A General Discrete Contour Model in Two, Three, and Four Dimensions for Topology-Adaptive Multichannel Segmentation

Jörg Bredno, Thomas M. Lehmann, *Member*, *IEEE*, and Klaus Spitzer

Abstract—We present a discrete contour model for the segmentation of image data with any dimension of image domain and value range. The model consists of a representation using simplex meshes and a mechanical formulation of influences that drive an iterative segmentation. The object's representation as well as the influences are valid for any dimension of the image domain. The image influences introduced here, can combine information from independent channels of higher-dimensional value ranges. Additionally, the topology of the model automatically adapts to objects contained in images. Noncontextual tests have validated the ability of the model to reproducibly delineate synthetic objects. In particular, images with a signal to noise ratio of SNR ≤ 0.5 are delineated within two pixels of their ground truth contour. Contextual validations have shown the applicability of the model for medical image analysis in image domains of two, three, and four dimensions in single as well as multichannel value ranges.

Index Terms—Discrete contour model, segmentation, volumetric, spatio-temporal , multichannel , topology-adaptive, tracking.

1 Introduction

One of the main tasks in image analysis is the reproducible extraction of quantitative information. In medical image processing, the segmentation remains the weakest part of automated analyses [1]. The main requirements of automated segmentation are the generality to guarantee the applicability to many different segmentation tasks [2], the robustness of the segmentation against noise and artifacts [3], [4], and the reproducibility [5], i.e., the reduction of inter and intraobserver variabilities [6]. Even though many applications exist for automated segmentation, so do only a few robust realizations [4].

So far, deficiencies are inherent to automated segmentation. They are increased when the image domain is of higher dimension, e.g., representing an image sequence or volumetric data. Here, segmentation methods are mostly not general, but specific to few applications [7].

In this paper, we introduce a model-based segmentation that can be used for two (2D), three (3D), and four-dimensional (4D) image domains and is applicable to single and multichannel value ranges. The model is tailored for, but not restricted to medical images. Our work is related to the models introduced by Cohen and Cohen [8] as well as McInerney and Terzopoulos [9]. In these precursor works, 2D and 3D discrete contour models are used to segment medical images. Our work coherently extends such models into a 4D image domain, allows us to incorporate image information from independent color channels, uses a new computation of external influences that provides improved robustness against noise, and further extends the methods for automated adaptation of topology (e.g., [10], [11]).

A model-based segmentation consists of a variable object's representation and an algorithm that finds the instance of this representation that best explains the given image [12]. Therefore, we first describe the representation of objects (Section 2.1), then the mechanically formulated influences that act on the model (Section 2.2), and an iterative segmentation resulting from the evolution of the model under these influences (Section 2.3). The model is validated noncontextual (Section 3.1) and contextual (Section 3.2), and our approach and the results of the validation are discussed (Section 4).

2 General Model for Segmentation

In the following, we assume that the segmentation task is the automated delineation of objects contained in an image domain $\underline{I}$ of dimension $d_I \in \{2, 3, 4\}$ with $\underline{I} \subset \mathbb{N}_0^{d_I}$. For each image element, which is referred to as picture element (pixel), volume element (voxel), or time-volume element (toxel) at position $\vec{x} \in \underline{I}$, a one or more-dimensional value $\mathbf{w} \in \underline{W}$ is defined. Here, $\underline{W}$ is the value range of dimension d_W with $\underline{W} \subset \mathbb{R}^{d_W}$. The image itself is defined as the function $f : \vec{x} \in \underline{I} \rightarrow \mathbf{w} \in \underline{W}$. We use $\mathbf{w}$ or $\vec{x}$ to denote vectors having the character of a list or those describing positions and directions, respectively.

2.1 Representation of Objects

Objects are coherent subsets of $\underline{I}$ represented by their contour $\mathcal{C}$. Depending on the application in which the model is used, one or more objects can be represented by $\mathcal{C}$. Simplex meshes are used for the representation so that the border, surface, or metasurface of objects in two, three, and

- *J. Bredno is with Philips Research Labs, Weisshausstrasse 2, D-52066 Aachen, Germany. E-mail: joerg.bredno@philips.com.*
- *T.M. Lehmann and K. Spitzer are with the Institute of Medical Informatics, Aachen University of Technology, (RWTH), D-52057 Aachen, Germany. E-mail: lehmann@computer.org and spitzer@mi.rwth-aachen.de.*

Manuscript received 22 May 2001; revised 26 July 2002; accepted 28 Oct. 2002.
Recommended for acceptance by A. Khotanzad.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 114183.

four dimensions is linearly approximated. The mesh consists of a set $\underline{V} = \{v_0, v_1, \ldots, v_{|\underline{V}|}\}$ of vertices v_i and a set $\underline{E} = \{e_1, e_2, \ldots, e_{|\underline{E}|}\}$ of finite elements e_j which are called edgels. For every v_i, the vector $\vec{v}_i$ defines the position in $\underline{I}$.

Edgels are affine simplices Γ of dimension $d_I - 1$. The affine simplex is a set of points resulting from a limited linear combination of supporting vertices. Dependent on its dimension, the 0-simplex is a point, the 1-simplex a line, the 2-simplex a triangle, and the 3-simplex a tetrahedron. Each edgel e_j is supported by d_I vertices $v_{j,1\ldots d_I}$ and defines the connection and orientation of vertices:

$$e_j = \Gamma(v_{j,1}, \ldots, v_{j,d_I}),\ v_{j,k} \in \underline{V}. \tag{1}$$

Here, the general affine n-simplex Γ is defined as:

$$\Gamma(v_0, v_1, \ldots, v_n) = \left\{ \vec{x}_e \mid \vec{x}_e = \kappa_0 \vec{v}_0 + \ldots + \kappa_n \vec{v}_n, 0 \le \kappa_k \le 1, \sum_{k=0}^{n} \kappa_k = 1 \right\}, \tag{2}$$

where the κ_k are the restricted linear coefficients to define all points $\vec{x}_e$ of which e_j consists. For the computation of influences, a unity normal vector $\vec{n}_j$ and the size $|e_j|$ is computed for every edgel. We refer to a straight line between two vertices that share an edgel as a route in the mesh and define $\underline{U}_i^v(\Delta)$ as the set of all vertices that are no more than Δ routes apart of v_i. $\underline{U}_i^e$ is defined as the set of all edgels that are supported by v_i. In order to represent a coherent subset of $\underline{I}$ with a simplex mesh, the following requirements must be fulfilled:

- Every v_i must support at least d_I edgels.
- Every e_j has exactly d_I neighboring edgels. Two edgels are neighbored if they share $d_I - 1$ supporting vertices.
- No two edgels e_j and $e_k, j \neq k$ exist that share d_I vertices.
- The combinatorial order of vertices $v_{j,1\ldots d_1}$ gives the orientation of e_j such that $\vec{n}_j$ points outwards.
- Edgels do not intersect.
- If only one object is represented by $\mathcal{C}$, each v_i must be reachable by every other v_j using a succession of routes.

The number of edgels supported by a single vertex and therefore its valence is not fixed in three and four dimensions, whereas in two dimensions, every vertex always supports two edgels. Such a general mesh is called simplicial complex [13]. The simplicial elements are combinatorially connected to represent arbitrary objects [14].

Instances of this model in two, three, and four dimensions yield the following representations:

- *2D:* On planar images, $\mathcal{C}$ forms an irregular polygon.
- *3D:* For a volumetric data set ($d_I = 3$), an object is represented by a triangulated surface. Image sequences also result in a 3D image domain with positions $(x, y, t)^\top$. A triangulated edgel then becomes a line that is translated and scaled for a short sequence in time t and is at the same time a segment of a moving polygonal contour. All segments of the polygonal contour at time t_c are obtained by cutting the triangulated surface with planes $t = t_c$, the movement and velocity of this part of $\mathcal{C}$ is estimated from the triangle's orientation (Fig. 1).
- *4D:* In four dimensions, $\underline{E}$ consists of tetrahedrons with supporting vertices having coordinates in space and time $(x, y, z, t)^\top$. The cut of all edgels with hyperplanes $t = t_c$ yields a triangulated representation of an object at $t = t_c$. Then, each edgel yields either zero, one, or two facets of the triangulated object. These triangles are scaled and translated over a short period of time.

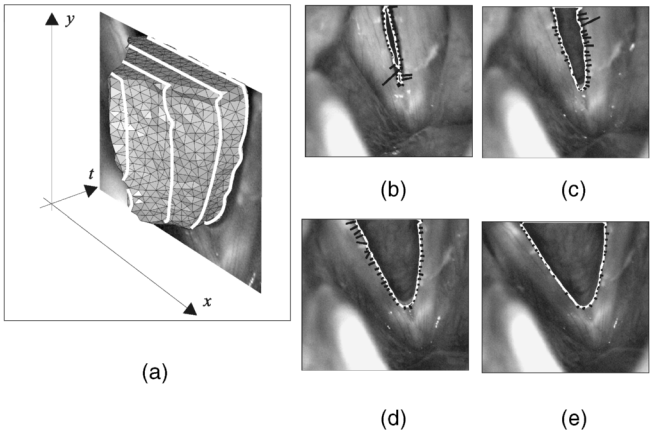

Fig. 1. Triangulated segmentation of a laryngoscopic video sequence. The glottal area which is framed by the vocal cords is delineated (a). The resulting contour (white) and vectors showing the local motion estimation (black) are displayed in frame 12, 17, 22, and 27 of the sequence in parts b, c, d, and e, respectively.

2.2 Mechanical Influences

The object's representation is combined with mechanical influences so that the edgels move to reach the border of relevant objects in the image. According to [15], the term "influence" is used as a generalization for internal and external energies, external force, image energy, image force, and many others because none of the latter terms is used unambiguously in literature.

The use of mechanical influences allows us to use formulations independent of the dimension d_I of the image domain. The model includes image or external influences depending on image values $\mathbf{w} \in \underline{W}$ that bind the active contour to objects in the image, deformation forces resulting from a modeling of material properties, and searching tendencies that move the edgels until they contact relevant image information. The searching tendency should be equal to the image influences at the border of objects. All influences result in forces acting on vertices. Influences on edgels are transferred to their supporting vertices. As such features have often been presented for deformable models, our focus lies on improved robustness and the general formulation in two, three, and four dimensions of the image domain.

2.2.1 Image Influence

Every e_j is influenced by all image values in the image subset $\underline{I}_j$ that contains all image elements in a defined vicinity of e_j:

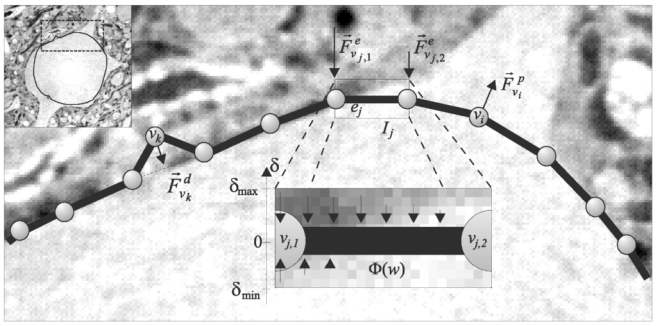

Fig. 2. Pressure force $\vec{F}^p_{v_i}$, deformation force $\vec{F}^d_{v_k}$, and external force $\vec{F}^e_{e_j}$ resulting from image potentials $\Phi(w)$ in the image subset $\underline{I}_j$ are shown for the segmentation of a micrograph (detail from the image in the upper left corner). The external force $\vec{F}^e_{e_j}$ is distributed to $\vec{F}^e_{v_{j,1}}$ and $\vec{F}^e_{v_{j,2}}$.

$$\underline{I}_j = \left\{\vec{x} \in \underline{I} \mid \vec{x} = \vec{x}_e + \delta \cdot \vec{n}_j,\ \vec{x}_e \in e_j, \delta_{\min} \le \delta \le \delta_{\max}\right\}. \quad (3)$$

This image subset contains all image elements that can be reached from positions $\vec{x}_e$ belonging to the edgel within the distance $[\delta_{\min}, \delta_{\max}]$ along $\vec{n}_j$. Usually, $\delta_{\min}$ is negative and $\delta_{\max}$ is positive so that $\underline{I}_j$ represents a "thickened edgel" in image domain. In our model, the dimension of the subset of image elements acting on an edgel is the dimension of the image domain itself, $\dim(\underline{I}_j) = d_I$. The definition of $\underline{I}_j$ requires a Cartesian image domain. This requirement does not necessarily hold for images that were acquired slice-wise. Here, the interslice resolution is usually lower than the intraslice resolution. Therefore, the z-axis is individually scaled before reading image values $f(\vec{x})$ to result in an approximately Cartesian image domain. In image sequences, there is no context that merges resolution in space and time. However, a scaling factor for the t-axis is used as well. It has to be chosen so that a triangulated representation is on the one hand able to track accelerated objects and, on the other hand still offers regularization between frames of the sequence.

To compute image influences, undirected image potentials $\mathbf{\Phi}(\mathbf{w})$ are assigned to the $\mathbf{w} \in \underline{W}$. The image potentials are chosen with regard to an application to describe the appearance of relevant objects in different channels of the image. For example, for a gray value image ($d_W = 1$), $\Phi(w)$ is a scalar function. For a red, green, and blue (RGB)-color image described by

$$\mathbf{w} = \left(w^{(r)}, w^{(g)}, w^{(b)}\right)^\top,$$
$$\mathbf{\Phi} = \left(\Phi^{(r)}(w^{(r)}), \Phi^{(g)}(w^{(g)}), \Phi^{(b)}(w^{(b)})\right)^\top$$

individually assigns potentials to color channels.

An edge template $\mathbf{k}(\delta)$, $\delta \in [\delta_{\min}, \delta_{\max}]$ in the dimension $\dim(\mathbf{\Phi}) = d_W$ is defined to describe the scale and strength of relevant image potentials that occur at the border of objects. The nonnormalized correlation of $\mathbf{k}(\delta)$ and $\mathbf{\Phi}(\mathbf{w})$ is used to define the external force $\vec{F}^e_{e_j}$ acting on e_j. Note that superscripts e, d, and p denote the kind of force (external, deformation, and pressure) whereas the subscript denotes where the force acts on (Fig. 2). The $\mathbf{k}(\delta)$ has to be chosen appropriately for every application of the model. Image influences should act along $\vec{n}_j$ [16] resulting in:

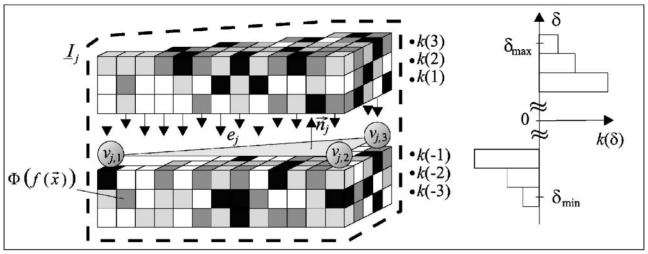

Fig. 3. Triangular edgel e_j of a 3D contour with prismatic image subset $\underline{I}_j$ and the considered image potentials $\Phi(f(\vec{x}))$.

$$\vec{F}^e_{e_j} = -\vec{n}_j \sum_{\vec{x}_e \in e_j} \sum_{\delta=\delta_{\min}}^{\delta_{\max}} \mathbf{k}(\delta) \cdot \mathbf{\Phi}\big(f(\vec{x}_e + \delta \cdot \vec{n}_j)\big). \quad (4)$$

All potentials in $\underline{I}_j$ are weighted according to $\mathbf{k}(\delta)$ dependent on their distance δ to the edgel (Fig. 3). The edge template is applied normal to e_j in all points $\vec{x}_e$ belonging to e_j. Hence, $\underline{I}_j$ becomes the kernel of the correlative filtering of image potentials with the edge template. In the physical analogy, forces always act along the gradient of potentials. In our extension, the computation of the external force is generalized and, hence, any correlative filter can be applied. The free choice of $\mathbf{k}(\delta)$ allows the search for significant gradients in the image as well as the region-based detection of objects. If all entries in $\mathbf{k}(\delta)$ are nonnegative, image influences are computed with region-based external forces [17]. If $\mathbf{k}(\delta)$ contains positive and negative entries, gradient information is evaluated. For example, $\mathbf{k}(\delta)$ is defined as a Gaussian derivative when referring to Canny's approach on edge detection [18]. Altogether, $\mathbf{k}(\delta)$ is a filter with the size adaptable to the scale of relevant gradients. The entries are adaptable to region- or gradient-based segmentation procedures and the direction is automatically adjusted by the contour itself. Therefore, the model runs on the original image data avoiding a priori filtering, which is required in many other approaches.

For edgels close to the border of the image domain, a shift along $\vec{n}_j$ might result in positions that are not contained within the image. For all these positions, the same padding value $\mathbf{w}_{\text{pad}}$ is assigned.

The resulting force $\vec{F}^e_{e_j}$ is then distributed onto the supporting vertices. This transfer additionally regards the location of image elements acting on the edgel. Vertices carry a higher portion of image influences that result from near image elements. In the mechanical analogy, this corresponds to an equality of forces as well as turning moments of image influences and supporting vertices [19]. Assuming that elements at position $\vec{x}_e$ of e_j are written as

$$\vec{x}_e = \sum_{l=1}^{d_I} \kappa_{j,l} \cdot \vec{v}_{j,l}, \quad (5)$$

where l is the index of supporting vertices of e_j so that $v_i = v_{j,l}$, the $\kappa_{j,l} \in [0, 1]$ are close to 1 if $\vec{x}_e$ is close to $v_{j,l}$ and the $\kappa_{j,l}$ decrease with increasing distance to $v_{j,l}$. The image influence $\vec{F}^e_{v_i}$ on a vertex v_i is therefore computed to:

$$\vec{F}^e_{v_i} = -\sum_{e_j \in \underline{U}^e_i} \vec{n}_j \sum_{\vec{x}_e \in e_j} \sum_{\delta=\delta_{\min}}^{\delta_{\max}} \mathbf{k}(\delta) \cdot \mathbf{\Phi}\big(f(\vec{x}_e + \delta \cdot \vec{n}_j)\big) \cdot \kappa_{j,l}. \quad (6)$$

Each v_i takes a share of the load of all external influences of the supported $e_j \in \underline{U}^e_i$.

2.2.2 Deformation Force

Laplacian smoothing [20] is applied to regularize the discrete contour. The mean curvature is reduced by shifting vertices normal to the contour and a uniform vertex spacing is achieved by a movement in tangent direction. Both effects are caused by a local deformation force $\vec{F}^d_{v_i}$ for each vertex in the mesh (Fig. 2).

$$\vec{F}^d_{v_i} = \frac{s_d}{|\underline{U}^v_i(\Delta)|} \sum_{v_k \in \underline{U}^v_i(\Delta)} \vec{v_k} - \vec{v_i}. \quad (7)$$

Every vertex is pulled to the barycenter of the vertices in its vicinity with adjustable strength s_d. Obtained from further experiments, we found $\Delta = 1$ to be sufficient. In other words, only 1-ring neighbors are considered. Then, (7) resembles a formulation of the scale-dependent umbrella-operator [21] that is independent of the dimension of image domain. However, the Laplacian is reduced without computing its absolute value. For image sequences, the reduction of the second-order derivative corresponds to an implicit velocity estimation [22] with finite forces applied to inert masses.

2.2.3 Searching Tendency

The tendency of the edgels to search for relevant image information originates from the balloon force, which has been introduced for active contour models by Cohen and Cohen [8]. This balloon force is computed for every e_j:

$$\vec{F}^p_{e_j} = p \cdot |e_j| \cdot \vec{n}_j. \quad (8)$$

It is directed along $\vec{n}_j$ and proportional to the size $|e_j|$ and the applied pressure p. Positive and negative pressures result in a searching tendency toward the outside and inside, respectively. Since the pressure force acts on the edgels, it has to be transferred to the supporting vertices:

$$\vec{F}^p_{v_i} = \frac{1}{d_I} \sum_{e_j \in \underline{U}^e_i} \vec{F}^p_{e_j}. \quad (9)$$

Every vertex carries the d_Ith part of all supported edgels (Fig. 2).

2.3 Iterative Segmentation

The model is used in an iterative segmentation algorithm to detect relevant objects in the image. In this iteration, vertices move until they contact structures or borders matching the description coded by $\mathbf{\Phi}(\mathbf{w})$ and $\mathbf{k}(\delta)$. The model starts from a given initialization $\mathcal{C}_0$, then moves vertices according to the computed forces and—if necessary—performs a local resampling and an automated adaptation of the contour's topology.

2.3.1 Initialization

Depending on positive or negative pressure, $\mathcal{C}_0$ is either a small seed that is inflated or a large frame at the image border that is shrunk. These initial contours are created with a fixed number of edgels and vertices. For a seed contour, fixed simplex meshes with vertices at positions $|\vec{v}_i - \vec{c}_0| = R_0$ are once created and, then, used with application-specific centers $\vec{c}_0$ and radii R_0. An initial 2D contour for segmentations with negative pressure consists of four vertices at the corners and four edgels at the borders of the image. For $d_I = 3$, vertices are positioned at the eight corners of the cubic image domain and each of the six sides is represented by two triangular edgels. In four dimensions, the corresponding structure consists of 16 initial vertices.

2.3.2 Movement of Vertices

In each iteration, all forces $\vec{F}_{v_i}$ are computed at first. Each individual vertex is explicitly shifted as reaction to the forces acting on it. This scheme requires no matrix inversion of any kind. Edgels move accordingly because they are attached to their supporting vertices. The transformation of forces into movements requires a normalization. The absolute values of forces depend on the size of edgels, whereas the movements of vertices should not. For every v_i, the supported size $|\underline{U}^e_i|$ is determined to specify the number of image elements in the part of the contour supported by v_i:

$$|\underline{U}^e_i| := \frac{1}{d_I} \sum_{e_j \in \underline{U}^e_i} |e_j|. \quad (10)$$

This number is used to normalize $\vec{F}^e_{v_i}$ and $\vec{F}^p_{v_i}$. To normalize $\vec{F}^d_{v_i}$, the average sampling size $\overline{L}$ is used. This is the average length of routes in the mesh. Thereafter, the normalized sum of forces acting on the vertices is determined:

$$\vec{F}^\Sigma_{v_i} = \frac{1}{\overline{L}} \vec{F}^d_{v_i} + \frac{1}{|\underline{U}^e_i|} \left(\vec{F}^p_{v_i} + \vec{F}^e_{v_i}\right). \quad (11)$$

Then, the vertices are moved according to $\vec{F}^\Sigma_{v_i}$. We model vertices without mass or momentum of inertia. For a finite movement, a damping force is needed. In the model, vertices experience liquid friction with a resulting force proportional to their velocity and directed against their movement. In the finite approximation defined by the steps $\Delta\mathcal{I}$ of the iteration, this results in a movement proportional to the normalized sum of forces:

$$\frac{\Delta\vec{v}_i}{\Delta\mathcal{I}} = \frac{1}{\beta} \vec{F}^\Sigma_{v_i}, \quad (12)$$

where the mechanical parameter β is the fixed coefficient of liquid friction if the velocity of vertices is approximated by $\frac{\Delta\vec{v}_i}{\Delta\mathcal{I}}$. Shifts $\Delta\vec{v}_i$ are not necessarily integer. Therefore, noninteger positions $\vec{v}_i$ are used such that vertices may lay at subpixel positions.

Instabilities that might arise in the explicit scheme are handled by limits on the value range of parameters for all influences. The bound $|p|/\beta \leq |\delta_{max} - \delta_{min}|$ ensures that no image information is disregarded due to a too strong searching tendency, the same bound applies to the maximal strength of image influences. Additionally, $s_d/\beta \leq 0.5$ prevents oscillations of the contour due to strong explicit smoothing. Some further heuristic extensions improve the stability of the contour at relevant structures in the image [23], [24]. The following rules are applied:

- If it is not directed against the pressure force, i.e., $\vec{F}^e_{v_i} \cdot \vec{F}^p_{v_i} > 0$, the external force is set to zero before v_i

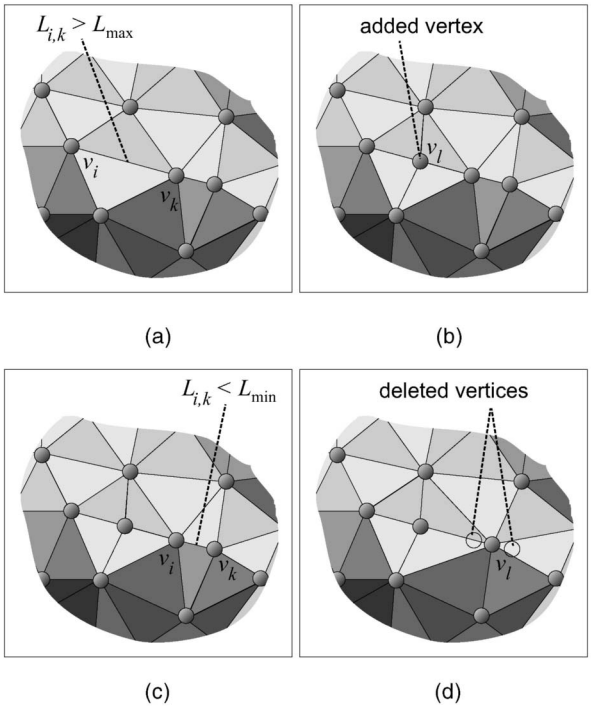

Fig. 4. The resampling process is exemplified for a 3D simplex mesh. Cartesian distances of neighboring vertices above $L_{\max}$ (a) result in the creation of a new vertex (b). If a vertex distance is below $L_{\min}$ (c), two vertices are merged to one (d).

is moved. Otherwise, local artifacts in the vicinity of the object's border could increase the danger of the contour breaking out due to the balloon force.

- If the external force is directed against the balloon force, the strength $|\vec{F}^e_{v_i}|$ is bounded by $2|\vec{F}^p_{v_i}|$. This prevents oscillatory instability of the contour in the vicinity of strong image influences.

2.3.3 Local Resampling

The movement changes the size of all edgels. In every iteration, the need for local resampling is checked to ensure that the length of routes, i.e., the Cartesian distance $L_{i,k}$ of vertices v_i and v_k sharing one edgel, lies within the range $[L_{\min}, L_{\max}]$. Whereas the deformation force ensures the uniform distribution of vertex distances, the additional local resampling adds or removes vertices and edgels if a local change in the size of $\mathcal{C}$ size has occurred. Additionally, the specification of $L_{\min}$ and $L_{\max}$ is used to estimate the average route length to $\overline{L} \approx \frac{1}{2}(L_{\min} + L_{\max})$.

Resampling is an intense field of research for triangulated surfaces, but not all results are general enough to be applicable to polygons or tetrahedral representations. We use nonuniform resampling methods [13]. The valence of vertices; that is, the number of supported edgels, changes in three and four dimensions during resampling. The following rules are applicable for $d_I \in \{2, 3, 4\}$:

1. If $L_{i,k}$ is above $L_{\max}$ (Fig. 4a), a new vertex v_l is created at position $\frac{1}{2}(\vec{v}_i + \vec{v}_k)$ (Fig. 4b). All edgels containing v_i as well as v_k are duplicated. Then, v_i is exchanged by v_l and v_k by v_l in the original and the duplicated edgel, respectively. Different methods to determine $\vec{v}_l$ have been introduced [25], [26]. However, in our model, resampling is always followed by the movement of vertices. Hence, the simplest method to compute $\vec{v}_l$ is applicable.
2. If $L_{i,k}$ is below $L_{\min}$ (Fig. 4c), again a new vertex v_l is created at position $\frac{1}{2}(\vec{v}_i + \vec{v}_k)$. All edgels containing both v_i and v_k are deleted. For edgels containing either v_i or v_k, the vertex is replaced by v_l. Then, v_i and v_k are deleted as well (Fig. 4d). The deletion of vertices and edgels results in local changes of the contour that could be reduced by a more sophisticated determination of $\vec{v}_l$. However, this is of minor improtance because the changed edgels will be moved again in the next iteration step.

Edgels affected by these operations are checked again for the occurrence of too long or too short routes. In three and four dimensions, an arbitrary order of operations can result in numerically unstable edgels due to the occurrence of strongly acute angels. Hence, the following order is established:

1. If an edgel has a size $|e_j|$ near to zero or if the computation of $\vec{n}_j$ showed numerical instability, then e_j is deleted and replaced by a vertex v_k in its center of gravity. Edgels sharing more than one vertex with e_j are deleted as well. For edgels that share one vertex with e_j, this vertex is replaced by v_k.
2. If at least one route of a numerically stable edgel is above $L_{\max}$, then the longest route of this edgel is divided.
3. If no route of an edgel is above $L_{\max}$, but at least one route below $L_{\min}$, then the shortest route of the edgel is removed.

In each iteration, only small changes of the contour occur. For the local resampling, the order of resamplings is not globally optimized. After the creation of $\mathcal{C}_0$, an initial resampling is needed before the first iteration. Here, many resampling steps take place that might affect edgels more than once. Then, a sorted list of routes is created [27], [28]. The longest route of the mesh is divided if above $L_{\max}$. All new routes resulting from this resampling are then included into the sorted list until no route is above $L_{\max}$. Then, the shortest route is removed if below $L_{\min}$, affected routes are resorted until no route is below $L_{\min}$.

2.3.4 Consistency and Adaptation of Topology

The movement of vertices and the local resampling can introduce intersections of edgels [17]. A self-intersection of the contour is characterized by $e_j \cap e_k \neq \emptyset$ if e_j and e_k share no vertex. In this section, we present a general multistep procedure to rearrange simplex meshes so that intersections are removed and a consistent mesh with eventually changed topology is created. This scheme requires the most part of computation time during segmentation. Whenever time is constrained, the following steps may be executed only in every second or third iteration still yielding robust results.

Models for segmentation with the ability to change their topology according to image data have been presented among others by Malladi et al. in their front propagation model [29]. This early approach is bound to monotone inflation or deflation and, therefore, offers only a low

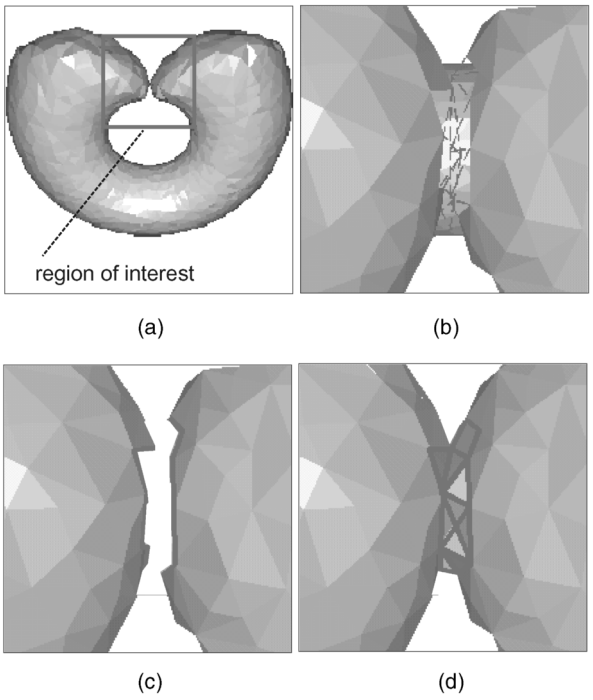

Fig. 5. Automated change of an object from an object homologous to a sphere (a) to an object homologous to a torus. Edgels that intersect after movement are deleted (b) and the resulting cut objects are extracted (c). Appropriate cut objects are closed with connecting edgels (d).

flexibility [21]. Lachaud presents a set of rules to change local vertex connectivities to adapt the topology of a triangulated 3D mesh. These results no longer rely on the "one pass—gone" monotony of front propagation methods. Our extension of topology adaptation methods handles simplex meshes in two, three, and four dimensions and employs a multistep procedure that is not restricted to the local change of vertex connectivities. For visualization, we first present the automated change for the segmentation of a torus in synthetic image data from a small seed contour that was homologous to the sphere. In Fig. 5a, the last consistent object representation prior to the intersection of edgels is shown. The method however, also applies to 2D and 4D representations.

In a first step, all edgels involved in intersections and all vertices that consequently do not support any edgels are removed (shown dotted in Fig. 5b). To detect intersections, the affine simplices of all edgels sharing no vertex are compared. An intersection is found if a resulting equation yields a valid solution for all κ_k. This step requires to solve nearly $|\underline{E}|^2$ sets of d_I linear equations with $2 \cdot (d_I - 1)$ degrees of freedom (compare 2). In order to reduce the resulting high-computational demands, the equation set is solved only if the bounding boxes of a pair of edgels overlap. This test for overlapping bounding boxes is fast end excludes a substantial fraction of all edgel pairs from the subsequent test of intersection. Furthermore, in the test, all routes of one edgel are intersected with the second edgel and vice versa, which eliminates the under-determination of above set of equations in three and four dimensions.

Thereafter, the resulting contour contains coherent successions of edgels with less than d_I neighbors. Successions of inconsistencies in the mesh are collected into different cut objects. A cut object consists of vertices in dimension d_I that are connected by affine simplices of dimension $d_I - 2$ (polygons drawn bold in Fig. 5c). All cut objects are stored in contours $\mathcal{S}_j$. If vertices exist that support edgels in more than one cut object $\mathcal{S}_j$, these vertices are deleted together with all supported edgels from the contour $\mathcal{C}$ and the cut objects $\mathcal{S}_j$ are updated.

Then, the $\mathcal{S}_j$ either have to be closed individually or combined with each other. To make the appropriate decision, a mean unity normal vector $\hat{\vec{n}}_j^{\mathcal{S}}$ is estimated for every cut object $\mathcal{S}_j$. Since a list of affine simplices of dimension $d_I - 2$ does not allow a straightforward computation of a normal vector of dimension d_I, the vector $\hat{\vec{n}}_j^{\mathcal{S}}$ is determined as the mean of the normal vectors for all edgels that would be created if the $e_j \in \mathcal{S}_j$ (of dimension $d_I - 1$) are combined with one vertex in the cut object's barycenter. A pair of cuts $\mathcal{S}_j$ and $\mathcal{S}_l$ are merged if

1. they are close to each other, i.e., the average of the Cartesian distances between all vertices in $\mathcal{S}_j$ and their nearest vertex in $\mathcal{S}_l$ is below $2 \cdot L_{\max}$,
2. they face each other, i.e., $\hat{\vec{n}}_j^{\mathcal{S}} \cdot \hat{\vec{n}}_l^{\mathcal{S}} \leq -\sqrt{3}/2$, if an angular deviation of 30° or less is tolerated, and
3. they are of similar size, i.e., the sum of sizes of all edgels in the cuts does not vary by a factor more than 1.25.

To perform a merge, the vertices from $\mathcal{S}_j$ and $\mathcal{S}_l$ are connected by new egdels that reconstruct a coherent simplex mesh (Fig. 5d). This creates a connecting tube-like structure and changes the topology of $\mathcal{C}$. To perform internal closures, edgels of dimension d_I are created that mutually connect the vertices from $\mathcal{S}_j$ to reconstruct the mesh. This procedure can result in separated objects in $\mathcal{C}$. Depending on the application, excessive objects either become part of the solution or they are deleted.

Whenever a merge or an internal closure was incorrect for the actual image data, the removed intersection will occur again in subsequent iterations. Then, the above decision is renewed with slightly changed input. Erroneous decisions may repeat for some iterations until the correct decision to change or conserve topology is made. Thereafter, the intersection will not appear anymore. This implicit self-assessment enables the model to flexibly change its topology according to objects contained in image data. In these operations, newly created edgels do not necessarily satisfy the sampling distance defined by $L_{\min}$ and $L_{\max}$. Therefore, closing of cut objects is always followed by local resampling.

In two dimensions, an intersection of two edgels (Fig. 6a) and their subsequent removal result in vertices that support only one or none edgel (Fig. 6b). Each of the four vertices that support one edgel is then a cut object of dimension $d_I - 2 = 0$. The approximated computation of the normal vector simply gives the direction of the single supported edgel. For this degenerated case, these four vertices are immediately connected by new edgels that do not intersect. Two different situations exist. Either a single consistent contour is created (Fig. 6c), or two unconnected contours occur (Fig. 6d) [11], [30]. This possibility enhances the variability and applicability of active contours [10]. Here, a random decision for the constellation of the two new edgels

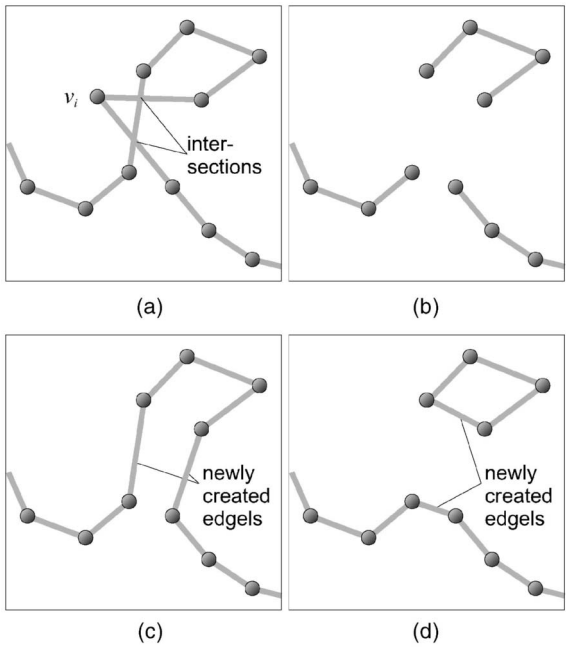

Fig. 6. Self-intersections of planar polygons (a) are solved by the deletion (b) and recreation of edgels. Either a single (c) or two consistent contours appear (d).

is made. As described above, the intersection reoccurs in the next iteration if the decision was wrong. Wrong decisions might succeed for a short period of time until the correct one is chosen once. Note that such simple heuristic rules are not applicable to 3D and 4D contours [31], [32].

A cut object of a triangulated surface in 3 dimensions does not lay in a plane. Cut objects $\mathcal{S}_j$ are sets of straight lines that connect 3D vertices. The normal vector of every cut object is estimated by the mean of the normal vectors of all triangles formed by two neighboring vertices of $\mathcal{S}_j$ and its center of gravity. If two cut objects $\mathcal{S}_j$ and $\mathcal{S}_l$ fulfill the criteria given above (Fig. 7a), their combination consists of the following steps that create a connecting triangle strip (Fig. 7b):

- The vertices in $\mathcal{S}_j$ and $\mathcal{S}_l$ with smallest Cartesian distance are identified.
- In an iteration, the smallest triangle connecting these vertices plus a neighboring vertex either on $\mathcal{S}_j$ or $\mathcal{S}_l$ is determined and created as a new edgel of $\mathcal{C}$. Accordingly, the chosen vertex on $\mathcal{S}_j$ or $\mathcal{S}_l$ is propagated to its neighbor.
- If this propagation has included all vertices of one of the cuts, triangles are inserted that connect the starting vertex of this cut with all still open routes of the other cut.

Cuts $\mathcal{S}$ without appropriate combination (Fig. 7c) are closed internally. Iteratively, the smallest possible triangle defined by three neighboring vertices in the cut is determined and an edgel is created accordingly. Then, the middle point of these three vertices is removed from the cut and a new line is inserted connecting the other two. This procedure takes place until the cut is closed (Fig. 7d).

For a 4D simplex mesh, a cut object is a triangulated surface, but the vertices of this surface are freely located in

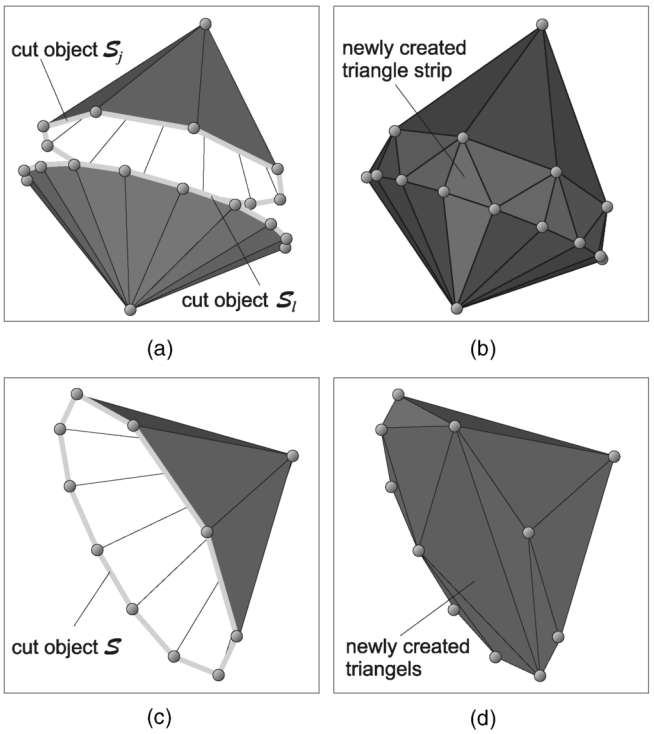

Fig. 7. Each of the two cut objects $\mathcal{S}_l$ and $\mathcal{S}_k$ shown in light gray forms the base of a cone-shaped object (a). Their combination results in a triangle strip connecting the objects (b). The closing of one single cut (c) results in a triangulated area (d).

the 4D image domain. For this case, we present the method to close single cut objects.

- For every v_i in the cut $\mathcal{S}$, the polygon defined by $U_1^v(v_i)$ is determined and the overall length of this polygon is calculated. The vertex with the shortest surrounding polygon is chosen.
- The chosen polygon is then treated according to the 3D case, where a single polygonal cut object was closed. Instead of triangular edgels, tetrahedral edgels are now created. The chosen vertex is always apex of the tetrahedron and the newly created triangle defines its base.
- The chosen vertex, together with the supported triangles is then removed from the cut object. The process takes place until only four vertices are left in $\mathcal{S}$, which are connected with a last tetrahedral edgel.

The operation to combine two cut objects works accordingly. Here, tetrahedrons are created where the apex vertex is positioned in one cut object and the triangular basis forms one element of the partner cut object.

2.3.5 Discontinuities and Boundary Conditions

In medical imaging, relevant structures are frequently imaged only in parts. For volumetric images or image sequences, objects might be cut in the first and last slice of the data set. Nevertheless, a segmentation must yield proper results in all parts of the image volume and in all slices. However, if an object is only partially imaged, two contradicting requirements must be satisfied. The border of the object is simultaneously determined by the aperture of data and by the cut structures that do not have a contour tangential to the image borders. Therefore, the contour

contains a sharp edge that is usually reduced by the deformation force yielding imprecise segmentations at the image border. To soften this conflict, vertices are allowed to occur outside the image domain within a tolerance range so that the border can be represented without strong bends. Of course, parts outside the image domain are discarded before subsequent analysis.

2.3.6 Stabilization and Convergence

For every vertex, the position of preceding iterations is stored. If the Cartesian distance between the actual position and the closest position in preceding iterations is below a threshold, the vertex is stabilized. For such vertices, influences are not calculated anymore. However, such vertices continue to contribute to the deformation force of neighboring vertices and therefore reduce the movement of the contour in their vicinity. This corresponds to an adaptive reduction of the contour's activity comparable to simulated annealing methods (e.g., [33], [34]). However, we do not impose a global reduction of the contour's activity, since this automatically results from the stability of local domains.

Stabilized vertices are reactivated whenever their neighborhood is resampled. Empirically, not all vertices reach a position yielding equality of forces [35]. Often, oscillatory instability of geometric deformable models prevents local convergence of vertices [11], [23], [24], which especially holds for region-based external forces. In our model, the positions occupied by vertices in some subsequent iterations are stored and regarded. This allows the detection of oscillating vertices which are subsequently stabilized.

3 Validation of the Model

Instances of the model in two, three, and four dimensions were implemented. Tests were performed to validate the abilities of the model in segmentation of image data. We present both a noncontextual validation that aims at giving reproducible quantitative measures describing the behavior of the model under variable known image characteristics, and a contextual validation to show different applications of the model for medical image analysis. The noncontextual tests were performed on synthetic image data and contextual tests show the segmentation of immunohistochemically stained micrographs, the volumetric segmentation of computed tomographies (CTs) of the spinal column, a segmentation of a 4D data set of the left ventricle from magnetic resonance images (MRIs), and the segmentation of laryngoscopic color image sequences showing the vocal cords.

3.1 Noncontextual Validation

According to Yu et al., the synthetic images consist of the metacircle or metasphere given by $r(\varphi) \leq \overline{r} + \Delta_r \sin(K_r \cdot \varphi)$ with rotationally symmetric 3D and 4D extensions where $\overline{r}$, Δ_r, and K_r denote the mean radius, the amplitude of the variation of the radius, and a constant for the frequency of radius variation, respectively [36]. The center of the metasphere is the center of the synthetic images and we used a fixed $K_r = 5$. The validation included the following synthetic objects:

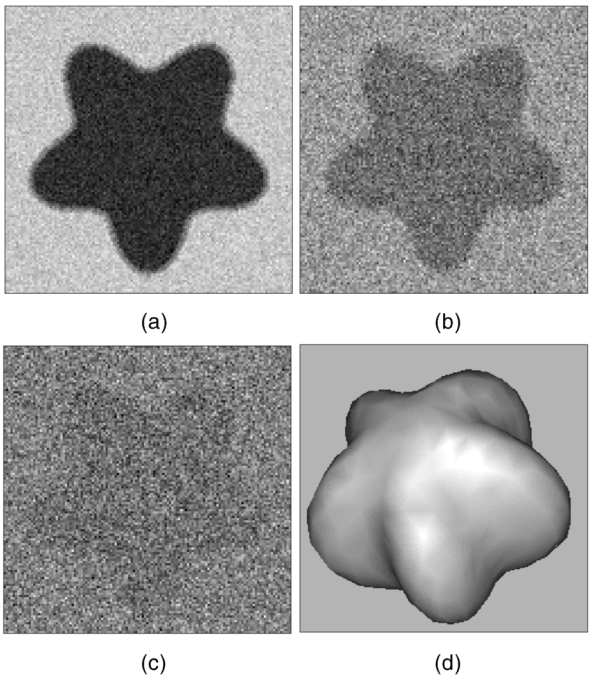

Fig. 8. Synthetic images used in the noncontextual validation. The SNR of the pixel values is 5, 1.67, and 0.5 in (a), (b), and (c), respectively. For visualization purposes, each image was individually optimized in contrast. The 3D object is visualized in (d).

- 2D images of size 128×128 pixels with the metacircle defined by $\overline{r} = 45$ and $\Delta_r = 10$.
- 3D volumetric data sets of size $64 \times 64 \times 64$ voxels with the metasphere define by $\overline{r} = 26$ and $\Delta_r = 5$.
- 4D spatio-temporal data sets of size $50 \times 50 \times 50 \times 50$ toxels with the 4D-metasphere defined by $\overline{r} = 18$ and $\Delta_r = 5$.

These templates were used to create images with synthetic Gaussian noise. The inside and outside of the object was filled with gray values from the normal distribution $\mu_{\text{in}} \pm \sigma$ and $\mu_{\text{out}} \pm \sigma$, respectively. To reduce the gradients normal to the border of the metasphere, which are usually inherent to synthetic images, the mean value was linearly interpolated over a fixed transition zone of 5 pixels width. Then, two different tests were performed:

- For fixed values $\mu_{\text{in}} = 150$ and $\mu_{\text{out}} = 160$, the additive noise was changed in the range $\sigma \in \{2, 4, \ldots, 20\}$ resulting in an amplitude-SNR of 5 (13.98dB) down to 0.5 (-6.02dB) (Fig. 8).
- For a fixed noise intensity of $\sigma = 10$, the background brightness was changed to $\mu_{\text{in}} \in \{140, 142, \ldots, 160\}$ with $\mu_{\text{out}} = \mu_{\text{in}} + 10$.

For each combination of parameters, 10 different images were created. Two different sets of parameters were created using the learning-from-examples paradigm [37]. During this process, an exemplary input image with known segmentation is used to reproducibly determine that set of parameters that results in a segmentation closest to the given reference [38]. The training was based on an exemplary 2D image characterized by $\mu_{\text{in}} = 150$, $\mu_{\text{out}} = 160$, and $\sigma = 10$. Table 1 summarizes the results for region and gradient-based images influences.

TABLE 1
Automatically Computed Parameters in the Noncontextual Validation

parameter	region-based setting	gradient-based setting
$L_{\min}$	5	5
$L_{\max}$	13	13
p	2	2
$\delta_{\min}$	0	-4
$\delta_{\max}$	2	4
$k(\delta)$	$\begin{cases} 1, & \delta_{\min} \le \delta \le \delta_{\max} \\ 0 & \text{else} \end{cases}$	$\begin{cases} -1, & \delta \in \{-4,-3\} \\ 1, & \delta \in \{3,4\} \\ 0 & \text{else} \end{cases}$
$\Phi(w)$	$0.27(w-154.1)$	$0.118w$
s_d	0.076	0.11

All segmentations were compared to the known standard. For the binary templates serving as reference, a binary contour image was created and then distance-transformed to give the Manhattan distance ($d_M = \sum \Delta x_i$, the sum of distances Δx_i in each coordinate direction) of each image element to the reference contour. For every vertex and the center of every edgel, the value in this distance image was read and averaged yielding the mean Manhattan distance $\overline{d}_M$.

In the resulting diagrams (Figs. 9 and 10), bullets and circles indicate the region-based and gradient-based segmentation, respectively. Every data point is the average of the distance measure for ten different synthetic images created with the same parameters.

For variable noise intensity, but known background brightness, region-based external influences result in smaller distances and, therefore, in a more precise delineation than gradient-based external forces. For region-based influences (black data points), the average distance enlarges with increasing noise intensity in two and three dimensions (Fig. 9). However, this effect is not observed in four dimensions. Here, additional images with the noise intensity $\sigma \in \{22, 24, \ldots, 30\}$ (SNR down to -9.54dB) were created and segmented. Even then, an increase in $\overline{d}_M$ was not seen. On the contrary, an optimum in precision was found for images with noise intensities similar to the training image used for the automated parameterization ($\sigma = 10$).

For gradient-based influences (unfilled circles), the delineation is in each case less precise compared to region-based influences. For 2D data, the increase of the

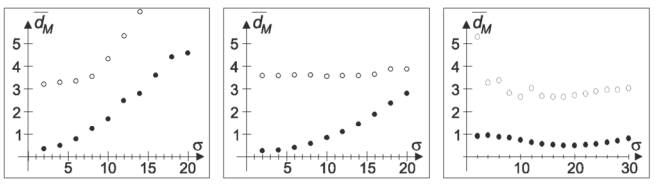

Fig. 9. Mean Manhattan distance $\overline{d}_M$ of segmentation and reference for region-based (bullet) and gradient-based (circle) image influences under varying noise intensity σ. Results are given for 2D, 3D, and 4D image domains in (a), (b), and (c), respectively.

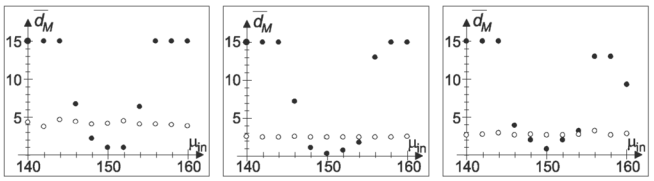

Fig. 10. Mean distance $\overline{d}_M$ of segmentation and reference for region-based (filled) and gradient-based (unfilled) image influences under varying background intensity μ_{in}. Results are given for 2D, 3D, and 4D image domains in parts (a), (b), and (c), respectively.

distance measure shows that segmentation completely fails for high levels of noise, i.e., $\sigma > |\mu_{\text{in}} - \mu_{\text{out}}|$. However, for 3D data, even though the delineation is imprecise with $3 \le \overline{d}_M \le 4$ for all tests, the automated delineation of the object is robust against noise. For 4D data, the segmentation fails for low noise intensities due to the unbalanced strong external forces. Again, an indistinct optimum for images similar to the training data is observed.

Results of an automated delineation are often given in dependency of the SNR. However, different distance- or overlap measures are used for different objects so that a direct comparison is nearly impossible. Methods that allow to either accept or reject delineations for varying images are not known yet. Exemplary, for different model-based segmentation methods, delineations were accepted for $\sigma \le 0.5|\mu_{\text{in}} - \mu_{\text{out}}|$ [39], [40], $\sigma \le 0.8|\mu_{\text{in}} - \mu_{\text{out}}|$ [41], and $\sigma \le 1.2|\mu_{\text{in}} - \mu_{\text{out}}|$ [42]. Gunn and Nixon additionally compared their model to the snake-model for which they report accepted segmentations for $\sigma \le 0.8|\mu_{\text{in}} - \mu_{\text{out}}|$. However, in most of these tests, no transition zone from the inside to the outside is modeled so that artificially high gradients occur in the images used for validation.

For acceptance, a threshold for $\overline{d}_M$ has to be selected that has major influence on the results. Despite the lack of standardized tests, a delineation is accepted if vertices are placed in a zone within the mean grey level transition. In order to give comparable results, we decided to accept segmentations with $\overline{d}_M \le 2$ and reject all others. For gradient-based influences, this results in no acceptable delineations. Note however, that our synthetic images include a linear transition zone instead of a binary border.

Delineations are much more robust against noise and far more precise for region-based influences. In 2D, our tests showed accepted segmentations for $\sigma \le 1.0|\mu_{\text{in}} - \mu_{\text{out}}|$, in 3D for $\sigma \le 1.6|\mu_{\text{in}} - \mu_{\text{out}}|$, and in 4D, for $\sigma \le 3|\mu_{\text{in}} - \mu_{\text{out}}|$. With increased dimension of image space, our formulation of image influences evaluates an increased number of image values, therefore giving an increased robustness against additive noise. Note that the model here was used with an automated and fully reproducible parameterization, but without any manual optimization.

Region-based influences depend on the known background intensity whereas gradient-based influences do not. Therefore, additional tests with varying μ_{in} were performed. If no image-adaptive parameterization (e.g., [43]) is used, segmentation might fail using region-based influences. However, fixed parameters are important for

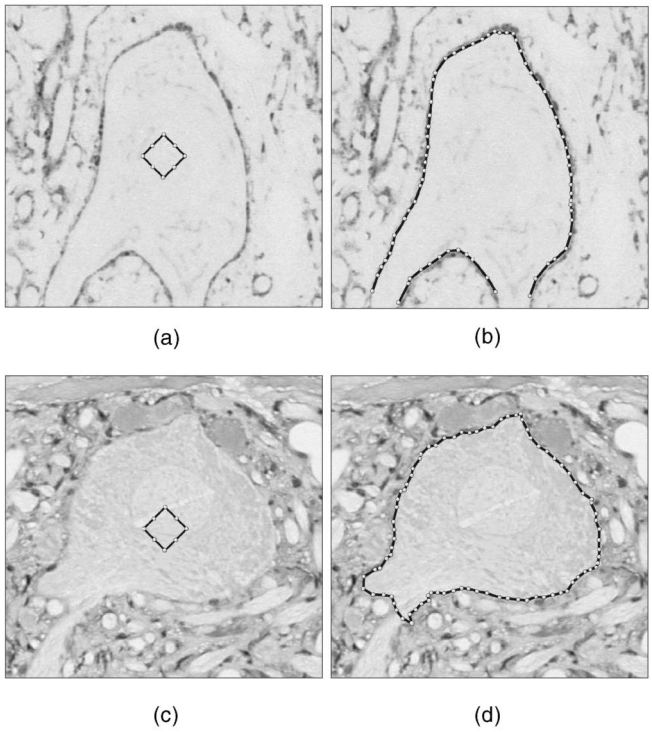

Fig. 11. Two examples of immunohistochemically stained micrographs of motoneurons with the initial contour $\mathcal{C}_0$ (a), (c) and the segmentation results of the 2D instance of the model (b), (d).

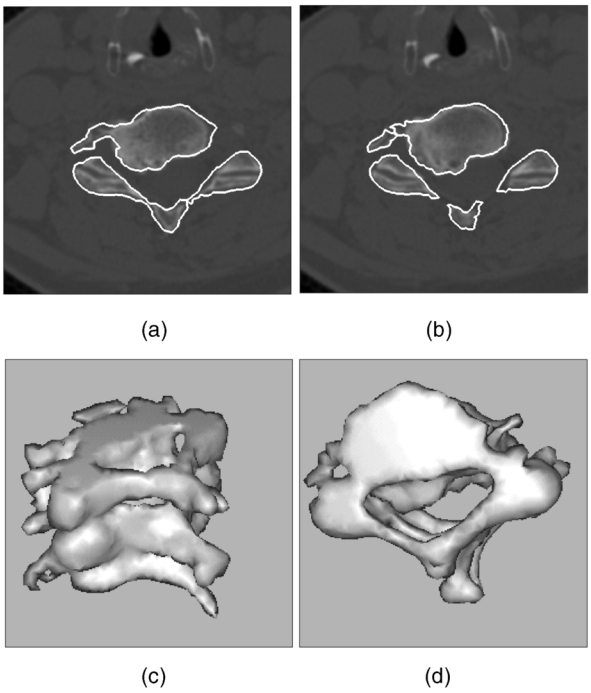

Fig. 12. Exemplary slice from a volumetric CT data set with clearly visible herniated disc with overlaid result contours (a), (b) and the volumetric segmentation result (c), (d). Note that the contours in (a) and (b) are obtained slicing the volumetric object representation.

the reproducibility of our tests. To be more precise, region-based influences result in a precise delineation if and only if the assumption of the background brightness learned in the training from examples is satisfied. In our tests, segmentations were accepted for $\mu_{\text{in}} \in [148, 152]$ (Fig. 10). In contrast, gradient-based influences result in less precise delineations, but the model is then independent of the image's background intensity.

3.2 Contextual Validation

The model was exemplary applied to medical images with $d_I \in \{2, 3, 4\}$ including single and multichannel value ranges. In each case, the segmentations were validated visually and using so-called silver standard images [44], i.e., synthetic images with realistic Fourier-synthesized textures, which are created for real objects with a priori known contours.

For the 2D validation, the model was used to segment arbitrarily shaped immunohistochemically stained motoneurons from the spinal cord of adult rats [45]. The task was to find the cell membrane applying internal pressure with gradient-based influences (Fig. 11). The segmentation of 1,825 cells was manually rated. In 93 percent, the automated delineation was accepted. The silver-standard tests were performed for four different shapes of cells with 50 exemplary images each. They result in acceptable segmentations for 98 percent of the 200 images with a mean Cartesian distance to the known standard of $\overline{d}_C = 1.67$.

As an exemplary volumetric segmentation, computed tomographies of the spinal column following herniation of an intervertebral disc (Figs. 12a and 12b) were segmented to support surgery planning. Region-based influences were used in a shrinking segmentation initiated at the border of the data set. All changes in topology necessary to segment the complex geometry of vertebrae were automatically accomplished by the model (Figs. 12c and 12d and the contours overlaid in Figs. 12a and 12b). The delineation of the contour of bony structures in three data sets was visually accepted for all cases. A silver-standard validation including 10 different synthetic images for every data set resulted in a mean distance of $\overline{d}_C = 2.9$. A large Hausdorff distance, that is, the maximal distance that was seen, indicated in this case that not all appendages of the vertebrae were automatically segmented. However, this does not influence surgery planning.

A spatio-temporal 4D magnetic resonance imaging sequence (True FISP, Figs. 13a and 13b) of the left ventricular cavity was segmented to present the 4D-instance of the model. The segmentation using internal pressure and region-based influences results after 85 iterations in a contour consisting of 921 vertices with 5,146 connecting edgels. The cut of this contour with hyperplanes defined by $t = t_c$ yields a contour of the left ventricle and additional information of the local movement of the heart wall (Figs. 13c and 13d). This local movement can be automatically extracted from the 4D normal vectors of the edgels. A silver-standard validation including 10 synthetic data sets gave a mean distance of $\overline{d}_C = 1.16$ compared to the given standard and therefore showed a reproducible delineation of the 4D contour.

Laryngoscopic color video sequences were segmented to validate the model on multichannel images. The sequences show the abduction of the arytenoid cartilage during inspiration. The glottal area framed by the vocal cords was delineated. The 3D segmentation (Fig. 1a) allows to compute the local movement of the vocal folds (Figs. 1b, 1c, 1d, and 1e) and gives important diagnostic

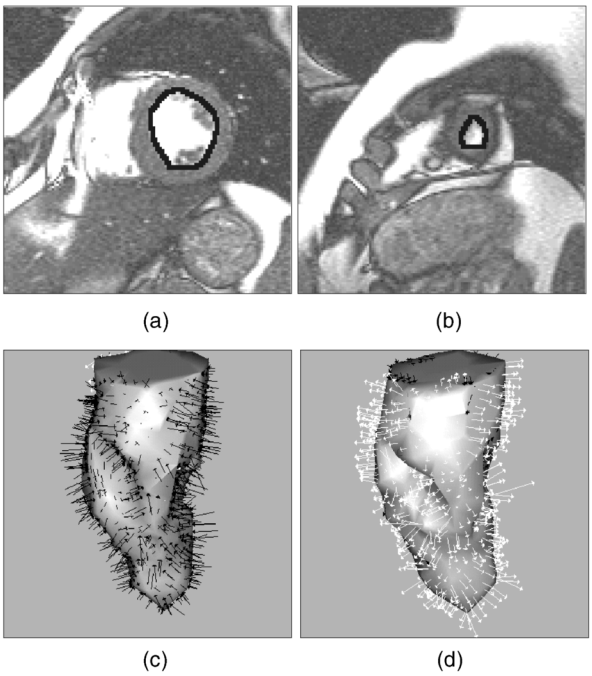

Fig. 13. Exemplary slices from a spatio-temporal MRI data set of the heart (a), (b) and the segmentation of the left ventricular cavity including local wall motion estimation ((c) midsystolic, and (d) middiastolic). The 3D results were obtained slicing the 4D object representation, and the 2D contours overlaid in (a) and (b) by a subsequent slicing of the 3D results.

information. The model was applied using internal pressure and gradient-based influences. Validation took place on three different sequences. Each result was visually accepted. A silver-standard validation with ten synthetic datasets for one sequence gave a mean distance of $\overline{d}_C = 3.26$. The less precise delineation is caused by gradient-based influences needed due to changing background intensities. However, Hausdorff errors of 14 pixels were found. These errors occur at the beginning of the opening sequence, because a narrow glottal area cannot be represented by a triangulated surface with internal reduction of the second-order derivatives.

On a 500MHz PC running Linux, the C++-implementation compiled without optimization, took 2s, 30s, and 120s for the segmentation of 2D micrographs, 3D CTs, and 4D MRIs, respectively.

4 Discussion

Active contour models are frequently reported. Our model integrates superior features into a consistent framework and contributes this synthesis and—where necessary—extensions of the features. Namely, we provide a model

- with spatio-temporal coherence in two, three, and four dimensions,
- that is suitable for multichannel data,
- operates on the original data without preprocessing,
- uses either gradient- or region-based image influences in a general and robust formulation,
- employs Laplacian smoothing to avoid the inversion of the rigidity matrix,
- provides automated topology adaption,
- and was proven to be applicable in contextual and noncontextual validation.

4.1 Spatio-Temporal Coherence

Most often, slicewise segmentation is used for image domains of more than two dimensions. Compared to object representations used in propagational methods, the triangulated representation for 2D+t image sequences and the coherent 4D tetrahedral representation for 3D+t data are, to the best of our knowledge, presented here for the first time and offer spatio-temporal coherence that is missing in many slicewise propagation methods. Yudong and Pelc have introduced a triangulated surface that describes vertex positions with a Fourier series [46]. Comprehensive considerations on tetrahedral meshes are given by Saha et al. for static volumetric objects [47], but tetrahedrons in 4D image data with a defined orientation in space and time were not mentioned so far. Spatio-temporal coherence in three and four dimensions generally increases the robustness of active contour models [48].

4.2 Multichannel Values

Usually, multichannel images are reduced to scalar gradient images before model-based segmentation. However, the computation of image influences on one or more independent color channels increases the robustness of the model for the detection of objects [49], [50]. So far, only few models allow a parallel assessment of different color channels [51].

4.3 Preprocessing

For the well-known snake-model [52] and many derivates, the image is converted into a scalar gradient image and the information about the direction of gradients is lost. Such global filters have the disadvantage of further weakening already weak image information [53], whereas a filter with local adaptive directions improves edge detection [54]. The correlative image filtering presented here is performed normal to the direction of the contour and therefore normal to relevant edges. This enables our model to operate on raw image data.

4.4 Adaptable and Robust Image Influences

In 1992, Cohen et al. presented a model where image information is taken into account over a whole deformable surface when segmenting 3D images [56]. Their discussion of respective advantages was repeated often by other authors, but it is ignored in many of the published formalisms. However, implementations that evaluate image information only at the current vertex position or on lines normal to the contour are likely to miss important image features. The computation of image influences presented here consistently evaluates all image elements in the neighborhood of edgels in every iteration, for any image dimension. The increased robustness against additive noise was proven by noncontextual and contextual tests [15]. Furthermore, including the equality of turning moments improves the local delineation quality and stabilization of edgels [19]. Such a robust computation is required to use the beneficial pressure force, which is otherwise difficult to balance against image influences [40], [55]. Note that our computation of influences allows automated balancing, even if the parameters are trained from only a small number of examples [38].

4.5 Laplacian Smoothing

The complete mechanical solution to compute deformation influences would incorporate the inversion of the rigidity matrix of the contour [9], [57]. Instead, the Laplacian smoothing is a rather simple method to compute deformation forces that only uses 1-ring neighborhoods decoupled from the whole mesh. The method yields two desired properties, that is, the reduction of the local second-order derivative and the regularization of the sampling avoiding computational overhead and has been proven suitable in application. Improvements of Laplacian smoothing, e.g., presented by Desbrun [58] offer no valuable advantages in iterative segmentation procedures where each step of the iteration includes one step of the smoothing computation.

4.6 Adaptation of Topology

So far, only few models deal with self-intersections of the mesh that can occur for many active contours. For some models, the occurrence of intersections is specified as problem, but rules how to prevent or solve them are not given (e.g., [59], [60]). Lachaud and Montanvert [21], Ngoi and Jia [61], or Gibson et al. [62] define simplified rules for a detection of intersections. However, approximative solutions of the self-intersection problem do not offer reliable adaptations of topology. Hence, the automated handling of self-intersections is an important feature of our model. Compared to front propagation approaches such as used by Malladi et al. [29] and the similar topology-adaptation presented by McInerney and Terzopoulos [10], a monotonous "one pass—gone" movement of the contour is no longer required. The approach of Juarez et al. [63] relies on a fixed number of vertices and their distance, which does not offer enough flexibility. Lachaud and Montanvert also present a topology adaptation for triangulated surfaces that deletes and recreates edgels according to a set of rules when intersections are detected. As a novelty, our model describes cut objects resulting from self-intersections in own data structures that are not bound to 1-ring neighborhoods in a triangulated surface. The cut objects are applicable for object representations in two, three, and four dimensions and handle large subparts of a contour in self-intersection. The adaptive rules for the closing of such inconsistencies, which are evaluated by the image data itself further increase robustness and flexibility. Note that the handling of bifurcations or similar well-known problems of topology only arise when a contour is handled in slices. Within the coherent 3D or 4D object representations used here, a special handling is not required.

4.7 Robustness and Applicability

Our noncontextual tests showed the ability to delineate objects under strong additive noise. Accepted segmentations with as high noise intensities as $\sigma = 1.6|\mu_{\text{in}} - \mu_{\text{out}}|$ for 3D images, and $\sigma = 3.0|\mu_{\text{in}} - \mu_{\text{out}}|$ for 4D images, have not been reported so far. This robustness against additive noise results from the new formulation of image influences. A large number of image elements is considered within 3D and 4D image subsets. Additionally, the equality of turning moments orients every linear edgel according to the object's border and results in precise locations of vertices. The contextual validation proved the applicability of the model to medical images that were chosen to be as diverse and various as possible. However, we have not compared our model to other possible segmentation models because, generally, applicable methods that segment 3D or 4D image data without manual initialization and manual choice of parameters are not readily available.

In conclusion, the formulation of the discrete segmentation model described here arises promising potential for the implementation of a general applicable model-based segmentation method.

REFERENCES

[1] P. Bamford and B. Lovell, "Bayesian Analysis of Cell Nucleus Segmentation by a Viterbi Search Based Active Contour," *Proc. Int'l Conf. Pattern Recognition,* pp. 133-135, 1998.

[2] C.M. Chen, H.H.-S. Lu, and Y.C. Lin, "A New Ultrasound Image Segmentation Algorithm Based on an Early Vision Model and Discrete Snake Model," *Proc. Int'l Soc. Optical Eng.,* vol. 3338, pp. 959-970, 1998.

[3] J.M. Pardo, D. Cabello, and J. Heras, "A Snake for Model-Based Segmentation of Biomedical Images," *Pattern Recognition Letters,* vol. 18, no. 14, pp. 1529-1538, 1997.

[4] V. Chalana, J.A. Hodgdon, and D.R. Haynor, "Unified Data Structures in a Software Environment for Medical Image Segmentation," *Proc. Int'l Soc. Optical Eng.,* vol. 3338, pp. 947-958, 1998.

[5] D.R. Haynor, "Performance Evalutation of Image Processing Algorithms in Medicine: A Clinical Perspective," *Proc. Int'l Soc. Optical Eng. (SPIE),* vol. 3979, p. 18, 2000.

[6] J.A. Lynch, S. Zaim, J. Zhao, A. Stork, C.G. Peterfy, and H.K. Genant, "Cartilage Segmentation of 3D MRI Scans of the Osteoarthritic Knee Combining User Knowledge and Active Contours," *Proc. Int'l Soc. Optical Eng.,* vol. 3979, pp. 925-935, 2000.

[7] P. Li, B.D. Corner, and S. Paquette, "Segmenting 3D Surface Scan Data of the Human Body by 2D Projection," *Proc. Int'l Soc. Optical Eng. (SPIE),* vol. 3958, pp. 172-177, 2000.

[8] L.D. Cohen and I. Cohen, "Finite-Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 11, pp. 1131-1147, Nov. 1993.

[9] T. McInerney and D. Terzopoulos, "A Dynamic Finite Element Surface Model for Segmentation and Tracking in Multidimensional Medical Images with Application to Cardiac 4D Image Analysis," *Computerized Medical Imaging and Graphics,* vol. 19, no. 1, pp. 69-83, 1995.

[10] T. McInerney and D. Terzopoulos, "Topology Adaptive Deformable Surfaces for Medical Image Volume Segmentation," *IEEE Trans. Medical Imaging,* vol. 18, no. 10, pp. 840-850, 1999.

[11] M.S. Horritt, "A Statistical Active Contour Model for SAR Image Segmentation," *Image and Vision Computing,* vol. 17, nos. 3-4, pp. 213-224, 1999.

[12] P.E. Undrill, K. Delibasis, and C.G. Cameron, "An Application of Genetic Algorithms to Geometric Model-Guided Interpretation of Brain Anatomy," *Pattern Recognition,* vol. 30, no. 2, pp. 217-227, 1997.

[13] L. Velho, L.H. de Figueiredo, and J. Gomes, "Hierarchical Generalized Triangle Strips," *Visual Computer,* vol. 15, no. 1, pp. 21-35, 1999.

[14] L.P. Kobbelt, "Discrete Fairing and Variational Subdivision for Freeform Surface Design," *Visual Computer,* vol. 16, nos. 3-4, pp. 142-158, 2000.

[15] T.M. Lehmann, J. Bredno, and K. Spitzer, "On the Design of Active Contour Models for Medical Image Segmentation—A Scheme for Classification and Construction," *Methods of Information in Medicine,* pending publication.

[16] S. Lobregt and M.A. Viergever, "A Discrete Dynamic Contour Model," *IEEE Trans. Medical Imaging,* vol. 14, no. 1, pp. 12-24, 1995.

[17] J. Ivins and J. Porrill, "Constrained Active Region Models for Fast Tracking in Color Image Sequences," *Computer Vision and Image Understanding,* vol. 72, no. 1, pp. 54-71, 1998.

[18] J.F. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, pp. 679-698, 1986.

[19] V. Metzler, J. Bredno, T.M. Lehmann, and K. Spitzer, "A Deformable Membrane for the Segmentation of Cytological Samples," *Proc. Int'l Soc. Optical Eng.,* vol. 3338, pp. 1246-1257, 1998.

[20] H. Delingette and J. Montangnat, "Shape and Tolopogy Constraints on Parametric Active Contours," *Computer Vision and Image Understanding,* vol. 83, no. 2, pp. 140-171, 2001.
[21] J. Lachaud and O. Montanvert, "Deformable Meshes with Automated Topology Changes for Coarse-to-Fine 3D Surface Extraction," *Medical Image Analysis,* vol. 3, no. 2, pp. 187-207, 1999.
[22] N. Peterfreund, "The Velocity Snake: Deformable Contour for Tracking in Spatio-Velocity Space," *Computer Vision and Image Understanding,* vol. 73, no. 3, pp. 346-356, 1999.
[23] F. Leymarie and M.D. Levine, "Tracking Deformable Objects in the Plane Using an Active Contour Model," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 6, pp. 617-634, June 1993.
[24] A.J. Bulpitt and N.D. Efford, "An Efficient 3D Deformable Model with a Self-Optimising Mesh," *Image and Vision Computing,* vol. 14, no. 8, pp. 573-580, 1996.
[25] A.D.C. Smith, D.L.G. Hill, D.J. Hawkes, C.E. Polkey, and T.C.S. Cox, "Surface Simplification for Shape Measurement-Application to the Human Brain," *Proc. Int'l Soc. Optical Eng.,* vol. 3338, pp. 652-662, 1998.
[26] R. Sitnik and M. Kujawinska, "Opto-Numerical Methods of Data Acquisition for Computer Graphics and Animation Systems," *Proc. Int'l Soc. Optical Eng.,* vol. 3958, pp. 36-43, 2000.
[27] A.E. Johnson and M. Hebert, "Control of Polygonal Mesh Resolution for 3-D Computer Vision," *Graphical Models and Image Processing,* vol. 60, no. 4, pp. 261-285, 1998.
[28] D. Kim, J. Kim, and H.S. Ko, "Unification of Distance and Volume Optimization in Surface Simplification," *Graphical Models and Image Processing,* vol. 61, no. 6, pp. 363-367, 1999.
[29] R. Malladi, J.A. Sethian, and B.C. Vemuri, "Shape Modeling with Front Propagation: A Level Set Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 2, pp. 158-175, Feb. 1995.
[30] R. Ronfard, "Region-Based Strategies for Active Contour Models," *Int'l J. Computer Vision,* vol. 13, no. 2, pp. 229-251, 1994.
[31] G.T. Herman and D.F. Robinson, "Digital Boundary Tracking," *Pattern Analysis and Applications,* vol. 1, no. 1, pp. 2-17, 1998.
[32] M. Alexa, "Merging Polyhedral Shapes with Scattered Features," *Visual Computer,* vol. 16, no. 1, pp. 26-37, 2000.
[33] R.P. Grzeszczuk, R.P. Levin, and D.N. Levin, "Brownian Strings: Segmenting Images with Stochastically Deformable Contours," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 10, pp. 1100-1114, Oct. 1997.
[34] D. Rueckert, P. Burger, S.M. Forbat, R.D. Mohiaddin, and G.Z. Yang, "Automatic Tracking of the Aorta in Cardiovascular MR Images Using Deformable Models," *IEEE Trans. Medical Imaging,* vol. 16, no. 5, pp. 581-590, 1997.
[35] C.L. Lam and S.Y. Yuen, "An Unbiased Active Contour Algorithm for Object Tracking," *Pattern Recognition Letters,* vol. 19, nos. 5-6, pp. 491-498, 1998.
[36] D.N. Yu, C. Xu, M.E. Rettmann, D.L. Pham, and J.L. Prince, "Quantitative Validation of a Deformable Cortical Surface Model," *Proc. Int'l Soc. Optical Eng.,* vol. 3979, pp. 1593-1604, 2000.
[37] S. Cagnoni, A.B. Dobrzeniecki, R. Poli, and J.C. Yanch, "Genetic Algorithm-Based Interactive Segmentation of 3D Medical Images," *Image and Vision Computing,* vol. 17, pp. 881-895, 1999.
[38] J. Bredno, T.M. Lehmann, and K. Spitzer, "Automatic Parameter Setting for Balloon Models," *Proc. Int'l Soc. Optical Eng.,* vol. 3979, pp. 1185-1194, 2000.
[39] C. Kauffmann, B. Godbout, and J.A. de Guise, "Simplified Active Contour Model Applied to Bone Structure Segmentation in Digital Radiographs," *Proc. Int'l Soc. Optical Eng.,* vol. 3338, pp. 663-672, 1998.
[40] J. Liang, T. McInerney, and D. Terzopoulos, "Interactive Medical Image Segmentation with United Snakes," *Lecture Notes in Computer Sciences,* vol. 1679, pp. 116-127, 1999.
[41] D. Fritsch, S. Pizer, L. Yu, V. Johnson, and E. Chaney, "Segmentation of Medical Image Objects Using Deformable Shape Loci," *Lecture Notes in Computer Sciences,* vol. 1230, pp. 127-140, 1997.
[42] S.R. Gunn and M.S. Nixon, "Improving Snake Performance Via a Dual Active Contour," *Lecture Notes in Computer Sciences,* vol. 970, pp. 600-605, 1995.
[43] T.M. Lehmann, J. Bredno, and K. Spitzer, "Texture-Adaptice Active Contour Models," *Lecture Notes in Computer Sciences,* vol. 2013, pp. 387-396, 2001.
[44] T.M. Lehmann, J. Bredno, and K. Spitzer, "Silver Standards Obtained from Fourier-Based Texture Synthesis to Evaluate Segmentation Procedures," *Proc. Int'l Soc. Optical Eng.,* vol. 4322, pp. 214-225, 2001.
[45] T.M. Lehmann, J. Bredno, V. Metzler, G. Brook, and W. Nacimiento, "Computer-Assisted Quantification of Axo-Somatic Boutons at the Cell Membrane of Motoneurons," *IEEE Trans. Biomedical Eng.,* vol. 48, no. 6, pp. 706-717, 2001.
[46] Z. Yudong and N.J. Pelc, "A Spatiotemporal Model of Cyclic Kinematics and Its Application to Analyzing Nonrigid Motion with MR Velocity Images," *IEEE Trans. Medical Imaging,* vol. 18, no. 7, pp. 557-569, 1999.
[47] P.K. Saha, D.D. Majumder, and A. Rosenfeld, "Local Topological Parameters in a Tetrahedral Representation," *Graphical Models and Image Processing,* vol. 60, no. 6, pp. 423-436, 1998.
[48] S. Malassiotis and M.G. Strintzis, "Tracking the Left Ventricle in Echocardiographic Images by Learning Heart Dynamics," *IEEE Trans. Medical Imaging,* vol. 18, no. 3, pp. 282-290, 1999.
[49] B. Heigl, D. Paulus, and H. Niemann, "Tracking Points in Sequences of Color Images," *Pattern Recognition and Image Analysis,* vol. 9, no. 4, pp. 648-653, 1999.
[50] M.P. Dubuisson-Jolly and A. Gupta, "Color and Texture Fusion: Application to Aerial Image Segmentation and GIS Updating," *Image and Vision Computing,* vol. 18, pp. 823-832, 2000.
[51] S.C. Zhu, T.S. Lee, and A.L. Yuille, "Region Competition: Unifying Snakes, Region Growing, Energy/Bayes/MDL for Multi-Band Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, pp. 884-900, 1996.
[52] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *Int'l J. Computer Vision,* vol. 1, no. 4, pp. 321-331, 1987.
[53] V. Chalana and Y. Kim, "A Methodology for Evaluation of Boundary Detection Algorithms on Medical Images," *IEEE Trans. Medical Imaging,* vol. 16, no. 5, pp. 642-652, 1997.
[54] B. Kochner, D. Schuhmann, M. Michaelis, G. Mann, and K.H. Englmeier, "Course Tracking and Contour Extraction of Retinal Vessels from Color Fundus Photographs: Most Efficient Use of Steerable Filters for Model Based Image Analysis," *Proc. Int'l Soc. Optical Eng. (SPIE),* vol. 3338, pp. 755-761, 1998.
[55] C. Xu and J.L. Prince, "Snakes, Shapes, and Gradient Vector Flow," *IEEE Trans. Image Processing,* vol. 7, no. 3, pp. 359-369, 1998.
[56] I. Cohen, L.D. Cohen, N. Ayache, "Using Deformable Surfaces to Segment 3-D Images and Infer Differential Structures," *Computer Vision, Graphics, and Image Processing: Image Understanding,* vol. 56, no. 2, pp. 242-263, 1992.
[57] C. Nastar and N. Ayache, "Nonrigid Motion Analysis in Medical Images: A Physically Based Approach," *Lecture Notes in Computer Sciences,* vol. 687, pp. 17-32, 1993.
[58] M. Desbrun, M. Meyer, P. Schröder, and A.H. Barr, "Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow," *Computer Graphics (SIGGRAPH '99 Proc.),* pp. 317-324, 1999.
[59] C. Lürig, L. Kobbelt, and T. Ertl, "Hierarchical Solutions for the Deformable Surface Problem in Visualization," *Graphical Models,* vol. 62, no. 1, pp. 2-18, 2000.
[60] P. Radeva, J. Guerrero, C. Molina, and R. Serneels, "A Physics Based Model of the Kohonen Ring," *Proc. Int'l Soc. Optical Eng. (SPIE),* vol. 3338, pp. 1345-1356, 1998.
[61] K.P. Ngoi and J.C. Jia, "An Active Contour Model for Colour Region Extraction in Natural Scenes," *Image and Vision Computing,* vol. 17, no. 13, pp. 955-966, 1999.
[62] S. Gibson, C. Fyock, E. Grimson, T. Kanade, R. Kikinis, H. Lauer et al., "Volumetric Object Modeling for Surgical Simulation," *Medical Image Analysis,* vol. 2, no. 2, pp. 121-132, 1998.
[63] E.L. Juarez, C. Dumont, and M.A. Abidi, "Object Modeling in Multiple-Object 3D Scenes Using Deformable Simplex Meshes," *Proc. Int'l Soc. Optical Eng.,* vol. 3958, pp. 144-152, 2000.

Jörg Bredno received the masters degree in mechanical engineering and the PhD degree in computer science from the Aachen University of Technology (RWTH), Aachen, Germany, in 1998 and 2002, respectively. Until 2001, he was with the Institute of Medical Informatics holding a scholarship from the Studienstiftung des deutschen Volkes (German Scholarship Foundation). He is currently working as a research scientist with the Imaging Systems Group at the Philips Research Laboratories in Aachen. He received the Springorum Medal from the Aachen University of Technology in 1999 for his work on ceramic coatings of hip endoprosthesises and the poster award from the German Workshop on Medical Image Processing in 2001 for his work on automated parameter settings for segmentation algorithms. His current research interests include the model-based analysis of medical images and the real-time ability of image analysis algorithms for clinical applications. He coauthored a textbook on programming of graphical user interfaces (tewi-Verlag, Munich, Germany, 1991).

Thomas M. Lehmann (S'94-M'99) received the masters degree in electrical engineering and the PhD degree (summa cum laude) in computer science from the Aachen University of Technology (RWTH), Aachen, Germany, in 1992 and 1998, respectively. In 1992, he was a research scientist at the Faculty of Electrical Engineering, RWTH Aachen. Since 1992, he has been with the Institute of Medical Informatics, Medical Faculty, RWTH Aachen, where he is currently head of the Department of Medical Image Processing at the assistant professor level. He coauthored a textbook on image processing for the medical sciences (Springer-Verlag, Berlin, Germany, 1997) and coedited the handbook of medical informatics (Carl Hanser Verlag, Munich, Germany, 2002). His research interests are discrete realizations of continuous image transforms, medical image processing applied to quantitative measurements for computer-assisted diagnoses, and content-based image retrieval from large medical databases. He received the DAGM-Preis '93. The award from the German Association for Pattern Recognition was given for his work on automatic strabometry using Hough transform and covariance filtering. In 1998, he received the Borcher's Medal from the RWTH Aachen for his work on medical image registration. He is chairman of the German Workshop on Medical Image Processing and vice president of the working group Medical Image Processing within the German Society of Medical Informatics, Biometry and Epidemiology (GMDS). He is chairman of the IEEE Joint Chapter Engineering in Medicine and Biology (IEEE Germany Section). He is a member of the International Association of Dentomaxillofacial Radiology (IADMFR) and the Society of Photo-Optical Instrumentation Engineering (SPIE). He serves on the international editorial board of Dentomaxillofacial Radiology. He is a member of the IEEE.

Klaus Spitzer received the masters degree in mathematics and the PhD degree (magna cum laude) in mathematics from the Friedrich-Wilhelms-University, Bonn, Germany, in 1983 and 1985, respectively. In 1983, he earned the MD degree (summa cum laude) from the Friedrich-Wilhelms-University, Bonn, Germany. From 1983 to 1993, he was a physician and neurologist at the Department of Neurology, University Hospital of Hamburg, Germany. He was a professor of neurology at the University of Hamburg from 1992 to 1994 and an associate professor of medical informatics from 1993 to 1995 at the University of Heidelberg, Germany. Since 1995, he has been a professor of medical informatics and the chair of the Institute of Medical Informatics, Aachen University of Technology (RWTH), Aachen, Germany. His research interests are knowledge-based systems in medicine, computer-based training, digital patient records, hospital information systems, and management of IT-systems.