Global optimization-based deformable meshes for surface extraction from medical images

Jussi Tohka

Digital Media Institute / Signal Processing

Tampere University of Technology

jussi.tohka@tut.fi

Abstract

This thesis deals with surface extraction from noisy volumetric images, which is a common problem in medical image analysis. Due to noise, the use of *a-priori* information about surface topology and shape is necessary for automatic surface extraction methods. Deformable surface models can incorporate such geometric knowledge into extraction process which is restated as an energy minimization problem. A drawback of deformable models is that the formulated minimization problem is difficult to solve because of numerous local minima and a large number of variables. This difficulty may lead to sensitivity to the initialization, complicating the unsupervised use of deformable models. The main contributions of this thesis are algorithms for solving the minimization problem globally. We propose two classes of algorithms for the task, *Dual surface minimization* (DSM) and a hybrid of real-coded genetic algorithms and a greedy algorithm (GAGR). By global optimization of the energy of the deformable models, we are capable of reducing the initialization sensitivity of deformable surface models, and hence enabling automation of surface extraction. Moreover, these methods for global optimization do not lead to unforseeable sensitivity to values of the model parameters, another problem common with deformable models. As our second contribution, we extend a shape modeling approach for two-dimensional contours to surfaces and analytically derive a shape model for the sphere (surface). We also consider surface extraction from positron emission tomography (PET) images as an application of the deformable model based on the DSM algorithm. This task is problematic because of high noise levels in PET as compared to the contrast of the images. Our automatic method based on the proposed deformable model reliably yielded extraction results of good accuracy as compared to the imaging resolution. The success in this application demonstrates the good properties of global optimization - based deformable models for automatic surface extraction.

i

ii

Preface

The research reported in this thesis has been carried out in the Institute of Signal Processing of Tampere University of Technology during 1999 - 2003. However, the foundations of this work were laid out in the Turku PET Centre where I worked as a research assistant during the summer of 1998. There I first learned about deformable models, or in other words, got bitten by 'snakes'. It was my future supervisor, Docent Ulla Ruotsalainen, PhD, who then introduced me to the world of medical image analysis. For this and for her invaluable advice and guidance, I am indebted to her. I am also deeply grateful to Professor Jaakko Astola for welcoming me to the Institute of Signal Processing as well as his support and encouragement.

The reviewers of this thesis, Professor Edward Delp (Purdue University, USA) and Dr. Hervé Delingette (INRIA, France) deserve heartfelt thanks for their careful reading and constructive feedback of the manuscript of this thesis.

I wish to thank Jouni Mykkänen, PhD, from University of Tampere, the co-author of three of the papers featured in this thesis, for fertile co-operation during these years. Mr. Jouni Luoma, PhD Sakari Alenius, MSc Anu Kivimäki, MSocSc Esa Wallius, MSc Antti Happonen and all the other past and present members of the M^2oBSI research group have provided their assistance when I have needed it. I sincerely thank all my colleagues in the Institute Signal of Processing for providing a stimulating working environment.

From October 2001 to May 2002, I worked at McConnell Brain Imaging Centre (BIC) at Montreal Neurological Institute Canada. I am grateful for Professor Alan Evans for inviting me to BIC. I thank my colleagues at BIC for sharing their know-how and for many discussions related to medical image analysis as well as to other issues.

I would like to thank my parents, Sinikka and Matti, for their encouragement and love during my life and studies. My friends have offered constant support during this research by giving me a chance to forget about it for a while. Most of all, I thank Sanna for her love and caring. I am grateful to Toni Hyvärinen for advice concerning the English language.

Tampere, September 2003   Jussi Tohka

Contents

List of publications

This thesis is based on the following publications. These are referred to in the text as [Publication x], where x is a roman numeral.

Publication-I J. Tohka and J. Mykkänen. Deformable Mesh for Automated Surface Extraction from Noisy Images. To appear in *International Journal of Image and Graphics*, special issue on *Deformable models for image analysis and pattern recognition*, 2003.

Publication-II J. Tohka. Global optimization of deformable surface meshes based on genetic algorithms. In E. Ardizzone and V. Di Gesu, editors, *Proc. of 11th International Conference on Image Analysis and Processing, ICIAP01*, pages 459 – 464, Palermo, Italy, IEEE-CS Press, September 2001.

Publication-III J. Tohka. Surface extraction from volumetric images using deformable meshes: A comparative study. In A. Hayden, G. Sparr, M. Nielsen and P. Johansen, editors, *Proc. of 7th European Conference on Computer Vision, ECCV02, Lecture Notes in Computer Science 2352.* pages 350 – 364, Copenhagen, Denmark, Springer Verlag, May 2002.

Publication-IV J. Mykkänen, J. Tohka, and U. Ruotsalainen. Delineation of Brain Structures from Positron Emission Tomography Images with Deformable Models. In R. Baud, M. Fieschi, P. Le Beaux, and P. Ruch, editors, *The new navigators: from Professionals to Patients*, vol 95, *(Proc. of Medical Informatics Europe, MIE2003)*, pages 33 -38, St. Malo, France, IOS-Press, 2003.

Publication-V J. Mykkänen, J. Tohka, J. Luoma, and U. Ruotsalainen. Automatic extraction of brain surface and mid-sagittal plane from PET images applying deformable models. Submitted to *Medical Image Analysis*.

The author's contribution to Publications I, IV and V was as follows. As the first author in [Publication I] J. Tohka has designed and implemented the algorithms, performed the mathematical derivations, performed most of the experiments reported, and written the manuscript for the most part. In [Publication IV], [Publication V] the author has had an important role in the development of methods to process positron emission tomography images and has also had an

active role in writing the manuscript. Articles [Publication IV] and [Publication V] have appeared as a part of the Ph.D. thesis of J. Mykkänen. An earlier version of [Publication V] is available as a technical report (Technical Report A-2003-1, Deparment of Computer and Information Sciences, University of Tampere, 2003).

List of abbreviations

2-D	two-dimensional
3-D	three-dimensional
CT	computerized tomography
DSM	Dual surface minimization
DSM-CRP	DSM-constrained reference point
DSM-FRP	DSM-floating reference point
DSM-OS	DSM - Outer surface
FDG	fluoro-2-deoxy-D-glucose
fMRI	functional magnetic resonance imaging
GA	genetic algorithm
GAGR	A hybrid algorithm of a real coded genetic algorithm and a greedy algorithm
GGVF	generalized gradient vector flow
GVF	gradient vector flow
MAP	maximum a posteriori
MR	magnetic resonance
pdf	probability density function
PET	positron emission tomography
RCGA	real coded genetic algorithm
SPECT	single photon emission computed tomography

Chapter 1

Introduction

1.1 Volumetric medical imaging

Several types of devices provide three-dimensional image data. For example, laser ranging systems produce images where each pixel intensity expresses the distance between a known reference frame and a visible point in the scene. These images are called range images or 2.5-dimensional (2.5-D) images. The focus in this thesis is on another type of three-dimensional image data, namely volumetric intensity images. These are collections of elementary volume elements called voxels and intensity values associated with them. Voxels are three-dimensional counterparts of pixels and they have the shape of a rectangular parallelepiped. Three-dimensional intensity images can be considered as generalizations of 'normal' two-dimensional intensity images.

Volumetric intensity images, or volumetric images, are most commonly encountered within medical imaging. They are acquired by the means of tomography; a method of producing a three-dimensional image of the internal structures of a solid object by the observation and recording of the differences in the effects on the passage of waves of energy impinging on those structures [18]. Medical imaging allows non-invasive examination of living beings and offers valuable information for clinicians in support for making critical decisions . However, applications of medical images go beyond this; They are important, for example, in drug development and in image-guided surgery.

Information provided by a medical image depend on the measured physical phenomenon and there are several imaging modalities each providing different information about the object imaged [69]. X-ray based computed tomography (CT) produces images of the photon attenuation of tissue. Magnetic resonance (MR) imaging measures the proton or water density. MR and CT images are
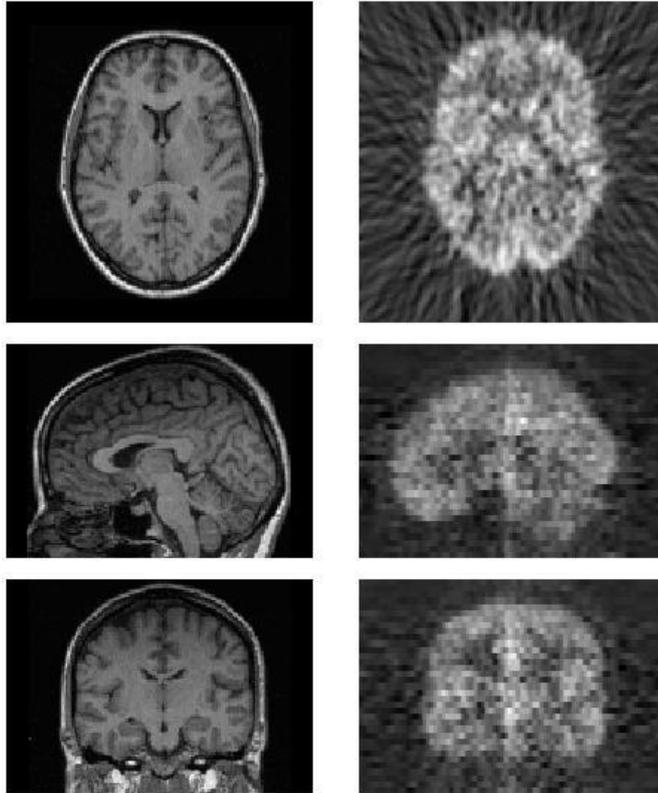
Figure 1.1: Examples of medical images of head, from top transaxial, sagittal, and coronal cross-section views. (Left) A T1-weighted MR image (left) and PET image (right) of the same subject. Images were provided by Turku PET Centre.

used for describing the anatomical structure of the imaged subject, because they can make distinctions between different types of tissue. On the other hand, imaging modalities like positron emission tomography (PET) and functional MR imaging (fMRI) can delineate information about the functional properties of the tissue. See Fig. 1.1 for examples of image cross-sections acquired with MR and PET.

1.2   Analysis of volumetric image data

Medical images can be used qualitatively for aid in making a diagnosis. However, their use in medical research requires extraction of quantitative and objective information from images. Several rather distinct entities can be measured

quantitatively. These include measuring volumes or characterizing shape of anatomical brain structures based on MR images, or computing the glucose consumption within biologically meaningful volumes based on FDG (fluoro-2-deoxy-D-glucose) PET images. Of course, before any values can be computed, structures of interest must be delineated from images. Structures of interest can be volumes or surfaces, and procedures for recovering them are called, respectively, image segmentation and surface extraction.

Image segmentation can be, in principle, performed manually by a trained clinician with suitable equipment. However, manual segmentation has several drawbacks:

- The amount of acquired data is enormous and performing the structure extraction manually, or even semi-automatically, can be costly, if feasible at all.

- When several experts are processing the images, the reproducibility and the comparability of the processed images are reduced. This is simply due to the divergent opinions and the individual working habits of the people involved. See [85] and page 216 in [69] for particular examples.

These considerations call for automatic methods to perform the structure extraction. However, automation of medical image analysis is complicated and it requires advanced techniques, because 1) images are noisy as compared to their contrast and 2) intensity values in an image do not solely define the (biologically meaningful) structure of interest, as their spatial organization is also very important. 3) Images are characterized by individual variability.

Noise in medical images is a sum of different components, e.g. measurement noise, natural intensity variation within structures of interest, and modality dependent imaging artifacts. Noise suppression and artifact correction without destroying valuable information is therefore not a simple issue. Types and levels of noise in images vary significantly between imaging modalities. Note that in medical image analysis the level of noise is more naturally compared to the contrast between structures of interest than to the intensity values themselves. This is because the features that allow us to discriminate between different structures of interest are differences in intensity values rather than intensity values themselves.

The spatial relationships, such as inclusion and adjacency, between different structures in a medical image are often *a-priori* known based on existing neuroanatomical knowledge. This has to be taken into account when segmenting images. The high-level prior knowledge simplifies the segmentation problem,

but at the same time algorithms capable of utilizing it can become more complicated than segmentation algorithms relying only on the image data.

Biological shapes are complex and individually variable and naturally this has its effect on medical images. Individual variability is reflected also on the intensity values of images. There does not exist an intensity value that globally characterizes a certain structure in the images of several subjects. This is, again, due to individual variability in subjects and sometimes also due to difficulties in the calibration of tomographs. To summarize, it is necessary to apply high-level information together with low-level clues to be able to develop automatic methods for medical image analysis.

1.3   Aims and structure of the thesis

The aim of the thesis is to develop fully automatic techniques for surface extraction from noisy volumetric images. Particularly, focus is on the extraction of surfaces homeomorphic to the sphere. The proposed methods are based on deformable surface models, and automation and tolerance to noise are achieved by using global optimization algorithms for minimization of their energy. The designed global optimization algorithms are the main contributions of the thesis. The extracted surfaces can be applied for segmentation of specific brain structures from medical images or for other tasks in medical image analysis requiring surface extraction.

More mathematically oriented definitions of basic concepts such as the image and the surface are provided in Chapter 2, where some standard methods for surface extraction are also summarized. Chapter 3 reviews the literature on deformable models. With the literature review our intention is to provide an unified framework of deformable models that is not blurred by computational considerations. In Chapter 3, we also present a more detailed description of the aims of the thesis based on the terminology introduced so far. Chapter 4 focuses on deformable models built on a particular surface representation, surface meshes. In this chapter, the first contributions of this thesis are presented. Chapter 5 presents two global optimization algorithms for surface extraction with deformable models. These are the primary contributions of this thesis. In Chapter 6 we provide an experimental comparison of deformable models based on our global optimization algorithms and some other other recent deformable models. The application of the developed surface extraction methods for the analysis of PET images is described in Chapter 7. Some limitations and advantages of the proposed algorithms are discussed in Chapter 8, where the main contributions of this thesis are summarized.

Chapter 2

Surface Extraction and Image Analysis

In this Chapter some basic concepts and terminology are defined. The purpose of these definitions is to make the problem setting easier to grasp. Preliminaries from topology are explained, but the reader is assumed to be familiar with the basic concepts from real analysis that can be found in e.g. [3].

2.1 Images and segmentations

2.1.1 Images

Volumetric digital images are collections of values describing the strength of some measured physical quantity at a (finite) set $\mathcal{D} \subset \mathbb{R}^3$ of loci. The measured quantity is referred to as the intensity. The loci in the set $\mathcal{D}$ are related to image voxels and we can assume that they are the voxel centers. For notational simplicity, we also use the symbol $\mathcal{D}$ for the set of voxels, although this is not strictly true.

Images are defined as maps from the set of loci $\mathcal{D}$ to the set of possible intensity values. However, to simplify the notation later on we define a volumetric image as a map $I : \mathbb{R}^3 \to [0, 1]$, which is piecewise constant and has a bounded *support* $supp(I) = \{\mathbf{x}|I(\mathbf{x}) \neq 0\}$. The support $supp(I)$ is called *image domain*. The range of image I is selected as $[0, 1]$ for convenience and in practise this only requires affine scaling of the intensity values of the observed image appropriately. The requirement that images are piecewise constant simply means that image intensity inside a particular voxel is constant. (We might as well consider images that are piecewise trilinear functions.). Intensity values

of an image I are nonzero only on a bounded subset of $\mathbb{R}^3$ because the support of I is bounded.

2.1.2   Segmentations

Segmentation [1] of the image I denoted by I_s provides the information about which voxels belong to a certain structure of interest. Let us denote by $\mathcal{L} = \{1, \ldots, m\}$ the set of labels of structures of interest. In medical image analysis labels are known prior to any processing. Also, labels are not interchangeable, meaning each label is identified with a particular structure of interest. In a segmentation a label (or labels) is assigned to each voxel. A *segmentation* of I is therefore a collection of subsets of $\mathcal{D}$, i.e. $I_s = \{R_l \subseteq \mathcal{D} | l \in \mathcal{L}\}$. Indeed, we sometimes wish to associate several labels with a single voxel. Consider an anatomical MR-image of a human head. We know that ventricles, nuclei and all the other brain structures are inside the brain, but the skull, for instance, is not inside the brain. This is easiest to model if we can assign both labels 'brain' and 'ventricles' to a voxel belonging to ventricles.

As already mentioned, we often know quite a lot about the spatial relations between structures of interest. Therefore, we can assert constraints on sets R_l, such as connectivity, adjacency or non-adjacency between R_a and R_b, inclusion and so on. Algorithms aiming at image segmentations that respect the given constraints can be coarsely classified into two classes. 1) Algorithms that try to solve the whole problem in one step in e.g. [50], and 2) algorithms that make use of intermediate goals, i.e. divide the segmentation problem into smaller sub-problems as in e.g. [55].

Structures of interest can be surfaces instead of image sub-volumes. If this is the case, we face another problem termed surface extraction, which is the topic of this thesis. Surface extraction and segmentation problems are closely related when considering an input in the form of a volumetric image. Particularly, knowing the bounding surfaces of the volumetric structure gives the information about the voxels inside that structure; the segmentation problem can be attacked via surface extraction. This leads to a segmentation strategy of the type 2) above. We can extract surfaces of interest in the image one by one starting from the easiest surfaces to extract, and then utilize information of the already found surfaces for succeeding (more difficult) surface extraction tasks. A practical example of application of this 'segmentation via surface extraction' paradigm is presented in Chapter 7, where we consider automatic analysis of PET images.

---

[1]The term segmentation is used for the process of segmenting images as well as the result of that process.

Also other applications of the surface extraction exist, some examples are cited in Chapter 2.3.

2.2 Surfaces

In this Section we consider surfaces from a topological point of view. The material of the section is mainly taken from [19]. However, most of it can be found in almost any text on topology (e.g. [36]) and also in the Internet resource [79].

2.2.1 Preliminaries from topology

Before we can define surfaces we need some definitions from topology. Recall that the Euclidean topology of $\mathbb{R}^3$ means a set of open subsets of the metric space $\mathbb{R}^3$ with the Euclidean distance as the metric. The Euclidean space $\mathbb{R}^3$ equipped with this topology becomes a topological space. A function $f : X \to Y$ between two topological spaces is *continuous* if the inverse image

$$f^{-1}(O) = \{x \in X | f(x) \in O\}$$

is open for every open set $O \subset Y$. This definition is equivalent to the standard definition of continuity when the Euclidean topology is considered. A *homeomorphism* between two topological spaces is a bijective continuous function between these spaces that has a continuous inverse. If there exist a homeomorphism between two topological spaces they are said to be homeomorphic, i.e. topologically equivalent.

Two-dimensional *manifolds* (2-manifolds) are topological spaces which have the property that each of their points has a local neighborhood that is homeomorphic to $\mathbb{R}^2$. Now, *surfaces* (in $\mathbb{R}^3$) can be defined mathematically as connected two-dimensional sub-manifolds of $\mathbb{R}^3$. The practical meaning of restricting surfaces to be sub-manifolds of $\mathbb{R}^3$ is that we do not allow such 2-manifolds as surfaces that cannot be drawn in $\mathbb{R}^3$ without self-intersections. An example of such a 2-manifold is the Klein's bottle. In this thesis, we consider only closed surfaces i.e. manifolds that are also closed and bounded as sets.

Particularly, our interest lies on surfaces that are homeomorphic to the sphere. Unlike Jordan curves all (non-selfintersecting) closed surfaces are not topologically equivalent. To decide whether two surfaces are homeomorphic, we need the notion of genus. The *genus* of a surface is defined as the largest number of non-intersecting simple closed curves that can be drawn on the surface without separating it. For example, the genus of the sphere is 0 and the

genus of the torus is 1. Two surfaces are homeomorphic if and only if they have the same genus. The genus is sometimes referred to as the number of handles or holes in the surface.

Note, however, that if there exists an homeomorphism f between surfaces S_1 and S_2, there need not to exist such a homeomorphism $\hat{f} : \mathbb{R}^3 \to \mathbb{R}^3$ that $\hat{f}(S_1) = S_2$. For more about the topic, see pp. 174 - 178 in [36] and references therein.

2.2.2   Simplicial complexes

A finite collection of points is said to be *affinely independent* if no affine space of dimension i contains more than $i + 1$ of the points and this is true for every i. For example, three points of $\mathbb{R}^3$ are affinely independent if they do not lie on the same line. A *k-simplex* υ in $\mathbb{R}^d$ is the convex hull of a set $U = \{\mathbf{x}_1, \ldots, \mathbf{x}_{k+1}\} \subset \mathbb{R}^d$ of $k + 1$ affinely independent points,

$$\upsilon = \{\mathbf{x} = \sum_{i=1}^{k+1} a_i \mathbf{x}_i | a_i \in [0, 1], \sum a_i = 1\}.$$

Hence in $\mathbb{R}^3$, 0-simplices are points of $\mathbb{R}^3$, 1-simplices are edges of $\mathbb{R}^3$ and 2-simplices are triangles in $\mathbb{R}^3$. The *dimension* of k simplex is k. A *face* τ of the simplex υ is a convex hull of any subset T of U. Note that τ of υ is itself a simplex. If τ is a face of υ, we denote $\tau \leq \upsilon$.

A simplicial complex is a collection of faces of simplices, any two of which are either disjoint or meet in a common face, cf. Fig. 2.1. Mathematically, it is a collection K of simplices such that

1. if $\upsilon \in K$ and $\tau \leq \upsilon$, then $\tau \in K$;

2. if $\upsilon \in K$ and $\varsigma \in K$, then $\phi \leq \upsilon$ and $\phi \leq \varsigma$, where $\phi = \upsilon \cap \varsigma$.

The *dimension* of K is the largest of dimensions of its simplices. If the dimension of K is k, K is a k-complex.

Simplicial complexes can be defined also without referring to the geometry in their construction, and particularly without specifying the space in which they lie. See e.g. [19] or [37] for details of these abstract simplicial complexes.

It remains to be explained how simplicial complexes relate to surfaces. The underlying space $|K|$ of a simplicial complex K in $\mathbb{R}^d$ is the union of its simplices together with the subspace topology inherited from $\mathbb{R}^d$, i.e.

$$|K| = \{x \in \mathbb{R}^d | x \in \upsilon \in K\}.$$
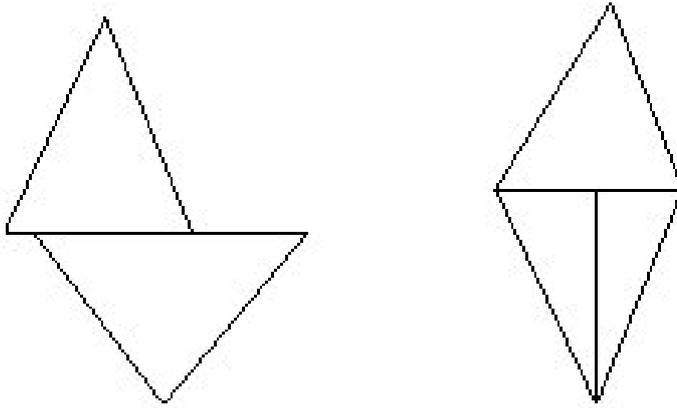
Figure 2.1: Sets of triangles that violate the condition 2 in the definition of simplicial complexes.

A triangulation of the topological space X is a simplicial complex K whose underlying space is homeomorphic to X. Now, since surfaces are topological spaces and particularly 2-manifolds, a triangulation of the surface is a simplicial complex homeomorphic to that surface. Particularly, the dimension of the simplicial complex must be equal to 2. Later on, we refer to 0-simplices of a triangulation as vertices, 1-simplices of a triangulation as edges and 2-simplices of triangulation as faces. Note that this definition of the face is somewhat narrower than the one that was given previously.

In this thesis our objective is to extract surfaces of a particular topological type from the images. The topological type of the surface can be deducted from a triangulation of it. Denote by p_k the number of k-simplices in a triangulation of a surface in $\mathbb{R}^3$, then

$$\sum_{k=0}^{2}(-1)^k p_k = 2 - 2g, \qquad (2.1)$$

where g is the genus of the surface. The above result is known as the *Euler-Poincaré* formula. It extends to some other topological entities besides triangulations of surfaces, an example of which are embeddings of the graphs [8, 34].

In medical image analysis, the topological type of the imaged object is often known more precisely than that can be expressed with simple set-theoretic constraints on the segmentations. In fact, looking at the surfaces present in the image, their topological type is known *a-priori* in many applications. This is the reason behind the interest in constraining surface topologies within medical image analysis. Of course, the expected surface topologies and also the

set theoretic constraints imposed on segmentations may be violated by severe pathologies.

2.3   Surface extraction

Surface extraction, or reconstruction, is a general problem that can be studied from several viewpoints. The problem is to recover a finite representation, for example a triangulation, of a surface of interest given some input data. Here, we assume that the initial form of the input is a volumetric image. Other forms of input data such as unorganized sets of points can as well be considered cf. [7, 38, 37].

2.3.1   Iso-surface algorithms

Probably the best known surface extraction algorithms are so-called iso-surface algorithms [62]. The algorithms belonging to this class take a function f from $\mathbb{R}^3$ to $\mathbb{R}$ as their input. The zero level set of the function f is assumed to be the surface to be approximated. Iso-surface algorithms work by dividing the space $\mathbb{R}^3$ into cubes, or into some other polyhedral primitives, and thereafter evaluating the function f at every vertex of every primitive. Based on these evaluated values, intersections of the surface and primitive polyhedra are inferred. For example, the Marching Cubes algorithm [49] uses only the information about the sign of the evaluated function values and a look-up table to triangulate the given surface.

The Marching Cubes algorithm offers a straight-forward method for extracting surfaces from volumetric images. Each voxel has simply to be classified either being inside or outside of the surface of interest, a segmentation of the image. If image segmentation is performed simply by thresholding intensity values, the process is sensitive to imaging noise, natural variation of intensity values across the image, and other image artifacts. Especially, the correct surface topology cannot be enforced. Note, that noise, blur, and other artifacts may cause the surface interest appear with wrong topological type in the image. Another drawback of the marching cubes algorithm is that the number of triangles in a triangulation is large.

2.3.2   Applications of surface extraction

Applications of surface extraction are various in medical imaging. An obvious application is the visualization of three-dimensional anatomical structures,

whose shapes are obviously difficult to conceive directly from volumetric images. The application that we have already introduced and are mainly concerned with in this thesis is image segmentation.

There exist also other, maybe not so obvious, applications. For example, the computation of the thickness of the human cerebral cortex can reveal important information about different disorders, the issue have been studied in the case of the Alzheimer's disease by Lerch *et al*. [47]. Their thickness computation depended on the extraction of cortical surfaces from MR images, see [51]. Another application of surface extraction is surgery simulation [13, 14] and there are others.

Chapter 3

Deformable Surface Models

3.1 Introduction

Purely bottom-up approaches for surface extraction, such as the Marching Cubes algorithm [49], suffer from their inability use information that cannot be directly inferred from image intensities as explained in the previous Chapter. This seriously complicates even their supervised use in medical image analysis. Deformable surface models are advanced techniques for surface extraction that can incorporate soft and hard constraints into the surface extraction problem. Deformable models and their applications are surveyed for example in [40, 56, 60, 82].

A deformable surface model consists of a geometric representation of surfaces, a template surface, and rules for the evolution of the surface based on image data. In this chapter we describe a general framework for governing the evolution of these surfaces. Some possibilities for surface representation are also mentioned.

3.2 Surface extraction as an energy minimization problem

Surface extraction is formulated as an energy minimization problem with deformable models. That is, to reconstruct a surface from an image we associate to each (admissible) surface a quantity called energy. The energy of the surface describes how well that particular surface matches with image data and how well it suites to our prior expectations of the surface to be extracted. The lower the energy the better the surface is and we aim to find the surface that has the

minimal energy. If S_1 is the surface extraction result, $\mathcal{S}$ is the set of admissible surfaces and $E(S, I)$ is the energy of the surface S given the image I, then the problem is the following

$$\text{Given the image } I, \text{ find such } S_1 \in \mathcal{S} \text{ that } S_1 = \arg\min_{S \in \mathcal{S}} E(S, I). \quad (3.1)$$

The energy $E(S)$ of the surface S is composed of the internal energy $E_{int}(S)$ which depends only of the properties of the surface S and of the external energy $E_{ext}(S, I)$, which couples S with the observed image I. In symbols,

$$E(S, I) = E_{int}(S) + E_{ext}(S, I), \quad (3.2)$$

where I is the observed image. Note, that we have two distinct ways of incorporating image independent information to the process. We can rule out surfaces by restricting the set $\mathcal{S}$ and we can favor some surfaces over the others based on the internal energy. The internal energy can, for example, quantify smoothness of surfaces and the set $\mathcal{S}$ can be composed of surfaces topologically equivalent to the sphere.

The energy minimization problem (3.1) should not be viewed in strictly mathematical sense. In this thesis, we are not concerned if the energy minimization problem has a unique solution. More interesting question is what kind of solutions we can obtain for the surface extraction problem using the computational framework offered by Eqs. (3.1) and (3.2).

3.3 Probabilistic rationale

The energy minimization problem (3.1) can be viewed as a maximum a posteriori (MAP) estimation problem as presented for example in [56]. For this, we associate a prior probability

$$p(S) = \frac{1}{Z_1} \exp(-E_{int}(S)) \quad (3.3)$$

for each surface. The likelihood of an image I given S is

$$p(I|S) = \frac{1}{Z_2(S)} \exp(-E_{ext}(S, I)). \quad (3.4)$$

Quantities Z_1 and $Z_2(S)$ are normalization factors, termed partition functions, required to make the probabilities proper. Using the Bayes formula we find that the negative of the logarithm of the posterior probability is

$$l(S|I) \doteq -log[p(S|I)] = E_{int}(S) + E_{ext}(S, I) + log(Z_1) + log(Z_2(S)). \quad (3.5)$$

The minimization of $l(S|I)$ then corresponds then to the minimization of the $E(S, I)$ if and only if Z_2 is independent of S as Figueirdo *et al.* [22] have stated. This is not always the case and therefore, in general, the minimization of the energy is not equivalent to the MAP-estimation.

The problem in the continuous limit is that $Z_1 \to \infty$, and hence our probabilities are not proper. However, this presents no real problem in practise, because all our computations are based on finite approximations of surfaces and images, see [61].

3.4   Pattern theory

The interpretation of the energy minimization problem as MAP estimation leads us to consider deformable surface models as a part of pattern theory. Pattern theory is a branch of applied mathematics born in late sixties which studies *regular structures* that can be found from e.g. anatomy (cf. [28]), linguistics, and physics. An introductory text about pattern theory is [27] and a more complete treatment can be found in [26].

A main concept in pattern theory is the notion of a deformable template, which contains an average model of the particular object being modeled and a set of transformations that can be applied to the average model. All the other instances of the object of interest are then obtained by applying these transformations to the average model. Each transformation is assumed to have a prior probability and the correspondence of the transformation and the observed data is modeled by a likelihood of the data given the transformation. This leads again to the MAP framework for the purpose of data-analysis. Deformable surface models can be therefore regarded as pattern theoretic deformable templates. Subsequently, deformable surface models can be defined and seen as a part of a broader concept. Deformable surface/contour models are not the only type of deformable templates encountered within image analysis, an overview of Bayesian methods in image analysis is presented in [61].

3.5   Physics-based rationale

The energy minimization problem in Eq. (3.1) has a physics based analogue. For this, we place a surface made of some elastic material into a force field. After a certain amount of time has passed, the surface has assumed a position in which its energy is minimized. The energy of the surface is composed of the internal term relating to elasticity and rigidity properties of the material of the

surface and the external term relating to the potential of the force field derived from the underlying image. In other words, we once again end up with the energy minimization problem (3.1).

Deformable models in this physics-based setting were originally conceived by Terzopoulos and his co-workers [43, 74, 75]. Especially, the snakes algorithm [43] for contour extraction has been a real starting point for the research and application of deformable models in the image analysis. The motivating idea was to consider surface/contour extraction as an ill-posed inverse problem [71]. The well-posedness of the problem is then restored by regularizing it by internal energy. This process in general setting is referred to as *Tikhonov regularization* [76].

It is an interesting question to ask what is the forward problem related to the inverse problem of contour/surface extraction, or more generally, to the inverse problem of vision. Terzopoulos has suggested that it is computer graphics and animation, and deformable surface models have been used also in this setting [73]. In the physics-based setting this is very natural especially if one considers the reasoning behind the numerical schemes used to solve the energy minimization problem. The computations leading to a solution of the energy minimization problem utilize forces acting on a surface. In other words, a dynamic Euler-Lagrange differential equation is derived that corresponds the energy minimization problem. The solution of the differential equation is also (a local) minimum of the corresponding energy function. Hence, we have a numerical scheme for solving the energy minimization problem by solving the corresponding differential equation. For an elegant treatment of these issues and for computational considerations, see [12]. The equivalence of certain variational problems and boundary value problems is a standard result in variational calculus [41].

3.6 Computational aspects of deformable surface models

Before any computations can be performed we have to decide how to discretize the problem. Most importantly surfaces have to be represented in some way that supports the computations. A taxonomy of deformable models based on surface representations has been presented in [60]. Surface representations were first divided into the two main categories: continuous and discrete. With discrete representations, geometry of surfaces is known only at a finite set of points. Discrete representations include particle systems [70] and surface meshes, which

are dealt in greater detail in the next Chapter.

Continuous surface representations can be either implicit or explicit. With explicit representations, surfaces are defined as functions $\mathbf{s_q} : \Omega \to \mathbb{R}^3$, where $\Omega \subset \mathbb{R}^2$. The vector $\mathbf{q}$ contains a finite set of parameters, which completely specify the function $\mathbf{s_q}$. There are several distinct explicit representations that have been applied with deformable surface models. For example superquadrics [72] and finite elements in [12] are considered to belong to this class of surface representations.

Implicit representations consider surfaces defined as zero level-sets of functions from $\mathbb{R}^3$ to $\mathbb{R}$. An extension of superquadrics, hyperquadrics, belong to this class [33]. Level-sets, introduced by Malladi *et al*. [52] and Casselles *et al*. [10], are flexible ways to represent surfaces implicitly. The flexibility of representation as compared to many others stems from the fact that the topology of the surface can changed implicitly with the level set formulation of the surface. This is convenient for many but not for all surface extraction tasks within medical image analysis. For applications of deformable models based on level sets within medical image analysis, see e.g. [4, 15, 67].

3.7   Related approaches

There exist a number of approaches related to deformable surface models for medical image segmentation. We briefly consider two examples here, namely elastic registration and deformable topological models. The aim of elastic registration [5] is to deform the image domain of an atlas to fit the observed image. An atlas is a labeled template image that represents idealized (average) image of the object of interest. The deformation from the atlas to the observed image is found by minimizing a cost function. The cost function is composed of the data term (corresponding external energy) and the term reflecting intrinsic costs of deformations (corresponding internal energy).

Deformable topological models [54, 55] could be characterized as an digital analogue to the elastic registration framework. An initial labeled image, or a segmentation in our terminology, is matched to the observed image by minimizing an energy function. The stochastic minimization process is carried out by studying only those (local) transformations that respect the pre-defined topological constraints.

3.8   Summary of deformable surface models

Methods based on solely image data are not sufficient for reliable and automatic segmentation of noisy images encountered in medical image analysis. However, the use of image independent information can remedy the situation and enable automatic segmentation and surface extraction. Several model-based techniques, applying image independent information, have been introduced so far. One example of these is deformable surface models. Deformable surface models apply geometric prior information in addition to image data for surface extraction. In practise, the surface extraction problem is converted to an optimization problem. In this chapter, we have reviewed interpretations of the optimization problem and briefly presented possible ways to discretize it.

3.9   Aims of the thesis revisited

At this point, we have not discussed how to solve the optimization problem (3.1). Indeed, it can be expected that the energy function is multi-modal (i.e. has several local minima) and minimizing it globally is therefore difficult. It is the primary objective of this thesis to propose algorithms targeted specifically to this optimization problem. By applying these global optimization algorithms for solving (3.1), we wish to reduce common problems with deformable models related to initialization and parameter sensitivities.

The term initialization sensitivity refers to the problem of the final surface extraction result depending heavily on the provided initialization. Since automatic means for generating close initializations are difficult design, the problem of initialization sensitivity obviously limits fully automatic use of deformable models.

Another common drawback of deformable surface models is that their formulation contains a number of user definable parameters. These parameters can be model parameters, e.g. parameters that control how much the internal energy is weighted relative to the external energy. In addition, there can be parameters related to the optimization procedure. Deformable models are and need to be sensitive to these parameters allowing of processing dissimilar data. However, it is problematic if a deformable model is highly sensitive to its parameter values or the relation between parameter values and the behaviour of a deformable model is unpredictable. After all, deformable models should be able to cope with images within a specified application with the same parameter values to be automatic and, as mentioned in Chapter 1, these images can be rather divergent. Besides trying to answer the initialization sensitivity problem

with global optimization, we try to confine the parameter sensitivity problem to its minimum. Our methods for the formulation and optimization of deformable surface models are free of user definable parameters, but most of these parameters are rather easy set based on the properties desired from the deformable model. Also, it will be demonstrated that dissimilar images can be processed successfully with the same set of parameter values.

Chapter 4

Deformable Surface Meshes

4.1 Meshes as surface representations

We shall now focus on deformable surfaces build upon a particular surface representation, namely surface meshes. Intuitively, a surface mesh is a collection of polygons (in $\mathbb{R}^3$) glued together to form a piecewise linear surface. A particular type of the surface meshes are triangulations which were defined already in Chapter 2. We will refer to triangulations as triangular meshes when using them with deformable models. Another type of surface meshes are simplex meshes introduced by Delingette [16, 17]. There are other types of surface meshes as well, but abovementioned ones are the most popular within deformable surface models. One reason for this probably is that they can represent surfaces of any genus.

4.1.1 Triangular meshes

Recall that triangular meshes are triangulations of surfaces. In what follows, it will be useful to distinguish between the connectivity of the mesh and the geometry of it. The connectivity refers to the way the elements of the mesh relate to each other and the geometry refers to the vertex positions of the triangular mesh. Two vertices are said to be *neighbours* if they belong to the same edge of the mesh. Similarly, two edges are neighbours if their intersection is not empty. Two triangles of the mesh are neighbours if they share an edge. These connectivity, or adjacency, relations can be modeled by graphs. If we take as a fact that a triangular mesh is a valid simplicial complex approximating a surface of the known topology, it is sufficient to consider only the information derived from the vertex-adjacency graph of the mesh. The vertex adjacency graph $\mathcal{G}$ related

to the mesh K is called the *graph of the mesh*. Moreover, we write K as a pair $(\mathbf{W}, \mathcal{G})$, where $\mathbf{W}$ is the set of vertex positions, or *mexels*. Note, that required facts about the connectivity of the mesh cannot necessarily be deducted quickly from the graph of the mesh. Therefore, this conceptually simple representation of the connectivity does not support efficient implementation.

4.1.2 Simplex meshes

The formal (geometric) definition for the simplex mesh is given in [17] and it is based on the definition of q-cells. A 0-cell is a point of $\mathbb{R}^d$ and a 1-cell is an edge of $\mathbb{R}^d$. Then, q-cell C is an union of $(q-1)$-cells such that

1. Every vertex belonging to C belongs to q distinct $(q-1)$-cells.

2. The intersection of two $(q-1)$-cells is either empty or a $(q-2)$ cell.

A k-simplex mesh of $\mathbb{R}^d$ is then a $(k+1)$-cell of $\mathbb{R}^d$. Our interest here is on 2-simplex meshes of $\mathbb{R}^3$, and by a simplex mesh we refer particularly to 2-simplex meshes of $\mathbb{R}^3$. Furthermore, 2-cells of a simplex mesh are called faces of the mesh.

Again, we define two mexels to be neighbours if there exists an edge connecting them. As with triangular meshes, we can represent a simplex mesh as a pair $(\mathbf{W}, \mathcal{G})$, where $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_N\}$ is the set of mexel positions and $\mathcal{G}$ is the graph of the mesh. Each mexel in a simplex mesh has exactly three neighbours and therefore graphs of simplex meshes are 3-regular or trivalent. We write $\mathbf{w}_{i_j}, j = 1, 2, 3$ for the three neighbours of the mexel $\mathbf{w}_i$.

For each simplex mesh there exists a triangular mesh whose graph is the dual graph (cf. [8] Chapter 4) of the graph of the simplex mesh. The inverse property holds also, namely for each triangular mesh there exists a simplex mesh whose graph is the dual graph of the graph of the triangular mesh. This property is entirely of topological nature, there does not exist a geometrical duality between simplex and triangular meshes [16]. The (topological) duality between simplex meshes and triangular meshes gives means for constructing a simplex mesh from a triangular one, the procedure is outlined in detail in [81]. Constructing a triangular mesh from a simplex mesh based on the duality is possible, but in practise it is instead preferable to triangulate each face of the simplex mesh. Examples of a triangular mesh, the dual simplex mesh, the dual of the simplex mesh, and the triangulated simplex mesh are shown in Fig. 4.1.

Simplex meshes have many favorable properties compared to triangular meshes that follow from the constant vertex connectivity, cf. [16]. On the
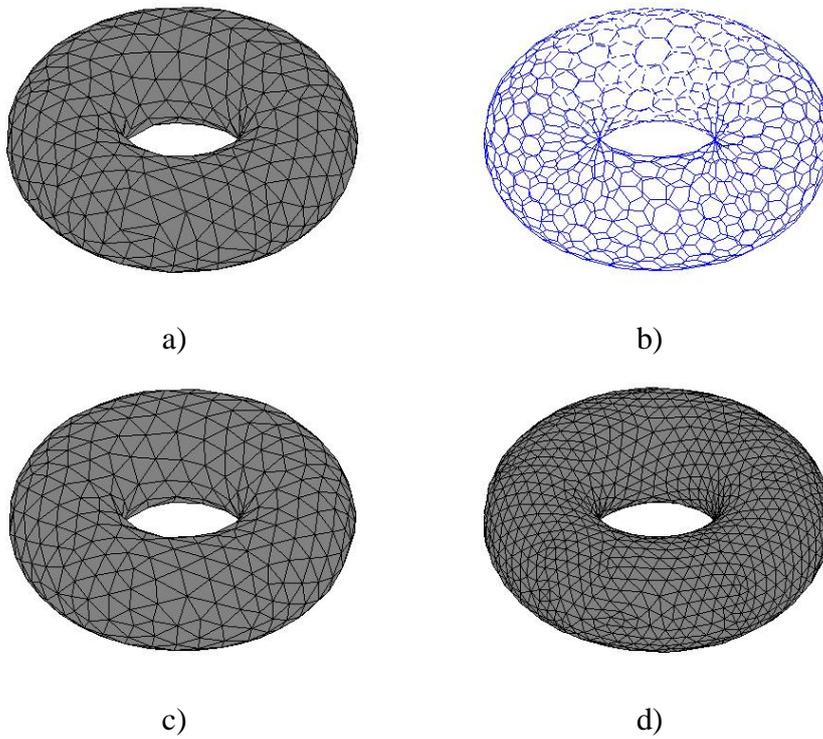
Figure 4.1: Example meshes. a) A triangular mesh of a torus. b) A dual simplex mesh of the mesh in a). c) A dual triangular mesh of the simplex mesh in b), note that meshes in a) and c) are clearly different although they share the connectivity. d) The triangulation of the simplex mesh in b). This triangular mesh does not have the same connectivity as meshes in a) and c).

other hand, simplex meshes have to be converted to triangular ones for area calculation or visualization.

We consider only simplex meshes from now on if not otherwise mentioned. However, optimization algorithms to be described in the next Chapter can be converted in a straight forward manner for the use with triangular meshes instead of simplex meshes.

4.2   Topology adaptation

Deformations of the template mesh can be of topological nature or of geometric nature. Geometric operations alter the mexel positioning and affect only the set of mexels. Topological operations are such that they modify the adjacency graph. They divide into two subgroups. The first subgroup consists of the operations which change the genus of the underlying surface. The second kind of operations modify the adjacency graph but do not change the genus of the underlying surface. These operations include vertex addition to produce denser mesh and vertex removal to reduce the mesh resolution. Because removing vertices diminishes the required storage for the mesh, it is often termed as surface simplification.

Operations to adapt the topology of simplex meshes are defined in [17, 16]. We shall later on use T_2^2 operation meant for increasing the mesh resolution. The operation is depicted in Fig. 4.2. As can be seen from the figure, the operation basically adds an edge to a simplex mesh.

Figure 4.2: T_2^2-operation. On left a face of a simplex mesh before the operation is depicted. On right two new faces of a simplex mesh resulting from the operation are shown.

Ways to adapt the connectivity of triangular meshes have been studied more extensively, especially for the purpose of surface simplification, cf. Chapter 4 in [19]. For deformable triangular meshes, Lachaud and Montranvert define a full set of topology adaptation operations including Eulerian operations of vertex

addition and removal as well as non-Eulerian operations to adapt the genus of the underlying surfaces [44].

McInerney and Terzopoulos suggest complete re-parameterization of the mesh instead of local, mesh-based operations for topology adaptation [57]. They decompose image domain into tetrahedral cells usually larger than the voxels of the image. The mesh topology is then re-created by using an iso-surface algorithm (cf. Chapter 2.3.1) based on a tetrahedral decomposition. The required signs of the function defining the implicit surface at every vertex of every tetrahedron are computed based on the original mesh.

4.3 Adaptation of the geometry

Geometric operations alter mexels positions without changing adjacency relations between them. Because they leave the connectivity of the mesh unchanged, we use symbol $\mathbf{W}$ for the whole mesh and assume that the graph of the mesh is known. The adaptation of the mesh geometry can be controlled in two differing ways. We call these the force based approach and the energy based approach. The force based approach stems from the physics based interpretation of deformable models (Chapter 3.5) and the energy based approach stems directly from the energy minimization framework described in Chapter 3.2. The two approaches can be considered as truly different in the sense that the traditional dualism (derived from variational calculus) between the energy and forces does not necessarily hold. That is, some force-based methods do not have an energy minimization interpretation and correspondingly some energy minimization algorithms cannot be implemented as force-based methods.

4.3.1 Force based approach

The evolution of the mesh is controlled by assigning an equation of motion to each mexel. In the general case the equation of motion is Newtonian:

$$m\ddot{\mathbf{w}}_i(t) + \gamma\dot{\mathbf{w}}_i(t) = \lambda\mathbf{F}_{int}(\mathbf{w}_i(t)) + \beta\mathbf{F}_{ext}(\mathbf{w}_i(t)), \qquad (4.1)$$

where $\lambda, \beta, \gamma, m \in \mathbb{R}$ and t is time. The purpose of the internal force $\mathbf{F}_{int} : \mathbb{R}^3 \to \mathbb{R}^3$ is to impose image independent soft constraints on the shape of the mesh to guarantee, for example, the smoothness of the resulting surface mesh. The external force $\mathbf{F}_{ext} : \mathbb{R}^3 \to \mathbb{R}^3$ draws the mesh towards salient image features. Discretizing Eq. (4.1) with respect to time yields (assuming $m = 1$)

$$\mathbf{w}_i^{t+1} = \mathbf{w}_i^t + (1 - \gamma)(\mathbf{w}_i^t - \mathbf{v}_i^{t-1}) + \lambda\mathbf{F}_{int}(\mathbf{w}_i^t) + \beta\mathbf{F}_{ext}(\mathbf{w}_i^t), \qquad (4.2)$$

with the initial mesh when time $t = 0$ given. When $\gamma = 1$, Eq. (4.2) reduces to a Lagrangian equation of motion. A more general equation of motion results when global forces acting on mexels are also considered. The force fields relating to the global forces are parameterized with only a few parameters and the framework is described in [59].

A simple internal force is defined as

$$\mathbf{F}_{int}(\mathbf{w}_i) = \frac{1}{3} \sum_{j=1}^{3} (\mathbf{w}_{i_j} - \mathbf{w}_i). \qquad (4.3)$$

The internal force (4.3) causes surface to shrink to a point if no external force is present. This is sometimes useful, because it reduces sensitivity to the initialization provided that initial meshes are set outside the surface of interest. Other choices for internal forces can be found in [17].

The construction of external force fields based on image data is challenging because external forces should guide the initial surface mesh to a good representation of the surface of interest. Advanced methods for constructing external force fields have been studied e.g. in [17, 83, 84]. These three methods are also summarized in [Publication III].

4.3.2   Energy based approach

In the energy based approach one minimizes the energy of the deformable mesh. The energy function is defined for every mesh in the set of admissible meshes and the resulting surface mesh is the minimum argument of this function. However, before defining the energy function, we consider simplex meshes with a global position parameter introduced in [Publication I].

4.4   Simplex meshes with a global position parameter

We consider simplex meshes $\mathbf{W}$ whose coordinate system is different from that of the image. Our motivation is to derive a common framework for several variants of an optimization algorithm to be introduced in the next Chapter.

Mexels $\mathbf{w}_i$ of $\mathbf{W}$ are set relative to *a reference point* $\mathbf{g}_W \in \mathbb{R}^3$. Actual positions of mexels in an image are $\hat{\mathbf{w}}_i = \mathbf{w}_i + \mathbf{g}_W$. A surface mesh is then represented as a pair $(\mathbf{W}, \mathbf{g}_W)$, where $\mathbf{W}$ is referred to as the *surface centered mesh*. The set $\hat{\mathbf{W}} = \{\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_N\}$ is the unique (image centered) *actual mesh* induced by $(\mathbf{W}, \mathbf{g}_W)$. Note that different pairs consisting of a surface centered

mesh and an associated reference point may induce exactly the same actual mesh.

We allow three distinct interpretations for reference points:

- We consider meshes only with a certain fixed reference points. In this case, we say that reference points are *fixed*. Setting the fixed reference point $\mathbf{g} = \mathbf{0}$ yields standard simplex meshes.

- We consider meshes with all the possible reference points, i.e. we treat the reference point as a variable independent of the accompanying surface centered mesh. In this case, reference points are *floating*.

- Given an actual mesh $\hat{\mathbf{W}}$, the reference point $\mathbf{g}_W$ can be a function of $\hat{\mathbf{W}}$. Here, we consider only reference points defined as

$$\mathbf{g}_W = \frac{1}{N} \sum_{i=1}^{N} \hat{\mathbf{w}}_i \qquad (4.4)$$

and we say that reference points are *constrained*.

4.5 Energy functions

The energy of the deformable mesh $(\mathbf{W}, \mathbf{g}_W)$ is now given by

$$
\begin{aligned}
E(\mathbf{W}, \mathbf{g}_W) &= \lambda E_{int}(\mathbf{W}) + (1 - \lambda) E_{ext}(\hat{\mathbf{W}}) \qquad (4.5) \\
&= \frac{1}{N} \sum_{i=1}^{N} [\lambda E_{int}^i(\mathbf{w}_i | \mathbf{w}_{i_1}, \mathbf{w}_{i_2}, \mathbf{w}_{i_3}) + (1 - \lambda) E_{ext}^i(\mathbf{w}_i + \mathbf{g}_W)],
\end{aligned}
$$

where $E_{int}(\cdot)$ is the internal energy, $E_{ext}(\cdot)$ is the external energy, and $\lambda \in [0, 1]$ is the regularization parameter. The internal energy controls the shape of $\mathbf{W}$. The external energy couples the actual surface mesh $\hat{\mathbf{W}}$ with salient image features. The parameter λ controls the trade-off between the external energy and the internal energy in such a way that incrementing the value of λ results in more weight to the internal energy.

4.5.1 Internal energy

The internal energy for the mexel $\mathbf{w}_i$ is

$$E_{int}^i(\mathbf{w}_i | \mathbf{w}_{i_1}, \mathbf{w}_{i_2}, \mathbf{w}_{i_3}) = \frac{||\sum_{j=1}^{3} \alpha_{i_j} \mathbf{w}_{i_j} - \mathbf{w}_i||^2}{A(\mathbf{W})}, \qquad (4.6)$$

where $\alpha_{i_j} \in \mathbb{R}$ are shape parameters and $A(\mathbf{W})$ is the area of the mesh $\mathbf{W}$. The area of the simplex mesh is computed by triangulating its faces.

The shape parameters describe the expected shape, or the reference shape, for deformable surface meshes. The shape parameters can be acquired from a given example mesh similarly as they were generated from example contours in the 2-D case [46]. However, as the construction of example meshes of the pre-defined quality is significantly harder than the construction of example contours, this might be cumbersome in the 3-D case. Instead, it is possible to analytically estimate the shape parameters of relatively simple reference shapes. The estimation is based on local properties of reference shapes and a surface mesh satisfying exactly these properties does not necessarily even exist. This relates to Mallet's *Discrete Smooth Approximation* (DSA) [53]. The mesh of the minimum internal energy is such that from the existing meshes its local properties best match to those which are posed by the shape parameters.

In [Publication I], two kinds of the analytically derived shape parameters were introduced. The simpler, thin-plate shape parameters are

$$\alpha_{i_j} = \frac{1}{3}, \tag{4.7}$$

for all i, j. These parameters state that the optimum position for each mexel is in the mass-centre of its neighbours. More complex, sphere shape parameters, are

$$\alpha_{i_j} = \frac{1}{3\cos(2\arctan\frac{2\sqrt{\pi\sqrt{3}}}{3\sqrt{N}})}, \tag{4.8}$$

for all i, j, where N is the number of mexels. Roughly speaking, these parameters set the optimal shape of the deformable mesh to be the sphere. For derivation and more detailed interpretation of the sphere shape parameters, see [Publication I].

The internal energy is scale invariant due to the normalization by the area of the mesh and it is straight-forward to see that the internal energy is invariant to the rotations of the mesh. However, the internal energy of a surface centered mesh is translation invariant only if

$$\sum_{j=1}^{3} \alpha_{i_j} = 1,$$

for all $i = 1, \ldots, N$. This is the case with the thin-plate shape parameters but not with the sphere shape parameters. However, when interpreted in the terms of actual meshes and when reference points are floating or constrained,

the internal energy is translation invariant also for the sphere shape parameters. Here lies the usefulness of constrained reference points. Floating reference points can also be used for incorporating the possibility to study translations of the mesh during the optimization of its geometry.

4.5.2 External energy

The input for the deformable mesh, an image I, is a preprocessed version of the image to be analyzed. Image to be analyzed is denoted by I^*. In I, voxels are given an intensity value based on their saliency inferred from local characteristics of I^*. We also normalize I to have intensity values from 0 to 1 with the voxel of the greatest saliency having the intensity value of 1. Based on the input image I, the mexel-wise external energy is defined simply as

$$E_{ext}^i(\mathbf{w}_i) = 1 - I(\mathbf{w}_i). \tag{4.9}$$

As relation between I and the external energy is simple, the input image I can be called *the energy image* as in [Publication IV] and in [Publication V].

Preprocessing is always an application specific task. If one is interested in locating surfaces defined by edges, a simple choice for the input image is

$$I = ||\nabla I^*||, \tag{4.10}$$

where the gradient can be computed by the three-dimensional Sobel operator [86].

Several other choices for the external energy function have been presented in the literature. For example, if the input image I is binary valued, one can set

$$E_{ext}^i(\mathbf{w}_i) = \frac{1}{D}||\mathbf{w}_i - \mathbf{c}_i||^2,$$

where $\mathbf{c}_i$ are the coordinates of the voxel centre nearest to $\mathbf{w}_i$ such that $I(\mathbf{c}_i) = 1$. The constant $D \in \mathbb{R}$ is used for normalizing the range of the external energy. The binary input image I can be obtained by applying an edge-detection algorithm to I^* [9, 58].

4.6 Comparing energies of meshes with different connectivity

Comparing energies of meshes with different number of mexels is not necessarily reasonable. This can be seen e.g. from [Publication II] where the energy

values of some meshes with different resolutions are printed. These show that the energy tends to increase with the mesh resolution. The reason is not due to neglecting the normalization factor $Z_2(\cdot)$ as in Eq. (3.5). Indeed, take a mesh $\mathbf{W}_N$ with N mexels and denote $I(\mathbf{w}_i)$ by x_i. For simplicity, further assume that no voxel contains more than one mexel. Then, by combining Eqs. (4.9) and (3.4), we obtain an expression for the partition function

$$Z_2(\mathbf{W}_N) \;\; = \;\; \int p(I|\mathbf{W}_N)dI \tag{4.11}$$

$$= \;\; \int_0^1 \cdots \int_0^1 e^{(-(1/N)(\sum(1-x_i)))} dx_1 \cdots dx_N \tag{4.12}$$

$$= \;\; (N(1 - e^{-1/N}))^N. \tag{4.13}$$

The value of the partition function depends on $\mathbf{W}_N$ only via N and we write $Z_2(\mathbf{W}_N) \doteq Z_2(N)$. Furthermore, ignoring the partition function from computations do not lead to trouble. This can be seen from Fig. 4.3 where the logarithm of $Z_2(N)$ is plotted against different values of N.
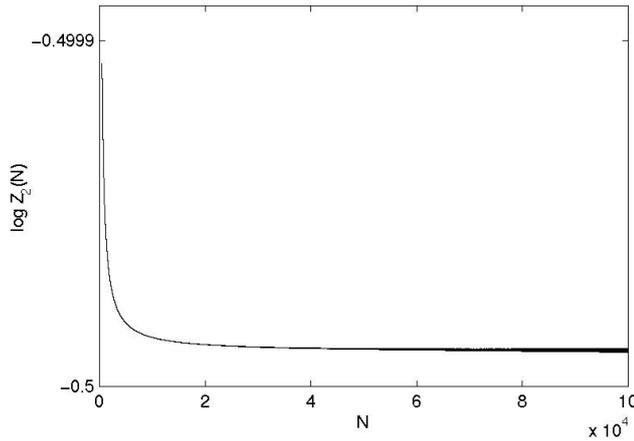
Figure 4.3: $\log Z_2(N)$ for values of N from 500 to 100000.

The problem is more fundamental. The external energy defined by Eq. (4.9) takes only a part of the image into account, namely those voxels in which mexels are situated. Obviously, we should take somehow the whole image into account to be able compare meshes discretized with different resolutions based on their energy. If it was possible to specify the parametric forms of the pdfs for intensity values of voxels belonging to the background and of voxels belonging

the object of interest, the use of regional information could remedy the problem [11, 22]. However, often it is not a trivial task to specify the required parametric forms of the pdfs. This is because all the intensity values relating to the object of interest (or the background) are not necessarily drawn from the same distribution. Our formulation of the external energy, Eq. (4.9), is more flexible as regarding to this issue. Particularly, we do not need knowledge about the parametric forms of the pdfs describing intensity values in images when constructing the input image for the deformable model.

To summarize this section, comparison of the quality of meshes with different number of mexels is not reasonable based on the energy functions defined in this thesis. This is why the global optimization approach and algorithms to be presented are not suitable for deciding the optimal resolution for extracted meshes. However in many medical imaging applications, especially in the ones we consider in this thesis, the number of mexels in extracted surfaces can be set *a -priori*. Hence, restricting the admissible set for the optimization algorithms to consist only of meshes of a fixed resolution does not present a serious problem.

Chapter 5

Algorithms for Energy Minimization

5.1   Global optimization approach

Minimization of the energy defined in Eq. (4.5) is not a simple task. Because images are assumed to be noisy, the external energy term is most probably multi-modal. Hence, algorithms aimed for local optimization (e.g. gradient descent methods) have as such little use within deformable surface meshes. The problem with them is that the surface extraction result depends too heavily on the provided initialization. In fact, the initialization sensitivity is a major problem also with the force based approach to control deformable meshes and with deformable surface models altogether as explained in Chapter 3.9.

   Several ways to deal with the initialization problem have been proposed. The conceptually simplest way is perhaps trying to provide an initial mesh which is already in a close vicinity of the surface of interest. However, while this may be possible for some applications, it does not provide a general solution to the problem due to the difficulty of the initialization itself. Another possible solution to the problem is to apply such an internal force/energy that results in shrinking surfaces as in Eq. 4.3. However, this leads easily to high sensitivity to parameter values. In other words, the same value of the regularization parameter λ is not necessarily applicable with every image within a specific application. This can prevent unsupervised use of methods relying on a shrinking behaviour.

A solution to the abovementioned problems can be achieved via global minimization of the energy [1]. The strategy has received relatively little attention with deformable surfaces unlike with deformable contours. One reason for this could be that algorithms based on dynamical programming, introduced by Amini *et al.* [2] for deformable contours, are not applicable with deformable surface meshes. This is because it is not possible to create a total order (cf. [29, page 374]) of mexels such that the internal energy of a single mexel would depend only on itself and immediate predecessors and successors of it. Dynamical programming for deformable contours has the pleasing property that it is able to find the global energy minimizer among a finite set of candidate contours in a polynomial time. Such strong convergence results rarely exist for other global optimization algorithms applied in practise [77]. In this thesis a global optimization algorithm is understood as an optimization algorithm capable to overcome local minima.

5.2 Coarse-to-fine minimization

Advanced methods for minimization of the energy of deformable meshes existing in the literature are often based on a multi-resolution or a coarse-to-fine scheme. The basic form of the scheme is depicted in Fig. 5.1. A similar coarse-to-fine approach can be applied also with force-based deformable meshes, see e.g. [44]. Note, that energies of meshes with different resolutions are not compared and the resolution of the final mesh is given as an input to the algorithm, cf. Chapter 4.6.

MacDonald *et al.* have applied deformable surface models with the basic multi-resolution scheme to extract cortical surfaces from MR-images [51]. In [50] the deformable surface meshes were combined with an elastic registration framework. The deformation of surface meshes was achieved by deforming underlying regular three-dimensional grid by minimizing an energy function in a coarse-to-fine manner. The method was demonstrated by segmenting MR-images of the thorax and head. Note, that both abovementioned methods expect their input images to be binary valued.

Yet another type of coarse-to-fine method was presented in [68]. There segments of the surface mesh underwent a deformation which locally minimized the energy of the segment. The iterative deformation was carried out in a hier-

---

[1]The unconstrained global minimum of the energy (4.5) is achieved by finding the voxel of highest intensity value and placing the mesh inside this voxel. That would obviously be undesired. Hence, meshes with area and/or volume lower than some threshold are not considered, although this is not necessarily mentioned explicitly.

Resolution increase

```
                        ┌──────────────┐                  ┌──────────────┐
┌──────────────┐        │              │      LMA         │              │
│ Initial coarse│───────▶│ Initialization│─────────────────▶│ Result of the│
│     mesh     │        │   for a LMA  │                  │     LMA      │
└──────────────┘        └──────────────┘                  └──────────────┘
                                                                  │
                                                                  ▼
                                                          ┌──────────────┐
                                                          │  Final dense │
                                                          │     mesh     │
                                                          └──────────────┘
```
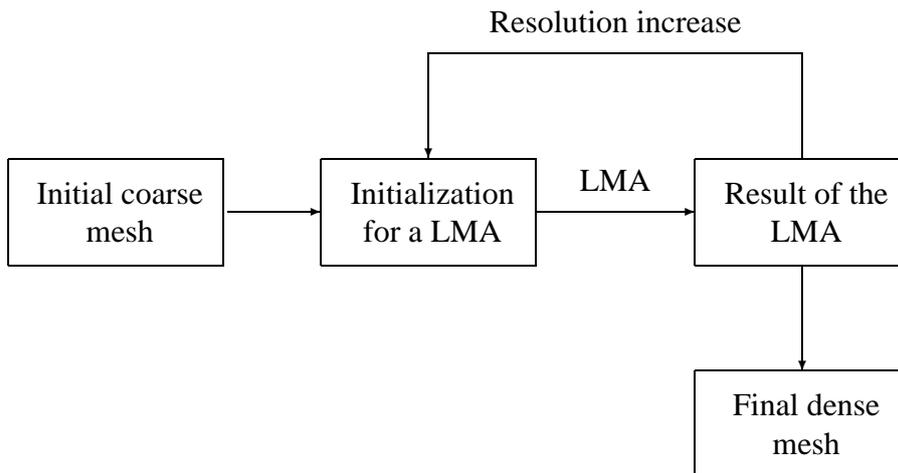
Figure 5.1: The basic multiresolution scheme. The abbreviation LMA stands for local mini-mization algorithm.

archical manner where the surface segments considered became smaller as the process advanced. The deformable mesh was applied particularly for extracting certain brain structures (ventricles and nuclei) from MR images.

All the optimization methods reviewed above are closely tied to a particular application [51] or a particular shape model requiring hand-crafted examples of surfaces to be generated [50, 68]. Although these methods do well with applications they are crafted for, generalizing these methods for other applications could be laborious.

5.3   Dual surface minimization

An algorithm for global optimization of deformable meshes called *Dual sur-face minimization* (DSM) was introduced in [Publication I]. The algorithm was inspired by the dual contour method by Gunn and Nixon [31, 32]. Several vari-ants of the DSM algorithm were developed. These variants were formulated based on different interpretations of the reference points that were introduced in the previous Chapter.

Here we present first the standard DSM algorithm for fixed reference points. Thereafter, we briefly explain modifications required if reference points are as-sumed to be floating. Lastly, a variant called DSM-OS (DSM-Outer surface)

is introduced. This modification has proven to be useful when extracting brain surfaces from PET images. The constrained reference point version of the algorithm is not described here, see [Publication I] for it.

5.3.1 Standard DSM algorithm

The algorithm is initialized with two surface meshes, one placed inside of the surface of interest and the other placed outside of surface of interest. We call these meshes, respectively, the inner mesh and the outer mesh. Both surface centered meshes have the same reference point that is located in the mass centre of the actual meshes. The reference point is fixed, it does not change position during the optimization.

The meshes for the next iteration are the mesh of the lower energy and the one obtained by locally minimizing the energy function starting from the mesh of the higher energy. The local and discrete minimization is performed by a greedy algorithm adapted from [80]. For each mexel, the greedy algorithm studies a set of new candidate positions (including the current mexel position) and selects the position that yields the lowest mexel-wise energy. The set of candidate positions is called *the search space*. The search space is constrained in such a way that the outer (resp. inner) mesh shrinks (resp. expands). The search space for greedy algorithm is 'directed' towards the reference point in the case of the outer mesh, or away from the reference point in the case of the inner mesh. Check for Fig. 5.2 for a depiction of the search space and [Publication I] for the exact definition of the search space. Directing the search space based on the reference point instead of local normals of the surface helps to avoid surface self-intersections. After the greedy algorithm has updated positions of all the mexels, the energies of the meshes are compared again and the greedy algorithm is initiated from the mesh of the higher energy. The DSM algorithm is iterated until the volume of the inner mesh has exceeded the volume of the outer mesh. At this point, the mesh with the lower energy is returned as the result of the algorithm. Obviously, this mesh has the lowest energy of those encountered during the algorithm.

The greedy algorithm, due to its local nature, can produce its initialization also as its result. In this case, the DSM algorithm is trapped in a local minimum and it has to be helped to escape from the minimum. The energy function of the mesh of the higher energy is then modified by adding a penalty for the current mexel positions to enable the escape from the local minimum. The penalty is increased gradually in small steps until the mesh has been forced out of the local energy minimum.

We are unable to prove that the DSM algorithm or variants would converge
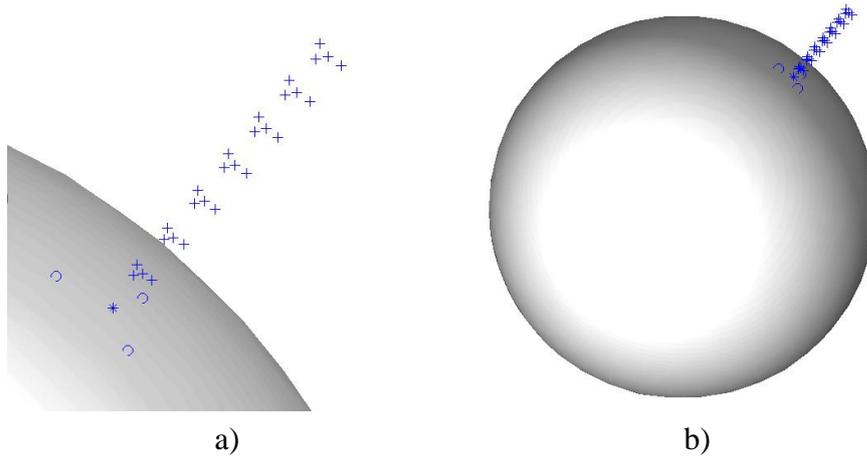
Figure 5.2: Depiction of the search space for the greedy algorithm optimizing the inner mesh. In a) a close-up from a global view in b) is shown. The position of the mexel marked by a star ($*$) is to be optimized. The positions marked by a plus sign ($+$) belong to the search space in addition to the current mexel position. The neighbours of the mexel marked by a star are marked by a circle ($\circ$).

to the (constrained) global minimum and for deformable mesh optimization, it is probably more important that the algorithm can overcome local energy minima. In the Appendix, we show that the algorithm will terminate when certain restricting assumptions are made. Experimentally, the algorithm is found to stop also when these assumptions are not made.

5.3.2   Floating reference point algorithm

The floating reference point (DSM-FRP) algorithm differs from the standard DSM algorithm in two aspects. Firstly, and more importantly, we consider surface centered deformable meshes with several possible reference points. Furthermore, these are not tied to the actual surface meshes. Secondly, the method to drive meshes out of local energy minima is different.

The algorithm works just as the standard algorithm expect that in each iteration several greedy algorithms are initiated. These all are initialized with the current surface centered mesh, but with different reference points. These reference points are selected from the neighborhood of the current reference point. Then, the mesh with the lowest energy resulting from greedy algorithms is selected as the result of that iteration of DSM-FRP algorithm. Note that the inner mesh and the outer mesh may have different reference points.

The DSM-FRP algorithm is in a local energy minimum if after an iteration both the surface centered mesh and the reference point are same than at the previous iteration. If a mesh is trapped in a local energy minimum, the surface mesh with the higher energy is simply shrank (outer mesh) or expanded (inner mesh). Shrinking or expansion of a mesh is realized by multiplying a mexels of the surface centered mesh with a suitable constant.

The DSM-FRP modification outperforms the standard DSM-algorithm when the images are very noisy. It is helpful for DSM-FRP if the shapes of the initial surface meshes remind the shape of the surfaces of interest, e.g. all of them are spheres.

5.3.3   Single surface modification, DSM-OS

Recall that reference points were applied to generate search spaces that guarantee shrinkage of the outer surface mesh and expansion of the inner surface mesh. Therefore, it is not necessary to apply two surface meshes with the DSM-algorithm as it was with the dual contour method. This property of the DSM algorithm leads to another modification of it, which we call the *outer surface modification* and abbreviate as DSM-OS. In DSM-OS, only one of the surfaces, the outer, is allowed to move. The other (inner) surface is just for deciding whether to terminate the algorithm and only the knowledge about its volume is required. The DSM-OS algorithm works just as the standard DSM-algorithm except obviously it has to store the surface mesh with the lowest energy encountered so far. It returns the mesh of the lowest energy of those encountered during the iterative algorithm. In a similar fashion, the inner surface modification of the algorithm, where only the inner surface would be allowed to change position, could be created.

The rationale for the DSM-OS modification is that sometimes it is known that approaching surface of interest from a certain direction (inside or outside) is favorable. For example, when extracting brain surfaces from PET images, it is better to approach the surface of interest from outside because noise levels are lower there (see Chapter 7 and [Publication I]).

However, since the DSM-OS algorithm has to force the mesh out of the 'global' minimum, it often requires more iterations to converge than the standard DSM algorithm. Thus, the standard algorithm can be considered to be more efficient. The standard DSM algorithm is also a better choice when no substantiated knowledge about the favorable direction of approach exists.

5.3.4   Demonstration

We illustrate the results obtainable with the DSM algorithm by extracting a surface from a noisy synthetic image with dimensions of $64 \times 64 \times 64$. Some cross-sections of the synthetic input image are shown in Fig. 5.4 (a). The initialization for the DSM algorithm is depicted in Fig. 5.4 (b) and differences between extracted surface and the true surface of interest are depicted in Fig. 5.4 (c). Three-dimensional visualizations of the true surface and the surface extracted from the noisy input image are shown in Fig. 5.5. The energy function minimized by the standard DSM algorithm consisted of the external energy as in Eq. (4.9) and the internal energy as Eq. (4.6) with the thin-plate shape parameters. The regularization parameter λ was set to 0.3. Values of the energy function of inner and outer meshes during the iteration of the standard DSM algorithm are shown in Fig. 5.3.
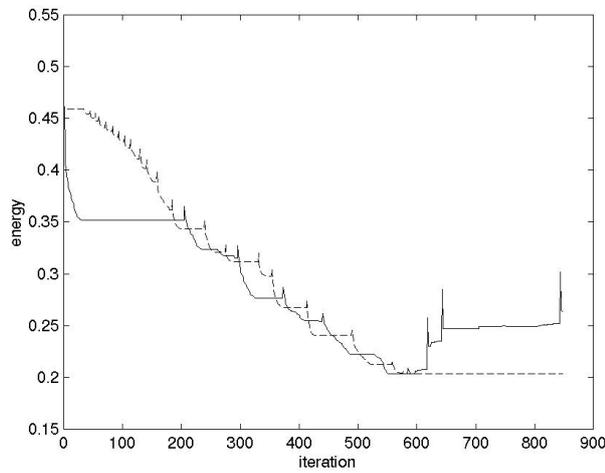
Figure 5.3: Energies of inner and outer meshes in each iteration of the DSM algorithm. Solid line represents the energy of the outer mesh and the dashed line represents the energy of the inner mesh.

5.4   Hybrid of a GA and a greedy algorithm, GAGR

5.4.1   Description of the algorithm

A hybrid of a genetic algorithm (GA) and a greedy algorithm for deformable mesh optimization was presented in [Publication II]. Genetic algorithms [25]

(a)	(b)	(c)
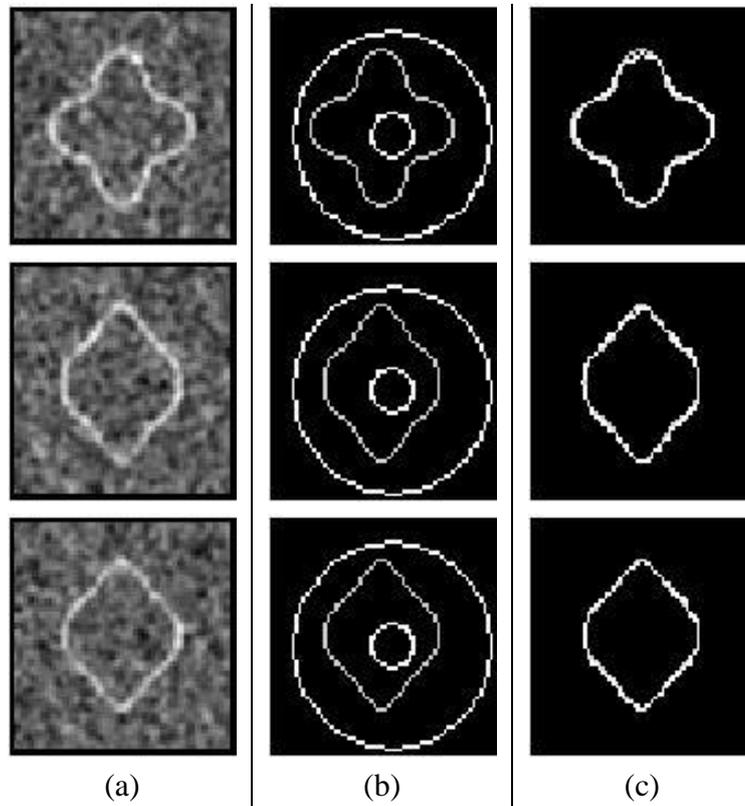
Figure 5.4: The input image (a), the dual initialization (b), and the extraction result using the DSM-algorithm (c). From top, the central cross-sections in xy, xz and yz planes are shown. In (b), cross-sections of the true surface of interest are shown in gray and cross-sections of initial surfaces for the DSM are shown in white. In (c), the true surface of interest is in gray and the extraction result is in white.

(a)   (b)
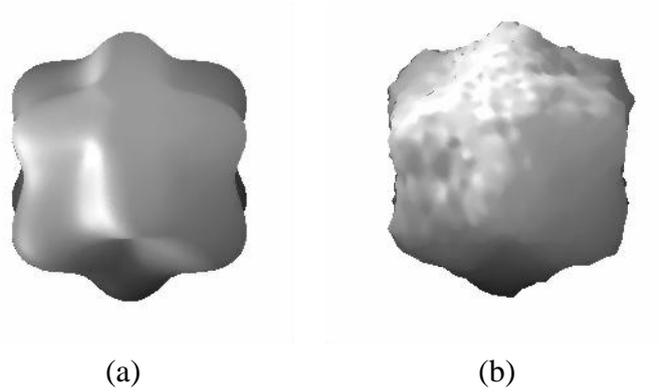
Figure 5.5: The extraction result. (a) the true surface and (b) the extracted surface from the noisy input image shown in Fig. 5.4. Note that the true surface is visualized based on the continuous expression of it while surface extraction was performed based on discrete data.

have been successfully applied to many challenging optimization problems. GAs are based on the genetic processes of biological organisms. Mimicking the natural selection and reproduction an initial population evolves to the solution of the problem at the hand. The basic structure of a GA is presented in Algorithm 1.

Algorithm 1

$t \leftarrow 0$
initialize a population $P(t)$ of surface meshes
evaluate $P(t)$ by computing the energy of each individual in it
while NOT termination condition **do**
  $t \leftarrow t + 1$
  select $P(t)$ from $P(t-1)$
  recombine $P(t)$
  evaluate $P(t)$
end while

We applied a GA as the first step of the optimization algorithm comprised of four steps:

1. Global minimization of the energy (Eq. 4.5) of the mesh by a GA;

2. Local minimization of the energy (4.5) by a greedy algorithm [80] starting from the mesh obtained from first step;

3. Adaptation of the resolution of the mesh by T_2^2-operator (cf. Chap. 4.2);

4. Minimization of the energy by the greedy algorithm starting from the mesh of the adapted resolution;

The method is abbreviated as GAGR later on. Note that only the standard simplex meshes, i.e. simplex meshes with the fixed reference point $\mathbf{g} = \mathbf{0}$, are considered.

For the first step, we applied a real coded genetic algorithm (RCGA) with the BLX-α crossover-operator [20, 35]. RCGAs represent variables (meshes in this case) with vectors consisting of floating point numbers. On the other hand, with traditional binary coded GAs variables are represented with binary strings. RCGAs are favored here because they can be more efficiently adapted to the particular problem domain, indeed we expect mexel positions to be real valued. The RCGA applied was terminated when 100 generations, each comprising of 4000 meshes, were evaluated. The early termination of the RCGA was for efficiency, usually GAs evolve for greater number of generations than 100. Early termination was reflected in the selection of the value for the parameter α, which was set to 0.3. Herrera *et al.* recommended setting α to 0.5 to avoid premature convergence [35]. By using the value $\alpha = 0.3$, convergence to a good initialization for the greedy algorithm was obtained during only 100 generations of RCGA.

See [Publication II] for a more detailed description of the algorithm. As with the coarse-to-fine scheme described in Chapter 5.2, energies of meshes with different resolutions are not compared in the GAGR algorithm either.

5.4.2 Demonstration

We illustrate the GAGR algorithm by applying it for the same surface extraction problem as the DSM algorithm was applied for in Chapter 5.3. The energy function to be minimized was exactly the same than in the case of the DSM algorithm. Meshes after each intermediate step of GAGR are shown Fig. 5.6. The extracted surface is compared to the true surface in Fig. 5.7. The values of the energy function after each intermediate step are listed in Table 5.1. Note that increment of the mesh resolution had the effect of increasing the energy of the mesh. The reasons behind this were briefly discussed in Chapter 4.5.

Step 1   Step 2

Step 3   Step 4
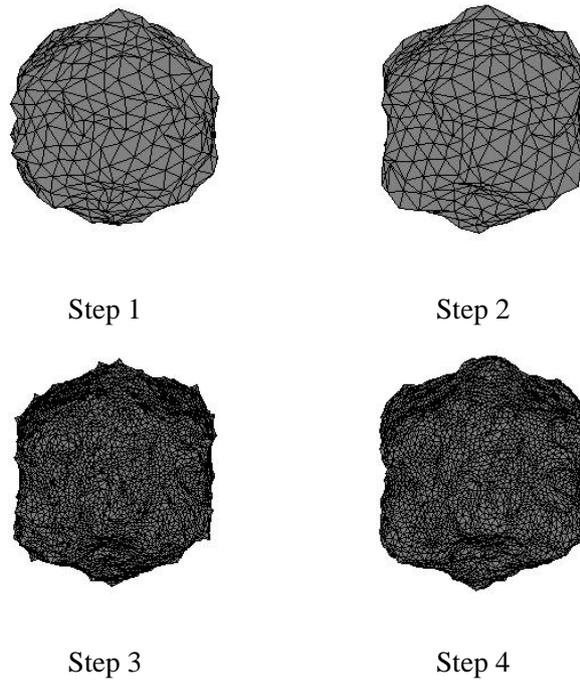
Figure 5.6: Meshes after each step of GAGR. Simplex meshes have been triangulated for the visualization.
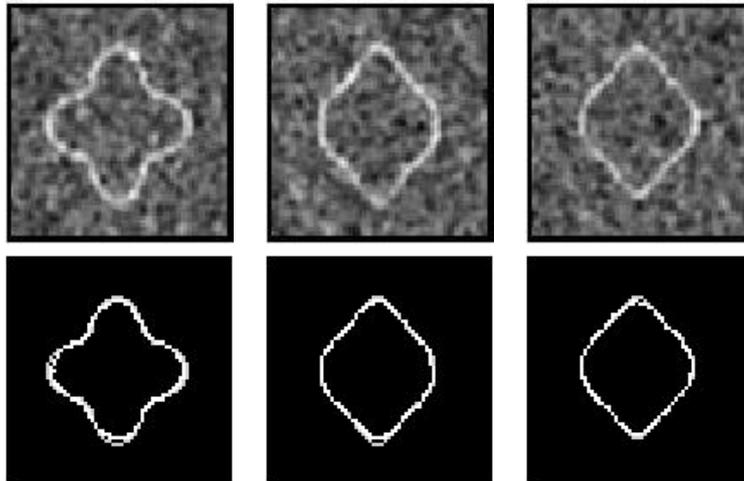
Figure 5.7: Cross-sections of the extracted surface using GAGR. Top row: The input image for the deformable model. Bottom row: The extraction result, the true surface of interest is in gray and the extraction result is in white. From left, the central cross-sections in xy, xz and yz planes are shown.

Table 5.1: The energies after each intermediate step of the GAGR algorithm. The energy after the step 0 is the lowest of energies of initial randomly generated sphere meshes.

Step	0	1	2	3	4
Energy	0.3642	0.2180	0.1680	0.2734	0.1961

Chapter 6

Comparison with the Force Based Approach

6.1 Objectives and methodology

A comparative study between the standard DSM algorithm, the GAGR algorithm, and a few recent force based methods [17, 83, 84] to control the mesh deformation was presented in [Publication III]. The purpose of the study was to find some characteristics of the methods as compared to others and to find out whether energy and force based schemes differed in any general way. Of course, ranking rather differing deformable models is not reasonable with an possible exception of a highly specialized application, and this was not the purpose of the study.

The construction of the external force field with force based deformable meshes can be regarded as an equivalent task to optimization with energy based deformable meshes. Hence, only these aspects were compared while other features (like the internal energy) of the applied deformable models were kept as simple as possible.

The approach of the study was to supply an image containing the surface of interest and a reasonable initialization for deformable models and see how well the surface of interest was captured by different methods. The internal force for each force based method was the same, only values of the model parameters and ways to construct the external force field based on image data were varied. Similarly, for the energy based methods, energy functions (besides the value of the regularization parameter) were same and only methods to minimize them were varied. The internal energy was as in Eq. (4.6) with the thin plate shape parameters (4.7). The internal force was as in Eq. (4.3) and it can be regarded
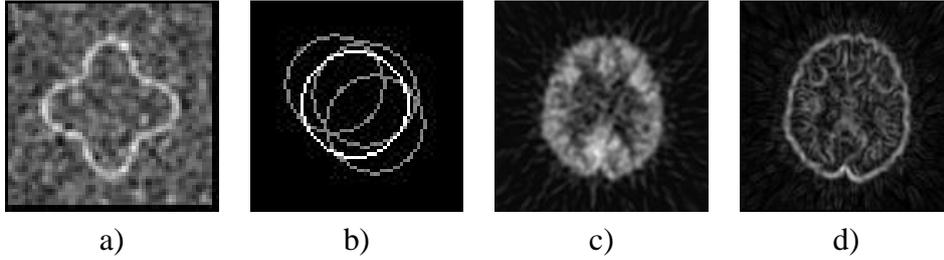
Figure 6.1: Examples of cross-sections of test images. a) The first image set consisted of surfaces degraded with white Gaussian noise of varying variances which were then blurred with a Gaussian filter. b) The second image set contained images where, in addition to the correct surface (drawn with intensity value of 1.0), were some false surfaces (drawn with the intensity value of 0.5). c) and d) PET-images form where the brain surfaces were to be extracted, in c) a transaxial cross-section of a PET image is shown and d) the corresponding input image computed by using Eq. (4.10) is shown.

as the counterpart of the applied internal energy function.

The DSM algorithm featured in the comparative study did not have all the properties described in [Publication I]. Namely, the oscillation control was missing. Embedding the oscillation control in the DSM algorithm removed some problems of the algorithm and improved its results in some cases, cf. [Publication I].

6.2 Material

The test material was composed of three image sets. Two image sets contained synthetic images and the third one contained FDG-PET brain images. The brain surface was to be extracted from PET images. Examples of image cross-sections are shown in Fig. 6.1, for details on the image sets cf. [Publication III].

With synthetic images, two criteria were used for quantitative evaluation of extraction results. The set of voxels belonging to true (digital) surface is denoted by TS and the set of voxels belonging to the extracted surface is denoted by ES. Furthermore, let the set of voxels inside the true surface be TV and the set of voxels inside the extracted digital surface be EV. Then,

$$\epsilon_1 = 1 - \frac{|TV \cap EV|}{|TV \cup EV|}, \tag{6.1}$$

$$\epsilon_2 = \max_{X \in TS} \min_{Y \in ES} dist(X, Y), \tag{6.2}$$

where distance between voxels X and Y ($dist(X, Y)$) is the Euclidean distance between their integer coordinates. The first error measure (6.1) captures well the goodness of overall result. The quantity $1 - \epsilon_1$ is known as the Tanimoto coefficient, which is a criterion that is often applied for the evaluation of segmentation results. The second criterion (6.2) reacts also to very local errors in surface reconstruction. This makes it useful in determining if high curvature parts of a surface are extracted well. However, the criterion (6.1) is perhaps more informative than (6.2) when extracted surfaces are applied for segmentation purposes.

6.3   Results

The quantitative results using the criteria defined above were presented in [Publication III] and they are not repeated here.

The GAGR algorithm performed well with the first set of images. Quantitative results obtained with it were similar to those obtained with the best of the force based methods (generalized gradient vector flow (GGVF) [83]). The standard DSM algorithm was as accurate as GGVF or GAGR when the noise level in the images was low. When the level of noise was increased , the quality of results obtained with DSM however degraded more than the quality of those obtained with GAGR and GGVF.

With the second set of images GAGR performed only slightly worse than the best force-based method, which this time was gradient vector flow GVF [84]. (GGVF did not perform well this time). The standard DSM algorithm experienced problems with some images. However, these problems turned out to be avoidable by adding a mechanism called oscillation control to the algorithm. When the oscillation control was designed and implemented, DSM performed well with the second image set as can be seen from results listed in [Publication I].

Extracted brain surfaces from PET images were of excellent quality in visual inspection when DSM or GGVF was applied. Accurate surface extractions with GGVF required a little bit more parameter tuning than with DSM, otherwise there were only minor differences between the methods. GAGR produced rather good results, however they were clearly less accurate than those with DSM or GGVF. With the other force based methods than GGVF, reliable brain surface extraction was not achieved.

The main advantage of the energy-based scheme over the force-based one seemed to be their reduced sensitivity to parameter values. Indeed, all the results of energy-based methods were obtained with the same parameter set al-

though analyzed images were quite divergent. This was not the case with force-based methods that required parameter tuning in between the experiments. Initialization sensitivity was a big problem only in few cases, like for the GGVF method with the second set of images. However, none of the methods was completely insensitive to its initialization. In other words, different initializations led usually to different results. These results were nevertheless of a similar quality in most cases.

Chapter 7

Deformable Meshes for Analysis of PET Brain Images

7.1 Positron emission tomography

Positron Emission Tomography (PET) is an imaging method to study physiological processes in living organs [63, 69]. In PET, tracer molecules are administered to the subject. A tracer is recognized by the body and it takes part in metabolic processes. It is also labeled by an unstable radionuclide. Unstable label nuclides emit positrons in order to achieve stability. These emissions are then detected through the annihilation process by a PET scanner. The rate of these emissions coming from a certain part of the body then relate to the concentration of the tracer molecule within that part of the body. Moreover, positron emitting compounds are produced in cyclotrons and they do not exist as such in nature. This removes the problems related to background radiation and allows gathering quantitative information about subject's physiology. Obviously, information that can be obtained depends on the applied tracer. For example, FDG (fluoro-2-deoxy-D-glucose) emulates glucose and hence FDG-PET images offer information about the glucose consumption. The FDG tracer has uptake all over the brain volume and the highest uptake of it is in cortical region of the brain. Besides tracers describing metabolic functions, there exist receptor type tracers. These have uptake mainly in the structures having highest density of the corresponding receptors. The receptor type tracer of interest here is ^{11}C labeled Raclopride (later on just Raclopride). Its uptake is mainly in the striatum.

7.2 Automatic surface extraction from PET brain images

7.2.1 Introduction

Increasing use of PET has created a need to develop automatic means for analysis and processing of PET images. In brain imaging, a PET image can be registered (rigidly) with the MR image of the same subject to allow gathering of physiological information corresponding to anatomical structures [39]. If the segmentation of MR image and image-registration are carried out in an automatic manner, this provides a way to automate the analysis of PET images. Other approach for analyzing PET images is to compare images from patient groups statistically, like in *statistical parametric mapping* [23].

However, it could be useful to extract surfaces from individual PET images without relying on the corresponding anatomical image as explained in [66] and in [Publication V]. This could be helpful for image registration and compensation for patient movement during a dynamic PET study. Reliable extraction of functional structures [1] directly from PET images could be useful also for quantitative analysis of PET images. However, as it is still rather unclear how this kind of analysis should be carried out and what are its possible applications and limitations, this is not speculated further in this thesis.

7.2.2 Challenges in PET image segmentation

As PET images are noisy, automatic surface extraction from them is difficult. Although advances in image reconstruction methods for PET have somewhat simplified the problem (cf. [1]) the images still have quite poor contrast to noise ratio and automating even simple segmentation tasks, such as the extraction of the brain surface, remains challenging.

Other challenge in processing of PET images follows from the fact that images acquired with different tracers are indeed quite divergent. For example, by comparing the FDG-PET image in Fig. 7.1 (a) to the Raclopride PET image shown in Fig. 7.2 (a) and (b) it can be seen that exactly the same procedure for brain surface extraction would not probably work for both tracers. However, saying this does not exclude the possibility that similar general principles could be used for processing PET images acquired with different tracers.

---

[1] A functional structure is a sub-volume in a PET image which has distinguishable uptake (i.e. image intensity) from its surroundings and some biological meaning

   In [Publication IV] and [Publication V] we have studied the extraction of
brain surfaces using the deformable model based on the DSM algorithm. In
[Publication V] the surface extraction procedure for FDG-PET images intro-
duced in [Publication IV] was generalized also for Raclopride PET images. As-
tonishingly, exactly the same surface extraction routine besides the generation
of the input images for the deformable model was adequate for both tracers. In
[Publication V], we further determined the mid-sagittal plane from PET images
by applying the automatically extracted brain surface. The mid-sagittal plane
coarsely divides the brain into the left and the right hemispheres. Hence, the
determination of the mid-sagittal plane makes it possible to study functional
differences between the hemispheres. We also studied the extraction of the
boundary between uptake levels in white matter and gray matter visible in FDG
images using the deformable model. This white matter surface together with
the brain surface delineates a volume that could be called functional cortex. In
the next section, an overview of the methods and results of the PET applications
of the deformable model based on the DSM algorithm is presented.

                  (a)   (b)   (c)
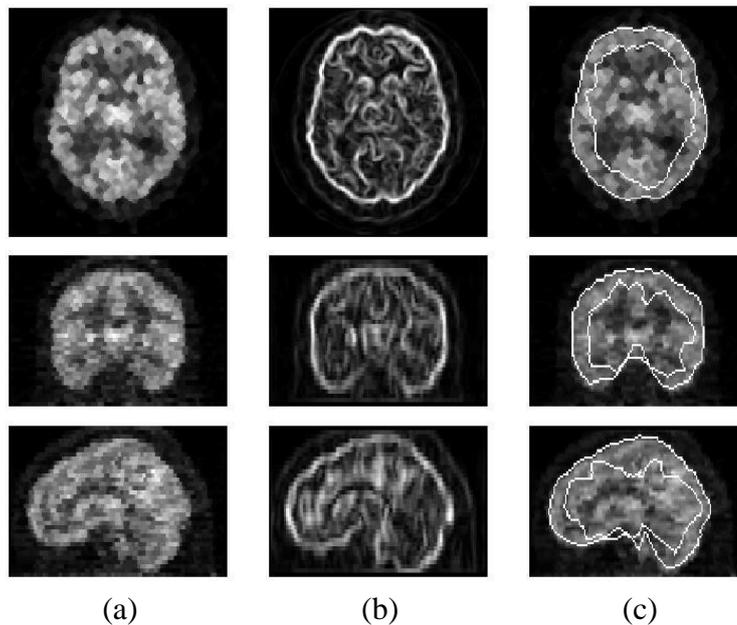
Figure 7.1: FDG-PET images. (a) Example cross-sections from a FDG-PET image. (b) The
input image for the deformable model generated from the image presented in (a). (c) The
extracted brain surface and the white matter surface from the image shown in (a), cf. Sect.
7.3.2 for information about the generation of the input image and about the surface extraction
procedure. From top transaxial, coronal, and sagittal cross-sections are shown.
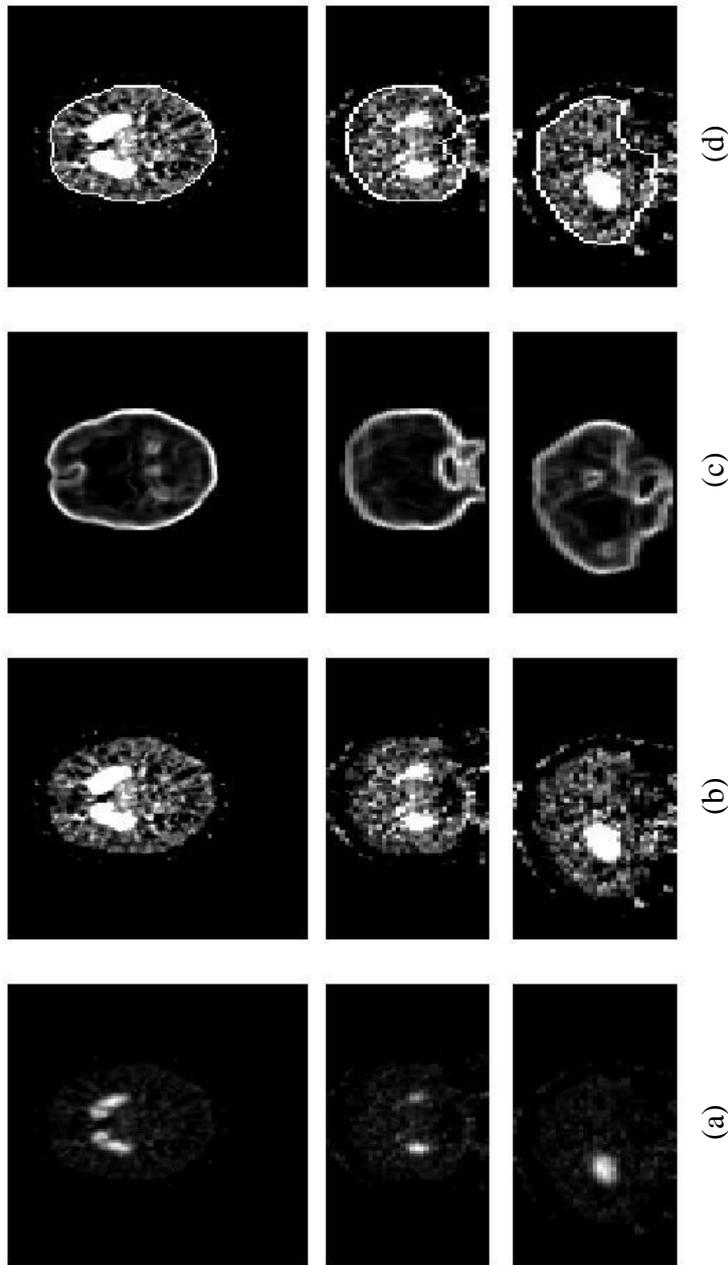
Figure 7.2: Raclopride PET images. (a) and (b) Example cross-sections from a Raclopride PET image. Intensity values are in linear scale in (a) and (b) is a nonlinearly rescaled version of (a) from which uptake levels outside the striatum are better seen. (c) The input image for the deformable model generated from the image shown in (a), cf. Sect. 7.3.2 for information about the generation of the input image and surface extraction. (d) The extracted brain surface from the image shown in (a) and (b). From top transaxial, coronal, and sagittal cross-sections are shown.

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

7.2.3   Methods and results

Material

The material for the study consisted of a FDG-filled Hoffman brain phantom (JB003, Nuclemed N.V./S.A., Roeselare, Belgium) and 17 FDG and four Raclopride brain images of healthy volunteers. Structures corresponding to cerebellum, cortex, basal ganglia and ventricles are represented in the phantom. All the PET acquisitions were made with GE Advance scanner (GE, Milwaukee, USA). The FDG-PET images were reconstructed with the iterative MRP method to the image cross-section size of 128 by 128 [1]. The Raclopride PET images were reconstructed with the FBP method to the image cross-section size of 128 by 128. Both types of images consisted of 35 transaxial cross-sections. The pixel by pixel Patlak model [65] was applied to the FDG sinograms to produce parametric images to be used for structure extraction. The Raclopride images were calculated to parametric images showing the Raclopride binding with a simplified reference model [30].

Definitions of input images

As mentioned in Chapter 4.5.2, image processing operations required for creating an input image for the deformable model are always application specific. Intensities in the input image should ideally be high at the voxels belonging to surface of interest and low elsewhere in the image. With FDG-PET images, edges in the original images seem to be a good and simple feature characterizing the surfaces of interest. We define the input image for FDG as

$$I_{FDG} = ||\nabla I_P^*||, \tag{7.1}$$

where I_P^* is a median filtered version of the original PET image to be processed. The gradient is computed by the three-dimensional Sobel operator [86]. In practise, the input image I_{FDG} can contain few aberrantly large intensity values, which reduce the contrast in other parts of the input image. Therefore, to improve the contrast, a certain (small) percentage of largest intensity values all receive the intensity value 1 in the normalized version of the input image. See Fig. 7.1 (b) for an example of the input image for the deformable model.

The pre-processing stage for Raclopride images is more complicated than for FDG images, see [Publication V] for it. After the preprocessing steps, voxels just outside the brain volume typically have low intensity values in images. Moreover, their values in the gradient magnitude image are expected to be relatively high. Hence, to extract the brain surface from Raclopride images, we

set

$$I_{Raclopride}(\mathbf{x}) = ||\nabla I_P^*(\mathbf{x})||(1 - I_P^*(\mathbf{x})) \tag{7.2}$$

where $I_{Raclopride}$ is the input image and I_P^* is the image after the pre-processing steps. The normalization of input images is nonlinear just as in the case of FDG. See Fig. 7.2 (b) for an example of the input image.

Brain surface extraction

Brain surface extraction was accomplished by using the deformable mesh with the internal energy with the thin-plate shape parameters and the external energy (4.9) calculated from the input images defined in Eqs. (7.1) and (7.2). The regularization parameter λ was 0.3. The DSM-OS algorithm was used for optimization and the initialization for it was created automatically. Same parameters for the deformable mesh were applied with all the images, FDG or Raclopride. Also, the initialization procedure was same for both tracers.

In visual inspection, the extracted brain surfaces from FDG images were accurate in all cases, cf. Fig 7.1 (c) for a typical example. The extracted brain surfaces from Raclopride PET images were also found reliably, cf. Fig. 7.2 (c) for an example. In one case, accuracy of the extracted brain surface could have been better at the lower and upper parts of the brain.

Mid-sagittal plane determination

The mid-sagittal plane of a brain image can be defined as a virtual geometric plane about which the (anatomical) brain in the image presents maximum bilateral symmetry [48]. We do not have knowledge about the exact anatomy of the studied subjects at our disposal but we can assume that the extracted brain surface from functional images is approximately the same as the corresponding anatomical brain surface. Hence, we can determine the mid-sagittal plane based on the extracted brain surface. We do this by adapting a procedure from [48]. Details of our implementation of the procedure and analysis of results can be found in [Publication V]. Examples of the brain surface and the mid-sagittal plane extracted from a FDG-PET and from a Raclopride PET image are shown in Fig. 7.3. All the other results were of approximately same quality, especially the mid-sagittal plane was successfully extracted from all the studied images.

Extraction of functional cortex

In FDG-PET images, the boundary between tracer uptake levels in the gray matter and the white matter is visible. This boundary, called white matter sur-
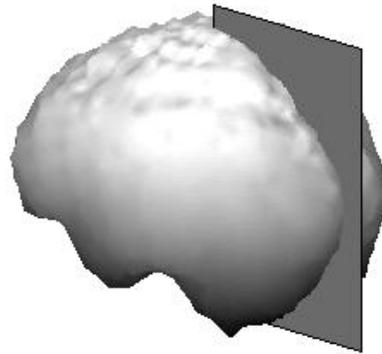
(a)
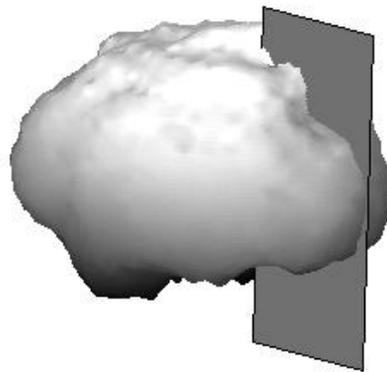
(b)

Figure 7.3: An extracted brain surface from a FDG-PET image (a) and from a Raclopride-PET image (b) and the mid-sagittal planes determined based on these brain surfaces.

face, together with the brain surface define what we call the functional cortex. In [Publication IV] and [Publication V] we have studied the extraction of the white matter surface from FDG-PET images. This was done with the deformable mesh based on the standard DSM algorithm. The energy function was the same as for brain surface extraction except the sphere shape parameters, Eq. (4.8), were applied for the internal energy and the value of λ was 0.2. Initial surfaces were generated automatically based on the extracted brain surface. The optimization was constrained in such a way that the strong energy minimum at the brain surface could not interfere with the optimization process. Example results can be seen in Fig. 7.1. Again for further explanation of the methodology and analysis of results we refer to [Publication IV] and [Publication V].

Chapter 8

Discussion

8.1   Summary of publications

In this section, we shall present a summary of the most important research contributions from each publication featured in this thesis. The methodological ones are then discussed in more depth in the succeeding sections.

In [Publication I], the *Dual surface minimization* algorithm for global optimization of deformable mesh geometry was introduced. Three variants of the standard DSM algorithm applicable in differing situations were also derived in the paper. The standard algorithm is an extension and generalization of the *dual contour method* presented in [32]. However, there are several differences between the dual contour method and the DSM algorithm as explained in [Publication I]. The shape modeling scheme due to Lai and Chin [46] was extended to three-dimensional shapes and the sphere prior shape for surfaces was analytically derived in [Publication I]. Experiments with synthetic as well as with PET images were reported.

In [Publication II] a new global optimization algorithm for deformable meshes (GAGR) was presented. It was tested with synthetic data. The superiority of the algorithm compared to a simple multi-start optimization algorithm was also demonstrated.

In [Publication III] a comparison between some recent methods to optimize the deformable mesh geometry was reported. Although comparison of the methods is a difficult task and results are open to various interpretations, observations about the strengths and weaknesses of different methods were made and reported in the study.

In [Publication IV] and [Publication V] an application of the deformable model from [Publication I] to the analysis of PET brain images was presented.

Segmentation results of PET images were excellent, taking into account the poor contrast to noise ratio of the images and the novelty of the task. Furthermore, the segmentation procedure was fully automatic. We were able to extract brain surfaces from images acquired using two fairly different tracers without any changes in the procedure except for pre-processing of the images. This demonstrates the capability of the deformable model to adapt to different kinds of data. The two studied tracers were FDG and Raclopride. The extracted brain surface from PET images was used for determination of the mid-sagittal plane. Basing this task only on the intensity values in the images could prove challenging at least for the Raclopride tracer.

8.2 Global minimization approach to deformable surfaces

In this study, the energy of deformable meshes has been minimized globally in order to avoid the requirement for a close initialization. A large body of work has been devoted for reducing initialization sensitivity of deformable surface models. Combination of local and global deformations as in [59, 72] and advanced construction of external force fields as in [84, 83] are instances of techniques that address this problem. However, techniques that are applied to the reduction of initialization sensitivity often come with an extra set of parameters and it might not be obvious how to set parameter values within a particular application. This easily leads to the problem of excessive parameter sensitivity, i.e. the difficulty of finding suitable parameter values for processing a large set of images. The parameter sensitivity problem is less addressed in the literature than the initialization problem.

Deformable meshes based on global optimization also come with an extra set of parameters for the user to tune. However, there are possibilities to tune them based on some rather profound principles instead of the trial and error method. The energy function itself should provide a description of surface properties that are desirable. It is often quite easy to quantify these properties via cost functions. Therefore, designing an energy function based on solid principles for a particular application is quite possible without an arduous trial and error cycle. The parameters for the minimization algorithms themselves relate only to them. Hence, tests for finding a good set of parameters for minimizing a particular energy function are easy to design. Some parameters related to the optimization algorithms can be derivable from the (internal) energy function with simple calculations. For example, the selection of good values for search

space parameters for the DSM algorithm is closely related to the internal energy function. Finally, we note that there have been some efforts [21, 22, 45] for automating parameter selection with 2-D active contours.

One disadvantage of the global optimization approach to deformable meshes is that the number of mexels in the surface mesh has to be fixed. This is because the external energies of meshes of differing resolutions are not comparable in a meaningful way (as explained in Chapter 4.5). However, the number of mexels can often be set based on neuroanatomical knowledge for applications in analysis of brain images. Therefore, this drawback of the approach concerning these applications is a minor one.

Global optimization approach to deformable surfaces requires such an external energy function to be found that characterizes globally surface of interest. Practically speaking this means that voxels of surface of interest should feature high intensity values in the input image compared to those voxels belonging to the background. It may not be obvious how to design such an external energy function for a given application and this may be claimed as a disadvantage of the approach. However, basically the same disadvantage holds for all methods trying to solve the initialization sensitivity problem. Also the global characterization of the surface of interest is only required within the set of admissible meshes. A practical example on the simplifying effect of the restricting the set of admissible meshes on the design of the external energy function is white matter surface extraction from PET images [Publication IV], [Publication V]. There, by constraining the set of admissible meshes, exactly the same external energy function could be used for extraction of the white matter surface and the brain surface, although intensities in the input image were much higher for the voxels of the brain surface. Implementation of these kind of constraints is easy, just placing a low (zero) value in the input image to the voxels that are known to belong to the background.

In some applications, it may be practical to search for 'the local minimum closest to the initialization' instead of the global minimum. Obviously, then a close initialization is required. The fundamental problem with this strategy, in addition to the requirement for a good initialization, is that the closest local minimum is only definable with respect to a particular optimization algorithm, it is the minimum to which the algorithm converges to. Further, it is usually not obvious which is the closest local minimum for a given initialization and a local minimization algorithm. Indeed, fractal-like images of the basins of attraction of Newton's method and other nonlinear minimization algorithms are famous examples of unstability of dynamical systems with respect to their initialization [24, 64].

8.3 Dual surface minimization

We have introduced variants of the Dual surface minimization algorithm for deformable mesh optimization in [Publication I]. The standard DSM algorithm yielded good results in the experiments where the noise level in the images was moderate. For more demanding conditions, we introduced the DSM-FRP modification. For using the sphere shape parameters (Eq. (4.8)) without compromising the translation invariance of the internal energy, the DSM-CRP (DSM-Constrained reference point) variant must to be used instead of the standard algorithm. Moreover, for specialized situations where the surface of interest is more favorably approached from outside we introduced the DSM-OS modification. By generalizing the idea of reference points from [46], it was possible to derive these variants of the DSM algorithm in the common framework.

These algorithms were inspired by the Dual contour method of Gunn and Nixon [32], but there are important differences between the dual contour method and the DSM algorithm. An obvious difference is the extension from 2-D to 3-D, but there also are other dissimilarities which were detailed in [Publication I]. We shall now summarize these briefly. In the direct generalization of the dual contour method, each mexel in the mesh of the greater energy would be pushed towards the corresponding mexel in the other surface mesh instead of pushing it towards (or away from) the reference point. This would necessitate the use of two surfaces and make the DSM-OS modification impossible and complicate the DSM-FRP modification. As both of these modifications were found useful, the change in the algorithm is well-grounded. In the dual contour method the contour of the greater energy was updated based on the evolution equation similar to ones used with force-based deformable meshes during each iteration. Instead with DSM, we applied a greedy local search for updating the meshes, mainly to make the DSM-FRP modification easier. However, as speculated in [Publication I], applying the greedy local search enables us to set constraints to local optimizations and hence reduces the parameter sensitivity.

The DSM algorithm is not suitable for all surface extraction tasks. For example, the extraction of curved tube-like structures could be difficult with it. This is due to the construction of search spaces which are directed towards (or away from) the reference point. The problem could remedied by directing the search in the direction of the normal of the surface at a particular mexel. However, as reported in the literature (e.g. [57]), this can easily lead to surface self-intersections that are undesirable.

No special attention has been given to computational efficiency while implementing the algorithm. Our current implementation is based on Matlab (Mathworks, Natick, MA, US) code compiled to C with the Matlab Compiler version

1.2. One iteration of the fixed reference point algorithm with standard parameter values and meshes of 1280 mexels lasts approximately 0.5 seconds on a 533 MHz processor (AlphaServer 4100 with 21164A (EV5.6) processors). With synthetic image experiments in [Publication I], the required number of iterations was typically 1000 and total computation time was 8 - 9 minutes. However, we are currently programming a faster implementation in C++, which should yield a considerable speed up for the algorithm.

8.4   GAGR-algorithm

In this study, we also examined the applicability of real coded genetic algorithms for the optimization task. The studied algorithm was a hybrid of a RCGA and a greedy algorithm. The hybridization with a greedy algorithm was performed for easing the computational burden of the RCGA. In addition, the selected BLX-α crossover operator tends to produce meshes that are not smooth. These are then smoothed by the subsequent application of the greedy algorithm for optimizing the mesh. Therefore, combining the particular crossover operator with the greedy algorithm seems to be a favorable choice. However, as noted in [Publication II], the crossover operator is not optimal, since a lot of computation time is wasted for evaluating meshes that are of irregular shape and therefore have high energy values. Representing surfaces with meshes is probably not an optimal choice with RCGA. With some other surface descriptions such as hyperquadrics [33], it might be easier to design a crossover operation that would confine the number of evaluated meshes of shapes too irregular to the minimum. Furthermore, the applied simple energy function cannot utilize the full power of RCGA, which could be used for optimizing much more complicated functions.

Nevertheless, the GAGR algorithm did remarkably well in the comparative study [Publication III]. On the average, it yielded the best experimental results of the five methods compared. Also, as mentioned in [Publication II], it was superior to a simple multi-start algorithm in the experiments performed. However, the computational cost of the algorithm is still high regardless of the hybrid nature and early termination of RCGA. For example, the DSM-algorithm with meshes of 1280 mexels typically took slightly over 1000 iterations to converge in the experiments reported in [Publication I]. It can be roughly (over)estimated that the energies of 32 meshes were calculated during each iteration of the DSM algorithm. Hence, the total count of meshes that were evaluated becomes approximately 32000. In the RCGA step of the GAGR algorithm, 100 populations of 4000 meshes were evaluated. This totals 400000 evaluated meshes. The total

computation time required by the GAGR algorithm is approximately 7 hours on a 533 MHz Alpha 21164A processor with a similar Matlab-based implementation than described for the DSM algorithm. As with the DSM algorithm, a considerable speed up could be expected from an implementation in optimized C code. Besides, also algorithmic ways to speed up the algorithm are possible. Nevertheless, for the deformable mesh optimization the GAGR algorithm is still very slow that decreases its attractiveness for the task.

There exists few applications of GAs in the framework of deformable models in literature. In [6], gray-coded [1] GAs were used to optimize two-dimensional active contours. There, hundreds of thousands of populations of contours were evaluated, which suggests very high demands for the machinery used for computation. Joshi *et al.* have used GAs for initializing deformable surfaces [42].

8.5 Shape modeling and internal energy

We extended a shape modeling scheme by Lai and Chin [46] to create shape models for surfaces. The difficulty in the 3-D case as compared to the case of shape modeling for (discrete) contours is that it is problematic building a mesh of pre-defined quality that represents the desired surface. Indeed, constructing a triangulation whose triangles would be even approximately equilateral is very challenging. Therefore, building a good shape model based on example meshes, or exemplars, is not straight-forward, although it has been explored in the literature [17, 50, 68, 78]. Other, more complicated representations of surfaces such as *m-reps* [42] could be more amenable to shape modeling than discrete surface meshes.

Our approach to shape modeling was somewhat different than the approach in [46]. Instead of considering exemplars, we constructed models for simple shapes by analyzing favorable local properties the mesh should posses. Obviously, very specialized models for the shapes of surfaces are impossible to construct in this way. Nevertheless, even using the simple sphere shape model, Eq. (4.8), instead of the thin-plate shape model, Eq. (4.7), improved the surface extraction results, cf. results Section in [Publication I]. More complicated shape models could be constructed by combining several sphere shape models. Another possibility would be to describe the shapes in terms of *simplex angles* studied in [16]. However, we shall not pursue these ideas further in this thesis.

---

[1]Gray codes are a form of binary codes.

Appendix A

Convergence of the DSM algorithm

A proof of the convergence of the DSM algorithm in a simplified case is presented in this Appendix. After the presentation of the proof, we shall consider how to extend the guaranteed convergence to more general cases. Only the fixed reference point algorithm is considered and convergence is only proved in the meaning *the algorithm will stop*. Notation used in this Appendix will be as in Chapter 4 and in [Publication I], particularly the surface centered mesh in the iteration t is $\mathbf{W}^t = \{\mathbf{w}_i^t\}$. In the each iteration there are two meshes, the inner and the outer mesh, but in order to avoid clutter we include this information in notation only when absolutely necessary. The fixed reference point is denoted by $\mathbf{g}$. Important definitions for the convergence proof are repeated from [Publication I] in a brief manner to make the Appendix easier to read.

In the each iteration of the DSM algorithm, we sequentially update all mexels $\mathbf{w}_1^t, \ldots, \mathbf{w}_N^t$ according to

$$\mathbf{w}_i^{t+1} = \arg \min_{\mathbf{w} \in S(\mathbf{w}_i^t)} E^i(\mathbf{w}, \mathbf{g} | \mathbf{w}_{i_1}^{t*}, \mathbf{w}_{i_2}^{t*}, \mathbf{w}_{i_3}^{t*}), \tag{A.1}$$

where t^* is t if $i_j > i$ and $t + 1$ otherwise.

The search space $S(\cdot)$ is defined differently in the case of the inner mesh and the outer mesh. For the inner mesh

$$
\begin{aligned}
S(\mathbf{w}_i) &= S_1(\mathbf{w}_i) \cup S_2(\mathbf{w}_i) \tag{A.2}\\
&= \{(1 + \frac{jL}{||\mathbf{w}_i||})\mathbf{w}_i : j = 0, \ldots, J\}\\
&\cup \{(1 + \frac{jL}{||\mathbf{w}_i||})[(1 - 2kD)\mathbf{w}_i + kD(\mathbf{w}_{i_a} + \mathbf{w}_{i_b})]\\
&\quad : (j = 1, \ldots, J), (k = 1, \ldots, K), (a, b = 1, 2, 3), (a \neq b)\},
\end{aligned}
$$

where L, J, K, D are user definable constants. And for the outer mesh

$$
\begin{aligned}
S(\mathbf{w}_i) &= S_1(\mathbf{w}_i) \cup S_2(\mathbf{w}_i) \\
&= \{(1 - \frac{jL}{||\mathbf{w}_i||})\mathbf{w}_i : j = 0, \ldots, J\} \\
&\cup \{(1 - \frac{jL}{||\mathbf{w}_i||})[(1 - 2kD)\mathbf{w}_i + kD(\mathbf{w}_{i_a} + \mathbf{w}_{i_b})] \\
&\quad : (j = 1, \ldots, J), (k = 1, \ldots, K), (a, b = 1, 2, 3), (a \neq b)\}.
\end{aligned}
\tag{A.3}
$$

In both cases we assume that the integer K is positive and the integer J is strictly positive. Also, we assume that $L > 0$ and $D > 0$. If K is set to zero, $S(\cdot) = S_1(\cdot)$ and we say that the algorithm is *simple*. If $K > 0$ then the algorithm is said to be *general*.

The algorithm is terminated when the volume inside of the inner mesh exceeds the volume inside of the outer mesh. The volume of the mesh is defined approximately as

$$
V(\mathbf{W}) = \frac{4}{3}\pi \cdot \left(\frac{\sum_{i=1}^{N} ||\mathbf{w}_i||}{N} \right)^3.
\tag{A.4}
$$

There are two reasons for the approximation, 1) it reduces the computation time for the algorithm and 2) it enables to prove the convergence of the algorithm with the search space definitions given in Eqs. (A.2) and (A.3). Note that the volume of the mesh is in one-to-one correspondence with the sum of the lengths of the mexels. Moreover, the bijection in question is an order-isomorphism, i.e.

$$
V(\mathbf{W}^1) \leq V(\mathbf{W}^2) \Leftrightarrow \sum_{i=1}^{N} ||\mathbf{w}_i^1|| \leq \sum_{i=1}^{N} ||\mathbf{w}_i^2||
$$

We use the symbol $\cong$ to denote this. Now we are ready to present a convergence result for the simple algorithm.

Proposition 1 *Provided that $K = 0$ and $JL < ||\mathbf{w}_i^t||/2$ for all i, $V(\mathbf{W}^t) \geq V(\mathbf{W}^{t+1})$ for the outer mesh and $V(\mathbf{W}^t) \leq V(\mathbf{W}^{t+1})$ for the inner mesh.*

Proof. We will present the proof in the case of the outer mesh, the proof for the inner mesh is similar.

First, using $K = 0$,

$$
\max_{\mathbf{w} \in S(\mathbf{w}_i^t)} ||\mathbf{w}|| = \max\{||(1 - \frac{jL}{||\mathbf{w}_i^t||})\mathbf{w}_i^t|| : j = 0, \ldots, J\} \leq ||\mathbf{w}_i^t|| \tag{A.5}
$$

for all i. The last inequality holds because $j \geq 0$, $L > 0$, $jL < ||\mathbf{w}_i^t||/2$ and hence $|1 - \frac{jL}{||\mathbf{w}_i^t||}| \leq 1$ for $j = 0, \ldots, J$. From (A.5), it follows that $||\mathbf{w}_i^t|| \geq ||\mathbf{w}_i^{t+1}||$ for all i. Hence, $\sum_{i=1}^N ||\mathbf{w}_i^t|| \geq \sum_{i=1}^N ||\mathbf{w}_i^{t+1}||$. Since the volumes

$$V(\mathbf{W}^{t+1}) \cong \sum_{i=1}^N ||\mathbf{w}_i^{t+1}||, \quad V(\mathbf{W}^t) \cong \sum_{i=1}^N ||\mathbf{w}_i^t||, \tag{A.6}$$

we obtain $V(\mathbf{W}^t) \geq V(\mathbf{W}^{t+1})$.

Note that in the above proof the condition that $JL < ||\mathbf{w}_i^t||/2$ is required for the outer mesh only. In practise, this condition prohibits oscillating behaviour that can occur if the mexels of the outer mesh come too close to the reference point. If the condition is still made stronger, we obtain the following lemma which is presented without its straightforward proof.

Lemma 1 *Let $K = 0$ and $JL < ||\mathbf{w}_i^t||$ for all i. Then for both inner and outer mesh, if $V(\mathbf{W}^{t+1}) \neq V(\mathbf{W}^t)$,*

$$|V(\mathbf{W}^{t+1}) - V(\mathbf{W}^t)| \geq \pi \frac{L^3}{N^3}.$$

As mentioned, the greedy algorithm can produce its initialization as a result in which case we said that the algorithm is in a local energy minimum. In that case, the energy function to be minimized was modified in order to help the mesh escape from the local minimum. The modified energy function is written as

$$E_{modified}^i(\mathbf{w}, \mathbf{g}|\cdot) = E^i(\mathbf{w}, \mathbf{g}|\cdot) + r\gamma\delta(\mathbf{w} - \mathbf{w}^t), \tag{A.7}$$

where $\delta(\cdot)$ is the Discrete Delta Function. The integer $r = 0, 1, 2 \ldots$ is gradually incremented until the result of the greedy algorithm differs from its starting mesh. The parameter γ determines the amount of the penalty added at a time.

Instantly from Eq. (A.7) it follows that if $E^i(\mathbf{w}, \mathbf{g}|\cdot)$ is bounded for all i and $\gamma > 0$ is not arbitrarily small, the mesh is forced to escape from a local minimum in a finite number of iterations. Particularly, all the energy functions considered in this thesis are bounded. Combining this remark with the Proposition 1 and the Lemma 1 and assuming reasonable values for all parameters, we obtain

Proposition 2 *The simple DSM-algorithm converges in a finite number of iterations.*

In this thesis, we will not extend formal considerations to cover general DSM algorithms. Instead, we consider the convergence of general DSM algorithms based on heuristic and experimental arguments. Particularly, it is clear based on experiments that for the convergence, the mechanism of the oscillation control has to be implemented, cf. [Publication I], [Publication III]. The oscillation control means just reducing a badly behaving general DSM algorithm to a simple one until the normal behaviour of the algorithm can be assumed to be restored. The bad behaviour is easily detected by studying the volumes of the meshes. The more problematic part is when it is reasonable to switch back from the simple algorithm to the general one. In our experiments with the general algorithm equipped with the oscillation control, we have never run into the problems with convergence of the algorithm. The formal consideration of the issue could point out how to implement the mechanism more efficiently. However, a convergence proof would probably require some assumptions, which might be lengthy to explain and might not be particularly interesting.

The reason for applying general algorithms is that with the simple algorithm the set of admissible meshes becomes too restricted. This means that meshes resulting from the simple algorithm often suffer e.g. unequal spacing of the mexels.

Bibliography

[1] S. Alenius, U. Ruotsalainen, and J. Astola. Using local median as the location of the prior distributon in iterative emission tomography image reconstruction. *IEEE Transactions on Nuclear Science*, 45(6(2)):3097 – 3107, 1998.

[2] A. Amini, T.E. Weymouth, and R.C. Jain. Using dynamic programming to solve variational problems in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):855 – 867, 1990.

[3] T. Apostol. *Mathematical Analysis*. Addison-Wesley, Reading, MA, US, 2nd edition, 1973.

[4] M. Audette, K. Siddiqi, and T. Peters. Level-set surface segmentation and fast cortical range image tracking for computing intrasurgical deformations. In *Proc. of Medical Image Computing and Computer Assisted Intervention, MICCAI99, Lecture Notes in Computer Science 1679, Cambridge, England*, pages 788 – 797. Springer, 1999.

[5] R. Bajcsy and S. Kovacic. Multiresolution elastic matching. *Computer Vision, Graphics and Image Processing*, 46(1):1 – 21, 1989.

[6] L. Ballerini. Genetic snakes for medical images segmentation. In *Proc of Mathematical Modeling and Estimation Techniques in Computer Vision, Proc. SPIE Vol. 3457*, pages 284 – 295, San Diego, CA, US, 1998.

[7] J.-D. Boissonat. Geometric structures fot three-dimensional shape reconstruction. *ACM Transactions on Graphics*, 3(4):266 – 286, 1984.

[8] R. Busacker and T. Saaty. *Finite Graphs and Networks*. McGraw-Hill book company, New York, 1965.

[9] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679 – 698, 1986.

[10] V. Caselles, R. Kimmel, G. Sapiro, and C. Sbert. Minimal surfaces: a geometric three dimensional segmentation approach. *Numer. Math.*, 77(4):423 – 451, 1997.

[11] C. Chesnaud, P. Réfrégier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1145 – 1157, 1999.

[12] L.D. Cohen and I. Cohen. Finite-element method for active contour models and bal-loons for 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1131 – 47, 1993.

[13] S. Cotin, H. Delingette, and N. Ayache. Real-time volumetric deformable models for surgery simulation. In *Proc. of Visualization in Biomedical Computing*, volume 1131 of *Lecture Notes In Computer Science*, pages 535 – 540, Hamburg, Germany, 1996.

[14] S. Cotin, H. Delingette, and N. Ayache. Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):62 – 73, 1999.

[15] E. Debreuve, G. Aubert M. Barlaud, I. Laurette, and J. Darcourt. Space-time sege-mentation using level set active contours applied to myocardial gated SPECT. *IEEE Transactions on Medical Imaging*, 20(7):643 – 659, 2001.

[16] H. Delingette. Simplex meshes: a general representation for 3D shape reconstruction. Technical Report 2214, INRIA, Sophia-Antipolis, France, March 1994.

[17] H. Delingette. General object reconstruction based on simplex meshes. *International Journal of Computer Vision*, 32(2):111–142, 1999.

[18] WWWebster dictionary. Merriam-Webster Incorporated, http://www.m-w.com.

[19] H. Edelsbrunner. *Geometry and Topology for Mesh Generation*, volume 6 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, New York, 2001.

[20] L. J. Eshelman and J. D. Schaffer. Real-coded genetic algorithms and interval schemata. In L.D. Whitley, editor, *Foundations of Genetic Algorithms 2*, pages 185 – 202. Morgan Kaufmann Publishers, 1993.

[21] M. Figueirdo and J. Leitao. Bayesian estimation of ventricular contours in angiographic images. *IEEE transactions on Medical Imaging*, 11(3):416 – 429, 1992.

[22] M. Figueirdo, J. Leitao, and A.K. Jain. Unsupervised contour representation and estima-tion using B-splines and a minimum description length criterion. *IEEE Transactions on Image Processing*, 9(6):1075 – 87, 2000.

[23] R. S. J. Frackowiak, K. J. Friston, C. D. Frith, R. J. Dolan, and J. C. Mazziotta, editors. *Human Brain Function*. Academic Press, 1997.

[24] J. Gleick. *Chaos. Making a New Science*. Cardinal, London, 1990.

[25] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989.

[26] U. Grenander. *General pattern theory*. Oxford University Press, New York, 1993.

[27] U. Grenander. *Elements of Pattern Theory*. The Johns Hopkins University Press, Balti-more, Maryland, 1996.

[28] U. Grenander and M.I. Miller. Computational anatomy: An emerging discipline. *Quaterly of Applied Mathematics*, 56(4):617 – 694, 1998.

[29] R. P. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. Addison-Wesley Publishing Company, New York, 3rd edition, 1994.

[30] R. N. Gunn, A. A. Lammertsma, S.P. Hume, and V.J. Cunningham. Parametric imaging of ligand-receptor binding in pet using a simplified reference region model. *Neuroimage*, 6(4):279 – 287, 1997.

[31] S.R. Gunn. *Dual Active Contour Models for Image Feature Extraction*. PhD thesis, University of Southampton, 1996.

[32] S.R. Gunn and M.S. Nixon. A robust snake implementation: a dual active contour. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1):63 – 68, 1997.

[33] S. Han, D. Geldof, and K. Boyer. Using hyperquadratics for shape recovery from range data. In *Proc. of IEEE Int. Conf. on Computer Vision (ICCV93)*, pages 492 – 496, Berlin, Germany, 1993.

[34] T. Harju. *Lecture Notes in Graph Theory*. Department of Mathematics, University of Turku, Finland, 2002. Available: http://users.utu.fi/harju/lectnotes.htm.

[35] F. Herrera, M. Lonzano, and J. L. Verdegay. Tackling real-coded genetic algorithms: Operators and tools for behaviorial analysis. *Artificial Intelligence Review*, 12(4):265 – 319, 1998.

[36] J. G. Hocking and G. S. Young. *Topology*. Addison-Wesley, Reading, MA, 1961.

[37] H. Hoppe. *Surface Reconstruction from Unorganized Points*. PhD thesis, University of Washington, 1994. Available from http://research.microsoft.com/~hoppe.

[38] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuezle. Surface reconstruction from unorganized set of points. *Computer Graphics (SIGGRAPH 92)*, 26(2):71 – 78, 1992.

[39] B. Hutton, M. Braun, L. Thurfjell, and D. Lau. Image registration: an essential tool for nuclear medicine. *European Journal of Nuclear Medicine*, 29(4):559 – 577, 2002.

[40] A.K. Jain, Y. Zhong, and M-P. Dubusson-Jolly. Deformable template models: A review. *Signal Processing*, 71(2):109 – 129, 1998.

[41] C. Johnson. *Numerical solution of partial differential equations by the finite element method*. Cambridge University Press, Cambridge, UK, 1987.

[42] S. Joshi, S. Pizer, P.T. Fletcher, A. Thall, and G. Tracton. Multi-scale deformable model segmentation based on medial description. In *Proc. of Information Processing in Medical Imaging (IPMI01), Lecture Notes in Computer Science 2082*, pages 64 – 77, Davis, CA, 2001.

[43] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321 – 331, 1988.

[44] J.-O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine three-dimensional surface extraction. *Medical Image Analysis*, 3(2):187 – 207, 1998.

[45] K. F. Lai and R. T. Chin. On regularization, formulation and initialization of the active contour models. In *Asian Conference on Computer Vision (ACCV93)*, pages 542 – 545, Osaka, Japan, 1993.

[46] K.F. Lai and R.T. Chin. Deformable contours - modelling and extraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(11):1084 – 1090, 1995.

[47] J. Lerch, J. Pruessner, A. Zijdenbos, S.J. Teipel, K. Buerger, H. Hampel, and A.C. Evans. Changes in cortical integrity in alzheimer's disesase. Poster presented at 8th International Conference on Alzheimer's Disease and Related Disorders, Stockholm, Sweden, 2002. Available from http://www.bic.mni.mcgill.ca/users/jason/.

[48] Y. Liu, R.T. Collins, and W.E. Rothfus. Robust midsagittal plane extraction from normal and pathological 3-D neuroradiology images. *IEEE Transactions on Medical Imaging*, 20(3):175 – 192, March 2001.

[49] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics (SIGGRAPH 87)*, 21(4):163 – 169, 1987.

[50] J. Lötjönen, P.-J. Reissman, I.E. Mangin, and T. Katila. Model extraction fron magnetic resonance volume data using the deformable pyramid. *Medical Image Analysis*, 3(4):387 – 406, 1999.

[51] D. MacDonald, N. Kabani, D. Avis, and A.C Evans. Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI. *Neuroimage*, 12(3):340 – 356, 2000.

[52] R. Malladi, J. Sethian, and B. C. Vemuri. Shape modelling with front progagation: A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158 – 175, 1995.

[53] J.-L. Mallet. Discrete smooth interpolation. *ACM Transactions on Graphics*, 8(2):121 – 144, 1989.

[54] J.-F. Mangin, V. Frouin, I. Bloch, J. Regis, and J. Lopez-Krahe. From 3D magnetic resonance images to structural representations of the cortex topography using topology preserving deformations. *Journal Mathematical Imaging and Vision*, 5(4):297 – 318, 1995.

[55] J.-F. Mangin, F. Tupin, V. Frouin, I. Bloch, R. Rougetet, J. Regis, and J. Lopez-Krahe. Deformable topological models for segmentation of 3D medical images. In *Proc. of Information Processing in Medical Imaging, IPMI95*, pages 153–164, Brest, France, 1995.

[56] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: A survey. *Medical Image Analysis*, 2(1):91 – 108, 1996.

[57] T. McInerney and D. Terzopoulos. Topology adaptive deformable surfaces for medical image volume segmentation. *IEEE Transactions on Medical Imaging*, 18(10):840 – 850, October 1999.

[58] O. Monga and R. Deriche. 3D edge detection using recuresive filtering: Application to scanner images. In *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 89)*, pages 28 – 35, San Diego, CA, 1989.

[59] J. Montagnat and H. Delingette. Globally constrained deformable models for 3D object reconstruction. *Signal Processing*, 71(2):173 – 186, 1998.

[60] J. Montagnat, H. Delingette, and N. Ayache. A review of deformable surfaces: topology, geometry and deformation. *Image and Vision Computing*, 19(14):1023 – 1040, 2001.

[61] D. Mumford. The bayesian rationale for energy functionals. In B. Romeny, editor, *Geometry driven diffusion in computer vision*, pages 141 – 153. Kluwer Academic, 1994.

[62] P. Ning and J. Bloomenthal. An evaluation of implicit surface tilers. *IEEE Computer Graphics and Applications*, 13(6):33 –41, 1993.

[63] M. Phelps and J. Mazziotta. Positron emission tomography: Human brain function and biochemistry. *Science*, 228(4701):799 – 809, 1985.

[64] C. A. Reiter. Visualizing steepest descent. *The Visual Computer*, 8(1):64 – 67, 1991.

[65] U. Ruotsalainen. *Quantification and data analysis in positron emission tomography: organ blood flow, graphical analysis and radiation dosimetry*. PhD thesis, Tampere University of Technology, 1997.

[66] U. Ruotsalainen, J. Mykkänen, J. Luoma, J. Tohka, and S. Alenius. Methods to improve repeatability in quantification of brain PET images. In *World Congress on Neuroinformatics: Part II Proceedings, ARGESIM Report no. 20*, pages 659 – 664, 2001.

[67] A. Sarti, C. Ortiz de Solorzano, S. Lockett, and R. Malladi. A geomatric model for 3-D confocal image analysis. *IEEE Transactions on Biomedical Engineering*, 47(12):1600 – 9, December 2000.

[68] D. Shen, E.H Herskovits, and C. Davatzikos. An adaptive focus statistical shape model for segmentation and shape modeling of 3D brain structures. *IEEE Transactions on Medical Imaging*, 20(4):257 – 270, April 2001.

[69] P. Suetens. *Fundamentals of Medical Imaging*. Cambrigde University Press, New York, 2002.

[70] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics (SIGGRAPH 92)*, 26(2):185 – 194, 1992.

[71] D. Terzopoulos. Regularization of inverse problems involving discontinuities. *IEEE Transactions in pattern analysis and machine intelligence*, 8(4):413 – 424, 1986.

[72] D. Terzopoulos and D. Metaxas. Dynamic 3D models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703 – 714, July 1991.

[73] D. Terzopoulos, J. Platt, A. Barr, and K. Fleicher. Elastically deformable models. *Computer Graphics (SIGGRAPH 87)*, 21(4):205 – 214, 1987.

[74] D. Terzopoulos and A. Witkin. Physically based models with rigid and deformable components. *IEEE Computer Graphics and Applications*, 8(6):41 – 51, 1988.

[75] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36(1):91 – 123, 1988.

[76] A. N. Tikhonov and V. A. Arsenin. *Solutions of ill-posed problems*. Winston, Washington, D. C., 1977.

[77] A. Törn and A. Žilinkas. *Global Optimization*. Lecture Notes in Computer Science 350, Springer-Verlag, 1989.

[78] J. Weese, M. Kaus, C. Lorenz, S. Lobregt, R. Truyen, and V. Pekar. Shape constrained deformable models for 3D medical image segmentation. In *Proc. of Information Processing in Medical Imaging (IPMI01), Lecture Notes in Computer Science 2082*, pages 380 – 387, Cambridge, England, 2001.

[79] E. Weinstein, editor. *Eric Weinstein's World of Mathematics*. http://mathworld.wolfram.com.

[80] D.J. Williams and M. Shah. A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, 55(1):14 – 26, 1992.

[81] C. Xu. *Deformable Models with Application to Human Celebral Cortex Reconstruction from Magnetic Resonance Images*. PhD thesis, the Johns Hopkins University, 1999.

[82] C. Xu, D.L. Pham, and J.L. Prince. Medical image segmentation using deformable models. In J.M. Fitzpatrick and M. Sonka, editors, *Handbook of Medical Imaging – Volume 2: Medical Image Processing and Analysis*, pages 129 – 174. SPIE Press, 2000.

[83] C. Xu and J.L Prince. Generalized gradient vector flow external forces for active contours. *Signal Processing*, 71(2):131 – 139, 1998.

[84] C. Xu and J.L. Prince. Snakes, shapes and gradient vector flow. *IEEE Transactions on Image Processing*, 7(3):359 – 369, 1998.

[85] A. Zijdenbos, R. Forghani, and A. Evans. Automatic quantification of MS lesions in 3D MRI brain data sets: Validation of INSECT. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI98), Lecture Notes in Computer Science 1496*, pages 439–448, Cambridge, MA, 1998.

[86] S. Zucker and R. Hummel. A three-dimensional edge operator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(3):324 – 331, 1981.

Publications