3D Mumford-Shah Based Active Mesh

Alexandre Dufour[1,2], Nicole Vincent[2], and Auguste Genovesio[1]

[1] Image Mining group - Institut Pasteur Korea
alexandre.dufour@pasteur.or.kr
[2] Systèmes Intelligents de Perception
Centre de Recherche en Informatique de Paris 5 - Université René Descartes

Abstract. Deformable mesh methods have become an alternative of choice to classical deformable models for 3D image understanding. They allow to render the evolving surface directly during the segmentation process in a fast and efficient way, avoiding both the additional time-cost and approximation errors induced by 3D reconstruction algorithms after segmentation. Current methods utilize edge-based forces to attract the mesh surface toward the image entities. These forces are inadequate in 3D fluorescence microscopy, where edges are not well defined by gradient. In this paper, we propose a fully automated deformable 3D mesh model that deforms using the reduced Mumford-Shah functional to segment and track objects with fuzzy boundaries. Simultaneous rendering of the mesh evolution allows faster tweaking of the model parameters and offers biologists a more precise insight on the scene and hence better understanding of biological phenomena. We present evaluations on both synthetic and real 3D microscopy data.

1 Introduction

3D segmentation and tracking by deformable models is a topic of active research in many domains including medical imaging [1][2][3] and biological imaging [4][5]. Until the early 90's, due to poor machine performances, 3D rendering of the scene used to be done in a post-processing step using surface extraction algorithms, inducing surface approximation errors and an additional time cost for the reconstruction. Advances in 3D computer graphic cards and rendering techniques have thus lead to the development of new techniques where segmentation and 3D rendering are done simultaneously, namely deformable mesh models. The idea, as originally proposed by Miller in [6], is to discretize an initial surface into a closed mesh, i.e. a closed set of polygons, and deform this surface by attracting its vertices toward the features of interest. Most deformable mesh models use edge information to drive the surface. This performs poorly on 3D volumes (depth-stacks of 2D images) obtained in confocal fluorescence microscopy due to several factors: first, a significant blur is introduced by the microscope's Point-Spread Function (PSF), especially along the depth axis. Moreover, like in 2D imaging, the stacks are corrupted by both multiplicative (Poisson) and additive (Gaussian) noise. This has many sources of explanation, including low exposure times, autofluorescence of the medium and shot noise induced by the camera's

J.F. Martínez-Trinidad et al. (Eds.): CIARP 2006, LNCS 4225, pp. 208–217, 2006.

CCD. Finally, in a tracking context, absorption of fluorophore by the entities induces an intensity decrease through time (phenomenon called *photo-bleaching*). To address this problem, a mesh model that uses region-based information instead of edge-based information is proposed. The method is able to detect and track an object with fuzzy boundaries and provides a fast and precise 3D rendering of the evolving surface thanks to the mesh formalism.

In section 2, we review existing work on deformable surfaces and mesh models. Then we detail in section 3 the characteristics of the proposed method. Evaluation on synthetic data and tests on real data are presented in section 4. We conclude the paper in section 5 and discuss possible extensions of the method.

2 Background and Related Work

In this section, an overview of the principles of 3D deformable surfaces is presented, followed by a brief review of existing 3D deformable mesh models.

2.1 3D Deformable Surfaces

To evolve from its initial state to the object boundary, a surface S is driven under the influence of two kinds of forces: data attachment (external) forces, which pull the surface toward specific features of the image, and regularization (internal) forces, which maintain smoothness and regularity of the surface. This evolution can be done by three different ways: a variational, a dynamic and a probabilistic approach.

- The **variational approach** is most popular. The 2D formulation is due to Kass *et. al* [7], and its 3D extension to Cohen and Cohen [8]. The evolving shape is defined as a parametrized surface $S(p) = (x(p), y(p), z(p))^\top$, where $p \in [0, 1]$ is the parametric domain. The surface is then linked to an energy functional, defined to be minimal when the surface fits the desired object correctly. This functional reads

$$F(S) = F_{internal}(S) + F_{external}(S), \tag{1}$$

where

$$F_{internal}(S) = \frac{1}{2}\int_0^1 \alpha \left|\frac{\partial S}{\partial p}\right|^2 + \beta \left|\frac{\partial^2 S}{\partial p^2}\right|^2 dp \tag{2}$$

and

$$F_{external}(S) = -\int_0^1 P(S(p))dp. \tag{3}$$

$F_{internal}$ is the deformation energy, where α is a "tension" parameter that tends to minimize the surface perimeter, and β a "rigidity" parameter that smooths the surface curvature.

The external energy (eq. 3) attracts the surface toward the image features through the scalar potential function P.

Eq. 1 is then minimized using a time-dependent gradient-descent method such as the well-known Euler-Lagrange equations, which give

$$\frac{\partial S}{\partial t} = \alpha \frac{\partial^2 S}{\partial p^2} - \beta \frac{\partial^4 S}{\partial p^4} + \nabla P(S(p,t)). \tag{4}$$

The final surface is obtained after convergence, i.e. when $\partial S / \partial t = 0$.

- In a **dynamic approach**, the shape is defined as a time-varying physical surface $S(p,t) = (x(p,t), y(p,t), z(p,t))^\top$ that evolves until it reaches a steady-state, following the Lagrange equation of motion, which gives

$$\gamma \frac{\partial S}{\partial t} + \mu \frac{\partial^2 S}{\partial t^2} + \alpha \frac{\partial^2 S}{\partial p^2} - \beta \frac{\partial^4 S}{\partial p^4} = \nabla P(S(p,t)) \tag{5}$$

where γ and μ are the surface mass and damping densities, respectively. The final surface is obtained when the system converges to equilibrium, i.e. $\partial S / \partial t = \partial^2 S / \partial t^2 = 0$. This approach allows to quantify both shape and motion evolution through time but complicates the method with two more parameters (γ and μ) to be set by the user.

- Finally, the model can be viewed as a fitting process in a **probabilistic framework** [9][10]. The shape is defined by a set $\mathbf{x}$ of parameters and their probability $p(\mathbf{x})$, being higher for low energy-shapes (this is usually done using a Gibbs distribution). Then, knowing the probability of the image $p(u_0)$ and that of the image knowing the initial shape $p(u_0/\mathbf{x})$, the probability of guessing $\mathbf{x}$ from u_0 is obtained thanks to the Bayes theorem: $p(\mathbf{x}/u_0) = p(u_0/\mathbf{x})p(\mathbf{x})/p(u_0)$. Finally, this probability is maximized to find an estimation of $\hat{\mathbf{x}}$ called the "maximum a posteriori". Probabilistic approaches have the advantage of giving an uncertainty measure of the fitting, but they are computationally much more expensive.

2.2 Deformable Meshes in Computer Vision

3D mesh models find applications in numerous domains including object reconstruction [11], realistic surgery simulation [12] and medical volume segmentation [2][13]. The continuous surface is discretized into a 3D mesh defined as a list of vertices $\mathbf{v}_i = (x_i, y_i, z_i)^\top$ organized in a closed set of oriented polygons representing the boundary of a volumetric object. The surface is then driven by the movement of the mesh vertices. In addition, two reference distances D_{min} and D_{max} are computed from the initial mesh in order to keep a regular mesh sampling during the evolution. The distance between two connected vertices $\mathbf{v}_i$ and $\mathbf{v}_j$ should always satisfy

$$D_{min} \leq dist(\mathbf{v}_i, \mathbf{v}_j) \leq D_{max}.$$

If $dist(\mathbf{v}_i, \mathbf{v}_j) < D_{min}$ or $dist(\mathbf{v}_i, \mathbf{v}_j) > D_{max}$, a local adaptation of the mesh must be applied to respect the above constraint (by splitting or merging problematic faces or edges).

In most models, the external potential P is chosen as an edge map of the image (eventually smoothed to be more robust to noise). This turns out to be efficient for many applications, but fails to detect fuzzy edges in 3D fluorescence microscopy images, where the surface simply shrinks (or inflates, depending on the parameters) and misses the object boundaries. This problem is partially solved in [13], where an additional external term is involved to attract the surface toward a particular iso-value (gray level) of the image. Nevertheless, the detected boundaries are still erroneous, since the object has a much lower intensity at the top and bottom of the volume (along the depth axis) than in the middle plane.

An efficient solution to detect objects with ill-defined boundaries is the Mumford-Shah functional [14], designed for segmentation and denoising in a variational context. The general form of this functional reads

$$F^{MS}(u,S) = \int_{\Omega} |u_0 - u|^2 d\omega + \int_{\Omega/S} |\nabla u|^2 d\omega + \mathcal{H}^{N-1}(S) \tag{6}$$

where Ω is the image domain, u_0 the original image, u the reconstructed image and $\mathcal{H}^{N-1}$ the (N-1) dimensional Hausdorff measure. In case the reconstructed image u should be formed only of regions $R_i \subset \Omega, i \in \{1, n\}$ of constant intensity c_i, then u can be considered piecewise-constant, i.e. $\int_{\Omega \setminus S} |\nabla u|^2 d\omega = 0$, which leads to the reduced Mumford-Shah functional

$$F^{RMS}(c_1, ..., c_n, S) = \sum_{i=1}^{n} \left[\int_{R_i} |u_0 - c_i|^2 d\omega \right] + \mathcal{H}^{N-1}(S). \tag{7}$$

These two functionals have been extensively utilized by region-based deformable models, in 2D (implicit and explicit) methods [15][16],[17][18] and in the 3D implicit case in [5]. To our knowledge, it has not yet been employed in the context of mesh models. We therefore introduce in the next section the new deformable mesh model and its underlying energy term based on the reduced Mumford-Shah functional.

3 Contributed Model

This section describes the proposed method. First, the geometrical aspects of the mesh are presented, then the evolution equation driving the mesh toward the object boundary is described. Finally, we describe the automatic initialization and tracking possibilities of the method.

3.1 Geometrical Mesh Properties

The *simplex mesh* formalism introduced by Delingette in [19] allows the mesh to be formed of different types of geometric primitives (triangles, squares and other polygons). Local mesh adaptation must therefore be done differently for each kind of primitive, which requires to store additional information in the mesh structure. In our approach, we prefer handling a uniform triangle-shaped mesh, which gives satisfactory results in our context. Moreover, a unique scheme is necessary to perform local mesh adaptations, and the mesh structure stores only the vertices and their connectivity.

As presented in section 2.2, the distance between two vertices should be regular over the whole mesh. Therefore we use the initial distance d_0 between two mesh vertices to define a freedom range $[D_{min}, D_{max}]$ such that

$$\begin{cases} D_{min} = d_0/2, \\ D_{max} = 2d_0. \end{cases} \tag{8}$$

The more this interval is chosen narrow, the more the mesh will be regular, and the more it is chosen large, the more freedom is given to the vertices. If the distance between two vertices $\mathbf{v}_i$ and $\mathbf{v}_j$ gets lower than D_{min}, we apply a local mesh adaptation by deleting these two vertices and creating a new one in their center that will share all the neighbors of $\mathbf{v}_i$ and $\mathbf{v}_j$. Inversely, if this distance gets higher than D_{max}, then we isolate the two vertices $\mathbf{v}_k$ and $\mathbf{v}_l$ forming a triangle with $\mathbf{v}_i$ and $\mathbf{v}_j$, separate $\mathbf{v}_i$ and $\mathbf{v}_j$ so that they are no longer neighbors and create a new vertex in their center that is linked to $\mathbf{v}_i$, $\mathbf{v}_j$,$\mathbf{v}_k$ and $\mathbf{v}_l$.

Lachaud proposed in [13] a multi-resolution approach to speed up convergence. The initial mesh is chosen with a low resolution (i.e. the initial distance between two vertices is very high compared to the image resolution), and evolved until convergence. Then the mesh is globally subdivided to obtain a higher-resolution mesh, and evolved again to fit the object boundary more precisely. The evolution-subdivision process is repeated until a suitable mesh resolution is reached (note that subpixel accuracy can be obtained if the distance between two vertices is lower than the image resolution).

The global subdivision strategy should be well chosen, in order to preserve the local mesh curvature as much as possible. We have thus chosen to apply a straightforward subdivision scheme: all faces are splitted into 4 smaller faces by creating a new vertex in the middle of each edge. This turns out to be the first step of the well-known *Loop* subdivision scheme, proposed for mesh refinement and smoothing in computer graphics in 1987 [20]. This scheme outperforms that of Lachaud in [13] which tends to sharpen the mesh corners, increasing its local curvature.

3.2 Evolution Equation

The Mumford-Shah functional is designed to fit into a variational framework, we therefore choose to evolve the surface by an energy-minimization scheme similar to that defined in section 2.1. Our images are typically formed of two phases (the object and the background), thus we may restrict eq. 7 to the two-phase case, which gives

$$F^{RMS}(c_{in}, c_{out}, S) = \int_{in(S)} |u_0 - c_{in}|^2 d\omega + \int_{out(S)} |u_0 - c_{out}|^2 d\omega + \mathcal{H}^{N-1}(S) \tag{9}$$

where c_{in} and c_{out} are variables being respectively the mean intensities of the original image u_0 inside and outside S. We finally construct our energy functional using the same internal energy defined in eq. 2 and the data attachment term of the above equation, and we obtain

$$\begin{aligned} F(c_{in}, c_{out}, S) = & \frac{1}{2}\int_0^1 \alpha \left|\frac{\partial S}{\partial p}\right|^2 + \beta \left|\frac{\partial^2 S}{\partial p^2}\right|^2 dp \\ & + \lambda_{in} \int_{in(S)} |u_0 - c_{in}|^2 d\omega \\ & + \lambda_{out} \int_{out(S)} |u_0 - c_{out}|^2 d\omega \end{aligned} \tag{10}$$

where λ_{in} and λ_{out} are strictly positive parameters. The Euler-Lagrange equations linked to the minimization of this new energy functional thus give

$$\frac{\partial S}{\partial t} = \alpha \frac{\partial^2 S}{\partial p^2} - \beta \frac{\partial^4 S}{\partial p^4} - \left[\lambda_{in}(u_0 - c_{in})^2 - \lambda_{out}(u_0 - c_{out})^2\right] \left|\frac{\partial S}{\partial p}\right| \overrightarrow{n} \tag{11}$$

where $\overrightarrow{n}$ is the surface normal pointing outward. During the evolution, the variables c_{in} and c_{out} have to be recomputed at every iteration.

3.3 Initialization

Deformable models usually rely on user intervention to initialize the model. We prefer employing an automated framework to determine the location and approximate size of the object. First, an automatic thresholding technique is applied (we used the well-known K-means algorithm). The biggest connected component is then extracted from the thresholded image, its mass center $\mathbf{v}_c$ is computed as the average of the objects points coordinates and the mean radius r is defined as the average distance between the mass center and all surface points. Finally, a regular polyhedron of radius r is created and placed in the image domain at the position $\mathbf{v}_c$. Actually, any kind of polyhedron can be used as an initial shape. In our case, a regular 20-face polyhedron was used, also known as the *regular icosahedron.*

3.4 Tracking

As for most deformable models, tracking is done in a straightforward way. Once the object has been segmented on the first frame, we reuse the obtained mesh to initialize the model for the next frame. The new initial mesh being very close from the new object's position, a few number of iterations will be sufficient to track the object movement. This however implies that a significant overlap exist between the object's position at two successive time-points. This overlap should be such that the difference $c_{in} - c_{out}$ is of same sign for both frames. Otherwise, if $c_{in} - c_{out}$ changes sign, the object is interpreted as the background and vice-versa, and the model starts to segment the dual of the object.

There are two possible walk-arounds to such behavior. The first simple way is to reduce the time-lapse between two successive frames during the image acquisition. The second solution is to fix c_{in} and c_{out} after segmentation of the first frame, which are good estimates of the object's and background's mean intensities as long as the global illumination does not vary through time. If photo-bleaching occurs, these estimates must be recalculated regularly, i.e. c_{in} and c_{out} are updated once after convergence on each new frame.

4 Experiments and Results

This section presents several experimentations and evaluations of our method. We shall first discuss results on generated noisy data, and then show results obtained on real microscopy data sets.

4.1 Synthetic Data

We simulated 50 stacks of cell-shaped objects moving in a biological medium, using a method close to that described in [5]. First, binary volumes of size $200 \times 200 \times 200$ voxels are generated, in which a sphere of random radius an center is created (representing the cell body). Then several smaller spheres are generated on the boundary of the bigger sphere to simulate cell protuberances. Then, dark structures are generated randomly within the cell body to simulate inhomogeneous fluorescent marking. Finally, the volumes are convolved by the microscope's PSF and disturbed by multiplicative (Poisson) and additive (Gaussian) noise representing imperfections of the imaging system. To take into account the low depth-resolution of the 3D microscope, we additionally downscale the volumes depth resolution to $200 \times 200 \times 40$. However, due to major improvements of new microscopy systems, space resolution is now almost isotropic. The left and middle columns of fig. 1 show two slices of an original (left) and noisy (right) stack.

The main calculations were done on a 1.7 GHz mobile cpu, while the rendering was simultaneously performed on a 128 Mb graphic card. The processing time of each stack was approximately 60 seconds including the update of c_{in} and c_{out} after every iteration (an example of segmentation result is shown on the right column of fig. 1). To illustrate the rendering possibilities of the method, we show in fig. 2 three different steps of the mesh evolution at its lower (initial) resolution: the left frame is a snapshot taken before the evolution, where one can see the icosahedron used to initialize the model; the middle frame shows in intermediate stage of the evolution; the right frame shows the converged mesh before it is refined and evolved again to fit the object more precisely.

We evaluated the segmentation by measuring the average distance between the segmentation residue and the original object, as proposed by Ciofolo in [21]. This measure is defined by

$$d_{err}(R, O) = \sum_{v_R \in R} min_{v_O \in O} \frac{dist(v_R, v_O)}{card(R)}$$

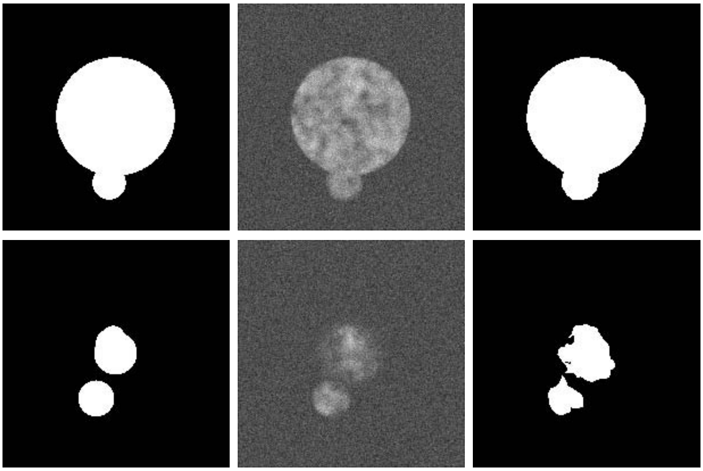

Fig. 1. Two different slices of a synthetic 200 × 200 × 40 stack. left: original image, middle: noisy image, right: result. The top row shows a slice in the middle of the cell, while the bottom row shows a higher slice of the cell. Note the effect of the microscope's PSF on the object boundary.

where v_O and v_R are image voxels forming respectively the original object O the and the segmentation residue R (i.e. the difference between the detected object and O). The average distance d_{err} over the 50 processed stacks is 2 voxels. This measure tends to show that the object boundary has been quite well detected despite the noise, since the errors are mostly located in the vicinity of the object.

4.2 Trial Microscopy Data Sets

Here we present experimental results on a 3D time sequence of a HEK cell changing shape through time while emitting protuberances at its surface. The cell was stained using the Yellow Fluorescent Protein (YFP), and the 3D stacks were acquired on a Zeiss LSM 5 microscope. Fig. 3 shows snapshots of the sequence as well as segmentation and tracking results. The two left-most columns show top and side views of the original stack (left) and segmentation result (right). Note that the model has detected correctly the upper and lower fuzzy boundaries of the cell. The two right-most columns show the same views for the next time-point of the sequence (left), where protuberances appear on the cell surface, and which are correctly detected by the mesh (right).

5 Conclusion and Future Works

A new deformable mesh model based on the reduced Mumford-Shah functional has been proposed, joining the theories of explicit active mesh, region-based information and 3D visualization. The method is able to detect an object with

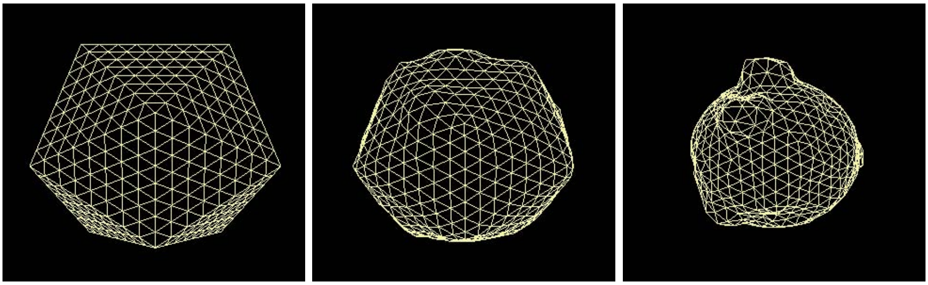

Fig. 2. Rendering of the low resolution mesh at various stages of the evolution: left: before evolution; middle: during evolution; right: after convergence (before refinement)

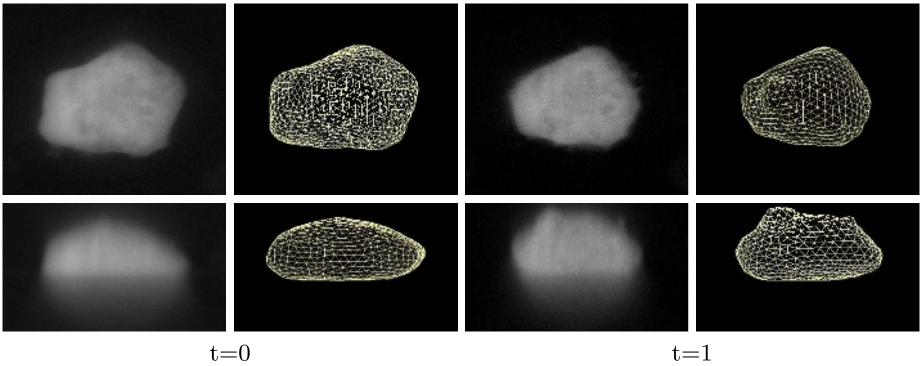

Fig. 3. Tracking results on a HEK cell emitting a protuberance at its top (volume size: 512x512x181): top row: XY plane view from the top of the cell; bottom row: XZ plane view from the side of the cell

fuzzy boundaries and follow its movement and shape changes through time while offering a simultaneous rendering of the evolving scene. Experimental results show that the method is suitable for 3D confocal image analysis, and will be employed in a biological context to study pathogenic diseases involving cellular structures changes through time.

The next step of our work is to detect and track multiple objects simultaneously. To achieve this, multiple meshes will evolve simultaneously within the image domain, and a collision detection strategy will be employed to distinguish correctly two or more objects that might touch each other in time.

References

1. Chakraborty, A., Staib, L., Duncan, J.: Deformable Boundary Finding in Medical Images by Intergrating Gradient and Region Information. IEEE Transactions on Medical Imaging **15**(6) (1996) 859–870

2. Xu, C., Pham, D., Rettman, M., Yu, D., Prince, J.: Reconstruction of the Human Cerebral Cortex from Magnetic Resonance Images. IEEE Transactions on Medical Imaging **18**(6) (1999) 467–480
3. Paragios, N.: A Level Set Approach for Shape-Driven Segmentation and Tracking of the Left Ventricle. IEEE Transactions on Medical Imaging **22** (2003) 773–776
4. Sarti, A., de Solorzano, C.O., Lockett, S., Malladi, R.: A Geometric Model for 3D Confocal Image Analysis. IEEE Transactions on Biomedical Engineering **47**(12) (2000) 1600–1609
5. Dufour, A., Shinin, V., Tajbaksh, S., Guillen, N., Olivo-Marin, J., Zimmer, C.: Segmenting and tracking fluorescent cells in dynamic 3d microscopy with coupled active surfaces. IEEE Transactions on Image Processing **14**(9) (2005) 1396–1410
6. Miller, J.: Geometrically deformed models for the extraction of closed shapes from volume data. Master's thesis, Rensselaer Polytechnic Institute, New York (1990)
7. Kass, M., Witkin, A., Terzopoulos, D.: Snakes : Active contour models. International Journal of Computer Vision **1** (1988) 321–331
8. Cohen, L., Cohen, I.: Finite-element methods for active contour models and balloons for 2-D and 3-D images. IEEE Transactions on Pattern Analysis and Machine Intelligence **15**(11) (1993) 1131–1147
9. Szeliski, R., Terzopoulos, D.: Physically based and probabilistic models for computer vision. In: Geometric Methods in Computer Vision. Volume 1570. (1991) 140–152
10. Staib, L., Duncan, J.: Boundary finding with parametrically deformable models. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(11) (1992) 1061–1075
11. Delingette, H.: General object reconstruction based on simplex meshes. International Journal on Computer Vision **32** (1999) 111–146
12. Cotin, S., Delingette, H., Ayache, N.: Real-time elastic deformations of soft tissues for surgery simulation. IEEE Transactions on Visualization and Computer Graphics **5** (1999) 62–73
13. Lachaud, J., Montanvert, A.: Deformable meshes with automated topology changes for coarse-to-fine three-dimensional surface extraction. Medical Image Analysis **3**(2) (1999) 187–207
14. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Com. Pure App. Math. **42** (1989) 577–684
15. Chan, T., Vese, L.: Active contours without edges. IEEE Transactions on Image Processing **10**(2) (2001) 266–277
16. Tsai, A., Yezzi, A., Willsky, A.: Curve Evolution Implementation of the Mumford-Shah Functional for Image Segmentation, Denoising, Interpolation, and Magnification. IEEE Transactions on Image Processing **10**(8) (2001) 1169–1186
17. Cremers, D., Tischhauser, F., Weickert, J., Schnorr, C.: Diffusion snakes: introducing statistical shape knowledge into the Mumford-Shah functional. International Journal on Computer Vision **50** (2002) 295–313
18. Zimmer, C., Olivo-Marin, J.C.: Coupled Parametric Active Contours. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(11) (2005) 1838–1842
19. Delingette, H.: Modelisation, Deformation et Reconnaissance d'Objets Tridimensionnels a l'Aide de Maillages Simplexes. PhD thesis, Ecole Centrale de Paris (1994)
20. Loop, C.: Smooth Subdivion Surfaces based on Triangles. PhD thesis, University of Utah (1987)
21. Ciofolo, C.: Segmentation des formes guidee par des modeles en neuro-imagerie. PhD thesis, Universite Rennes I (2005)