GLOBAL DEFORMABLE SURFACE OPTIMIZATION USING ADAPTIVE CONSTRAINTS AND PENALTIES

JUSSI TOHKA[1,2]

[1]Institute of Signal Processing, Tampere University of Technology, P.O. Box 553, FIN-33101, Finland,
[2]Laboratory of Neuro Imaging, Department of Neurology, David Geffen School of Medicine, University of California Los Angeles, 90095 CA, USA
e-mail: jussi.tohka@tut.fi

ABSTRACT

Deformable models are able to solve surface extraction problems challenged by image noise because image-independent constraints are used to regularize the shape of the extracted surface. However, this ability of deformable models is shadowed by their application specificity, initialization sensitivity and the difficulty of the selection of proper values for user definable parameters. To overcome these problems restricting the automation of surface extraction, we present a new algorithm, named AdaCoP, for the global minimization of the energy of deformable surfaces. It iteratively performs constrained local minimizations of the energy. It avoids the detection of the same local minimum multiple times by constraining the local optimizations in an adaptive manner. AdaCoP escapes from local minima by imposing an adaptive penalty energy to it. These constraints and penalties prevent the convergence to the local minima already found. The performance of the AdaCoP algorithm is relatively independent on the nature of the underlying image as well as the shape of the surface to be extracted. The performance of the algorithm is evaluated by extracting surfaces from synthetic images. Moreover, the good properties of the algorithm are demonstrated by considering applications within the automated analysis of positron emission tomography images. Although AdaCoP cannot be proven to converge to the global minimum, it is insensitive to its initialization and it therefore provides a way to automate surface extraction problems within medical image analysis.

Keywords: energy minimization, medical image analysis, positron emission tomography, segmentation, surface extraction .

INTRODUCTION

The quantitative analysis of volumetric medical images is essential for biological and medical research. The analysis and interpretation of these three-dimensional (3-D) images usually requires surface extraction and/or image segmentation before quantitative values of interest can be computed. Manual and semiautomatic methods for image segmentation may suffer from inter and intra observer variability of segmentation results. Moreover, the large size of images makes it burdensome to process them manually or even in a semiautomatic way. Consequently, there is a large need for automatic algorithms for image segmentation and surface extraction in medical imaging community.

Deformable models are widely applied techniques for surface extraction within medical image analysis (McInerney and Terzopoulos, 1996, Montagnat *et al.*, 2001). The surface extraction with deformable models is re-formulated as an energy minimization problem. The energy function to be minimized is a weighted sum of two energies: the internal energy reflects prior information about the shape of surface of interest and the external energy is derived from image data. By the internal energy deformable models are able to cope with image noise and discontinuous object boundaries that are typical to medical images.

Deformable models may show a drawback despite of their wide applicability. Namely, the energy function has numerous local minima, which causes deformable models to be sensitive to their initialization. This complicates their fully automatic use for surface extraction. Several distinct approaches have been taken to overcome the initialization sensitivity problem. Koikkalainen and Lötjönen (2002) have suggested techniques to automate the initialization process and Xu and Prince (1998a,b) have proposed an advanced method to compute external forces to reduce the initialization sensitivity. The level-set technique (Malladi *et al.*, 1995) can be applied if the topology of the surface of interest is not *a priori* known. However, level-set based deformable models for the extraction of surfaces with a fixed topology appear to be sensitive to their initialization (Han *et al.*, 2003). In this study, we take a direct approach to reduce the initialization sensitivity and aim to minimize the energy of the deformable surface globally.

We have previously proposed the dual surface minimization algorithm (DSM) for the global optimization problem (Tohka and Mykkänen, 2004). The algorithm, although not proven to converge to the global minimum, has been found effective in reducing the initialization sensitivity. Hence, it can be well applied to surface extraction tasks challenged by image noise (Mykkänen *et al.*, 2003, Kivimäki *et al.*, 2004, Tohka *et al.*, 2004). As a downside, the DSM algorithm makes implicit assumptions about the shape of the surface to be extracted. In particular, the extraction of surfaces with a complex geometry does not succeed well with the algorithm. In addition, the algorithm is restricted to the extraction of surfaces topologically equivalent to the sphere.

In this study, we introduce a new global optimization algorithm that makes less assumptions about the shapes of the surfaces to be extracted than DSM and that can extract surfaces with handles and holes as long as the surface topology is *a priori* known. If fixing the surface topology is not appropriate then other methods, such as level-set based deformable surfaces (Malladi *et al.*, 1995) are better suited. The new algorithm, termed AdaCoP (adaptive constraints and penalties), differs fundamentally from the DSM algorithm although both algorithms are meant for the global optimization of the energy of deformable surfaces. The main technical difference is that DSM is based on iterated local search whereas AdaCoP is based on the principles of constrained optimization in the continuous domain. AdaCoP uses an algorithm for constrained local optimization in an iterative manner. The convergence to a same local minimum over and over again is prevented by adapting the constraints for optimization. The algorithm is helped to escape from local minima by inducing an adaptive penalty energy. (That is why we name the algorithm as AdaCoP (adaptive constraints and penalties)).

The organization of the paper is as follows: First, we describe simplex meshes that are applied for surface representation. Then, we lay out the assumptions concerning the energy functions of the deformable surface. After that, we describe the AdaCoP algorithm. We experiment the algorithm with synthetic and medical images. Finally, we discuss the algorithm related to other relevant methods for deformable mesh optimization.

DEFORMABLE MODEL

SIMPLEX MESHES

In this study, surfaces are represented using simplex meshes (Delingette, 1999). A simplex mesh is defined as a pair $(\mathbf{W}, \mathscr{G})$, where $\mathbf{W} = \{\mathbf{w}_1, \dots, \mathbf{w}_m\}$, $\mathbf{w}_i \in \mathbb{R}^3$ is the set of vertex coordinates (*i.e.*, mexels) and $\mathscr{G}$ is the vertex adjacency graph of the mesh. Simplex meshes have the desirable property that the graph of the mesh $\mathscr{G}$ is 3-regular. That is, each vertex i has exactly three adjacent vertices denoted by i_1, i_2, i_3. Simplex meshes are the topological duals of triangular meshes in a sense the dual graph of the graph of a simplex mesh is the graph of some triangular mesh. However, there does not exist similar geometrical dualism between simplex and triangular meshes.

Our interest is in the global optimization of simplex mesh geometry. This means that the graph of the mesh $\mathscr{G}$ during the deformation must be fixed (Tohka, 2003, Chapter 4.6), *i.e.*, the topology of the surface to be extracted and the number of vertices in the mesh must be specified prior to surface extraction. In medical image analysis, the surface topologies are often known prior to surface extraction, and then it is advisable to constrain the topological type of the deformable surface. Indeed, it is often necessary that a surface extraction algorithm produces a surface with the pre-defined topology. In addition, the number of vertices in the mesh can often be selected based on prior information about the surfaces to be extracted. Hence, these limitations are not severe. Because $\mathscr{G}$ is constant, we identify $\mathbf{W}$ as the surface and assume that $\mathscr{G}$ is implicitly known. The three neighbors of the mexel $\mathbf{w}_i$ are denoted $\mathbf{w}_{i_1}, \mathbf{w}_{i_2}, \mathbf{w}_{i_3}$. The unit normal to the surface at $\mathbf{w}_i$ is denoted by $\mathbf{n}_i$ and it is proportional to $\mathbf{w}_{i_1} \times \mathbf{w}_{i_2} + \mathbf{w}_{i_2} \times \mathbf{w}_{i_3} + \mathbf{w}_{i_3} \times \mathbf{w}_{i_1}$, where $\times$ denotes cross product. It is assumed that the surface normals point inwards.

Simplex meshes are preferred here over triangular meshes because surface normals are unambiguously definable at the vertices of the simplex mesh (Delingette, 1999) and our algorithm depends on the definition of the surface normal at the vertices of the mesh. With triangular meshes the surface normals are unambiguously definable on the faces of the mesh, but not at the vertices of the triangular mesh.

ENERGY FUNCTIONS

The energy of a surface quantifies how well the surface couples with the underlying image and how well it satisfies our prior assumptions about its shape. The energy of $\mathbf{W}$ is

$$E(\mathbf{W}) = \lambda E_{int}(\mathbf{W}) + (1 - \lambda)E_{ext}(\mathbf{W}), \qquad (1)$$

where $E_{int}(\cdot)$ is the internal energy, $E_{ext}(\cdot)$ is the external energy, and the regularization parameter $\lambda \in [0, 1]$.

The internal energy reflects the prior assumptions about the surface shape. It acts as a regularizer of the surface shape, for example by penalizing surfaces that are not smooth. The regularization of the energy function by the internal energy is the key to noise tolerance of deformable surfaces.

The derivation of the AdaCoP algorithm does not require the full specification of the internal energy function. Moreover, deformable meshes with quite different internal energy functions can be optimized with the algorithm. Therefore, to maintain a high level of generality, the internal energy function is not specified at this point. However, it is helpful if the internal energy $E_{int}(\mathbf{w}_i|\mathbf{W})$ as a function of $\mathbf{w}_i$ with all the other vertex coordinates given is easy to compute. In addition, the internal energy should be scale, rotation, and translation invariant.

The external energy incorporates the information about the image into the surface extraction process. We define the external energy as

$$E_{ext}(\mathbf{W}) = \frac{1}{m}\sum_{i=1}^{m} E_{ext}(\mathbf{w}_i) = \frac{1}{m}\sum_{i=1}^{m}[1 - I(\mathbf{w}_i)], \quad (2)$$

where $I : \mathbb{R}^3 \to [0,1]$ is a piecewise constant function with a finite support. The function I called *the energy image* is constructed based on the image to be processed. The energy image may be viewed as a look-up-table for the external energy values for each image voxel. The values of the energy image should be high, on the average, for those voxels which are likely to be intersected by the surface to be extracted and low, on the average, for other voxels. For example, if the interest is in locating of the surface defined by the changes in image intensity, then the gradient operator followed by appropriate scaling of intensity values is used to construct the energy image. Note that the external energy is defined on the continuum ($\mathbb{R}^3$) as opposed to a discrete integer grid.

ADACOP ALGORITHM

OVERVIEW

The aim of the AdaCoP algorithm is to find a global minimizer of the energy (1), or more specifically to find a strong local minimum of (1) within a set of admissible meshes. The AdaCoP algorithm starts with an initialization placed inside or outside the surface of interest. For convenience, it is assumed that the initialization is placed outside the target surface. Modifications to the algorithm are straight-forward if the initialization is placed inside the target.

The surface $\mathbf{W}^{t+1}$ at the iteration $t+1$ is obtained by a local minimization algorithm initialized with $\mathbf{W}^t$ (the LM-step). In the LM step, constraints are set to prevent the volume inside the surface from increasing. After the step, the energy of the current surface is compared to the minimal energy value found during the iterations $1,\ldots,t$. If there is an improvement, the current surface and the current energy value are stored. Indeed, because of the hill climbing ability of AdaCoP, the current mesh has to be compared to the best mesh so far rather than to the mesh at the previous iteration. It may happen that the difference between $\mathbf{W}^t$ and $\mathbf{W}^{t+1}$ is negligible. In this case we say that the algorithm is in a local minimum. For escaping from local minima, the AdaCoP algorithm is equipped with a hill-climbing ability. The hill-climbing is realized by adding a penalty energy term to the energy function. The penalty applies to vertex positions close to their current positions and the penalty is adaptively grown until the result of the LM step differs from its initialization. When the mesh has changed its position, the penalty is removed and the iterations of the LM step continue in the normal fashion until the algorithm is in another local minimum. The AdaCoP algorithm stops when the volume inside $\mathbf{W}^t$ is lower than a given threshold. The surface of the lowest energy is returned as the result. A pseudo-code for AdaCoP is presented in Algorithm 1.

Algorithm 1 AdaCoP Algorithm

Initialize $\mathbf{W}^0$, set $t \leftarrow 0$
Set best minimum $\leftarrow E(\mathbf{W}^0)$, set $\mathbf{W}^* \leftarrow \mathbf{W}^0$
while Area inside $\mathbf{W}^t$ larger than the threshold **do**
   Obtain $\mathbf{W}^{t+1}$ by a local energy minimization starting from $\mathbf{W}^t$
   if $E(\mathbf{W}^{t+1}) <$ best minimum **then**
     set $\mathbf{W}^* \leftarrow \mathbf{W}^{t+1}$
   end if
   if $\mathbf{W}^t \neq \mathbf{W}^{t+1}$ **then**
     Set penalty term to zero
   else
     Increase the penalty term
   end if
   Set $t \leftarrow t+1$
end while
Return $\mathbf{W}^*$.

In the following sub-sections the LM step, constraints that are set, and the hill-climbing are presented in more detail. The surface at the iteration t is denoted by $\mathbf{W}^t = \{\mathbf{w}_i^t\}$. Further, we set

$$\mathbf{W}_i^t = \{\mathbf{w}_1^{t+1},\ldots,\mathbf{w}_{i-1}^{t+1},\mathbf{w}_{i+1}^t,\ldots,\mathbf{w}_m^t\}.$$

The symbol $\mathbf{n}_i^t$ denotes the unit normal to the surface at $\mathbf{w}_i^t$ and it is computed based on $\mathbf{W}_i^t$.

LOCAL MINIMIZATION: LM STEP

Starting from the surface at the iteration t, $\mathbf{W}^t = \{\mathbf{w}_i^t\}$, the surface at the iteration $t+1$ is obtained by an iteration of an iterative conditional modes (ICM) like algorithm (Besag, 1986). Each vertex position $\mathbf{w}_i^t$ is sequentially updated by minimizing the energy $E(\mathbf{w}_i|\mathbf{W}_i^t)$ with respect to $\mathbf{w}_i$. Given a feasible search direction $\mathbf{d}_i^t \in \mathbb{R}^3$ ($||\mathbf{d}_i^t|| = 1$), we set

$$\mathbf{w}_i^{t+1} = \mathbf{w}_i^t + s_{min}\mathbf{d}_i^t \qquad (3)$$

$$s_{min} = \arg \min_{s \in [0,1]} E(\mathbf{w}_i^t + s\mathbf{d}_i^t|\mathbf{W}_i^t) \doteq \arg \min_{s \in [0,1]} f(s) \, . \quad (4)$$

The optimization along the unit-vector $\mathbf{d}_i^t$ ensures that the volume of the surface does not increase. It is explained in the next sub-section how $\mathbf{d}_i^t$ is chosen.

To find s_{min}, one dimensional line search problem has to be solved. For this, there are several possible methods. We apply a straight-forward one of detecting all critical points and selecting the one with the lowest value of $f(s)$ as s_{min}. The critical points are 1) zeros of the derivative of $f(s)$, 2) points of discontinuity of $f(s)$, and 3) points $s = 0$ and $s = 1$. Since the external energy is piecewise constant, its derivative is equal to zero, and

$$f'(s) = \frac{\partial E(\mathbf{w}_i^t + s\mathbf{d}_i^t|\mathbf{W}_i^t)}{\partial s} = \lambda \frac{\partial E_{int}(\mathbf{w}_i^t + s\mathbf{d}_i^t|\mathbf{W}_i^t)}{\partial s} \, , \quad (5)$$

where the derivative is defined. This means that zeros of $f'(s)$, if exist, can be solved based on the internal energy only. The function $f(s)$ is not continuous where $\mathbf{w}_i^t + s\mathbf{d}_i^t$ crosses voxel boundaries. These points, there exist at most three of them, are easily detected. The value of $f(s)$ must be evaluated at both sides of these critical points, because $f(s)$ is not continuous.
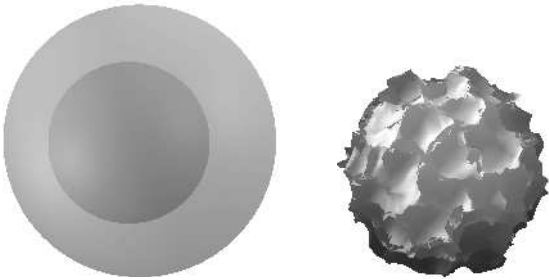

Fig. 1. *A failed extraction of a sphere surface from noiseless image with surface normals as search directions. Left: Initialization (transparent surface) and the correct surface. Right: The extraction result.*

SELECTION OF SEARCH DIRECTION

A feasible search direction must guarantee reduction of the volume inside the surface. A possible choice would be the surface normal at $\mathbf{w}_i^t$. Unfortunately, this choice leads easily to surface self-intersections destroying the information about the orientation of the surface that is essential to AdaCoP, *cf.* Fig. 1.

Instead, we generate a sequence of search directions $\{\mathbf{d}_i^t|t = 1,2,\ldots\}$, which 1) ensure that the volume of the surface diminishes, 2) do not change radically from an iteration to another, and 3) are similar between neighboring vertices. The requirements 2) and 3) are to prevent formations of surface self-intersections, or more precisely, to make the intersections less likely. A sequence of slowly adaptive *unconstrained search directions* $\mathbf{u}_i^t$ is defined by

$$\mathbf{u}_i^t = \frac{1}{4}(1-\omega)(\mathbf{d}_i^{t-1} + \sum_{j=1}^3 \mathbf{d}_{i_j}^{t-1}) - \omega \frac{\nabla E_{int}(\mathbf{w}_i^t|\mathbf{W}_i^t)}{||\nabla E_{int}(\mathbf{w}_i^t|\mathbf{W}_i^t)||} \, , \quad (6)$$

where $\omega \in \mathbb{R}$ is a user-definable parameter. In the right hand side of Eq. 6, the first term maintains the search directions between the iterations and neighboring vertices similar, and the second term is an approximation of the steepest descent direction which alters the unconstrained search direction. From $\mathbf{u}_i^t$, the search direction $\mathbf{d}_i^t$ is generated by a projection technique (Rosen, 1960). With the convention that $\mathbf{d}_i^t = \mathbf{n}_i^t$ if $\mathbf{u}_i^t = \mathbf{0}$, we select

$$\mathbf{d}_i^t \propto \begin{cases} \mathbf{u}_i^t & \text{if} \quad \mathbf{u}_i^t \cdot \mathbf{n}_i^t \geq \varepsilon||\mathbf{u}_i^t|| \\ \mathbf{u}_i^t + c\mathbf{n}_i^t & \text{otherwise} \end{cases} , \quad (7)$$

where $c = \frac{\varepsilon}{\sqrt{1-\varepsilon^2}}\sqrt{||\mathbf{u}_i^t||^2 - (\mathbf{n}_i^t \cdot \mathbf{u}_i^t)^2} - \mathbf{n}_i^t \cdot \mathbf{u}_i^t$, and $\cdot$ denotes the scalar product. As it is verified in Appendix, for this selection and $\varepsilon \in (0,1)$, it holds that

$$\mathbf{d}_i^t \cdot \mathbf{n}_i^t \geq \varepsilon||\mathbf{d}_i^t|| \, . \quad (8)$$

(In this study, we set always $\varepsilon = 0.1$.) This means that the search direction $\mathbf{d}_i^t$ points inwards from the surface at the current mexel position $\mathbf{w}_i^t$ if the surface is locally smooth enough. In practice, this ensures that the surface volume does not increase with iterations. Indeed, while this kind of condition seems natural, we have been unable to formally prove that Eq. 8 ensures the decrease in the volume of the mesh. This difficulty relates to ambiguity of the computation of the volumes of simplex meshes.

HILL-CLIMBING ABILITY

The LM step sometimes produces its initialization as its result, *i.e.*, $\mathbf{W}^t = \mathbf{W}^{t+1}$ for some t. In this case, the algorithm is at a local minimum, and it must be helped to escape from the minimum. For this, a penalty term $p(s)$ that decreases with s is added to the right hand side of Eq. 4, and s_{min} is solved as

$$s_{min} = \arg \min_{s \in [0,1]} E(\mathbf{w}_i^t + s\mathbf{d}_i^t | \mathbf{W}_i^t) + rp(s) . \quad (9)$$

The integer $r = 1, 2, \ldots,$ is increased until $\mathbf{W}^t \neq \mathbf{W}^{t+r}$. We define

$$p(s) = \begin{cases} \gamma & \text{when} \quad s < 0.5 \\ 0 & \text{when} \quad s \geq 0.5 \end{cases} \quad (10)$$

For us, $\gamma = 0.02$ has worked well. When this penalty term is active, $s = 0.5$ is a critical point in addition to the ones mentioned previously.

PROPERTIES OF THE ALGORITHM

The AdaCoP algorithm is global in a sense that it avoids getting trapped in local energy minima due to its hill-climbing ability. This is important in deformable model optimization since this property allows initialization far away from the surface of interest. The algorithm cannot be proven to converge to the global minimum, but it is capable to find a very good local energy minimum.

The energy of the deformable surface decreases with each LM step assuming that the penalty term is not active. This follows from resemblance to the ICM algorithm (Besag, 1986).

In practice, the volume inside the deformable surface does not increase with iterations. This, again in practice, ensures the convergence of the algorithm in a sense that the algorithm will stop. Because the information about the oriented surface normals is used to select 'volume decreasing' search directions, the self-intersections of the surface can break down the algorithm. Instead of forbidding self-intersections explicitly, which would be of high computational cost (MacDonald *et al.*, 2000), we select search directions in such a way that the surface evolves in a controlled manner during the optimization.

INTERNAL ENERGY FOR EXPERIMENTS

We fix the internal energy function and derive the required formulae to optimize the energy function with AdaCoP. This simple internal energy function, which encourages smooth surfaces, is applied in our experiments. The internal energy (Eq. 11) states that a mexel should lie near the centre of mass of its three neighbours.

Let $A_i = \{i_1, i_2, i_3\}$ denote the set of vertices adjacent to the vertex i. The internal energy function is defined as

$$E_{int}(\mathbf{W}) = \frac{1}{m} \sum_{i=1}^{m} E_{int}^i(\mathbf{w}_i | \mathbf{w}_j, j \in A_i)$$
$$= \frac{1}{m} \sum_{i=1}^{m} \frac{||\mathbf{w}_i - \frac{1}{3}\sum_{j \in A_i} \mathbf{w}_j||^2}{\mu(\mathbf{W})} , \quad (11)$$

where the normalization factor $\mu(\mathbf{W})$ is a measure of the area of the surface required for scale invariance. We calculate $\mu(\mathbf{W})$ as the average area of faces of simplex mesh. For computational simplicity, $\mu(\mathbf{W})$ is approximated by a constant computed based on the current vertex positions. Then, using the chain rule, the gradient of $E_{int}(\mathbf{w}_i | \mathbf{W}_i^t)$ becomes

$$\frac{2((1 + \frac{1}{3})\mathbf{w}_i + \sum_{j \in A_i}[-\frac{2}{3}\mathbf{w}_j^t + \frac{1}{3^2}\sum_{k \in A_j, k \neq i} \mathbf{w}_k^t])}{\mu(\mathbf{W}^t)}$$

and $\mathbf{d}_i^t$ can be computed according to Eq. 7. The derivative of $f(s)$ (Eq. 5) has at most one zero in the range $s \in [0,1]$ and it is

$$\mathbf{d}_i^t \cdot (-\mathbf{w}_i^t + \sum_{j \in A_i}[\frac{1}{2}\mathbf{w}_j^t - \frac{1}{12}\sum_{k \in A_j, k \neq i} \mathbf{w}_k^t)] \doteq s^*$$

if it lies in the desired interval. Hence, in order to find s_{min} the energy needs to be evaluated at $s = 0, s = 1, s = s^*$, intersections of $\mathbf{w}_i^t + s\mathbf{d}_i^t$ with the voxel boundaries within $s \in [0,1]$, and $s = 0.5$ if the penalty term is active.

EXPERIMENTS AND RESULTS

In this section, the performance of the AdaCoP algorithm is evaluated by extracting of surfaces from synthetic images. The parameter values for AdaCoP are $\gamma = 0.02, \lambda = 0.5$ and $\omega = 0.25$ unless otherwise mentioned.

In the first experiment, the surfaces of an ellipsoid bended along the *y*-axis and a torus were to be extracted. The surfaces were drawn to $128 \times 128 \times 128$ image with the intensity value of one, the background having intensity of zero. (These were the energy images, which means that we were interested in locating surfaces characterized by high intensity values in the original images.) In Fig. 2, the correct surfaces
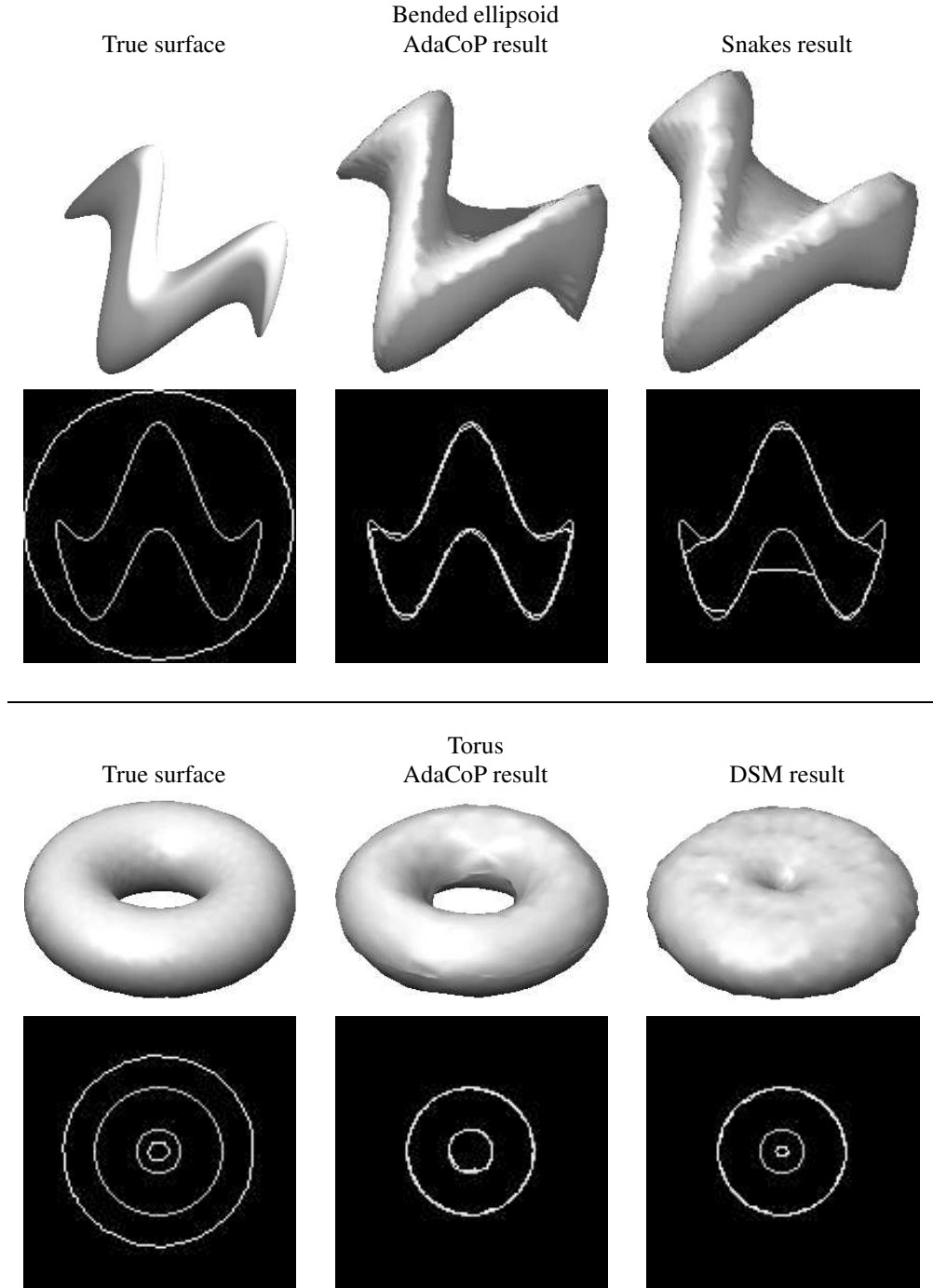
Fig. 2. *Bended ellipsoid and torus extraction results. The column 'true surface' shows the correct surface and intersections of the correct surface (in gray) and initial surface (in white) with the central xy-plane of the image. The result columns show surfaces extracted by different algorithms and their intersections with the central xy-plane (in white) along with the intersections of the correct surface (in gray). For the extraction of bended ellipsoid (resp. torus), we applied meshes with 1280 (958) mexels.*
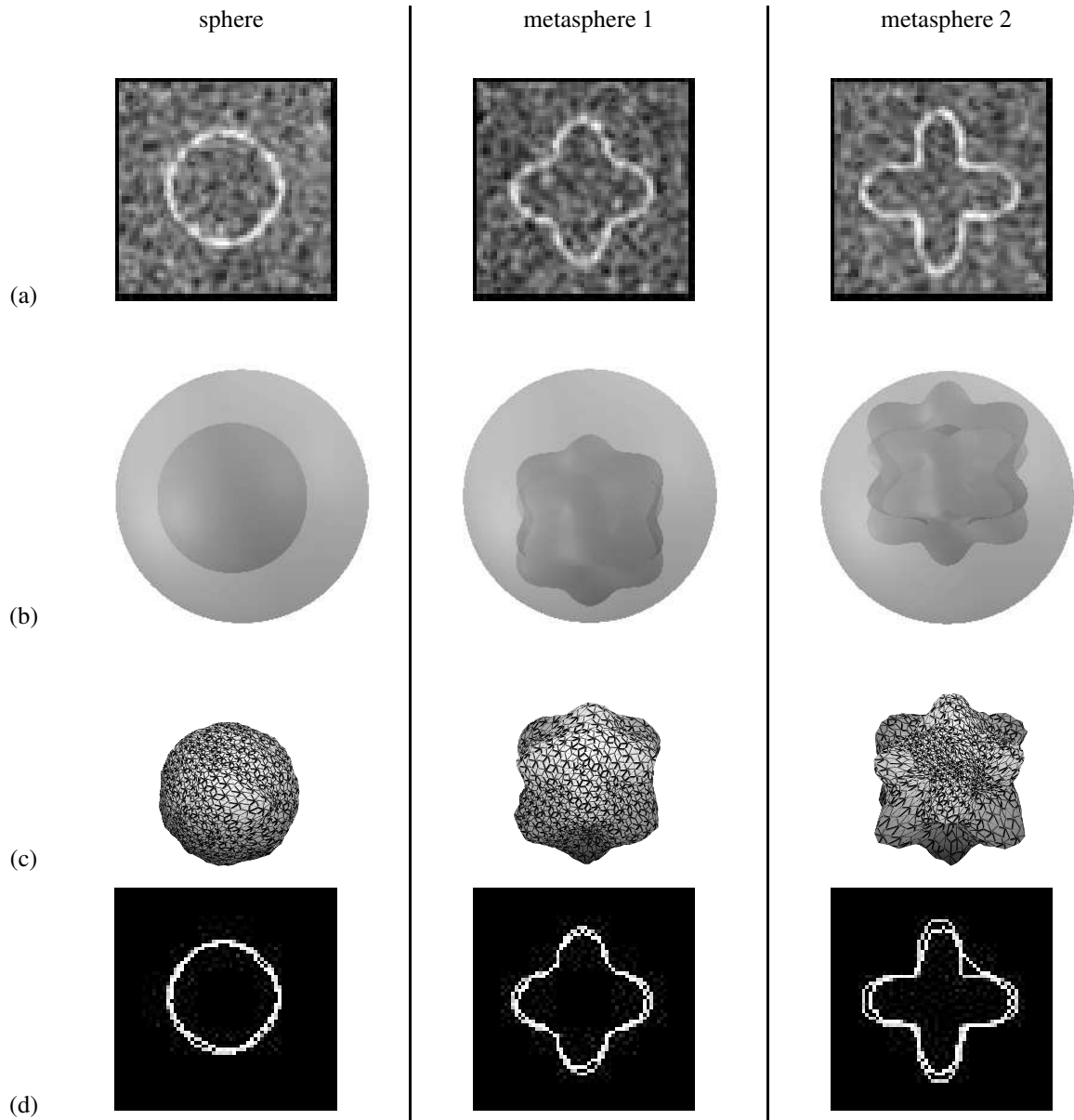
	sphere	metasphere 1	metasphere 2

(a)

(b)

(c)

(d)

Fig. 3. *Example results from experiments with noisy images. All shown results are from the initialization that yielded the median value of error. (a) Central cross-sections (in xy-plane) of noisy energy images, the noise variance was 0.2. (b) The synthetic surfaces inside the initializations for the algorithm. (c) The resulting meshes extracted from noisy images starting from initializations in (b). (d) Intersections of the resulting meshes with the central xy-plane (in white) compared to the ground-truth (in gray).*

Table 1. *The values of the error measure obtained by the AdaCoP and the DSM algorithm. The parameter ω was 0.01. The results of DSM are from (Tohka and Mykkänen, 2004). σ^2 denotes the variance of the Gaussian noise in the image.*

	sphere				metasphere 1				metasphere 2			
σ^2	0	0.2	0.6	1.0	0	0.2	0.6	1.0	0	0.2	0.6	1.0
AdaCoP	0.03	0.07	0.12	0.21	0.03	0.09	0.16	0.20	0.05	0.12	0.22	0.27
DSM	0.03	0.07	0.11	0.18	0.03	0.08	0.12	0.15	0.07	0.13	0.18	0.27

and the extraction results using AdaCoP, DSM-outer surface (DSM-OS), and a 3-D version of standard snakes (Kass *et al.*, 1988) are shown. Starting from the same initialization, placed outside the surface of interest, AdaCoP captured non-convex parts of the bended ellipsoid much better than the snakes algorithm as can be seen in Fig. 2. The parameter values for snakes were obtained by the trial and error method and the result shown in Fig. 2 was with the best parameter set. The capture range of snakes was also increased by Gaussian filtering the energy image before the external force computation. As shown in Fig. 2, the extraction of the torus with the DSM-OS algorithm did not succeed. On the other hand, the AdaCoP algorithm was able to extract the torus of our experiment. The initial surface for both algorithms was a torus placed outside the surface of interest.

In the second experiment, the noise sensitivity of AdaCoP and (the standard) DSM algorithm was compared. We used the same synthetic images and experimental setting as Tohka (2002) and Tohka and Mykkänen (2004). The images contained closed surfaces (sphere, metasphere 1, or metasphere 2 in Fig. 3) drawn to a $64 \times 64 \times 64$ grid with intensity value of one. Thereafter, white Gaussian noise was added to the images. The energy images were created by filtering noisy images with a Gaussian filter, and then linearly scaling images to have intensities from zero to one.

Quantitative results were computed using the following error measure: Let *EV* (extracted volume) be the set of voxels that are inside of or intersected by the extracted surface and let *TV* (true volume) be the the set of voxels on or inside of the true digital surface. Then, the error $e = 1 - \frac{|TV \cap EV|}{|TV \cup EV|}$.

Each surface was extracted starting from five different initializations placed outside of the target surface. The quantitative error measure was computed for all five surfaces and the median values of it are listed in Table 1. For comparison, the results with the DSM algorithm are also shown in Table 1. Some examples of extracted surfaces are shown in Fig. 3. The AdaCoP algorithm achieved good results with each surface with the two lowest levels of noise (variances 0 and 0.2). With the two higher noise levels, the results obtained by DSM were slightly better than AdaCoP's results. This could be expected since the purpose of AdaCoP was not to achieve the noise robustness of DSM but to allow the extraction of more complex surfaces. The results obtained by AdaCoP were consistent between initializations, which is important for repeatable and reliable surface extraction.

In the second experiment, meshes with 1280 mexels were applied and the value of ω was

0.01. A surface extraction with a Matlab-based implementation of AdaCoP took from 6 to 15 minutes with a Linux PC with a 3.0 GhZ processor.
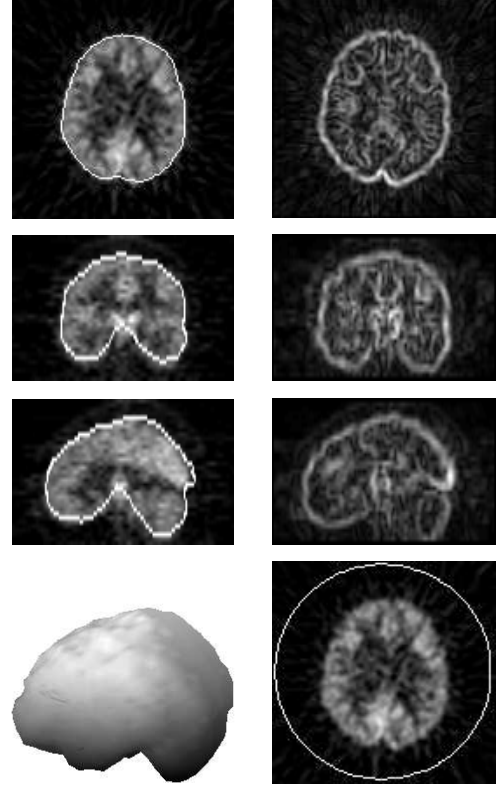
APPLICATIONS

Fig. 4. *PET brain surface extraction. Left column from top: central transaxial, coronal, and sagittal cross-sections of the extracted brain surface overlaid on the corresponding image slices and a 3D rendering of the extracted brain surface mesh. Right column from top: Slices of the energy image and the transaxial cross-section of the initial surface. The simplex mesh contained 1280 mexels and the parameter values for AdaCoP were λ = 0.5, γ = 0.02, ω = 0.01. The image dimensions were 128 × 128 × 35 and the voxel size was 1.72 mm × 1.72 mm × 4.25 mm.*

In this section, potential applications of the AdaCoP-based deformable surfaces for automatic medical image analysis are presented. The demonstrations are drawn from the automated segmentation of positron emission tomography (PET) brain images displaying information about the physiological properties of the brain. For more about PET imaging and its applications in medicine, see Phelps and Mazziotta (1985) and Passchier *et al.* (2002). In Fig. 4, brain surface extraction from a noisy

PET image acquired using fluoro-2-deoxy-D-glucose (FDG) as the radiopharmaceutical is demonstrated. Since the brain surface is characterized by strong image edges, the energy image was computed using the 3-D Sobel edge operator (Zucker and Hummel, 1981). Despite of large amount of noise in the image, AdaCoP captured details of the surface surprisingly well. In addition, the initialization could be placed far away from the brain surface.
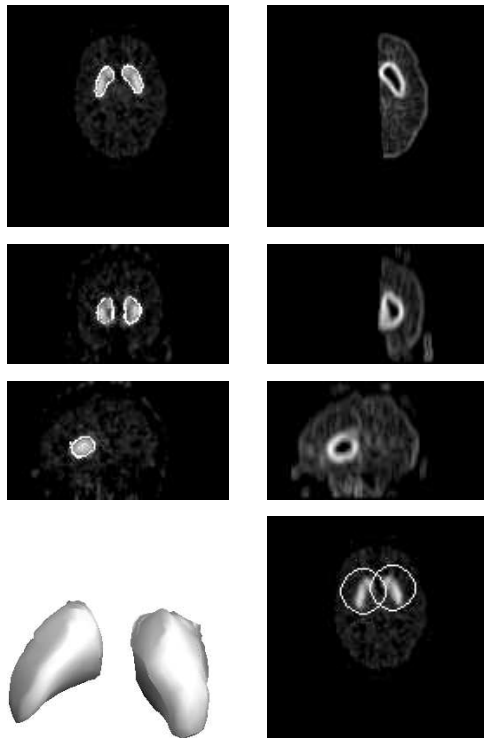

Fig. 5. *Extraction of striatum from Raclopride PET images. Left column from top: transaxial, coronal, and sagittal cross-sections of the extracted surfaces overlaid on the corresponding image slices and a 3D rendering of the extracted striatum. Right column from top: Slices of the energy image for the right hemisphere and the transaxial cross-sections of the initial surfaces. The parameter values for AdaCoP were $\lambda = 0.5$, $\gamma = 0.02$, $\omega = 0.25$. The image dimensions were $128 \times 128 \times 35$ and the voxel size was 2.3 mm $\times$ 2.3 mm $\times$ 4.25 mm.*

In Fig. 5, the extraction of left and right striatum from binding potential (Passchier *et al.*, 2002) [11]C Raclopride-PET images is depicted. The automatic extraction of striatum has potential applications for drug development (Tohka *et al.*, 2004). The energy images, computed again using the 3-D Sobel edge operator, were computed separately for left and right hemispheres, allowing the extraction of left and right striatum separately. The hemispheres were separated using an automatic algorithm for mid-sagittal plane determination (Mykkänen *et al.*, 2003). As can be

seen in Fig. 5, the surfaces of left and right striatum were well extracted by AdaCoP. The main challenge in this extraction task lies in the heterogeneity of the intensity values within the striatum which is mainly due to image noise and the partial volume effect.

DISCUSSION

We have introduced a new algorithm for the global minimization of the energy of deformable surface models. Efficient and effective global optimization strategies are important for deformable models, because local approaches, such as snakes (Kass *et al.*, 1988), are overly sensitive to their initializations. Due to this, a good initial surface mesh in a close vicinity of the surface of interest must be provided by the user. This restricts the automatic use of locally-based deformable models for surface extraction from volumetric image data.

The initialization sensitivity problem of deformable models has been addressed in various ways. For example, Xu and Prince (1998a,b) suggested an external force which allows for initializations far from the surface of interest. Application specific shape models have been used to reduce the number of local optima of the energy function (Shen *et al.*, 2001). However, the construction of a statistical shape model requires a large number of examples of the typical shape of the surface of interest. The construction of such examples, that have to be handcrafted, can be costly and time-consuming.

The global optimization approach to reduce the initialization sensitivity taken in this work has the advantage of being conceptually simple and well-defined at the mathematical level. The challenge is algorithmic: The optimization problem formulated is a difficult one having a large number of variables (from thousands to tens of thousands). Only few authors have considered the global optimization of deformable surfaces; with the exception of coarse-to-fine methods tailored for a particular application (MacDonald *et al.*, 2000). This is in contrast to deformable contours for which a number of global optimization methods have been suggested *e.g.*, (Storvik, 1994, Geiger *et al.*, 1995, Akgul and Kambhamettu, 2003). The optimization approaches for deformable contours are, however, often completely unsuitable (dynamical programming) or computationally too burdensome (evolutionary approaches) for deformable surface optimization.

We have previously proposed the dual surface minimization algorithm (DSM) for deformable mesh optimization. The algorithm is not suitable for the

extraction of surfaces with a non-spherical topology or complex shape as was demonstrated in the Experiments Section. The purpose of the current work was to derive an algorithm, termed AdaCoP, which could be used to extract a larger variety of surfaces than DSM while retaining the algorithm robust to image noise. The basic idea of AdaCoP is similar than in DSM. The deformable surface approaches the surface of interest by successively detecting local energy minima. Still, the two algorithms are quite different. (i) In AdaCoP the optimization problem is posed in continuous domain whereas in DSM the search space is discretized. Optimization in continuous domain is perhaps more elegant solution, and nevertheless setting the resolution of the discrete search space can be problematic. (ii) The evolution of the deformable surface is controlled differently in DSM and AdaCoP. In DSM, each mexel tends towards a single point, termed the reference point. This is the source of incapability of the DSM algorithm to capture complicated shapes properly. The AdaCoP algorithm, on the other hand, uses a strategy of slowly updating the search direction for each vertex. This enables us to control of the tradeoff between the shape complexity and image noise using a single parameter (ω). Increasing the value of ω means that there is more freedom for surface shape. As was demonstrated, the deformable surface based on the AdaCoP algorithm is able extract surfaces with a challenging shape as well as simpler surfaces from very noisy images. Only the value of ω was varied between the experiments. The values for other parameters were the same in all the experiments, which demonstrates that AdaCoP does not rely on extensive parameter tuning as opposed to many other deformable surface methods.

A new algorithm, termed AdaCoP, to minimize the energy of deformable meshes globally has been proposed. The importance of the global optimization of deformable meshes lies in the possibilities that it allows for unsupervised surface extraction from noisy volumetric images. This, for example, is required when extracting quantitative information from a large medical image database. We have evaluated the proposed algorithm quantitatively with synthetic image data and shown that the algorithm can operate in fairly distinct conditions. In addition, we have presented potential applications of the AdaCoP algorithm within the medical image analysis. As these experiments suggest, the algorithm reduces the initialization sensitivity of deformable surfaces and therefore allows completely automatic surface extraction in a variety of possible applications.

APPENDIX

In this Appendix, we show by a direct calculation that the condition in Eq. 8 follows from the selection of search direction according to Eq. 7. Indices are dropped for convenience. Let $\mathbf{u}$ be an arbitrary vector, $\mathbf{n}$ be such that $||\mathbf{n}|| = 1$, and select $\mathbf{d}$ according to Eq. 7. Let $\varepsilon \in (0,1)$. Then

$$\mathbf{d} \cdot \mathbf{n} = (\mathbf{u} + c\mathbf{n}) \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n} + c$$
$$= \frac{\varepsilon}{\sqrt{1-\varepsilon^2}} \sqrt{||\mathbf{u}||^2 - (\mathbf{u} \cdot \mathbf{n})^2} \, , \qquad (12)$$

$$||\mathbf{d}||^2 = ||\mathbf{u}||^2 + 2c(\mathbf{u} \cdot \mathbf{n}) + c^2$$
$$= ||\mathbf{u}||^2 - (\mathbf{u} \cdot \mathbf{n})^2 + \frac{\varepsilon^2}{1-\varepsilon^2}(||\mathbf{u}||^2 - (\mathbf{u} \cdot \mathbf{n})^2)$$
$$= \frac{1}{1-\varepsilon^2}(||\mathbf{u}||^2 - (\mathbf{u} \cdot \mathbf{n})^2) \, . \qquad (13)$$

By combining Eqs. 12 and 13, we obtain $\mathbf{d} \cdot \mathbf{n} = \varepsilon ||\mathbf{d}||$.

REFERENCES

Akgul Y, Kambhamettu C (2003). A coarse-to-fine deformable contour optimization framework. IEEE Trans Patt Anal Mach Intell 25:174-86.

Besag J (1986). On the statistical analysis of dirty pictures. J R Stat Soc Series B 48:259-302.

Delingette H (1999). General object reconstruction based on simplex meshes. Int J Computer Vision 32:111-42.

Geiger D, Gupta A, Costa L, Vlontzos J (1995). Dynamic programming for detecting, tracking and matching deformable contours. IEEE Trans Patt Anal Mach Intell 17:294-302.

Han X, Xu C, Prince J (2003). A topology preserving level set method for geometric deformable models. IEEE Trans Patt Anal Mach Intell 25:755-68.

Kass M, Witkin A, Terzopoulos D (1988). Snakes: Active contour models. Int J Computer Vision 1:321-31.

Kivimäki A, Tohka J, Anttila M, Ruotsalainen U (2004). Extraction of heart volumes from dynamic fdg-pet emission images for movement corrections. Eur J Nucl Med Mol Imaging 31:S406 (Supplement 2).

Koikkalainen J, Lötjönen J (2002). Model library for deformable model-based segmentation of 3-D brain MR-images. In: Proc. of Medical Image Computing and Computer Assisted Intervention (MICCAI), vol. 2488 of *Lecture Notes in Computer Science*. Springer-Verlag, pp. 540-7.

MacDonald D, Kabani N, Avis D, Evans A (2000). Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI. Neuroimage 12:340-56.

Malladi R, Sethian J, Vemuri BC (1995). Shape modelling with front progagation: A level set approach. IEEE Trans Patt Anal Mach Intell 17:158-75.

McInerney T, Terzopoulos D (1996). Deformable models in medical image analysis: A survey. Med Image Anal 2:91-108.

Montagnat J, Delingette H, Ayache N (2001). A review of deformable surfaces: topology, geometry and deformation. Image Vision Comput 19:1023-40.

Mykkänen J, Tohka J, Luoma J, Ruotsalainen U (2003). Automatic ectraction of brain surface and mid-sagittal plane from pet images applying deformable models. Tech. Rep. A-2003-1, Dept. of Computer and Information Sciences, University of Tampere, Finland. Available at `http://www.cs.uta.fi/reports/r2003.html`.

Passchier J, Gee A, Willemsen A, Vaalburg W, van Waarde A (2002). Measuring drug-related receptor occupancy with positron emission tomography. Methods 27:278-86.

Phelps M, Mazziotta J (1985). Positron emission tomography: Human brain function and biochemistry. Science 228:799-809.

Rosen J (1960). The gradient projection method for nonlinear programming part 1: Linear constraints. J SIAM 8:181-217.

Shen D, Herskovits E, Davatzikos C (2001). An adaptive focus statistical shape model for segmentation and shape modeling of 3D brain structures. IEEE Trans Med Imaging 20:257-70.

Storvik G (1994). A bayesian approach to dynamic contours through stochastic sampling and simulated annealing. IEEE Trans Patt Anal Mach Intell 16:976-86.

Tohka J (2002). Surface extraction from volumetric images using deformable meshes: A comparative study. In: Proc. of 7th European Conference on Computer Vision, Lecture Notes in Computer Science 2352, pp. 350-64

Tohka J (2003). Global Optimization-based Deformable Meshes for Surface Extraction from Medical Images. Ph.D. thesis, Tampere University of Technology, Finland.

Tohka J, Mykkänen J (2004). Deformable mesh for automated surface extraction from noisy images. Int J Image Graphics 4:405-32.

Tohka J, Wallius E, Hirvonen J, Hietala J, Ruotsalainen U (2004). Improved reproducibility in dopamine D_2-receptor studies with automatic segmentation striatum from PET-images. In: IEEE Nuclear Science Symposium Conference Record (NSS-MIC 2004). In press. Available at `http://www.cs.tut.fi/~jupeto/striatum_5p.pdf`.

Xu C, Prince J (1998a). Generalized gradient vector flow external forces for active contours. Signal Process 71:131-9.

Xu C, Prince J (1998b). Snakes, shapes and gradient vector flow. IEEE Trans Image Process 7:359-69.

Zucker S, Hummel R (1981). A three-dimensional edge operator. IEEE Trans Patt Anal Mach Intell 3:324-31.