrace{\nabla f(\mathbf{x}', t')}_{3 \times 1} \underbrace{\nabla f(\mathbf{x}', t')^T}_{1 \times 3} \, d\mathbf{x}' dt'}_{\text{convolution-operation}}, \qquad (38)$$

where $h(\mathbf{x}, t)$ was a spatial-temporal Gaussian function and ∇ was the gradient operator and T represented the transpose operation. The eigenvector associated with the smallest eigenvalue of $\mathbf{R}(\mathbf{x}, t)$ depicted the direction of motion for the pixel at $(\mathbf{x}, t)$. This eigenvalue was denoted by $\lambda_3(\mathbf{x}, t)$ (for details on the tensor calculation, see Jahne's book[131]). The discrete representation of tensor Eq. (38) at location (x_1, x_2, t) was:

$$R(x_1, x_2, t) = h(x_1, x_2, t) \otimes \underbrace{R'(x_1, x_2, t)}_{3 \times 3}, \qquad (39)$$

where $\otimes$ was the convolution operator and $R'(x_1, x_2, t) = [f_{x_1} f_{x_2} f_t] \times [f_{x_1} f_{x_2} f_t]^T$, where f_{x_1}, f_{x_2} and f_t were the partial derivatives for sequence f in the spatial and temporal directions. $\lambda_3(x_1, x_2, t)$ was the smallest eigenvalue of $R(x_1, x_2, t)$. It was found that compared to the simple frame differencing, the eigen-based method was more robust to noise.

Zhang *et al.*'s algorithm used the curve evolution to segment out the moving objects based on the smallest eigenvalue of $\mathbf{R}(\mathbf{x}, t)$, i.e. $\lambda_3(\mathbf{x}, t)$. If $C = C(s)$, $s \in [0, 1]$, was the closed curve, then the curve evolution equation amounts to evolving C over time $\mathbf{C}_t$ by using the following PDE:

$$\mathbf{C}_t = [g(\lambda_3)(V_0 + \kappa) - (\nabla g(\lambda_3) \cdot \mathbf{N})]\mathbf{N}, \qquad (40)$$

where $g(v)$ was a monotonically decreasing function that approached zero when v was large, and was associated with the external force; V_0 was a constant that made the curve move inward or outward; κ was the curvature of the evolving curve, that made the curve smooth; ∇ was the gradient operator; and $\mathbf{N}$ was the normal[c] to the curve. The first term in this speed function decreased to zero and stopped from

---

[c]The normal computation is the same as previously discussed in Sec. 4.1, however the symbols are different. Sometimes $\mathcal{N}$ is used instead of $\mathbf{N}$.

evolving when the curve hit the motion boundaries. The second term, $(\nabla g(\lambda_3) \cdot \mathbf{N})$, was used to track object boundaries.

5.1.2. *The Eulerian representation for object segmentation in motion imagery*

Zhang and his co-workers (see Zhang *et al.*[115] and Gao[113]) implemented the curve evolution by embedding curve C in a surface $u(\mathbf{x}; t)$. Specifically, at $t = 0$, the curve was the level set given by $\{\mathbf{x} | u(\mathbf{x}; 0) = 0\}$. The evolved curve was obtained from the level set $\{\mathbf{x} | u(\mathbf{x}; T) = 0\}$, when the evolution of $u(\mathbf{x}; t)$ stopped at $t = T$. The evolution equation for the surface $u(\mathbf{x}; t)$ was derived from the PDE Eq. (40) as:

$$\frac{\partial u}{\partial t} = \left[g(\lambda_3)(V_0 + \kappa) - \left(\nabla g(\lambda_3) \cdot \frac{\nabla u}{|\nabla u|} \right) \right] |\nabla u|. \qquad (41)$$

The initial curve in this method was the largest possible curve, which contained all moving objects. As this initial curve evolved, it moved towards the moving objects. The initial surface, $u(\mathbf{x}; 0)$, which embedded the initial curve, was given by the minimum distance between the location $\mathbf{x}$ and the initial curve. This distance was assigned a negative sign if the location $\mathbf{x}$ was inside the curve. Zhang *et al.*'s method used the narrow-band approach, originally developed by Sethian.[112]

Two examples were taken from Zhang *et al.*'s work (see Zhang *et al.*[115]) to show the representative results. In the first example (see Fig. 11), a surveillance sequence that consisted of a moving car (known as the "moving car sequence") was taken as frame numbers 7, 8 and 9, as shown in Figs. 11(a)–(c). Note here, the movement of the car is towards the viewer. Figure 11(f) shows the segmented result using Zhang *et al.*'s algorithm.[115] The image sequence contained low noise, clutter and had high object-background contrast. As a result, it was relatively easy to segment. In the second representative example (see Fig. 12) of object segmentation in the moving "outdoor sequence," the goal was to detect the walking person near the bottom right in the scene. The frames used in this sequence were 601, 602 and 603, as shown in Figs. 12(a)–(c). This sequence contained the moving objects and random motions due to trees. Figure 12(f) shows the segmented results from this "outdoor sequence" using Zhang *et al.*'s technique. As seen in this figure, the sequence was primarily hard to segment for two reasons. First, it had a lot of noise and second, there was random motion due to the trees. A simple technique such as frame difference had difficulty differentiating between these two types of motion. However, the tensor based frame-differencing embedded in PDE and the level set framework technique were effective in suppressing both the noise and the trees' random motion.

Zhang *et al.*[115] compared their technique with Paragios *et al.*'s[109] and Grimson *et al.*'s[108] approach. Paragios *et al.*'s[109] technique of curve evolution applied the frame difference images associated with an image sequence to achieve the segmentation. To enhance the frame difference images, a non-linear transform was used, which was motivated by statistical analysis in the boundary area. If the frame difference was large at a pixel position and not too large at some of its neighboring
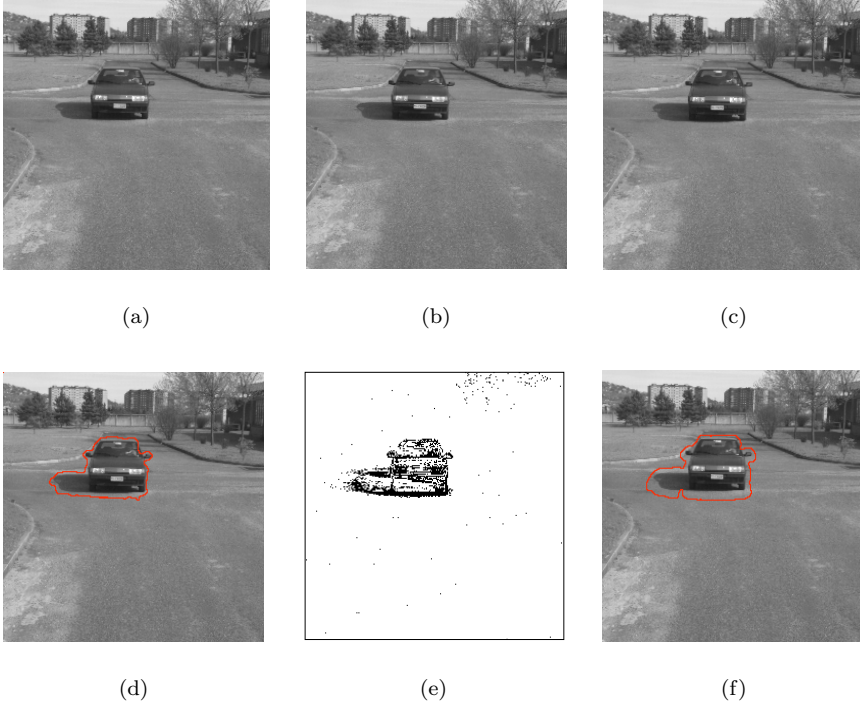
Fig. 11. Segmentation results for the "moving car sequence." (a) Frame number 7. (b) Frame number 8. (c) Frame number 9. (d) Result of the method of Paragios *et al.*[109] (e) Result of the method of Grimson *et al.*[108] (f) Result of Zhang *et al.*'s[115] method. (Reproduced with permission from IEEE.)

positions, then the pixel was considered as a point on a moving object boundary. This technique worked well when the noise level was low and the contrast was high. On the image sequence in Fig. 11, Paragios *et al.*'s technique produced good results. However, when the noise level was high and the contrast was low, or when there was random motion, Paragios *et al.*'s method did not perform well. This can be seen in Fig. 12(d), where Paragios *et al.*'s technique produced several false moving objects in the scene.

The technique by Grimson *et al.*[108] used an adaptive background estimation method using simple frame differencing. Intuitively, this worked by dynamically maintaining estimating background images. Every pixel $f(x_1, x_2, t)$ in the current frame was compared to the pixel in the same position for the estimated background images. This set of background images was denoted by $\pi_i(x_1, x_2, t)$, where $i = 1, \ldots, k$ and k was the total number of estimated background images. The pixel in location (x_1, x_2, t) was identified as a moving "object point," if it was not similar to all of the estimated background images $\pi_i(x_1, x_2, t)$. If the pixel was similar to one of the estimated background images, then the background image pixel $\pi_i(x_1, x_2, t)$ was updated by this current pixel location $f(x_1, x_2, t)$. To measure the similarities
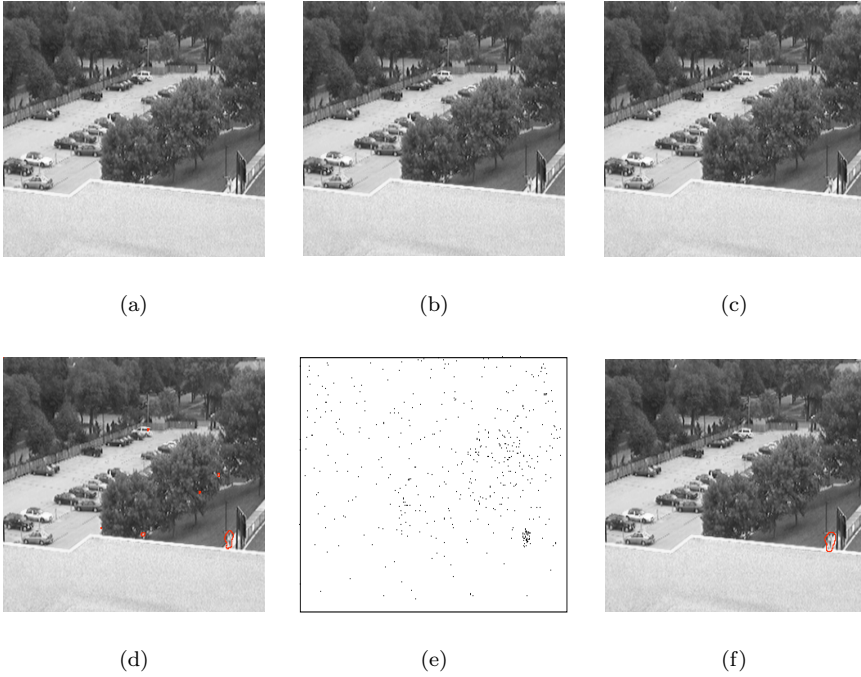
Fig. 12.   Segmentation of an "outdoor sequence." (a) Frame number 601. (b) Frame number 602. (c) Framce number 603. (d) Result of the method of Paragios *et al.*[109] (e) Result of the method of Grimson *et al.*[108] (f) Result of Zhang *et al.*'s[115] method. (Reproduced with permission from IEEE.)

between the current pixel at (x_1, x_2, t) and the background image pixel $\pi_i(x_1, x_2, t)$, the probability $P(x_1, x_2, t|i)$ was computed based on the Gaussian assumption. As seen in Figs. 11(e) and 12(e), post-processing is usually needed to link such "object points" into connected meaning objects. The advantages of this technique were their simplicity and speed; each pixel in the background image was estimated by a first-order recursive filter in time. The disadvantage was that the background images estimated were not always accurate near object boundaries. Also, the objects estimated tended to be noisy since each pixel was processed independently from its neighbors. Readers interested in segmentation of color image sequences, the extension of Zhang *et al.*'s technique, can see Gao[113] and Zhang *et al.*[115]

Pros and cons of Zhang *et al.*'s technique. The following were the major advantages of Zhang *et al.*'s technique. (1) It was insensitive to noise and global motion. This was due to the combination of successfully using three techniques: global motion compensation, robust frame differencing and level set based curve evolution. (2) It produced all segmented moving objects simultaneously. (3) Due to the fast marching method of curve evolution and fast global motion compensation, the whole system was relatively fast. (4) It was simple and straightforward to compute the tensor and eigenvalues. The following were the major weaknesses of Zhang

et al.'s technique. First, if there are several motions with large varying speeds, this technique encountered a problem. Since the λ_3 had a large range of values, it was very difficult to set the parameters, such as constant V_0, and selection of g on the curve evolution equation, to segment out all of the moving objects in the same frame. Second, since this was a method based not only on motion, the boundary of segmentation was not exact on the boundary of motion. Having discussed the frame difference method for object segmentation, we will next present a different approach for object segmentation in motion imagery, based on region competition.

5.2. *Motion segmentation via PDE and level sets (Mansouri/INRS)*

Mansouri *et al.*[117] recently demonstrated an algorithm for motion segmentation in image sequence, based on the region competition approach originally developed by Zhu and Yuille (see Zhu *et al.*[116]). The goal was to partition each frame of the sequence into regions of distinct motion. Although the results demonstrated by Mansouri *et al.* were illustrated using affine motion models, the proposed algorithm can accommodate any motion model. Motion segmentation was formulated in the two-motion case (moving object on a fixed or moving background) as a Bayesian estimation, and subsequently, energy minimization. The Euler–Lagrange descent equations corresponding to this energy minimization problem resulted in a curve evolution equation, which was then translated into the corresponding level set partial differential equation. Generalization to the multiple motion case (many moving objects) was done by analyzing the level set PDE in the two-motion case. Thus, this work was an excellent example of the fusion of regional information with the PDE and the level set framework. This research work had three characteristics: (1) The segmentation formulation used "region competition" in the PDE and the level set framework. (2) The motion segmentation algorithm was based purely on motion, unlike other techniques which also relied on intensity boundaries to refine motion boundaries. Nevertheless, the algorithm was extendable to have the capability of using intensity boundaries. (3) The motion segmentation algorithm could accommodate any motion model.

In the two-motion case, i.e. where there is a moving object on a fixed or moving background, the PDE was as follows: If $\mathcal{C}$ represented the closed curve, then the flow of the curve, represented as the rate of change $d\mathcal{C}(s)/dt$ of $\mathcal{C}$ with time, was given by a partial differential equation consisting of three terms: regional energy, curvature energy and gradient energy. Thus the PDE equation was given as:

$$
\frac{d\mathcal{C}(s)}{dt} = - \left[\underbrace{\mathcal{R}_{\mathrm{o}}^2(\mathcal{C}(s)) - \mathcal{R}_{\mathrm{b}}^2(\mathcal{C}(s))}_{\text{regional-energy}} + \underbrace{\lambda_L \kappa_{\mathcal{C}}(s)}_{\text{curvature-energy}} - \underbrace{\lambda_I \left(H_I \frac{\nabla I}{\|\nabla I\|} \right)}_{\text{gradient-energy}} \right] \cdot \mathcal{N}(s) ,
$$

(42)

where H_I was the Hessian of the image function I, $\mathcal{N}$ was the outward-pointing unit normal to $\mathcal{C}$, $\mathcal{R}_o^2 - \mathcal{R}_b^2$ represented the regional velocity term. This term was the difference between the squared motion residuals which was given as: $\mathcal{R}_o^2 - \mathcal{R}_b^2$ was actually given in terms of $\mathcal{R}(\mathbf{T}_o, \mathbf{X})_o^2 - \mathcal{R}(\mathbf{T}_b, \mathbf{X})_b^2$, where $\mathbf{T}_o$ and $\mathbf{T}_b$ were the transformations for the object and background regions, respectively. $\mathbf{X}$ represented the point with coordinates (x, y) of the object $(\mathcal{R}_o^2)$ and the background $(\mathcal{R}_b^2)$. The curvature velocity was given by $\kappa_{\mathcal{C}}(s)$, pre-multiplied by λ_L, the smoothing constant. This curvature velocity of the closed contour followed directly from the prior one on $\mathcal{C}$ which minimized curve length (as had been popular to do so). The last term was the velocity due to the image gradient and favored alignment of motion boundaries with intensity boundaries. This was given as $-H_I \nabla I / \|\nabla I\|$, pre-multiplied by the constant λ_I. The interpretation is as follows: (1) A boundary point $\mathcal{C}(s)$ undergoing the same motion as the object would satisfy $\mathcal{R}_o^2(\mathcal{C}(s)) \leq \mathcal{R}_b^2(\mathcal{C}(s))$, i.e. its motion residual will be smaller (in absolute value) for object motion than for background motion. Neglecting the contribution of curvature and intensity boundaries, this would induce a velocity vector $d\mathcal{C}(s)/dt$ in the direction of the outward normal $\mathcal{N}$, *growing* the curve $\mathcal{C}$ so as to englobe the point $\mathcal{C}(s)$. (1) If $\mathcal{C}(s)$ was a point undergoing the same motion as the background, then, neglecting the effect of curvature and intensity boundaries, $d\mathcal{C}(s)/dt$ would be oriented opposite to the normal $\mathcal{N}$, and curve $\mathcal{C}$ would *shrink*, relinquishing the point $\mathcal{C}(s)$. (3) The term $\kappa_{\mathcal{C}}(s)$ would tend to straighten out the curve, by pulling in convexities ($\kappa_{\mathcal{C}} > 0$), and pushing out concavities ($\kappa_{\mathcal{C}} < 0$). The Eulerian level set representation of Eq. (42) was given by:

$$\frac{\partial u}{\partial t}(\mathbf{X}) = -\left[\underbrace{\mathcal{R}_f^2(\mathbf{X}) - \mathcal{R}_b^2(\mathbf{X})}_{\text{regional-term}} + \underbrace{\lambda_L \kappa_{\mathcal{C}}(\mathbf{X})}_{\text{curvature-term}} - \underbrace{\lambda_I \left(H_I \frac{\nabla I}{\|\nabla I\|}\right)}_{\text{gradient-term}}\right] \|\nabla_u\|. \quad (43)$$

$$\underbrace{}_{\text{velocity}}$$

Note that Eq. (43) can be compared to the standard equation by Sethian: $\partial u/\partial t = V(\kappa)\|\nabla_u\|$, where $V(\kappa)$ was the velocity with which the curve propagated in the direction of the gradient of the level set function u.

Figure 13 shows the result of motion segmentation using Mansouri *et al.*'s[117] algorithm on frames 30 and 40 of the "*ping-pong* sequence," using affine motion models with $\lambda_L = 2500$ and $\lambda_I = 0$. These motion models were computed prior to the segmentation, using point correspondences and clustering in affine parameter space (see Mansouri *et al.*[117] for details). A total of three distinct motion classes were found: the first, corresponding to the (still) background; the second, to the ball, having an upwards direction of motion; and finally, the third motion class was that of the hand and was a rotational motion around the elbow in the counterclockwise direction. The top-left and top-right pictures show frames 30 and 40 of "*ping-pong* sequence," respectively. In the middle-left picture, the initial zero level set curves are shown. The final segmentation is shown in the remaining pictures. In
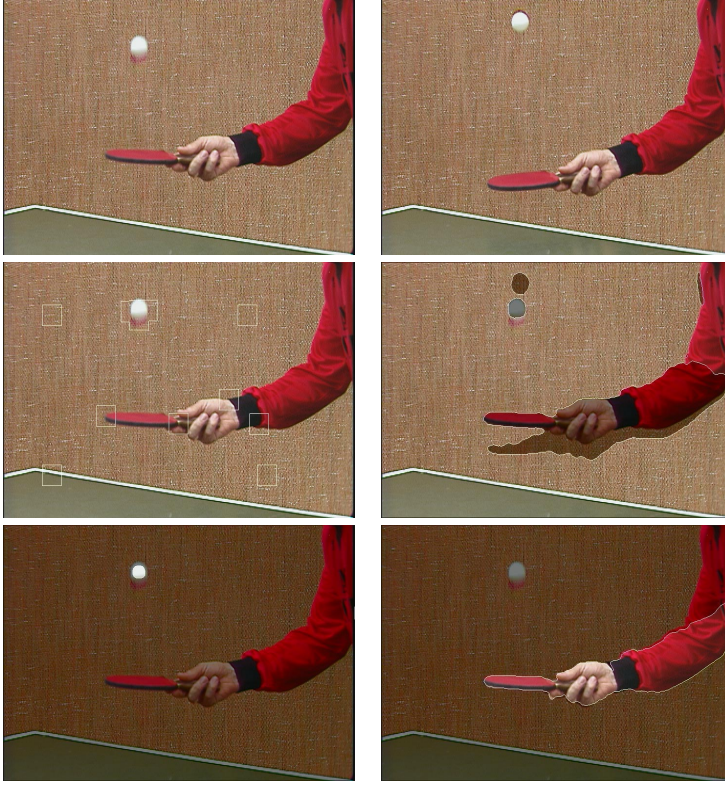
Fig. 13. Segmentation results of the "*ping-pong* sequence" using PDF implemented in the level set framework. Top left: Frame 30 of the "*ping-pong* sequence." Top right: Frame 40 of the "*ping-pong* sequence*". Middle left: Initial segmentation: Three squares are associated with one region and there are three regions: background, ball and hand. Each of these square boundaries is the zero level set function corresponding to the respective regions. Middle right: Final segmentation of the background. Bottom left: Final segmentation of the ball. Bottom right: Final segmentation of the hand. (Courtesy of Abdol-Reza Mansouri, INRS Telecommunications, Montreal, Quebec, Canada.)

each of them, for ease of viewing, the region segmented is shown without alteration, while the remaining regions have been dimmed. The middle-right picture shows the background region; note that in addition to the ball and the hand, the occlusion regions have also been excluded from the background. The bottom-left picture shows the ball region, and finally, the bottom-right picture shows the hand region. Note that all these regions have been very precisely estimated, despite the fact that no use was made of intensity boundaries in this segmentation.

Pros and cons of region competition in PDE/level set framework for segmentation in motion imagery. The major advantages of this technique were as follows. (1) Although Mansouri *et al.*'s algorithm[117] has the capability to rely on intensity boundaries to guide the motion segmentation, it did not need to. Indeed,

the vast majority of their results have been obtained through segmentation based on motion alone, which demonstrated that very accurate motion segmentation can be obtained, even where intensity boundaries were hardly visible. (2) The algorithm can be combined with any motion detection algorithm. All that was needed for this to work successfully was the number of distinct motion classes and their parameters. (3) Initial conditions for the proposed level set PDE were quite arbitrary, and the initial level set functions need not be in the vicinity of the solution. The major weakness of Mansouri *et al.*'s algorithm was that the computational burden of the algorithm grew linearly with the number of motion classes. Thus, the larger the number of distinct motion classes, the slower the resulting segmentation. Recently, Aubert and his co-workers (see Besson *et al.*[104]) also developed a segmentation and tracking method for the motion sequence based on the scheme presented in Sec. 4.4. Interested readers can explore this method further.

6. Miscellaneous Applications of PDEs in Image Processing

This section presents additional applications of PDE-based approaches. In Sec. 6.1, we discuss the PDE-based approach for filling missing information. Section 6.2 presents the mathematical morphology implementation using the PDE-based approach. Section 6.3 presents the low pass filtering operation.

6.1. *PDE for filling missing information for shape recovery using mean curvature flow of a graph*

Sarti *et al.*[127] recently fused the local and global information in PDE/level set framework to estimate the missing edge information. If c was the edge extractor or edge indicator function, ϕ the level set function, then the flow equation derived by Sarti *et al.* was: If $S : (x, y) \to (x, y, \phi)$ was defined in the same domain M of the image I, then the differential area of the graph S in the Euclidean space was given as $dA_E = \sqrt{1 + \phi_x^2 + \phi_y^2}\, dxdy$. Applying the edge indicator function or data consistency term, $c(x, y)$ to this and then computing the area A_c under the surface was given as:

$$A_c = \int_M c(x, y)\sqrt{1 + \phi_x^2 + \phi_y^2}\, dxdy \,, \tag{44}$$

where M was the domain. Note, this comes about from the *mean curvature flow* which was the steepest descent of the *area functional*. This equation was minimized using the steepest descent by applying multivariate calculus to obtain the flow equation as:

$$\frac{\partial \phi}{\partial t} = c(x, y)\frac{(1 + \phi_x^2)\phi_{yy} - 2\phi_x\phi_y\phi_{xy} + (1 + \phi_y^2)\phi_{xx}}{1 + \phi_x^2 + \phi_y^2} + (c_x\phi_x + c_y\phi_y) \,, \tag{45}$$

where c_x and c_y are the edges in the x and y directions, respectively. Figure 14 shows the results of running the above algorithm. Other work of Sarti in the area of level sets can also be explored (see Sarti *et al.*[128−130]).
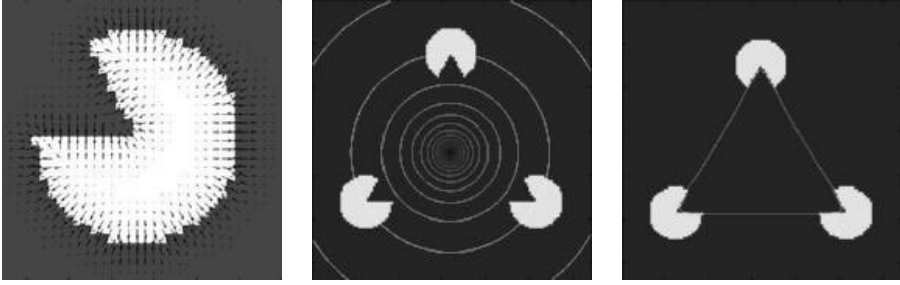
Fig. 14. Filling missing information for shape recovery. Left: Kanizsa Triangle initial image. Middle: Evolution process: Right: Final triangle boundary with missing information completed. (Courtesy of A. Sarti, Mathematics Department, LBNL, University of California, Berkeley and DEIS, University of Bologna, Italy.)

Pros and cons of Sarti *et al.*'s method. First, Sarti *et al.*'s method did not compute the flow of the surface using the *geometry of the levels*, but it did compute the flow depending upon the *geometry of the surface itself*. Second, the above equation was the *weighted mean curvature flow of a graph*, while Sarti *et al.*'s work[86] was the *level set flow* on a 3D volume.

6.2. *Mathematical morphology via PDE*

The mathematical morphology field has been used for shape analysis and shape quantification for some time (see Sapiro *et al.*,[66] Arehart *et al.*,[67] Catte *et al.*[68] and Sochen *et al.*[69]). It was in the early 1990's when the role of level sets started to move into mathematical morphology. Here, we will just show the dilation transformation modeling. Let some shape $\mathcal{C}(s,t)$ evolved by a mathematical morphology operation (say, dilation), by some structuring element $\mathcal{B}(\lambda)$ of size λ, to a new shape $\mathcal{C}(s,t+\lambda)$. This evolution was represented as $\mathcal{C}(s,t+\lambda)-\mathcal{C}(s,t) = \Gamma(s,t,\lambda)\mathcal{N}(s,t)$, where Γ was the distance along $\mathcal{N}$ that a point on the boundary moved in a dilation operation with a structuring element of size λ. Since $\Gamma(s,t,0)=0$:

$$\frac{\partial \mathcal{C}}{\partial t} = \lim_{\lambda \to 0} \frac{\Gamma(s,t,\lambda) - \Gamma(s,t,0)}{\lambda} \mathcal{N}\,. \tag{46}$$

The above equation was written as:

$$\frac{\partial \mathcal{C}}{\partial t} = \frac{\Gamma(s,t,\lambda)}{\partial \lambda} \mathcal{N}\,. \tag{47}$$

The above equation was simply $\beta(s,t)\mathcal{N}$, where β was the differential deformation of a shape $\mathcal{S}$ at a point P due to the dilation with a convex structuring element. This was also the maximal projection of the structuring element onto the normal $\mathcal{N}$ of the boundary at P. Note that β depended on the orientation of the target to the tangent shape $\mathcal{T}(s,t)$. This led to the evolution of the equation as:

$$\frac{\partial \mathcal{C}}{\partial t} = \beta(\mathcal{T}(s,t))\mathcal{N}$$

$$\mathcal{C}(s,0) = \mathcal{C}_0(s)\,.$$

$$(48)$$

We can similarly model the erosion process (for details, see Sochen *et al.*[69]).

6.2.1. *Erosion with a straight line via PDE*

If I was the original image, then the erosion of image I with the straight line segment was given as:

$$I_e = |\hat{n} \cdot \nabla I|\,, \tag{49}$$

where $\hat{n}$ was the vector representing the direction vector $(\cos(\theta)\sin(\theta))$ and θ was the angle between the given line segment and the x-axis. The above equation was written as:

$$I_e = \|\hat{n}\| \, \|\nabla I\| \cos(\alpha)\,, \tag{50}$$

where α was the angle between $\hat{n}$ and ∇I. Since $\|\nabla I\|$ was computed in the level set bed as it was a PDE, thus the morphologic erosion was solved via PDE. Note, here α was computed by taking the inverse of the tangent of the ratio of the y and x components and then subtracted by θ.

6.3. *PDE in the frequency domain: a low pass filter*

Banks[118] showed the interpretation of the finite difference method (FDM) for the PDE as a low pass filter (LPF). This was shown using the frequency domain analysis. We will discuss this key derivation of PDE as a LPF using the Fast Fourier Transform (FFT) method (for details, see Bloor *et al.*[119] and Brown *et al.*[120]). The ability to model the fitting problem into a PDE using the Euler–Lagrange method and then applying the finite difference method (FDM) to solve the PDE was one of the key characteristics of the PDE-based methods. The smoothing problem was posed as a surface fitting problem or minimization of the surface ϕ to the data points, say P_k. This was posed as a Least Squares problem where they minimized the function I as:

$$I = \alpha \underbrace{\sum_{k=1}^{N_p} [\phi(u_k, v_k) - P_k]^2}_{\text{closeness}}$$

$$+ \beta \int_{v_1}^{v_{n+1}} \int_{u_1}^{u_{n+1}} \underbrace{\left\| \left\| \frac{\partial^n \phi(u,v)}{\partial u^n} \right\|^2 + \left\| \frac{\partial^n \phi(u,v)}{\partial v^n} \right\|^2 \right\|}_{\text{smoothness}} du\,dv\,, \tag{51}$$

where α, β were the smoothing constants. n was the degree of the surface and $\partial^n \phi(u,v)/\partial u^n$ was the nth derivative of the surface ϕ with respect to the variable

u and $\partial^n \phi(u,v)/\partial v^n$ was the nth derivative of the surface ϕ with respect to the variable v. N_p was the total limit. Note, the surface fitting consisted of two terms: an error or closeness term which reduced the fitting error between the data points and the surface passing through them, and the second term was the smoothness term reducing the variations of the surface. Note the smoothness term was also the curvature reduction term. The second term was the smoothness term and we will focus on that. The second term can be expressed as a function of the derivatives of u and v. Taking the simplest case $n = 1$, the smoothing function (given as L) was given as: $L = \phi_u^2 + \phi_v^2$. The Euler–Lagrange equation is thus given as: $\partial^2 \phi/\partial u^2 + \partial^2 \phi/\partial v^2 = 0$, and we know the finite difference solution of this PDE. Now to show the PDE to be a LPF, imagine a continuous surface $f(u)$ being sampled at N intervals on regularly spaced grids $u = ih$, where $i = 0, \ldots, (N-1)$. By taking the Fourier Transform of the sampled signal f_n with frequency k, this is given as:

$$\hat{f}_k = \sum_{n=0}^{N-1} f_n \exp\left(\frac{-2\pi i n k}{N}\right). \tag{52}$$

If $\phi_{i,j}$ was the discrete function and we sample it into M points in two directions u and v, then the discrete Fourier Transform of $\phi_{m,n}$ was given as:

$$\hat{\phi}_{k,l} = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} \phi_{m,n} \exp\left(-2\pi i\left[\frac{mk}{M} + \frac{nl}{N}\right]\right). \tag{53}$$

We know that using the finite difference method, the value of the Euler–Lagrange equation was given as:

$$[\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j} + \phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}] - \lambda\phi_{i,j} = -\lambda P_{i,j}. \tag{54}$$

Substituting $\phi_{m,n}$ into the above equation and equating the coefficients of the (k,l) frequency mode, we get:

$$\hat{\phi}_{k,l} \frac{-\lambda \hat{P}_{k,l}}{[\sin^2(\pi k/M) + \sin^2(\pi l/N) + \lambda/4]}. \tag{55}$$

Equation (54) can be interpreted as an input-output with a transfer function $-\lambda \hat{P}_{k,l}/[\sin^2(\pi k/M) + \sin^2(\pi l/N) + (\lambda/4)]$. The output $\hat{\phi}_{k,l}$ was being smoothed in the sense that the amplitudes of the high frequency modes have been reduced with respect to low frequency, which is simply the LPF.

7. Advantages, Disadvantages, Conclusions and the Future of 2D and 3D PDE-Based Methods in Medical and Non-Medical Applications

This section is divided into the following parts. Section 7.1 presents a general purpose PDE/level set framework where most of the image segmentation issues fit in. Section 7.2 presents an example demonstrating the segmentation application using PDE and the level set framework. Section 7.3 discusses the advantages of PDE in

conjunction with level sets. Section 7.4 presents the disadvantages of the PDE in the level set framework. Finally, Sec. 7.5 presents a discussion on the conclusions and the future for PDE and level set based approaches.

7.1. *PDE framework for image processing: implementation*

Figure 15 shows a general purpose PDE and level set based image processing segmentation system, where, given the data set and the raw contour, one can model the segmentation by primarily computing the regional statistics term (called statistical modeling), gradient term and curvature analysis of the moving contour. An example of such an implementation can be seen by Suri *et al.*[3−6] They are all fused into one model and the segmentation problem can be solved as an energy minimization problem. This was used to frame the PDE equation which was solved in the level set framework to estimate the converged boundary. Thus, the PDE and level set

Segmentation Modeling: Energy Minimization + PDE + Level Set Framework
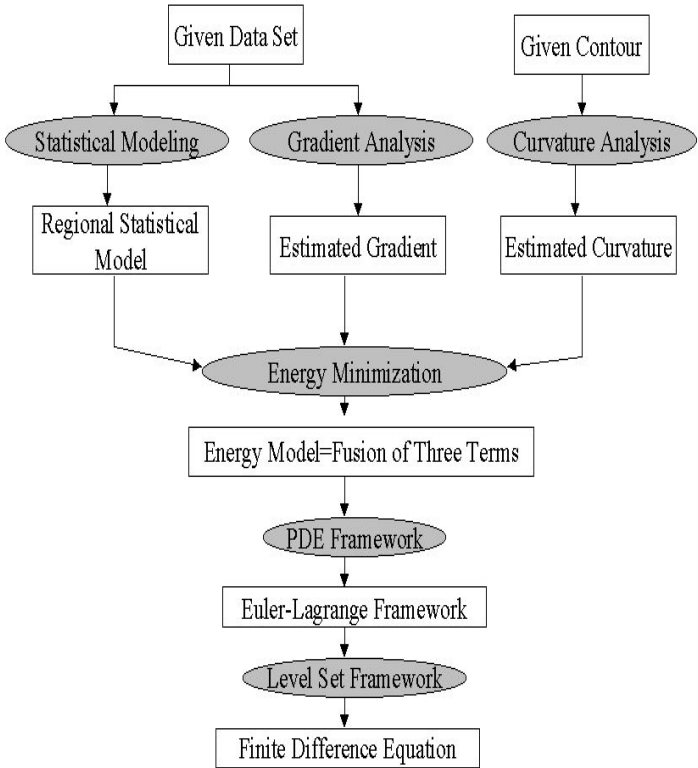

Fig. 15.   The generic level set framework consists of three stages: (i) energy minimization; (ii) PDE formation; and (iii) implementation in the level set framework. The input to the energy minimizaion algorithm consists of three terms: (a) design of the regularizer or statistical modeling; (b) gradient estimation technique; and (c) the curvature analysis.

framework have the following abilities: first, to transform a segmentation modeling problem into a partial differential equation and to embed the regularizers into these models; second, solving these partial differential equations using the finite difference methods; third, linkage between the partial differential equations and the level set framework for implementing the finite difference methods; and fourth, the ability to extend the PDE framework from 2D to 3D or higher dimensions. Having presented the abstract framework, we will next present a segmentation example using a finite difference method and then summarize the advantages/disadvantages of the PDE and the level set based segmentation modeling issue in Secs. 7.3 and 7.4.

7.2. *A segmentation example using a finite difference method*

Here, speed control functions and their integration in terms of the level set function ϕ to estimate the ϕ over time are presented (for details, see Suri[1,3]). The time step restrictions for solving the partial differential equation will not be discussed here (the reader can refer to the work by Osher and Sethian[48] and the recent work by Barillot and his co-workers[105]). Using the finite difference method (see also Sethian[89]), the level set Eq. (27) was given in terms of time n as (for details, see Suri[2,3]):

$$\phi_{x,y}^{n+1} = \phi_{x,y}^n - \Delta t \{V_{\text{reg}}(x,y) + V_{\text{grad}}(x,y) - V_{\text{cur}}(x,y)\}, \tag{56}$$

where $\phi_{x,y}^n$ and $\phi_{x,y}^{n+1}$ were level set functions at pixel location (x,y) at times n and $n+1$, Δt was the time difference, and $V_{\text{reg}}(x,y)$, $V_{\text{grad}}(x,y)$ and $V_{\text{cur}}(x,y)$ were the regional, gradient and curvature speed terms, respectively. Now, these terms are presented as follows. First, the regional speed term expressed in terms of the level set function (ϕ) was given as: $V_{\text{reg}}(x,y) = \max(V_p(x,y),0)\nabla^+ + \min(V_p(x,y),0)\nabla^-$, where terms $V_p(x,y)$, ∇^+ and ∇^- were given as: $V_p(x,y) = w_R/\gamma R_{\text{ind}}(x,y)$, $R_{\text{ind}} = 1 - 2u(x,y)$, $\nabla^- = [\nabla_x^- + \nabla_y^-]^{1/2}$ and $\nabla^+ = [\nabla_x^+ + \nabla_y^+]^{1/2}$.

$$
\begin{aligned}
\nabla_x^+ &= [\max(D^{-x}(x,y),0))^2 + \min(D^{+x}(x,y),0))^2] \\
\nabla_y^- &= [\max(D^{-y}(x,y),0))^2 + \min(D^{+y}(x,y),0))^2],
\end{aligned}
\tag{57}
$$

where $u(x,y)$ took a value between zero and one. This could be coming from, say, a fuzzy membership function or any other clustering technique. R_{ind} was the region indicator function that was in the range between -1 to $+1$. Second, the gradient speed term, called the edge strength of the object boundaries, was expressed in terms of the level set function (ϕ) as the x and y components of the gradient speed as: $V_{\text{grad}}(x,y) = V_{\text{grad x}}(x,y) + V_{\text{grad y}}(x,y)$, where:

$$V_{\text{grad x}}(x,y) = \max(p^n(x,y),0)D^{-x}(x,y) + \min(q^n(x,y),0)D^{+x}(x,y),$$

$$V_{\text{grad y}}(x,y) = \max(q^n(x,y),0)D^{-y}(x,y) + \min(q^n(x,y),0)D^{+y}(x,y), \tag{58}$$

$$p^n(x,y) = \nabla_x(w_e \nabla(G_\sigma * I)) \quad \text{and} \quad q^n(x,y) = \nabla_y(w_e \nabla(G_\sigma * I)),$$

where w_e was the weight of the edge and was also a fixed constant. $p^n(x,y)$ and $q^n(x,y)$ were defined as the x and y components of the gradient strength at a pixel

location (x, y). Note that the regional and edge speed terms depended upon the forward and backward difference operator which were defined in terms of the level set function ψ, defined as:

$$D^{-x}(x, y) = \frac{(\phi(x, y) - \phi(x - 1, y))}{\Delta x} \quad \text{and} \quad D^{+x}(x, y) = \frac{(\phi(x + 1, y) - \phi(x, y))}{\triangle x},$$

$$D^{-y}(x, y) = \frac{(\phi(x, y) - \phi(x, y - 1))}{\Delta y} \quad \text{and} \quad D^{+y}(x, y) = \frac{(\phi(x, y + 1) - \phi(x, y))}{\Delta y},$$

(59)

where $\phi(x, y)$, $\phi(x - 1, y)$, $\phi(x + 1, y)$, $\phi(x, y - 1)$, $\phi(x, y + 1)$ were the level set functions at pixel locations (x, y), $(x - 1, y)$, $(x + 1, y)$, $(x, y - 1)$, $(x, y + 1)$, being the four neighbours of (x, y). Third, the curvature speed term expressed in terms of the level set function (ϕ) was given as: $V_{\text{cur}}(x, y) = \epsilon \kappa^n(x, y)[(D^{0x}(x, y))^2 + ((D^{0y}(x, y))^2]^{1/2}$, where ϵ was a fixed constant, $\kappa^n(x, y)$ was the curvature at a pixel location (x, y) at nth iteration as: $\kappa^n(x, y) = (\phi_{xx}^2 \phi_y^2 - \phi_x^2 \phi_y^2 \phi_{xy}^2 + \phi_{yy}^2 \phi_x^2)/(\phi_x^2 + \phi_y^2)^{3/2}$ and $D^{0x}(x, y)$ and $D^{0y}(x, y)$ were defined as:

$$D^{0x}(x, y) = \frac{(\phi(x + 1, y) - \phi(x - 1, y))}{2\Delta x}$$

and

$$D^{0y}(x, y) = \frac{(\phi(x, y + 1) - \phi(x, y - 1))}{2\Delta y}.$$

(60)

Thus, to numerically solve Eq. (56), all that was needed was: (i) the gradient speed values (p, q), (ii) the curvature speed κ at pixel location (x, y) and (iii) the membership function $u(x, y)$ for a particular class K. In the next section, how these speeds control mathematical functions will be discussed and how they are used to compute the field flow (level set function, ϕ) in the "narrow band" using the "fast marching method," also called the "optimization technique."

7.3. *Advantages of PDE in the level set framework*

The PDE based method in the level set framework offers a large number of advantages. Since the number of advantages is large, we will present them in a list format as follows. (1) *Capture Range*: The greatest advantage of this technique is that this algorithm increases the capture range of the "field flow," which increases the robustness of the initial contour placement. (2) *Effect of Local Noise*: When the regional information is integrated into the system, then the local noise or edge does not distract the growth process. This technique is non-local and thus the local noise cannot distract the final placement of the contour or the diffusion growth process. (3) *No Need of Elasticity Coefficients*: These techniques are not controlled by the elasticity coefficients, unlike the classical parametric contour methods. There is no need to fit the tangents to the curves and compute the normals at each vertex. In this system, the normals are embedded in the system using the divergence of the

field flow. These methods have the ability to model the incremental deformations in the shape. (4) *Suitability for Medical Image Segmentation*: These techniques are very suitable for medical organ segmentation since they can handle any of the cavities, concavities, convolutedness, splitting or merging. (5) *Finding the Local and Global Minima*: There is no problem finding the local minima or global minima issues, unlike the optimization techniques of parametric snakes. (6) *Normal Computation*: These techniques are less prone to the normal computational error which is very easily incorporated in the "classical balloon force" snakes for segmentation. (7) *Automaticity*: It is very easy to extend this model from semi-automatic to completely automatic because the region is determined on the basis of prior information. (8) *Integration of Regional Statistics*: These techniques are based on the propagation of curves (just like the propagation of ripples in the tank or propagation of the fire flames) utilizing the region statistics. (9) *Flexible Topology*: These techniques adjust to the topological changes of the given shape such as joining and breaking of the curves. (10) *Wide Applications*: This technique can be applied to unimodal, bimodal and multi-modal imagery, which means it can have multiple gray scale values in it. These PDE/level set based methods have a wide range of applications in 3D surface modeling. (11) *Speed of the System*: These technique can be implemented using the fast marching methods in the narrow band and thus can be easily optimized. (12) *Extension*: This technique is an easy extension from 2D to 3D. (13) *Incorporation of Regularizing Terms*: This can easily incorporate other features for controlling the speed of the curve. This is done by adding an extra term to the region, gradient and curvature speed terms. (14) *Handling Corners*: The system takes care of the corners easily unlike the classical parametric curves, where it needs special handling at the corners of the boundary. (15) *Resolution Changes*: This technique is extendable to multi-scale resolutions, which means that at lower resolutions, one can compute regional segmentations. These segmented results can then be used for higher resolutions. (16) *Multi-phase Processing*: These techniques are extendable to multi-phase, which means that if there are multiple level set functions, then they automatically merge and split during the course of the segmentation process. (17) *Surface Tracking*: Tracking surfaces are implemented using level sets very smoothly. (18) *Quantification of 3D Structures*: Geometrical computations can be done in a natural way, for example, one can compute the curvature of 3D surfaces directly while performing normal computations. (19) *Integration of Regularization Terms*: Allows easy integration of vision models for shape recovery such as fuzzy clustering, Gibbs model, Markov Random Fields and Bayesian models (see Paragios *et al.*[63]). This makes the system very powerful, robust and accurate for medical shape recovery. One can segment any part of the brain depending upon the membership function of the brain image. So, depending upon the number of classes estimated, one can segment any shape in 2D or 3D. (20) *Concise Descriptions*: One can give concise descriptions of differential structures using level set methods. This is because of the background mesh resolution controls. (21) *Hierarchical Representations*: The level set offers a natural scale space for hierarchical

representations. (22) *Reparameterization*: There is no need for reparameterization for curve/surface estimation during the propagation, unlike in the classical snakes model. (23) *Modeling in a Continuous Domain*: One can model the segmentation process in a continuous domain using PDEs. Thus the formalism process is greatly simplified which is grid independent and isotropic. (24) *Stability Issues*: With the help of research in numerical analysis, one can achieve highly stable segmentation algorithms using PDEs. (25) *Existence and Uniqueness*: Using PDEs, one can derive not only successful algorithms but also useful theoretical results, such as existence and uniqueness of solutions (see Alvarez *et al.*[23]).

7.4. *Disadvantages of PDE in level sets*

Even though level sets have dominated several fields of imaging science, these front propagation algorithms have certain drawbacks. They are as follows. (1) *Convergence Issue*: Although the edges will not be blurry when one performs the diffusion imaging, the issue of convergence always remains a challenge. In diffusion imaging, if the step size used is small, then it takes longer to converge. (2) *Design of the Constant Force*: The design of the constant force in the PDE is another challenge yet to be overcome. This involves computation of regional statistics in the region of the moving contour. There is a trade-off between the robustness of the regional design, computational time for the operation and the accuracy of the segmentation. The design of the model plays a critical role in segmentation accuracy and remains a challenge. Another challenge occurs if the design force is internal or external (for details, see Suri *et al.*[6]). (3) *Stability Issues*: The stability issues in PDEs are also important during the front propagation. The ratio of $\Delta t/\Delta x$, called the CFL number (Courant number, named after the author Courant *et al.*[136]), is another factor which needs to be carefully designed. (4) *Initial Placement of the Contour*: One of the major drawbacks of the parametric active contours has been its initial placement. This does not have either enough capture range or enough power to grab the topology of the shapes. Both of these drawbacks are removed by level sets provided the initial contour has been placed symmetrically with respect to the boundaries of interest. This ensures that the level sets reach object boundaries almost at the same time. On the contrary, if the initial contour is much closer to the first portion of the object boundary compared to the second portion, then the evolving contour crosses over the first portion of the object boundary. This is because the stop function does not turn out to be zero. One of the controlling factors for the stop function is the gradient of the image. The relationship of the stop function to the gradient is its inverse and also depends upon the index power m in the ratio $1/1 + |\nabla G_\sigma * I(x, y)|^m$. For stopping the propagation, the denominator needs to be large, which means the image forces due to the gradient need to be high. This means index m needs to be high. In other words, if m is high, then the gradient is high, which means the weak boundaries are not detected well and will be easily crossed over by the evolving curve. If m is low (low threshold), then the level set

will stop at noisy or at isolated edges. (5) *Embedding of the Object*: If some objects (say, the inner objects) are embedded in another object (the outer object), then the level set will not capture all the objects of interest. This is especially true if the embedded objects are asymmetrically situated. Under such conditions, one needs multiple initializations of the active contours. This means there can be only one active contour per object. (6) *Gaps in the Object Boundaries*: This is one of the serious drawbacks of the level set method and has been pointed out by Siddiqi *et al.*[77] Due to the gaps in the object, the evolving contour simply leaks through the gaps. As a result, the objects represented by incomplete contours are not captured correctly and fully. This is especially prominent in realistic images, such as in ultrasound and in multi-class MR and CT images. (7) *Problems Due to Shocks*: Shocks are among the most common problems in level sets. Kimia and co-workers in Refs. 124–126 developed such a framework by representing shape as the set of singularities (called shocks) that arise in a rich space of shape deformations as classified into the following four types: (i) first-order shocks are orientation discontinuities (corners) and arise from protrusions and indentations; (ii) second-order shocks are formed when a shape breaks into two parts during a deformation; (iii) third-order shocks represent bends; and (iv) fourth-order shocks are the seeds for each component of a shape. These shocks arise in level sets and can sometimes cause serious problems. (8) *Challenge in Segmentation*: Although the level set segmentation method succeeds in the object and motion segmentation, it has weakness in segmenting many other kinds of images. This can occur in images that do not have a homogeneous background; for example, when images are composed of many different regions, such as in natural scenery images (containing streets, mountains, trees, cars and people). The method based on curve evolution will not produce the correct regions as desired. Such a segmentation problem is a challenge yet to be overcome.

7.5. *Conclusions and the future in PDE-based methods*

The class of differential geometry, also called PDE in conjunction with level sets, has been shown to dominate image processing, in particular medical imaging, in a major way. We still need to understand how the regularization terms can be integrated into the level sets to improve segmentation schemes. Even though the application of level sets has gone well in the fields of medical imaging, biomedicine, fluid mechanics, combustion, solidification, CAD/CAM, object tracking/image sequence analysis and device fabrication, we are still far away from achieving stable 3D volumes and a standard segmentation in real-time. By standard, we mean that which can segment the 3D volume with a wide variation in pulse sequence parameters (see Suri[137]). We will likely see in the near future the modeling of front propagation that takes into account the physical constraints of the problem, for example, minimization of variation geodesic distances, rather than simple distance transforms. We will likely also see more incorporation of likelihood functions and adaptive fuzzy models to prevent leaking of the curves/surfaces. A good example of

integration of the low level processes into the evolution process could be given as: $\partial\phi/\partial t = L(x,y)(\beta_0 - \beta_1\kappa)|\nabla\phi|$, where $L(x,y) = 1 - \max(S_1, S_2, S_i, \ldots, S_n)$, where S_i is the low level process from edge detection, optical flow, stereo disparity, texture, etc. The better the $S_i(x,y)$, the more robust could be the level set segmentation process. We also hope to see more papers on level sets where the segmentation step does require a re-initialization stage (see Zhao *et al.*[122] and Evans *et al.*[123]). It would also, however, be helpful if we can incorporate a faster triangulation algorithm for isosurface extraction in 3D segmentation methods.

We also see a massive effort by the computer vision community to integrate regularization terms to improve the robustness and accuracy of 3D segmentation techniques. In this paper, we have shown the role of the PDE and level set method for image smoothing, image diffusion or image denoising. Also shown was how curve/surface propagation hypersurfaces based on differential geometry are used for the segmentation of objects in still imagery. We have also shown the relationship between the parametric deformable models and curve evolution framework; incorporation of clamping/stopping forces to improve the robustness of these topologically independent curves/surfaces. We have discussed considerably segmentation of an object in motion imagery based on PDE and the level set framework. In this paper, we have also presented research in the area of coupled PDEs for edge preservation and smoothing. Some coverage has also been given on PDE in miscellaneous applications. Finally, this paper concludes with the advantages and the disadvantages of segmentation modeling via geometric deformable models (GDM), PDE and level sets.

Acknowledgments

References

1. J. S. Suri, "Two dimensional fast MR brain segmentation using a region-based level set approach," *Int. J. Engin. Medicine and Biology* **20**(4), (July/August, 2001) pp. 84–95.
2. J. S. Suri, "Leaking prevention in fast level sets using fuzzy models: An application in MR brain," in *Internation Conference in Information Technology Biomedicine* (November 2000), pp. 220–226.

3. J. S. Suri, "White matter/gray matter boundary segmentation using geometric snakes: A fuzzy deformable model," in *Proc. Int. Conference on Advances in Pattern Recognition, Lecture Notes in Computer Science (LNCS)* No. 2013, eds. S. Singh, N. Murshed, and W. Kropatsch (Springer-Verlag, Rio de Janeiro, Brazil), March 11–14, 2001, pp. 331–338.

4. J. S. Suri, S. K. Setarehdan, and S. Singh, "Advanced algorithmic approaches to medical image segmentation: State-of-the-art applications in cardiology," in *Neurology, Mammography and Pathology*, ISBN 1-85233-389-8, in press (2001).

5. J. S. Suri, S. Singh, and L. Reden, "Computer vision and pattern recognition techniques for 2D and 3D MR cerebral cortical segmentation: A state-of-the-art review," accepted for publication in *Int. J. Pattern Analysis and Applications* (2002).

6. J. S. Suri, S. Singh, S. Laxminarayana, X. Zeng, K. Liu, and L. Reden, "Shape recovery algorithms using level sets in 2D/3D medical imagery: A state-of-the-art review," in *IEEE Trans. Information Technology in Biomedicine (ITB)* (2002) in press.

7. J. S. Suri, *et al.*, *State-of-the-art PDE and level sets algorithmic approaches to static and motion imagery segmentation* (400 Pages), book submitted for review, 2001.

8. S. Haker, Geometric PDEs in Computer Vision, Ph.D. Thesis, *Dept. Comput. Science and Engin.* (University Minnesota, Minneapolis, MN, 1999).

9. A. Chambolle, "Partial differential equations and image processing," in *Proc. First IEEE Int. Conference on Image Proc.*, Austin, Texas, (November 1994), pp. 16–20 .

10. J.-M. Morel and S. Solimini, *Variational Methods in Image Segmentation* (Boston, MA, Birkhauser, 1995), ISBN 0-8176-3720-6.

11. G. Sapiro, *Geometric Partial Differential Equations and Image Analysis* (Cambridge University Press, Cambridge, MA, 2001), ISBN 0-521-79075-1.

12. J. Weickert, "Fast segmentation methods based on partial differential equations and the watershed transformation," *Mustererkennung*, eds. P. Levi, R.-J. Ahlers, F. May, and M. Schanz (Springer, Berlin, 1998), ISBN 3-519-02606-6, pp. 93–199.

13. J. Weickert and C. Schnörr, "PDE-based preprocessing of medical images," *Künstliche Intelligenz* **3**, 5 (2000); Revised version of Technical Report 8/2000, *Comput. Science Series*, University of Mannheim, Mannheim, Germany, February 2000.

14. C. Schnörr, "Unique reconstruction of piecewise smooth images by minimizing strictly convex non-quadratic functions," *J. Math. Imaging Vision* **4**(2), 189 (1994).

15. C. Schnörr, "A study of a convex variational diffusion approach for image segmentation and feature extraction," *J. Math. Imaging Vision* **8**(3), 271 (1998).

16. F. Catte, P.-L. Lions, J. M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion-I," *SIAM J. Numer. Anal.* **29**(1), 182 (1992).

17. F. Catte, P.-L. Lions, J. M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion-II," *SIAM J. Numer. Anal.* **29**(3), 845 (1992).

18. A. Witkin, "Scale-space filtering," in *Int. Joint Conference on Artificial Intelligence* (1983), pp. 1019–1022.

19. P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Patt. Anal. Mach. Intell.* **12**(7), 629 (1990).

20. P. Perona, "Orientation diffusions," *IEEE Trans. Image Proc.* **7**(3), 457 (1998).

21. G. Gerig, O. Kubler, R. Kikinis, and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Trans. Med. Imaging* **11**(2), 221 (1992).

22. L. Alvarez, P.-L. Lions, and J. M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.* **29**(3), 845 (1992).

23. L. Alvarez, F. Fuichard, P.-L. Lions, and J. M. Morel, "Axioms and fundamental equations on image processing," *Arch. Ration. Mech.* **123**(3), 199 (1993).

24. J. Canny, "A computational approach to edge detection," *IEEE Trans. Patt. Anal. Mach. Intell.* **8**(6), 679 (1986).

25. I. Pollak, A. S. Willsky, and H. Krim, "Image segmentation and edge enhancement with stabilized inverse diffusion equations," *IEEE Trans. Image Proc.* **9**(2), 256 (2000).

26. B. B. Kimia and K. Siddiqi, "Geometric heat equation and non-linear diffusion of shapes and images," in *IEEE Comput. Society Conference on Comput. Vision and Pattern Recognition* (1994), pp. 113–120.

27. B. B. Kimia and K. Siddiqi, "Geometric heat equation and nonlinear diffusion of shapes and images," *Comput. Vision Image Understanding* **64**(3), 305 (1996).

28. G. Sapiro, A. Tannenaum, Y. L. You, and M. Kaveh, "Experiments on geometric image enhancement," in *Proc. First IEEE Int. Conference on Image Proc.*, Austin, TX, 1994.

29. G. Sapiro and V. Caselles, "Histogram modification via differential equations," *J. Diff. Eqs.* **135**(2), 238 (1997).

30. G. Sapiro and V. Caselles, "Contrast enhancement via image evolution flows," *Graphical Models and Image Proc.* **59**(6), 407 (1997).

31. Y. L. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Trans. Image Proc.* **5**(11), 1539 (1996).

32. G. Sapiro, "From active contours to anisotropic diffusion: Connections between basic PDEs in image processing," in *Proc. IEEE Int. Conference on Image Proc.*, Vol. 1 (September 1996), pp. 477–480.

33. J. Shah, "A common framework for curve evolution, segmentation and anisotropic diffusion," in *Proc. IEEE Proc. Conference on Comput. Vision and Pattern Recognition* (June 1996), pp. 136–142.

34. V. Caselles, R. Kimmel, and G. Shapiro, "Geodesic active contours," *Int. J. Comput. Vision* **22**(1), 61 (1997).

35. J. Weickert, *Anisotropic Diffusion in Image Proc.* (Teubner-Verlag, Stuttgart, Germany, 1998), ISBN 3-519-02606-6; see also the article: "A review of nonlinear diffusion filtering," in *Scale-Space Theory in Comput. Vision* (Utrecht, The Netherlands, 1997), pp. 3–28.

36. M. Black, G. Sapiro, D. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. Image Proc.* **7**(3), 421 (1998).

37. S. R. Arridge and A. Simmons, "Multi-spectral probabilistic diffusion using Bayesian classification," in *Scale-Space Theory in Comput. Vision, Lecture Notes in Comput. Science*, Vol. 1252, eds. B. ter Haar Romeny, L. Florack, J. Koendernick, and M. Viergever (Springer, Berlin, 1997), pp. 224–235.

38. I. Bajla and I. Hollander, "Nonlinear filtering of magnetic resonance tomograms by geometry-driven diffusion," *Mach. Vision Appl.* **10**(5–6), 243 (1998).

39. P. J. Olver, G. Sapiro, and A. Tannenbaum, "Classification and uniqueness of invariant geometric flows," *Comptes Rendus De L'Acadmie Des Sciences./Serie I Mathmatique, Paris* **319**, Serie I, (1994), pp. 339–344.

40. O. Scherzer and J. Weickert, "Relations between regularization and diffusion filtering," *J. Math. Imaging Vision* **12**(1), 43 (2000).

41. B. ter Haar Romeny, L. Florack, J. Koendernick, and M. Viergever, "Scale-space theory in computer vision," *Lecture Notes in Computer Science*, Vol. 1252 (Springer, Berlin, 1997).

42. M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, "Scale-space theories in computer vision," *Lecture Notes in Computer Science*, Vol. 1682 (Springer, Berlin, 1999).

43. B. ter Haar Romeny, *Geometry-Driven Diffusion in Computer Vision* (Kluwer, Boston, MA, 1994), ISBN 0-7923-3087-0.

44. P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim, "Robust regression methods for computer vision: A review," *Int. J. Comput. Vision* **6**(1), 59 (1991).

45. J. S. Suri, R. M. Haralick, and F. H. Sheehan, "Left ventricle longitudinal axis fitting and LV apex estimation using a robust algorithm and its performance: A parametric apex model," *Proc. Int. Conference in Image Proc.*, Vol. III of III (IEEE, Santa Barbara, CA, 1997), ISBN 0-8186-8183-7/97, pp. 118–121.

46. G. I. Sanchez-Ortiz, D. Rueckert, and P. Burger, "Knowledge-based tensor anisotropic diffusion of cardiac magnetic resonance images," *Med. Image Anal.* **3**(1), 77 (1999).

47. J. S. Suri and J. Gao, Image Smoothing Using PDE, "Scale-space and mathematical morphology," submitted for *Int. Conference in Visualization, Imaging and Image Proc.* (2001).

48. S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations," *J. Comput. Phys.* **79**(1), 12 (1988).

49. J. A. Sethian, "An analysis of flame propagation," Ph.D. Thesis, *Dept. Mathematics* (University of California, Berkeley, CA, 1982).

50. S. Angenent, D. Chopp, and T. Ilmanen, "On the singularities of cones evolving by mean curvature," *Commun. Partial Differential Equations (CPDE)* **20**(11&12), 1937 (1995).

51. D. L. Chopp, "Flow under curvature: Singularity formation, minimal surfaces and geodesics," *Experimental Math.* **2**(4), 235 (1993).

52. D. L. Chopp, "Numerical computation of self-similar solutions for mean curvature flow," *Experimental Math.* **3**(1), 1 (1993).

53. J. A. Sethian, "Numerical algorithms for propagating interfaces: Hamilton-Jacobi equations and conservation laws," *J. Diff. Geo.* **31**(1), 131 (1990).

54. J. A. Sethian, "Curvature flow and entropy conditions applied to grid generation," *J. Comput. Phys.* **115**(1), 440 (1994).

55. W. Mulder, S. J. Osher, and J. A. Sethian, "Computing interface motion in compressible gas dynamics," *J. Comput. Phys.* **100**(1), 209 (1992).

56. M. Sussman, P. Smereka, and S. J. Osher, "A level set method for computing solutions to incompressible two-phase flow," *J. Comput. Phys.* **114**(1), 146 (1994).

57. C. Rhee, L. Talbot, and J. A. Sethian, "Dynamical study of a premixed V-flame," *J. Fluid Mechanics* **300**, 87 (1995).

58. D. Adalsteinsson and J. A. Sethian, "A unified level set approach to etching, deposition and lithography I: Algorithms and two-dimensional simulations," *J. Comput. Phys.* **120**(1), 128 (1995).

59. R. T. Whitaker, "Algorithms for implicit deformable models," *Int. Conference on Comput. Vision (ICCV)* (June 1995), pp. 822–827.

60. Ross T. Whitaker, "A level-set approach to 3D reconstruction from range data," *Int. J. Comput. Vision (IJCV)* **29**(3), 203 (1998).

61. Ross T. Whitaker and David E. Breen, "Level-set models for the deformation of solid objects, proceedings of implicit surfaces," *Eurographics/Siggraph* (June 1998), pp. 19–35.

62. N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Trans. Patt. Anal. Mach. Intell. (PAMI)* **22**, 266 (2000).

63. N. Paragios and R. Deriche, "Coupled geodesic active regions for image segmentation: A level set approach," in *The Sixth European Conference on Computer*

Vision (ECCV), Vol. II (Trinity College, Dublin, Ireland, 26th June–1st July, 2000), pp. 224–240.

64. P. Kornprobst, R. Deriche, and G. Aubert, "Image sequence analysis via partial differential equations," *J. Math. Imaging Vision (JMIV)* **11**(1), 5 (1999).

65. O. Faugeras and R. Keriven, "Variational principles, surface evolution, PDEs level set methods and the stereo problem," *IEEE Trans. Image Proc.* **7**(3), 336 (1998).

66. G. Sapiro, R. Kimmel, D. Shaked, B. B. Kimia, and A. M. Bruckstein, "Implementing continuous-scale morphology via curve evolution," *Patt. Recog.* **26**(9), 1363 (1997).

67. A. Arehart, L. Vincent, and B. B. Kimia, "Mathematical morphology: The Hamilton–Jacobi connection," in *Int. Conf. Comput. Vision (ICCV)* (1993), pp. 215–219.

68. F. Catte, F. Dibos, and G. Koepfler, "A morphological scheme for mean curvature motion and applications to anisotropic diffusion and motion of level sets," *SIAM J. Numer. Anal.* **32**(6), 1895 (1995).

69. N. Sochen, R. Kimmel, and R. Malladi, "A geometrical framework for low level vision," *IEEE Trans. Image Proc.* **7**(3), 310 (1998).

70. R. Malladi, R. Kimmel, D. Adalsteinsson, G. Sapiro, V. Caselles, and J. A. Sethian, "A geometric approach to segmentation and analysis of 3D medical images," *Proc. IEEE/SIAM Workshop on Mathematical Morphology and Biomedical Image Analysis (MMBIA)* (San Francisco, CA, June 1996), pp. 244–252.

71. R. Malladi and J. A. Sethian, "Image processing via level set curvature flow," *Proc. Natl. Acad. Sci. (PNAS), USA* (1995) pp. 7046–7050.

72. R. Malladi and J. A. Sethian, "Image processing: Flows under min/max curvature and mean curvature," *Graphics Models Image Proc. (GMIP)* **58**(2), 127 (1996).

73. R. Malladi, J. A. Sethian, and B. C. Vemuri, "A fast level set based algorithm for topology independent shape modeling," *J. Mathematical Imaging and Vision, Special Issue on Topology and Geometry in Computer Vision*, eds. A. Rosenfeld and Y. Kong, **6**(2&3), 269 (1996).

74. R. Malladi and J. A. Sethian, "A unified approach to noise removal, image-enhancement and shape recovery," *IEEE Trans. Image Proc.* **5**(11), 1554 (1996).

75. S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Conformal curvature flows: From phase transitions to active vision," *Arch. Ration. Mech. Anal.* **134**, 275 (1996).

76. A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum, "Snake model for segmentation of medical imagery," *IEEE Trans. Med. Imaging* **16**(2), 199 (1997).

77. K. Siddiqi, Y. B. Lauriere, A. Tannenbaum, and S. W. Zucker, "Area and length minimizing flows for shape segmentation," *IEEE Trans. Image Proc.* **7**(3), 433 (1998).

78. R. Malladi and J. A. Sethian, "An O(N log N) algorithm for shape modeling, applied mathematics," *Proc. Natl. Acad. Sci. (PNAS), USA* **93**(18), 9389 (1996).

79. J. Gomes and O. Faugeras, "Level sets and distance functions," in *Proc. sixth European Conference on Computer Vision (ECCV)* (2001), pp. 588–602.

80. M. Grayson, "The heat equation shrinks embedded plane curves to round points," *J. Diff. Geo.* **26**, 285 (1987).

81. J. A. Sethian, "Algorithms for tracking interfaces in CFD and material science," *Annual Review of Computational Fluid Mechanics*, (1995).

82. J. A. Sethian and J. D. Strain, "Crystal growth and dentric solidification," *J. Comput. Phys.* **98**(2), 231 (1992).

83. G. Sapiro, "Color Snakes," *Comput. Vision Image Understanding (CVIU)* **68**(2), 247 (1997).

84. J. S. Suri, "Fast WM/GM boundary segmentation from MR images using the relationship between parametric and geometric deformable models," Chapter 8, in

Advanced Algorithmic Approaches to Medical Image Segmentation: *State-of-the-Art Applications in Cardiology, Neurology, Mammography and Pathology*, eds. Suri, Setarehdan and Singh, in press, to be published in January 2001.

85. X. Zeng, L. H. Staib, R. T. Schultz, and J. S. Duncan, "Segmentation and measurement of the cortex from 3D MR images using coupled-surfaces propagation," *IEEE Trans. Med. Imaging* **18**(10), 927 (1999).

86. A. Sarti, C. Ortiz, S. Locket, and R. Malladi, "A geometric model for 3D confocal image analysis," *IEEE Trans. Biomed. Engin.* **47**(12), 1600 (2000).

87. J. A. Sethian, "A review of recent numerical algorithms for hypersurfaces moving with curvature dependent flows," *J. Diff. Geo.* **31**, 131 (1989).

88. J. A. Sethian, "Theory, algorithms and applications of level set methods for propagating interfaces," *Acta Numerica* **5**, 309 (1996).

89. J. A. Sethian, "Level set methods and fast marching methods: Evolving interfaces in computational geometry, fluid mechanics," *Comput. Vision and Material Science* (Cambridge University Press, Cambridge, UK, 1999), 2nd Edition, ISBN 0-521-64204-3.

90. S. Cao and S. Greenhalgh, "Finite-difference solution of the Eikonal equation using an efficient, first-arrival, wavefront tracking scheme," *Geophysics* **59**(4), 632 (1994).

91. S. Chen, B. Merriman, S. Osher, and P. Smereka, "A simple level set method for solving stefan problems," *J. Comput. Phys.* **135**(1), 8 (1997).

92. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vision* **1**(4), 321 (1988).

93. T. Pavlidis and Y. Liow, "Integrating region growing and edge detection," *IEEE Trans. Patt. Anal. Mach. Intell.* **12**(3), 225 (1990).

94. S. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing and Bayes/MDL for multiband image segmentation," *IEEE Trans. Patt. Anal. Mach. Intell.* **18**(9), 884 (1996).

95. J. S. Suri, "Computer vision, image processing and pattern recognition in left ventricle segmentation: Last 50 years," *J. Patt. Anal. Appl.* **3**(3), 209 (2000).

96. L. M. Lorigo, O. Faugeras, W. E. L. Grimson, R. Keriven, R. Kikinis, and Carl-Fredrik Westin, "Co-Dimension 2 geodesic active contours for MRA segmentation," in *Proc. 16th Int. Conference of Information Proc. Medical Imaging*, Visegrad, Hungary, Lecture Notes in Computer Science, Vol. 1613, (June/July 1999), pp. 126-139.

97. L. M. Lorigo, W. Eric L. Grimson, O. Faugeras, R. Keriven, R. Kikinis, A. Nabavi, and Carl-Fredrick Westin, "Two geodesic active contours for the segmentation of tubular structures," in *Proc. Comput. Vision and Pattern Recognition (CVPR)* (June 2000), pp. 444–451.

98. J. S. Suri and R. Bernstein, "2D and 3D display of aneurysms from magnetic resonance angiographic data," *6th Int. Conference in Computer Assisted Radiology* (1992), pp. 666–672.

99. N. Paragios and R. Deriche, "Geodesic active regions: A new paradigm to deal with frame partition problems in computer vision," *J. Visual Communication and Image Representation (JVCIR), Special Issue on Partial Differential Equations in Image Processing, Computer Vision and Computer Graphics* (to appear).

100. H. Tek and B. B. Kimia, "Deformable bubbles in the reaction-diffusion space," in *Proc. 5th Int. Conference in Comput. Vision (ICCV)* (Cambridge, MA, 1995), pp. 156–162.

101. C. Xu and J. L. Prince, "Generalized gradient vector flow external forces for active contours," *Int. J. Signal Processing* **71**(2), 131 (1998).

102. O. Amadieu, E. Debreuve, M. Barlaud, and G. Aubert, "Inward and outward curve evolution using level set method," in *Proc. IEEE Int. Conference in Image Proc.*, Vol. III, (October 1999), pp. 188–192.
103. F. Santosa, "A level set approach for inverse problems involving obstacles, ESAIM: Control," *Optimisation and Calculus of Variations, European Series in Applied and Industrial Mathematics* **1**, 17 (1996).
104. S. J. Besson, M. Barlaud, and G. Aubert, "Detection and tracking of moving objects using a new level set based method," *Int. Conference on Pattern Recognition* (Barcelona, September 2000), pp. 1112–1117.
105. C. Baillard, P. Hellier, and C. Barillot, "Segmentation of 3D brain structures using level sets," *Research Report 1291, IRISA* (Rennes Cedex, France, 2000), p. 16.
106. C. Baillard, C. Barillot, and P. Bouthemy, "Robust adaptive segmentation of 3D medical images with level sets," *Research Report 1369, IRISA* (Rennes Cedex, France, 2000), p. 26.
107. C. Baillard, P. Hellier, and C. Barillot, "Cooperation between level set techniques and dense 3D registration for the segmentation of brain structures," in *Int. Conference on Pattern Recognition* **1**, 991 (2000).
108. W. E. L. Grimson, C. Stauffer, R. L. L. Romano, P. Viola, and O. D. Faugeras, "Forest of sensors: Using adaptive tracking to classify and monitor activities in a site," *Proc. 1998 DARPA Image Understanding Workshop* **1**, 33 (1998).
109. N. Paragios and R. Deriche, "PDE-based level-set approach for detection and tracking of moving objects," *Proc. Int. Conference in Comput. Vision* (1998), pp. 1139–1145.
110. K. Zhang and J. Kittler, "Global motion estimation and robust regression for video coding," *ICIP'98* **3**, 944 (1998).
111. B. Jahne, H. Haussecker, H. Spies, D. Schmundt, and U. Schurr, "Study of dynamical processes with sensor-based spatiotemporal image processing techniques," *European Conference in Computer Vision (ECCV)* (1998), pp. 322–335.
112. J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Comput. Vision and Material Science* (Cambridge University Press, Cambridge, UK, 1999), 2nd Edition, ISBN 0-521-64204-3.
113. J. Gao, "Image sequence segmentation," Ph.D. Thesis, *Dept. Electrical Engin. Comput. Science* (University of Wisconsin, Milwaukee, WI, 1999).
114. J. Zhang and J. Gao, "Image sequence segmentation using curve evolution," *Conference Record of the Thirty-Third Asilomar Conference on Signals, Syst. Comput.* **2**, 1426 (1999).
115. J. Zhang, J. Gao, and W. Liu, "Image sequence segmentation using 3D structure tensor and curve evolution," *IEEE Trans. Circuits and Systems for Video Technology*, 2001 (to appear).
116. S. C. Zhu and A. Yullie, "Region competition: Unifying snakes, region growing and Bayes/MDL for multiband image segmentation," *IEEE Trans. Patt. Anal. Mach. Intell.* **18**(9), 884 (1996).
117. Abdol-Reza Mansouri and J. Konrad, "Multiple motion segmentation with level sets," *IEEE Trans. Imging Proc.*, 2001 (to appear).
118. S. Banks, *Signal processing, image processing and pattern recognition* (Prentice-Hall, Englewood Cliffs, NJ, 1990), ISBN 0-13-812579-1.
119. M. I. G. Bloor, M. J. Wilson, and H. Hagen, "The smoothing properties of variations schemes for surface design," *Comput. Aided Geo. Design* **112**(4), 381 (1995).
120. J. M. Brown, M. I. G. Bloor, M. S. Bloor, H. N. Nowacki, and M. J. Wilson, "Fairness of B-spline surface approximations to PDE surfaces using the finite-element method,"

in *Computer-Aided Surface Geometry and Design*: *The Mathematics of Surfaces IV*, ed. Bowyer, 1994.

121. J. Weickert, J. Heers, C. Schnorr, K. J. Zuiderveld, O. Scherzer, and H. S. Stiehl, "Fast parallel algorithms for a broad class of nonlinear variational diffusion approaches," *Real-Time Imaging* **7**(1), 31 (2001).

122. H. K. Zhao, T. Chan, B. Merriman, and S. Osher, "A variational level set approach to multiphase motion," *J. Comput. Phys.* **127**(1), 179 (1996).

123. L. C. Evans and J. Spruck, "Motion of level sets by mean curvature: Part-I," *J. Diff. Geo.* **33**, 635 (1991).

124. B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker, "Shapes, shocks and deformations, I: The components of shape and the reaction-diffusion space," *Int. J. Comput. Vision (IJCV)* **15**(3), 189 (1995).

125. K. Siddiqi, K. J. Tresness, and B. B. Kimia, "Parts of visual form: Ecological and psychophysical aspects," *Perception* **25**(4), 399 (1996).

126. P. Stoll, H. Tek, and B. B. Kimia, "Shocks from images: Propagation of orientation elements," in *Proc. Comput. Vision and Pattern Recognition (CVPR)* (June 15–16, 1997), pp. 839–845.

127. A. Sarti, R. Malladi, and J. A. Sethian, "Subjective surfaces: A method for completing missing boundaries," *Proc. Natl. Acad. Sci. USA* **12**(97), 6258 (2000).

128. A. Sarti, K. Mikula, and F. Sgallari, "Nonlinear multiscale analysis of 3D echocardiographic sequences," *IEEE Trans. Med. Imaging* **18**(6), 453 (1999).

129. A. Sarti and R. Malladi, "A geometric level set model for ultrasound analysis," LBNL-44442, University of California, Berkeley, CA, 1999.

130. A. Sarti, C. Ortiz, S. Locket, and R. Malladi, "A unified geometric model for 3D confocal image analysis in cytology," Report Number: LBNL-41740, University of California, Berkeley, CA, April 1998.

131. B. Jahne, *Digital Image Proc.*: *Concepts*, *Algorithms and Scientific Applications*, 4th Edition, ISBN 0-387-569-413 (Springer Verlag, 1997).

132. L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica* **D60**, 259 (1992).

133. R. T. Whitaker and S. M. Pizer, "A multi-scale approach to non-uniform diffusion," *Comput. Vision, Graphics and Image Proc.*: *Image Understanding* **51**(1), 99 (1993).

134. E. J. Pauwels, P. Fiddelaers, and L. J. Van Gool, "Shape-extraction for curves using geometry-driven diffusion and functional optimization," in *Proc. ICCV* (June 1995), pp. 396–401.

135. F. Dibos and G. Koepfler, "Image denoising through a level set approach," *Proc. Int. Conf. Image Proc.* **3**, 264 (1998).

136. R. Courant, K. O. Friedrichs, and H. Lewy, "On the partial difference equations of mathematical physics," *IBM J.* **11**, 215 (1967).

137. J. S. Suri, "Complex 2D and 3D segmentation issues and variational methods in medical imagery," under submission in *Int. J.* (2002).

Jasjit S. Suri received his B.S. in Computer Engineering with distinction from MACT, Bhopal, his M.S. in Computer Sciences from University of Illinois, and his Ph.D. in Electrical Engineering from University of Washington, Seattle. He has been working in the field of Computer Engineering/Imaging Sciences for more that 18 years. He has published more than 80 papers in imaging. He is a life time member of research engineering societies: Tau-Beta-Pi and Eta-Kappa-Nu, Sigma-Xi, NY Academy of Sciences, EMBS, SPIE, ACM and is also a senior member of IEEE. He is on the editorial board of several International Journals such as: Real Time Imaging, Pattern Analysis and Applications, Engineering in Medicine and Biology, Radiology, Journal of Computer Assisted Tomography, IEEE Transactions of Information Technology in Biomedicine and IASTED. He has chaired image processing sessions at several international conferences and has given more than 30 international presentations. Dr Suri has written a book on Medical Imaging and holds several US Patents. Dr Suri has been listed in Who's Who five times, is a recipient of President's Gold medal in 1980 and has received 50 scholarly and extra-curricular awards during his career. Dr Suri's major interest are: Computer Vision, Graphics and Image Processing (CVGIP), Object Oriented Programming and Image Guided Surgery. Dr Suri has been with Philips Medical Systems, Inc. (previously known as Marconi Medical Systems Inc.), Cleveland, since Dec. 1998.
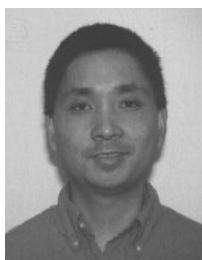
Dee H. Wu was born in Middletown, CT in 1965, and received his doctorate with specialization in Biomedical Engineering from Case Western Reserve University in Cleveland, Ohio. He also has an M.Sc. in Systems Engineering and a B.Sc. in Mathematics. Dr Wu also received the Alden fellowship, and is a member of Tau Beta Pi society, International Society for Magnetic Resonance, and the Society for Functional Mapping of the brain. Dr Wu currently serves as a peer reviewer for the Journal of Magnetic Resonance Imaging. Dr Wu's work has covered a wide variety of technical areas including magnetic resonance pulse programming, image reconstruction and analysis. He has been working over ten years in the area of functional MRI (fMRI), Diffusion/Perfusion, interventional, and cardiac imaging. He has been an author in 10 peer-reviewed articles and filed over ten patents. He has been a staff scientist at Philips Medical Systems, Inc. (previously known as Marconi Medical Systems, Inc.) for the past four years.

Laura Reden was born in Yonkers, New York. After graduating with a B.S. degree in Chemistry from the College of New Rochelle, she joined FONAR Corporation, pioneer of MRI technology. She worked for nine years in research and development and as a Field Applications Specialist, operating and training users on some of the first MRI systems ever deployed, in Europe, India and the U.S. In this work, Laura interacted with clinicians from multiple disciplines. In 1990, she joined Marconi Medical Systems, Inc. (then Picker International), engaging in a variety of user contact functions. Today, Laura is a Clinical Testing Specialist, involved with testing any new aspect of MRI development, both in-house and in the field. This activity is concerned with user experience of the total system: hardware, software, new techniques and accessories. Laura attends a wide variety of seminars on MRI, including fMRI, breast imaging and advanced neuro techniques for such applications as stroke. Laura has co-authored journal articles and is a U.S. co-patentee.
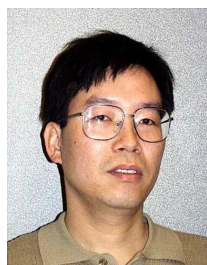
Jianbo Gao received his B.S. and M.S. in Electrical Engineering from Harbin Shipbuilding Engineering Institute, Harbin, China, in 1988 and 1991, respectively. He received his Ph.D. in Electrical Engineering from the University of Wisconsin, Milwaukee in 1999. Currently, he is an algorithm and software engineer in KLA-Tencor, and works on defect classification, machine vision inspection, etc. His research interests include image filtering, classification and segmentation, motion segmentation, etc.

Sameer Singh was born in New Delhi, India and graduated from Birla Institute of Technology, India with a Bachelor of Engineering degree with distinction in Computer Engineering. He received his Master of Science degree in Information Technology for Manufacturing from the University of Warwick, U.K. and a Ph.D. in speech and language analysis from the University of the West of England, U.K. His main research interests are in image processing, medical imaging, neural networks and pattern recognition. He is the Director of the Pattern Analysis and Neural Networks group at Exeter University. He serves as the Editor-in-Chief of the Pattern Analysis and Applications journal by Springer, Editor-in-Chief of the Springer book series on "Advances in Pattern Recognition," Chairman of the British Computer Society Specialist group on Pattern Analysis and Robotics, Editorial Board member of Neural Computing and Applications journal, and Editorial Board member of the Perspectives in Neural Computing book series by Springer. He is a Fellow of the Royal Statistical Society, and a Member of BMVA-IAPR, IEE and IEEE.
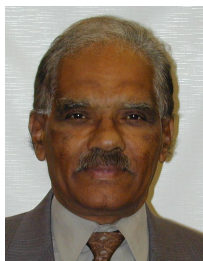
Swamy Laxminarayan is currently the Chief Information Officer at the National Louis University (NLU) in Chicago. Prior to coming to NLU, he was an Adjunct Professor of Biomedical Engineering at NJIT, Newark, New Jersey and a Clinical Associate Professor of Medical Informatics and Director and Chair of VocalTech University. Until recently, he was the Director of Health Care Information Services as well as Director of Bay Networks, authorized educational center at NextJen internet, Princeton, New Jersey. He also serves as a Visiting Professor of Biomedical Information Technology at the University of Brno, Slovak Republic, and an Honorary Professor at Tsinghua University, China. He is an internationally recognized scientist, engineer, and educator with over 200 technical publications in areas of wide ranging as biomedical information technology, computation biology, signal and image processing, biotechnology and physiological system modeling. He has been involved in Internet and information technology application for well over a decade with significant contributions to the applications to the discipline in medicine and health care. Professor Laxminarayan has won numerous international awards and has lectured widely as an invited speaker in over 35 countries. He has been closely associated with the IEEE Engineering and Medicine and Biology Society in various admistrative and executive committee roles, including his previous appointments as Vice President of the society and currently as Editor-in-Chief of IEEE Transactions on Information Technology and Biomedicine. Among the many awards and honors he has received, he is one of the 1995 recipients of the Purkynje Award, one of Europe's highest forms of recognition, given for pioneering contributions in cardiac and neuro physiological modeling work and his international bioengineering leadership. In 1994, he was inducted into the College of Fellows of the American Institute of Medical and Biological Engineering (AIMBE) for "outstanding contributions to advance computing and high performance communication applications in biomedical research and education." He recently became the recipient of the IEEE 3rd Millennium Medal.