Segmentation of Medical Images under Topological Constraints

by

Florent Ségonne

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December 2005

Author .....
Department of Electrical Engineering and Computer Science
December 12, 2005

Certified by .....
Bruce Fischl
Associate Professor of Radiology, Harvard Medical School
Thesis Supervisor

Certified by .....
W. Eric L. Grimson
Bernard Gordon Professor of Medical Engineering
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Segmentation of Medical Images under Topological Constraints

by

Florent Ségonne

Submitted to the Department of Electrical Engineering and Computer Science
on December 12, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Major advances in the field of medical imaging over the past two decades have provided physicians with powerful, non-invasive techniques to probe the structure, function, and pathology of the human body. This increasingly vast and detailed amount of information constitutes a great challenge for the medical imaging community, and requires significant innovations in all aspect of image processing.

To achieve accurate and topologically-correct delineations of anatomical structures from medical images is a critical step for many clinical and research applications. In this thesis, we extend the theoretical tools applicable to the segmentation of images under topological control, apply these new concepts to broaden the class of segmentation methodologies, and develop generally applicable and well-founded algorithms to achieve accurate segmentations of medical images under topological constraints.

First, we introduce a digital concept that offers more flexibility in controlling the topology of digital segmentations. Second, we design a level set framework that offers a subtle control over the topology of the level set components. Our method constitutes a trade-off between traditional level sets and topology-preserving level sets. Third, we develop an algorithm for the retrospective topology correction of 3D digital segmentations. Our method is nested in the theory of Bayesian parameter estimation, and integrates statistical information into the topology correction process. In addition, no assumption is made on the topology of the initial input images. Finally, we propose a genetic algorithm to accurately correct the spherical topology of cortical surfaces. Unlike existing approaches, our method is able to generate several potential topological corrections and to select the maximum-a-posteriori retessellation in a Bayesian framework. Our approach integrates statistical, geometrical, and shape information into the correction process, providing optimal solutions relatively to the MRI intensity profile and the expected curvature.

Thesis Supervisor: Bruce Fischl
Title: Associate Professor of Radiology, Harvard Medical School

Thesis Supervisor: W. Eric L. Grimson
Title: Bernard Gordon Professor of Medical Engineering

Thesis Committee: Polina Golland
Title: Assistant Professor of E.E.C.S

Thesis Committee: Alan Willky
Title: Professor of E.E.C.S.

Acknowledgments

My gratitude goes above and beyond, to Bruce Fischl, my advisor and mentor. My work began with Bruce, who at the very outset has provided continual guidance, support and encouragement. It has been a great honor and privilege to have worked on this PhD under his supervision.

I am also grateful to the insightful, and often most timely, instructions of my advisor Eric Grimson, whose enthusiasm has always been a great source of my motivation.

Over the past three years, my life has been filled with joy because of a few special people, to whom I owe my most sincere and affectionate gratitude. They are Anukool, Carsten, Craig, Guy, Joan, Keyu, Lilla, Maya, Patrick, Rahul, Richard, and Sonya, whose emotional support, cheerful disposition, and memorable dialogues with me has been an indispensible part of my life.

I would also like to thank my piano teacher, David Deveau, who has infused so much passion and wisdom into his teaching, and whose love for music has animated my very own.

Finally, I am forever indebted to my parents and my two brothers for their love, faith in me, and unconditional support throughout these years. This dissertation is dedicated to them.

December, 12^{th}, 2005

Florent Ségonne

Contents

List of Figures

0.1 Notations

In this dissertation, we make use of the following notations, which might be used as a reference throughout this dissertation. Bold fonts denote vectorial variables.

- Cortical surface: the cortical surface will be denoted by $\mathcal{C}$.
- Spherical representation: The spherical representation of the cortical surface will be denoted by $\mathcal{S}$.
- A mapping from the cortical surface $\mathcal{C}$ onto the sphere $\mathcal{S}$ is denoted by $\mathcal{M}$.
- A rectangular decomposition of a compact surface $\mathcal{C}$ is denoted by $\mathcal{D}(\mathcal{C})$. We note v, e, and f the number of vertices, edges and faces in the decomposition $\mathcal{D}(\mathcal{C})$ respectively.
- The Euler-characteristic χ of a surface $\mathcal{C}$ is the number $\chi(\mathcal{C}) = v - e + f$, computed from any polyhedral decomposition of the surface $\mathcal{D}(\mathcal{C})$.
- The genus of a surface $\mathcal{C}$ is $g(\mathcal{C})$. It is related to the Euler-characteristic χ through the formula: $\chi = 2(K - g)$, where K is the number of connected components of the surface.
- An evolving active contour is represented by Γ.
- The level set representation of an evolving active contour Γ is denoted by ϕ.
- The outward normal and mean curvature of a surface are denoted by $\mathbf{n}$ and H respectively. The velocity field of an active contour is denoted by $\mathbf{v}$.
- $N_n(\mathbf{x})$ is the set of grid points which are n-adjacent to $\mathbf{x}$; we define $N_n^*(\mathbf{x}) = N_n(\mathbf{x}) \setminus \{\mathbf{x}\}$.
- $C_n(\mathbf{x}, X)$ is the set of n-connected components of $X \setminus \{\mathbf{x}\}$ that are n-adjacent to $\mathbf{x}$. $C_n(X)$ is the set of n-connected components of X.
- $N_n^k(\mathbf{x}, X)$ denotes the geodesic neighborhood of a point $\mathbf{x} \in X$ of order k; $N_n^{\cdot}(\mathbf{x}, X)$ denotes $N_n^k(\mathbf{x}, X)$ where k equals 2, 3, 2, and 1 when n is 6, 6^+, 18, and 26 respectively.
- The topological numbers of a digital point $\mathbf{x}$ under the topology $(n,\overline{n})$ are defined by: $T_n(\mathbf{x}, X) = |C_n(\mathbf{x}, N_n^{\cdot}(\mathbf{x}, X))|$ and $T_{\overline{n}}(\mathbf{x}, X) = |C_{\overline{n}}(\mathbf{x}, N_{\overline{n}}^{\cdot}(\mathbf{x}, X))|$.
- The extended topological numbers of a digital point $\mathbf{x}$ under the topology $(n,\overline{n})$ are defined by: $T_n^+(\mathbf{x}, X) = |C_n(\mathbf{x}, X)|$ and $T_{\overline{n}}^+(\mathbf{x}, X) = |C_{\overline{n}}(\mathbf{x}, \overline{X})|$

Chapter 1

Introduction

The field of medical imaging has undergone revolutionary advances over of the past two decades. New medical imaging technologies have provided physicians powerful, non-invasive techniques to probe the structure, function, and pathology of the human body. A few years ago, only a small number of non-invasive techniques was available to radiologists. Besides much experience/practice, deep insight, even intuition, was required for clinical diagnostic imaging. In recent years, the improvement and the development of many image acquisition techniques, the enhancement of the general quality of the acquired images, advances in image processing and development of large computational capacities, have considerably eased this task.

Acquisition of medical images in two (2D), three (3D), or higher dimensions, has become a routine task for clinical and research applications. Image acquisition techniques include magnetic resonance imaging (MRI), magnetoencephalography (MEG), 3D ultra sound imaging, computed tomography (CT), positron emission tomography (PET), single photon emission computed tomography (SPECT), functional MRI (fMRI), and diffusion weighted imaging (DWI). This increasingly vast and detailed amount of information needs to be interpreted in a timely and accurate manner in order to provide better diagnosis and treatment options for a family of clinical applications. It requires significant innovation in all aspects of image processing, such as image segmentation, image registration, visualization, compression and communication, among others.

In medical image processing, the automated recognition of "meaningful" image components, anatomical structures, and other regions of interest, is a fundamental task commonly referred to as image segmentation. Image segmentation greatly facilitates visualization and manipulation of specific structures. It is a critical step that often dictates the outcome of the entire clinical or research analysis.

One approach to image segmentation is to have a trained anatomist or technician manually label some regions of interests. However, manual approaches are considerably time consuming. For instance, the labeling of some or all the structures in the brain can take up to a week for high-resolution images. Also, manual or interactive segmentations, which are often restricted to 2D slice-wise processing, often suffer from inconsistency across segmented slices. Finally, studies have shown a large amount of variance among manual segmentations, an effect which seems to increase the risks related to inter- and intra-observer reliability [52, 29].

Quantitative analysis of medical images requires reproducible, accurate and efficient segmentation methods. In medical imaging studies, the segmentation of a large number

of images is often necessary for obtaining meaningful (i.e. statistically significant) results. Therefore, automated segmentation is desirable. However, the challenge is that images are usually corrupted by several artifacts, such as image noise, image intensity inhomogeneity or non-uniformity, and partial volume averaging effect. In recent years, many segmentation algorithms have been proposed and designed to account for such unwanted artifacts. While these techniques produce repeatable and accurate results, few of them guarantee that the obtained segmentations respect the true anatomy of the structures. Too often, segmentations contain small geometric inaccuracies that alter the true anatomy of the modelled structures. For instance, cortical segmentations often include handles that incorrectly connect different regions of the cortex.

In medical imaging, the overall shape of a region of interest is prescribed by medical knowledge; it is usually known *a priori*. Segmentation techniques should be able to produce results that reflect the anatomy of the structures. Several clinical and research applications (e.g. visualization, surgical planning, surface-based processing of functional data, surface-based atlas, inter-subject registration, ...) depend critically on the accuracy and correctness of the representations.

However, accurate segmentation under anatomical consistency is challenging. Mathematically, the anatomical consistency refers to the notion of *topological* correctness of the segmented shape. One of the difficulties arises from the *continuous* nature of topological notions: they are difficult to transpose into a practical discrete framework that is applicable to the segmentation of medical images. Also, the anatomical consistency of a segmentation refers to the global connectivity of the geometric shape as well as to its local properties. This concept which is both local and global is difficult to model and integrate into the segmentation process. Overall, few methods have been proposed to precisely locate structures of interest while ensuring the correctness of their topology (i.e. the correct anatomy).

The objective of this research is to improve and extend the theoretical tools applicable to the segmentation of images under topological constraints, to apply these concepts to broaden the class of segmentation methodologies, and to develop generally applicable and well-founded algorithms to achieve accurate segmentations of medical images under topological constraints. Although the focus is on the segmentation of MR brain images, the set of potential applications extends well beyond the field of medical imaging.

1.1 Segmentation of Brain Structures from Magnetic Resonance Images

Excluding pathological cases, the shape of most macroscopic brain structures can be continuously deformed into a sphere. In mathematical terms, these structures have the topology of a sphere[1]. Particularly, this implies that most brain structures consist of *one single connected* object that does not possess any *handles* (i.e. holes) or *cavities*. This is the case for the subcortical structures, such as left and right ventricle, putamen, pallidum, amygdala, hippocampus, thalamus, and caudate nucleus (Fig. 1-1-a), but it also holds for the cortex under some specific conditions. The human cerebral cortex is a highly folded ribbon of gray matter (GM) that lies inside the cerebrospinal fluid (CSF) and outside the white matter (WM) of the brain. Locally, its intrinsic "unfolded" structure is that of a 2D sheet, several millimeters thick. In the absence of pathology and assuming that the midline

[1]Topology is a mathematical discipline that aims at characterizing the connectivity of geometrical shapes. Necessary background in topology is reviewed in Chapter 2.

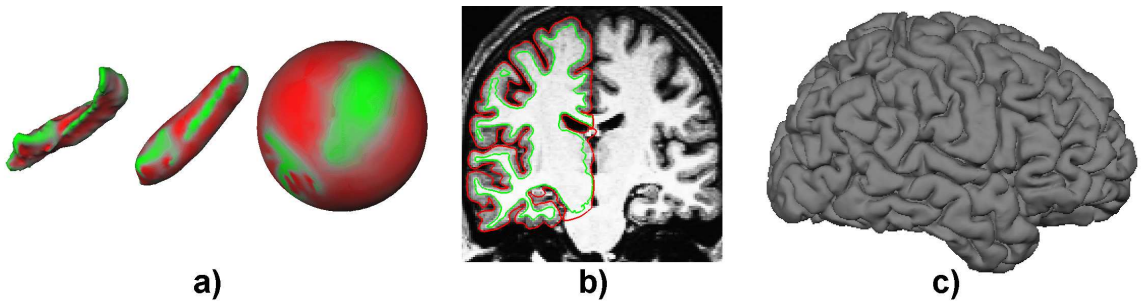

Figure 1-1: a) Subcortical structures have a spherical topology. For instance, the shape of the hippocampus can be continuously deformed onto a sphere. b) The human cerebral cortex is a highly folded ribbon of gray matter that lies inside the cerebrospinal fluid and outside the white matter of the brain. The green surface represents the interface between WM and GM, and the red surface (i.e. the pial surface) models the interface between GM and CSF. When the midline connections between the left and right hemisphere are artificially closed, these two surfaces have the topology of a sphere. c) 3D rendering of the highly folded pial surface. Opposite regions of a sulcus are often self-touching.

hemispheric connections are artificially closed, each cortical hemisphere can be considered as a simply-connected 2D sheet of neurons that carries the simple topology of a sphere[2] (Fig. 1-1-b).

Many recent segmentation algorithms are able to identify and precisely locate diverse brain structures, albeit without ensuring the validity of final topology (i.e. the one of a sphere). Medical images often contain various artifacts that are difficult to predict and model. For instance, the finite resolution of images makes it particularly difficult to accurately locate the cortical pial surface that separates GM from CSF (Fig. 1-1-b). Opposite banks of the cortical gray matter often appear connected, and the interface between GM and CSF becomes invisible. This type of artifact is referred to as the partial volume effect, and is illustrated in Fig 1-2-a. Consequently, segmentation methods that ignore this constraint of "separation" often produce segmentations with erroneously connected regions. These regions form handles in the segmentation that are hard to detect and correct retrospec-

[2]The true topology of the cortical surface is not the one of a sphere, due to the midline connections between the left and right hemisphere, such as the anterior and the posterior commisures.

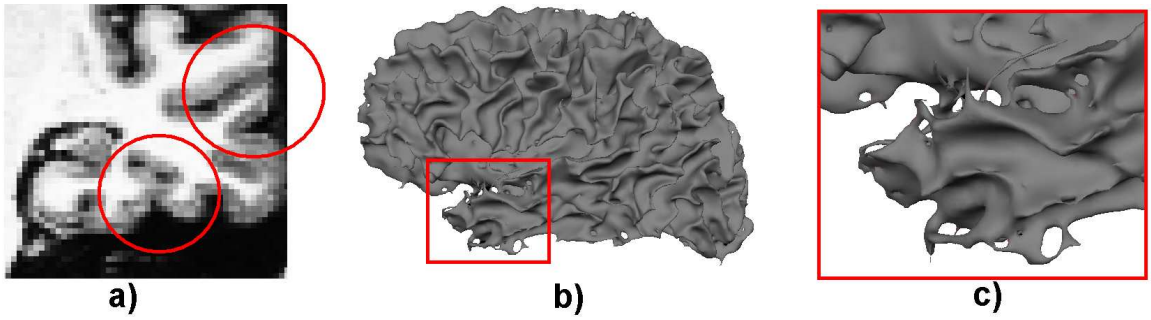

Figure 1-2: a) Due to the partial volume effect, it becomes hard to distinguish opposite banks of a the gray matter. b) Segmentation algorithm that do not constrain the topology often produce cortical segmentations with several topological defects (i.e. handles, cavities, disconnected components). c) A close-up of a topologically-incorrect cortical surface representation.

tively [44, 28, 43, 56]. Figure 1-2 shows an example of an incorrect cortical segmentation with several handles.

Many clinical and research applications require accurate segmentations that respect the true anatomy of the targeted structures. However, only a few techniques have been proposed to achieve accurate and topologically-correct segmentations. Yet, these methods are limited by the several artifacts present in the images, and by the few topological concepts that are applicable to the segmentation of medical images.

In the remainder of this chapter, we describe the significance of accurate and anatomically-consistent segmentation. We present the difficulties and challenges in MR Brain image segmentation and report the objective and contributions of this thesis. An overview of the dissertation concludes this chapter.

1.2 Importance of Accurate and Anatomically-Consistent Segmentation

Achieving accurate and topologically correct representations of different structures is of critical importance for various clinical and research applications.

The cortex, which is a highly folded ribbon of gray matter, is composed of cortical neurons (pyramidal and granular/Stellate cells) that are oriented horizontally to the surface. Its functional organization is essentially 2-dimensional with functional units of cortical activity organized in groups of neurons, oriented perpendicularly to the surface. Consequently, studies characterizing the functional organization of the brain require accurate models of the cortical surface [17]. These studies analyze how different regions of the cortex are connected and are related to each other.

Small geometric errors in a segmentation can easily change the apparent connectivity of the segmented structure, posing a problem to studies that aim at analyzing the connectedness of different regions. In particular, distances between regions can be changed, such that points appear much closer than they are (i.e. underestimation of true cortical (geodesic) distances). This is often the case in cortical representations, where small handles in the final segmentation artificially connect opposite banks of a sulcus. The accuracy and correctness of the representations contribute the success of such studies.

Many neurodegenerative disorders, psychiatric disorders, and healthy aging are frequently associated with structural changes in the brain. These changes, which can cause alterations in the imaging properties of the brain tissue, as well as in morphometric properties of brain structures, can be captured and detected by elaborate segmentation techniques [29, 55, 96, 95, 22]. For instance, the thickness of the cerebral cortex carries important information relative to aging [83]. Accurate 3D models are necessary for estimating the true cortical thickness (methods that only use 3D slices to estimate cortical thicknesses always lead to overestimated distances).

We provide a non-exhaustive list of potential applications that would benefit from accurate and topologically-correct representations.

- **Visualization.** The accurate and topologically-correct segmentation of various structures allows the direct visualization of the 3D geometrical shape of the structures. Also, functional data can be easily projected and displayed onto the surfaces for illustration, visual analysis, and post-processing [17, 99, 16].

- **Spherical Coordinate System and Surface-based atlas.** The analysis of cortical

data is greatly facilitated by the use of accurate 2D models of the cortical sheet [17, 99]. These models alleviate most problems of the 3D embedding space (such as the underestimation of true cortical distances or the overestimation of cortical thicknesses). Also, models of the brain surface allow for the establishment of a global 2D coordinate system onto the whole cortical surface; these coordinate systems can then be used to generate 2D spherical atlases. The display and the analysis of anatomical (e.g. thickness) and functional (e.g. fMRI, MEG) data of the brain cortex is greatly improved by the use of surface-based atlases [23, 99, 31, 27]. In addition, surface-based atlases can be used to evaluate and diagnose precisely brain abnormalities. Recent studies have found that Alzheimer's disease was associated with the thinning of GM volume and the enlargement of cortical sulci [91]. This information can be carefully integrated into statistical surface-based atlases to help the early detection of Alzheimer's pathology [22].

- **Shape Analysis.** Structural changes in the brain can be captured and detected by elaborate segmentation techniques. These changes can be used to characterize many neurodegenerative disorders, psychiatric disorders, and healthy aging [29, 55, 96, 95, 22].

- **Surface-based processing of functional data.** The functional organization of the human cerebral cortex is essentially 2-dimensional. Projecting functional data (e.g. fMRI, MEG) onto valid representations of the cortical surface greatly facilitates and improves the identification and localization of various functional brain areas [17].

- **Inter-subject Registration.** Accurate and topologically-correct representations of specific structures of interest provide important anatomical landmarks (e.g. sulcal depth, cortical thickness, sulcal crest lines) that can be used to improve the registration of different subjects [31, 92, 98]

1.3 Challenges in Magnetic Resonance Image Segmentation

The segmentation of medical images under topological constraints is a difficult task for a variety of reasons.

First, segmentation algorithms operate on the intensity or texture variations of the image and are therefore sensitive to the artifacts produced by the image acquisition process. These limitations include image noise, image intensity inhomogeneity or non-uniformity, RF inhomogeneities, partial volume averaging effect, and subject motion. In the case of cortical segmentations, the partial volume effect makes it particularly difficult to accurately locate the pial surface of the cortex (Fig. 1-2-a). Due to its highly folded nature, opposite banks of the sulcus often appear connected, and small gaps between adjacent folds of the neocortical gray matter become invisible in finite resolution MR images. The detection of the substructures is also difficult as many of them are defined by weakly visible boundaries. For instance, the intensity pattern of the thalamus in MRI images looks similar to the neighboring white matter. Segmentation methods cannot solely rely on the MR images in order to distinguish these structures, and prior information is often necessary to capture common attributes of anatomical structures in a population. In general, segmentation techniques that do not integrate any topological constraints generate segmentations that often contain some small deviations from the true anatomy of the structures of interest. These deviations are called

topological defects and can be of three types: cavities, disconnected components, or handles (i.e. holes) that incorrectly connect parts of the volumes (Sect. 2.2.4). We note that several good reviews of segmentation techniques can be found in the literature [103, 9, 75].

Next, the integration of topological constraints significantly increases the complexity of the task. Topology is both a global and a local concept; small and local modifications of a geometric shape can change its global connectivity. Also, topology is intrinsically a continuous concept and topological notions are difficult to adapt into a discrete framework. Therefore, the amount of techniques available and applicable to the segmentation of images are quite limited.

Only a few automatic techniques have been proposed to produce topologically-correct segmentations. The current state of the art in medical image segmentation under topology control is reviewed in details in Chapter 2. Here we provide a brief summary of the relevant existing techniques. In general, they can be divided in two categories.

One set of approaches incorporates directly the topological constraints into the segmentation process. Active contours and digital models, reviewed in Chapter 2, have been extensively used for this purpose. An initial model, carrying the correct topology, is deformed, usually by gradient descent, while preserving its topology. However, these methods are quite sensitive with regard to initialization, and large geometrical errors are often produced due to the strict topology preservation. The strict topology preservation is often too restrictive for most applications. The generation of cavities or disconnected components as well as the formation of handles is prevented. The primary concern is handles that are difficult to retrospectively correct [44, 28, 43, 56]. On the other hand, changes in the number of connected components or cavities during the deformation of an object are less problematic. We tackle these limitations in Chapter 3 by introducing a concept based on the theory of digital topology. In addition, in Chapter 4, we describe a flexible segmentation technique that offers a subtle topological control.

More recently, another set of approaches has been recently proposed to retrospectively correct the topology of already segmented images. These techniques, which do not enforce any topological constraints during the segmentation process, can focus on more accurate segmentations. However, the performance of these methods have not proved to be sufficiently satisfactory. First, they cannot be used to correct the topology of arbitrary segmentations, as they make assumptions about the topology of the initial input image. Also, they do not use any statistical information to locate and correct the topological defects, and the corresponding corrections may not correspond to the ones a trained operator would make. Particularly, for each topological defect, they only evaluate a small number of potential solutions, and most often fail to produce optimal solutions. We focus on the retrospective topology correction of medical images in Chapters 5 and 6. Chapter 5 introduces a novel framework to retrospectively correct the topology of digital images, while Chapter 6 is dedicated to the topology correction of cortical surfaces.

1.4 Motivation

Achieving accurate and topologically-correct segmentations of medical structures is a crucial step for many post-processing tasks in medical imaging. While existing methods can be effective, they have a number of limitations. The work presented in this dissertation is motivated by these limitations.

- **Topologically constrained segmentation methods are too restrictive.** Methods that aim at directly segmenting a structure of interest using strict topological constraints require an initialization of the model that is close to its final configuration. This is essentially due to the fact that topological barriers can easily lead to large geometrical errors, which are difficult to correct retrospectively. Medical images often contain artifacts (noise, image inhomogeneities, ...) or unexpected structures (tumors), which are hard to predict. Topologically constrained segmentation methods, which strictly preserve the topology of the initial contour, are too restrictive: they prevent the formation of cavities and disconnected components as well as the generation of handles. A finer degree of control over topological changes would certainly be of importance.

- **Retrospective topological corrections may not be optimal.** Most retrospective methods do not make full use of all available information. Most methods assume that the topological defects in the segmentation are located at the thinnest parts of the volume and aim at correcting the topology by minimizing the amount of modifications in the original segmentation. Most often, the resulting topological corrections do not correspond to the ones that a trained operator would make. This is illustrated by Fig. 1-3.

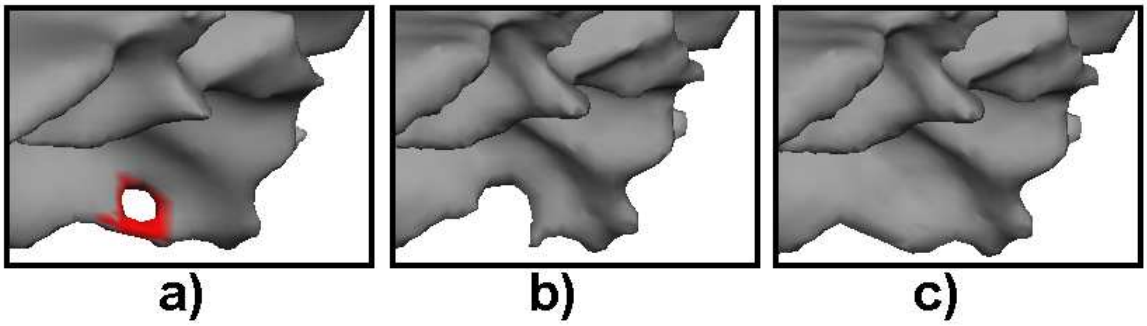

Figure 1-3: Minimal topological corrections might not be optimal. Methods that aim at correcting the topology of a segmentation by minimizing the amount of modifications in the segmentation might not achieve valid corrections. a) A topological defect is identified in red on the original topologically-incorrect cortical surface. b) An *incorrect* topological correction that was only based on the size of the defect. In this case, cutting the handle corresponds to a "smaller" modification of the surface than filling the corresponding hole. However, this topological correction is not the correct one and results in an inaccurate cortical representation. c) Correct topological correction realized by a trained operator.

- **Most methods to retrospectively correct the topology cannot deal with arbitrary segmentations.** Retrospective methods assume a fully-connected volume and cannot deal with segmentations that contain cavities of disconnected components. Subcortical segmentations are difficult to segment if solely based on intensity

properties. Imaging artifacts, anatomical variability, varying contrast properties, and poor registration, often result in segmentations that contain a few topological defects. While a small number of modifications is usually sufficient to correct the topology, no assumptions can be made on the topology of the initial segmentation.

- **At most, two candidate solutions per topological defect are generated and evaluated.** Methods to retrospectively correct the topology of an object fail to generate multiple potential solutions in order to select an optimal solution. They only produce two solutions. In the case of a handle, these two solutions correspond to either cutting the handle or filling the corresponding hole. Other solutions, such as the ones a trained operator would make are not generated. This problem is usually inherent to the proposed framework that is not adapted to the generation of multiple candidate solutions. Figure 1-4 illustrates the difficulty of finding an optimal solution when the topological defect is complex.

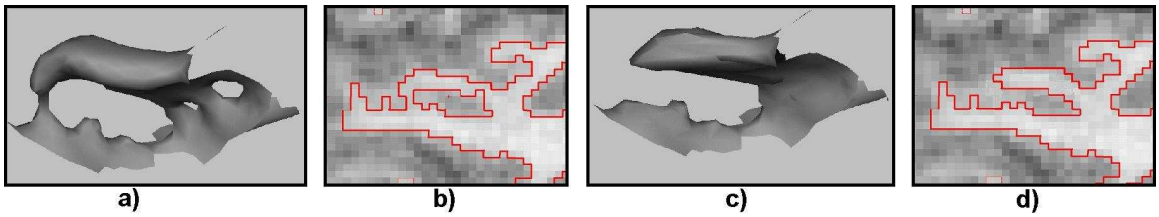

Figure 1-4: Some topological defects are quite complex and extremely hard to optimally correct. Existing methods only produce a few potential topological solutions and the chosen solution rarely is optimal. a) A complex topological defect formed by 3 handles. b) One sagital MRI slice of the topological defect, illustrating the complexity of the underlying MRI intensity profile. c) Optimal solution realized by our method developed in Chapter 6 under the supervision of a human expert. Note that the solution does not correspond to a simple cut of the main handle. d) Sagital cut of the MRI volume showing the location of the surface of the corrected defect. The optimal solution was optimized relative to the underlying MRI intensity profile and the expected local curvature of the surface. A human expert asserted the correctness and optimality of the solution.

Also, in medical imaging, depending on the application, different types of information are required and different data structures are used; the most common cases are surfaces modeled by tessellations, and volumes that use 3-dimensional Cartesian grids. Even though the segmentation of a structure of interest relies in part on intensity information extracted from one or several 3-dimensional images, the data structure might not use the 3-dimensional grid to encode the final segmentation, e.g. tessellations. It is important to develop methods for both types of data structures, 3-dimensional volumes or surfaces.

1.5 Objective and Contributions of this Thesis

In this dissertation, we propose type-specific (i.e. surface-based or volume-based) methods to address previous limitations. We extend the set of tools applicable to the segmentation of medical images under topology control, and develop techniques to achieve accurate segmentations under topological constraints. Our contributions can be broadly divided in two categories.

A - Topologically-constrained segmentation of medical images

We first focus on methods that integrate topological constraints directly into the segmentation process. These methods which strictly preserve the topology of the models are quite sensitive with regard to their initializations, and to the presence of noise and unexpected structures in the images. We address these limitations in the following:

- **Definition of a novel digital concept.**
  Topological concepts are hard to adapt into a discrete framework and the number of available tools are limited. We introduce a novel digital concept, the concept of *multisimple point*, which offers more flexibility in controlling the topology of segmentations.

- **A new active contour framework.**
  Deformable models have been extensively used for the purpose of image segmentation. We improve the control of topological changes with the level set method. Our method, the *genus-preserving level sets*, offers a subtle topological control over the topology of the level set components, and constitutes a trade-off between traditional level sets and topology-preserving level sets [46].

B - Retrospective topology correction of medical image segmentations

In the second part, we focus on the retrospective topology correction of already segmented structures. Existing methods to correct retrospectively the topology of segmentations suffer from some important limitations. They do not integrate any statistical or geometric information into the topology correction process and they do not guarantee the optimality of the topological corrections. Our contributions address these limitations:

- **Bayesian formulation**
  We integrate statistical and geometric information into the topology correction process using a Bayesian framework. The correction of a topological defect makes use of additional information, such as the underlying MRI intensity profile or the expected local curvature of the corrected defect. As a consequence, accurate solutions can be generated.

- **Optimal correction of the topology**
  While existing methods only evaluate a small number of potential topological corrections, and fail to produce optimal solutions, we introduce (in Chapter 6) a method that generates multiple potential solutions. The optimal correction is selected as the maximum-a-posteriori solution in a Bayesian framework.

In addition, we propose two algorithms for the retrospective topology correction of the two most common data structures used in medical images: 3D digital images and surfaces.

- **Topology correction of 3D binary images**
  We develop an algorithm for the retrospective topology correction of 3D digital segmentations. Our method is phrased within the theory of Bayesian parameter estimation, and integrates statistical information into the topology correction process. In addition, no assumption is made on the topology of the initial input images.

- **Topology correction of cortical surfaces**
  We propose a genetic algorithm to accurately correct the spherical topology of cortical surfaces. Unlike existing approaches, our method is able to generate several potential topological corrections and to select the maximum-a-posteriori retessellation in a Bayesian framework. Our approach integrates statistical, geometrical and shape information into the correction process, providing optimal solutions relatively to the MRI intensity profile and the expected curvature.

1.6 Thesis Overview

This thesis is organized in three parts. The first part consists of the necessary background to understand the contributions of our work. It corresponds to Chapter 2. Some elementary notions of topology are introduced and we describe how to apply these notions to discrete imaging. We summarize the current state of the art in medical image segmentation under topological constraints. Existing techniques to achieve accurate segmentations under some topological constraints are presented, and their limitations are described.

The second part, consisting of Chapters 3 and 4, presents our contributions to topologically constrained segmentation. We focus on the limitations inherent to methods that strictly preserve the topology of an object.

For this purpose, we introduce in Chapter 3 a new concept of digital topology that extends and generalizes the notion of simple points (this concept is clearly defined in Sect. 2.3.1). Simple points guarantee that the topology of a digital object is preserved during a deformation. However, the generation of cavities or disconnected components as well as the formation of handles are prevented. The extension of the concept of *simple point* to multi-label images, that we call *multisimple point*, ensures that no topological defects are generated while splitting or merging the components of the object.

Based on this concept, in Chapter 4, we then introduce an active contour framework, where level sets evolve under topological control. Level set components are allowed to merge, split, be destroyed or generated, but not to produce any handle during the evolution. The resulting active contour evolution is more flexible than topology-preserving deformations. This algorithm, which introduces a finer degree of control over topological changes in level-set-based image segmentation, fills the gap between the original level set framework and topology-preserving level set methods.

In the third part of this work, we focus on the retrospective topology correction of segmentations. It consists of Chapter 5 and Chapter 6.

In Chapter 5, we develop a fully automated volume-based method to correct the topology of any binary volumetric segmentation under any digital connectivity. The novelty of

our approach comes from the fact that any initial segmentation, containing disconnected regions, handles, or cavities, will be corrected. A multiple region growing process allows us to simultaneously work on different parts of the volume and to incorporate statistical information.

Finally, in Chapter 6, we specifically focus on the cortical reconstruction problem that is intrinsically more challenging than the topology correction of subcortical structures. We introduce a technique that directly extends the approach taken by Fischl et al. in [28], addressing most of its limitations (self-intersection, optimality, vertex elimination). Our approach integrates statistical, geometrical and shape information into the correction process, providing optimal solutions relative to the MRI intensity profile and the expected curvature.

At the end of each chapter, we will refer to the main contributions that were presented and indicate the associated publications. The proofs of Chapter 4 are provided in Appendix A. Appendix B lists all of our publications. All the algorithms developed in this thesis are part of the freely available cortical surface reconstruction and flattening software Freesurfer, associated with [16, 30, 31, 27, 28, 29, 32, 84, 85, 86].

Section I
Background

Chapter 2

Preliminaries: Theory of Topology in Medical Imaging

In this chapter, we present background material of central importance for this dissertation. We introduce some elementary notions of topology and show how these notions can be adapted into a discrete framework and applied to the segmentation of medical images under topological constraints. Particularly, we introduce the essential concept of Euler-characteristic, and we describe the theory of digital topology. Isocontours methods are also presented. Next, we describe the current state-of-the-art segmentation of images under topological constraints. We first present methods that aim at directly integrating the topological constraint into the segmentation process, followed by approaches that try to correct retrospectively the topology of already segmented images. The limitations of both approaches are explicated.

2.1 Introduction

This chapter presents background material of central importance for this dissertation. We first introduce some general notions of topology and present the strong connections of topology with differential geometry. The Euler-characteristic and the genus of a surface are defined. Next, we show how the continuous theory of topology can be applied to the segmentation of medical images under topological constraints. Particularly, we describe how topological notions can be adapted to the two most common data structures used in medical imaging: 3D voxel grids and surfaces. Also, we present methods for extracting topologically-consistent isocontours from digital images. Finally, we present the state-of-the-art segmentation of medical images under topological constraints. The most common segmentation algorithms are described and their limitations clearly reported. Some of the material presented in this section were taken from Mathworld [63] and the work of Bertrand [7, 8]. We refer the reader to the following text books [70, 37, 48] for a complete introduction to topology and algebraic topology.

2.2 General Notions of Topology

2.2.1 A Continuous Theory

Topology is a branch of mathematics that studies the properties of geometric figures that are preserved through deformations, twistings and stretchings, without regard to size and

absolute position. In topology, the important mathematical notions are those of *continuity* and of *continuous transformations*; tearing, which would generate discontinuities, is prohibited.

Topology studies the properties of spatial objects by abstracting their inherent connectivity while ignoring their detailed form. One of the central ideas is that geometric objects, such as circles, curves, surfaces, can be treated as objects in their own right, independently of how they are represented or how they are embedded in space. The exact geometry of the objects, their location and the details of their shape are irrelevant to the study of their topological properties. In essence, this amounts to characterizing the topology of an object by its number of disconnected components, holes and cavities, but not by their position. For instance, a circle is topologically equivalent to any closed loop, no matter how different these two curves may appear. Similarly, the surface of a coffee mug with a handle is topologically the same as the surface of a doughnut (this type of surface is called a one-handled torus).

However, by ignoring the embedding space, it then becomes impossible to distinguish a torus from a knotted torus (Fig. 2-1-a). This has led mathematicians to define several levels of topological equivalence depending on the chosen set of continuous transformations. Given a specific set of transformations, two geometric figures represent the same topological object, or the same equivalence class, if both of them can be deformed into a third one by using continuous transformations from the considered set only.

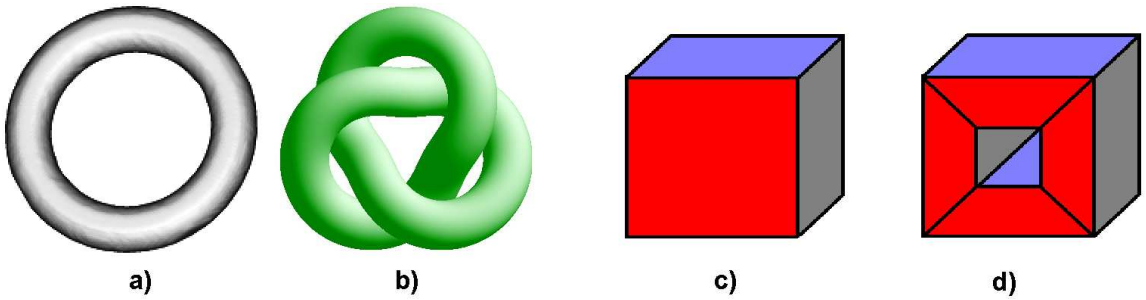

Figure 2-1: a-b) Two tori that are homeomorphically equivalent. They share the same intrinsic topology. However, they do not share the same homotopy type as one can not be continuously transformed into the other. c) A geometric object with a spherical topology; its Euler-characteristic is $\chi = v - e + f = 8 - 12 + 6 = 2$. d) A geometric object with a toroidal topology and an Euler-characteristic of $\chi = v - e + f = 16 - 32 + 16 = 0$.

2.2.2 Notions of Topological Equivalence

In this work, we are interested in locating anatomical structures from medical images. These structures are geometric entities, which are often referred to as topological spaces in the mathematics community. These shapes can be represented equivalently as surfaces or as volumes. Depending on the context, a topological space X might refer to the volume or surface representation. Using these two dual representations, two distinct levels of topological equivalence are usually considered. But we first need to introduce some mathematical definitions.

Definition 2.1 Homeomorphism
An homeomorphism M from a space X into a space Y is a continuous, one-to-one transformation with a continuous inverse M^{-1}.

Homeomorphisms have some important properties that we will use later in this dissertation (particularly in Chapter 6). The Jacobian $J_{\mathcal{M}} = |\frac{\partial \mathcal{M}}{\partial \mathbf{x}}|$ of the transformation[1] is non-singular (i.e. strictly positive or strictly negative). This is, of course, the multidimensional analog of monoticity. Another important property of the Jacobian is the fact that it relates the n-dimensional volumes of X and Y: $d^n\mathbf{y} = J_{\mathcal{M}} d^n\mathbf{x}$, where $\mathbf{y} = J(\mathbf{x})$.

Definition 2.2 Homotopy
An homotopy is a continuous transformation from one function into another. An homotopy between two functions f and g from a space X into a space Y is a continuous map $G : X \times [0,1] \rightarrow Y$ with $G(\mathbf{x},0) = f(\mathbf{x})$ and $G(\mathbf{x},1) = g(\mathbf{x})$, where $\times$ denotes set pairings. One says that two maps f_0 and f_1 are homotopic if there exists a homotopy connecting them, and one writes $f_0 \simeq f_1$.

In simpler terms, two objects are said to be homotopic if one can be continuously deformed into the other. For instance, a line segment is homotopic to a point, and a circle is homotopic to a solid torus. We note that, contrary to homeomorphism, homotopy does not consider the dimension of the topological objects. For instance, the unit ball in $\mathbb{R}^n$, $\{\mathbf{x} \in \mathbb{R}^n / \ \|\mathbf{x}\| \leq 1\}$, is homotopically equivalent to the point $\{\mathbf{x} = 0\}$. Other levels of topological equivalence can be defined by considering the dimensionality of the topological objects. Homotopy is one of the main concepts of Algebraic Topology. For more details on homotopies and algebraic topology, we refer the reader to an excellent book on algebraic topology [48].

Using these two sets of continuous transformations, two levels of topological equivalence are usually considered:

- *Intrinsic Topology*: the intrinsic topology of an object is defined by the set of properties that are preserved by *homeomorphic* transformations defined on the surface of the considered object. Under this set of equivalence, the embedding space is ignored: a knotted solid torus has the same intrinsic topology as a simple torus; and a hollow sphere is of the same topology as two spheres.

- *Homotopy type*: the homotopy type of an object is the set of properties that are preserved by *homotopic* transformations. Formally, we define two spaces X and Y to share the same homotopy type, or to be homotopy equivalent, if there are maps $f : X \rightarrow Y$ and $g : Y \rightarrow X$, such that the composition $f \circ g$ is homotopic to the identity map of Y ($f \circ g \simeq \mathbb{I}_Y$), and the composition $g \circ f$ is homotopic to the identity map of X ($g \circ f \simeq \mathbb{I}_X$). Homotopy, which was first formulated by Poincarré around 1900, provides a measure of an object's topology by considering the embedding space. At this level of topological equivalence, a torus is topologically different from a knotted torus, since one cannot be continuously transformed into the other (Fig. 2-1-a,b).

In this dissertation, the required level of topological equivalence is provided by homotopy. The anatomical structures to be segmented define smooth 2D compact (i.e. closed)

[1]The fact that the Jacobian exists might seem confusing at first. Indeed, we have only assumed that the mapping $\mathcal{M}$ was an homeomorphism, and we did not specify that the mapping $\mathcal{M}$ was differentiable. However, in dimension 1, 2, and 3, any pair of homeomorphic smooth manifolds are diffeomorphic! This surprising property does not hold in higher dimension.

orientable manifolds[2] embedded in the real 3D Euclidean space. For such "simple" surfaces, the study of their differential properties provides deep insights about their topology, as the topology of such surfaces have profound connections with differential geometry.

2.2.3 Topology and Differential Geometry

Differential geometry is the study of Riemannian manifolds. Differential geometry, which deals with metricable notions on manifolds, has some surprising and fundamental links with topology. The connections arise from a set of theorems of elementary geometry (we refer the reader to the book on elementary differential geometry of O'Neill for a proof of these theorems [71]). We first introduce a few notations and definitions.

Definition 2.3 Rectangle and 2-segment
A rectangle R is a region of the 2D plane $R : a \leq u \leq b\ ,\ c \leq v \leq d$, with $(u, v) \in \mathbb{R}^2$. The interior R° of the rectangle R is the open set $a < u < b\ ,\ c < v < d$. A 2-segment is a transformation from a rectangle R into $\mathbb{R}^3$ that is a one-to-one regular mapping from the interior R° of the rectangle R into $\mathbb{R}^3$.

Definition 2.4 Rectangular decomposition of a surface $\mathcal{C}$
A rectangular decomposition of a surface $\mathcal{C}$ is a finite collection of one-to-one 2-segments whose images cover $\mathcal{C}$ in such a way that if any two intersect, they do so in either a single common vertex or single common edge.

Theorem 2.1 Rectangular decomposition
Every compact surface $\mathcal{C}$ has a rectangular decomposition $\mathcal{D}(\mathcal{C})$.

Theorem 2.2 Euler-characteristic of a rectangular decomposition
If $\mathcal{D}(\mathcal{C})$ is a rectangular decomposition of a compact surface $\mathcal{C}$, let v, e, and f be the number of vertices, edges, and faces in $\mathcal{D}(\mathcal{C})$. Then the integer $(v - e + f)$ is the same for every rectangular decomposition of $\mathcal{C}$. This integer $\chi(\mathcal{C})$ is called the Euler-characteristic of $\mathcal{C}$.

The fact that a rectangular decomposition is used to compute the Euler-characteristic of the surface is merely a convenience for the proof of the theorem. Arbitrary polygons could as well have been used to decompose $\mathcal{C}$. In the resulting polygonal decomposition, the different polygons would still be required to fit neatly, but they would not have the same number of sides. An arbitrary polygonal decomposition is called a tessellation, while, when only triangles are used, the decomposition is called a triangulation of $\mathcal{C}$.

Theorem 2.3 Topological invariance of the Euler-characteristic
If $\mathcal{C}_\mathcal{M}$ and $\mathcal{C}_\mathcal{N}$ are two compact orientable surfaces, $\chi(\mathcal{C}_\mathcal{M}) = \chi(\mathcal{C}_\mathcal{N})$ if and only if $\mathcal{C}_\mathcal{M}$ and $\mathcal{C}_\mathcal{N}$ are homeomorphic.

Thm. 2.3 is of central importance. It states that *the Euler-characteristic of a surface is a topological invariant*. Two surfaces that have the same Euler-characteristic share the same *intrinsic* topology. However, we note that the Euler-characteristic does not define the homotopy type of a surface, since the embedding space is being ignored. Particularly, this implies that a discrete representation of a surface using a polygonal decomposition with the

[2] A manifold is a topological space such that each of its points has a neighborhood that is homeomorphic to an open planar disk.

desired Euler-characteristic might be self-intersecting in the 3D embedding space. We will discuss this important point later.

The Euler-characteristic is of great practical interest because it can be calculated from any polyhedral decomposition $\mathcal{D}$ of the surface by $\chi = v - e + f$, where v, e and f denote respectively the number of vertices, edges and faces of the polyhedron $\mathcal{D}$. The Euler-characteristic of a sphere $\mathcal{S}$ is $\chi(\mathcal{S}) = 2$ (Fig. 2-1-c). This implies that any surface $\mathcal{C}$ with $\chi(\mathcal{C}) = 2$ is topologically equivalent (i.e. homeomorphic) to a sphere and therefore does not contain any handles. Surfaces with an Euler-characteristic $\chi(\mathcal{C}) \neq 2$ have a topology that is different from that of a sphere. However, the Euler-characteristic does not provide any information about the localization of the topological differences. Also, Thm. 2.2 states that the way a surface is decomposed (i.e. tessellated) does not influence its topology. Any polyhedral decomposition of a surface will encode for the same intrinsic topology.

In fact, any compact, connected, and orientable surface is homeomorphic to a sphere with some number of handles. This number of handles is a topological invariant called the *genus*. For example, a sphere is of genus 0 and a torus is of genus 1. The genus g is directly related to the Euler-characteristic χ by the formula $\chi = 2 - 2g$. In the case of multiple surfaces involving K connected components, the total genus is related to the total Euler-characteristic by the formula: $\chi = 2(K - g)$.

2.2.4 On Topological Defects

We have already mentioned that an anatomical structure can be either represented by a volumetric representation or by a surface representation, the two descriptions being dual representations. In this work, we call a topological defect any deviation from the spherical topology. Since we are considering 2D, smooth, orientable, and compact surfaces that are embedded in the 3D Euclidean space, 3 types of topological defects can be encountered:

- Disconnected components: in the presence of image artifacts, segmentations often contain several connected components, which might either constitute parts of the same structure or erroneous pieces of a segmentation.

- Cavities: cavities could be either the result of unexpected anatomical structures that are located inside the volume of interest, such as tumors, or, most frequently, the result of image artifacts. Cavities are usually easy to detect and correct retrospectively if interpreted as connected background components.

- Handles or holes: a handle or hole in a volume or a surface is identified whenever there exists a continuous loop that cannot be homotopically deformed onto a point within the manifold itself. These loops are called non-separating loops and constitute a fundamental concept of algebraic topology [48]. Particularly, these are used to define the fundamental group of an object [48, 62].

Finally, we note that for each defect present in an object (i.e. the foreground object) there exists a corresponding defect in the background: a disconnected foreground component can be interpreted as a background cavity; a foreground cavity is a disconnected background component; and a handle in a foreground component defines another handle in the background component.

This foreground/background duality provides a methodology to correct a topological defect [56, 44] (i.e. any deviation from the spherical topology). For instance, the presence of a handle in an object could be corrected by either cutting the handle in the foreground object, or cutting the corresponding handle in the background object. Cutting the background handle can be interpreted as filling the corresponding hole. We will make use of this dual representation in Chapter 5.

2.3 Topology and Discrete Imaging

In order to apply topological concepts to a discrete framework and to define the topology type (i.e. homotopy type) of digital segmentations, the notion of continuity is transposed into discrete spaces and discrete objects, such as images and triangulations. This is obtained by replacing the notion of continuity with the weaker notion of connectivity.

We describe how topological notions can be adapted to the two most common data structures used in medical imaging: 3D data structures and surfaces. Particularly, we present the main concept of digital topology, which provides a topological framework over the set of 3D digital images (see Sect. 2.3.1). In this work, we are interested in generating accurate surface representations of anatomical structures. For this purpose, we generate surfaces from segmented 3D volumes. This is achieved by using isocontour surface extraction methods (see Sect. 2.3.3).

2.3.1 Digital Topology

Digital topology provides an elegant framework that transposes the continuous concepts of topology to discrete images. In this theory, binary images inherit a precise topological meaning. In particular, the concept of homotopic deformation, which is required to assign a topological type to a digital object, is clearly defined through the notion of *simple point*. An extensive discussion of these concepts can be found in [62].

In this framework, a 3D image is interpreted as a graph. The vertices of the graph are the digital points (voxels in the image) and the edges are defined through neighborhood relations between points. We note that the resulting "discrete" topology is not an instance of mathematical topology à la Munkres [70] and is somehow independent of standard topology theory.

In this section, some basic notions of digital topology are presented. The following definitions are used to define discrete equivalents of topological continuous notions, such as continuity and homotopic deformations. All definitions are from the work of Bertrand, to which we refer for more details and proofs [7].

A - Connectivity

A 3D binary digital image I is composed of a foreground object X and its inverse, the complement $\overline{X}$. We first need to define the concept of *connectivity*, which specifies the condition of adjacency that two points must fulfill to be regarded as connected. Three types of connectivity are commonly used in 3D: 6-, 18- and 26-connectivity. Two voxels are 6-adjacent if they share a face, 18-adjacent if they share at least an edge and 26-adjacent if they share at least a corner (Fig. 2-2-a). In the following, we note $N_n(\mathbf{x})$ the n-neighborhood of a point $\mathbf{x}$, i.e. the set of grid points which are n-adjacent to $\mathbf{x}$. We also set $N_n^*(\mathbf{x}) = N_n(\mathbf{x}) \setminus \{\mathbf{x}\}$. $C_n(X)$ denotes the set of all n-connected components of

X and $C_n(\mathbf{x}, X)$ the set of all n-connected components of X n-adjacent to a point $\mathbf{x}$. The cardinality of a set S is denoted $\#S$.

In order to avoid topological paradoxes, different connectivities, n and $\overline{n}$, must be used for one digital object X and its complement $\overline{X}$. This leaves us with four pairs of compatible connectivities: (6,26), (6,18), (26,6) and (18,6). In order to distinguish the 6-connectivity for X associated with the 18-connectivity for $\overline{X}$ from the 6-connectivity associated with the 26-connectivity for $\overline{X}$, a 6^+-notation is used for the (6,18) pair of connectivities. Figure 2-2-b illustrates with a simple 2D example one type of topological paradoxes, known as the Jordan's curve paradox. Jordan's theorem states that every simple closed curve divides the plane in two compartments, one inside the curve and one outside of it, and that it is impossible to pass continuously from one to the other without crossing the curve. However, if the 8-connectivity is used for the blue and red curves in Fig. 2-2-b, they continuously intersect each-other. In order to avoid such topological paradoxes, a pair of compatible connectivities (4,8) must be used for both curves. The result yields one continuous curve and one discontinuous curve. In 3D, the same topological paradoxes appear; therefore, compatible connectivities must be used for the foreground and the background objects.

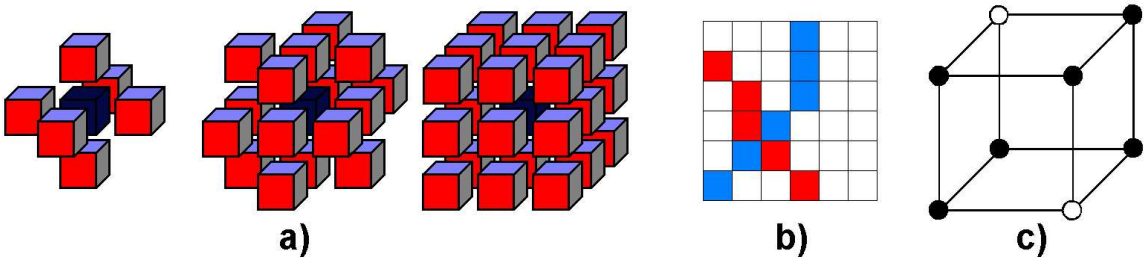

Figure 2-2: a) The three different types of connectivity in 3D. b) A simple two-dimensional example of the Jordan's curve paradox: under the 8-connectivity, the two curves are continuous even though they intersect each other. In order to avoid such topological paradoxes, a pair of compatible connectivities (4,8) must be used for one curve and the other (one curve becomes discontinuous). The same topological paradoxes appear in three dimensions and compatible connectivities must be used for the foreground and background objects. c) Under the 6-connectivity, $(n, \overline{n}) = (6, 26)$, the closed loop formed by the black dots cannot be homotopically deformed into a single point. It would remove a hole in the digital object, since the white dots are $\overline{n}$-connected.

One important consequence of the previous requirement is that digital topology does not provide a consistent framework for multi-label images. No compatible connectivities can be chosen for neighboring components of the same object. Therefore, digital topology is strictly limited to binary images.

B - Path, Topological Defect and Elementary Deformation

An n-path π is a sequence (possibly empty) of points $\mathbf{x}_0 ... \mathbf{x}_k$ with $\mathbf{x}_i$ n-adjacent to $\mathbf{x}_{i-1}$ for $i = 1, ..., k$. The path is elementary if all points in the sequence are different except possibly $\mathbf{x}_0 = \mathbf{x}_k$. An elementary n-path is simple if each point of π has, at most, two n-adjacent points in π. If $\mathbf{x}_0 = \mathbf{x}_k$, then π is closed. Finally, we note that any path from $\mathbf{x}_0$ to $\mathbf{x}_k$ contains an elementary path and a simple path from $\mathbf{x}_0$ to $\mathbf{x}_k$.

Once the concepts of connectivity and path have been introduced, the notion of topological defect can be defined. Similarly to the continuous formulation (see Sect. 2.2.4), topological defects in a digital volume are constituted of holes (i.e. handles), cavities, and

disconnected components. While a cavity in an object X (resp. $\overline{X}$) is easily detected as a finite $\overline{n}$-connected (resp. n-connected) component of $\overline{X}$ (resp. X), the notion of a hole is not simple to define. The presence of a hole in X (resp. $\overline{X}$) is detected whenever there is a closed n-path (resp. $\overline{n}$-path) in X (resp. $\overline{X}$) that cannot be deformed in X (resp. $\overline{X}$) to a simple point.

However, we need to clearly define the notion of deformation of a closed n-path. In the following, $P,\ P',\ P_1,\ P_2$ and $\mathbf{p},\ \mathbf{u}_1,\ \mathbf{u}_2$ denote paths and points respectively. Let $\mathbf{p} \in X$ be a point, called the base point. Let $\gamma \subset X$ and $\gamma' \subset X$ be two closed m-paths. We say that γ' is an elementary n-deformation of γ, noted $\gamma \sim \gamma'$, if γ and γ' are of the form:

$$\gamma = \mathbf{p}P_1\mathbf{u}_1.P.\mathbf{u}_2P_2\mathbf{p},\ \gamma' = \mathbf{p}P_1\mathbf{u}_1.P'.\mathbf{u}_2P_2\mathbf{p}$$

and:

- for $n = 6$, we have $\mathbf{u}_1.P.\mathbf{u}_2$,$\mathbf{u}_1.P'.\mathbf{u}_2$ are included in a unit square (a 2×2 square).
- for $n = 6^+, 18, 26$, we have $\mathbf{u}_1.P.\mathbf{u}_2$,$\mathbf{u}_1.P'.\mathbf{u}_2$ are included in a unit cube (a $2 \times 2 \times 2$ cube).

Figure 2-2-c illustrates with a simple example why a 2×2 square needs to be used with the 6-connectivity. If a $2 \times 2 \times 2$ cube were to be used, the 26-connected white points would be crossed over. Finally, we say that γ' is an n-deformation of γ if there is a sequence of closed n-paths $\gamma_0 ... \gamma_k$, such that $\gamma = \gamma_0$ and $\gamma_{i-1} \sim \gamma_i$ for $i = 1, ..., k$.

C - Geodesic Neighborhoods and Topological Numbers

Definition 2.5 Geodesic Neighborhood *The geodesic neighborhood of a point* $\mathbf{x} \in X$ *of order* k *is the set* $N_n^k(\mathbf{x}, X)$ *defined recursively by:*

$$\begin{cases} N_n^1(\mathbf{x}, X) = N_n^*(\mathbf{x}) \cap X \\ N_n^k(\mathbf{x}, X) = N_{26}^1(\mathbf{x}, X) \cap \bigcup \{N_n(y)\ , y \in N_n^{k-1}(\mathbf{x}, X)\} \end{cases},$$

$N_n^k(\mathbf{x}, X)$ is the set composed of all points $\mathbf{y}$ of $N_{26}^*(\mathbf{x})(\mathbf{x}, X) \cap X$, such that there exists an n-path π from $\mathbf{x}$ to $\mathbf{y}$ of length less than or equal to k and all points of π, except possibly $\mathbf{x}$, belong to $N_{26}^*(\mathbf{x})(\mathbf{x}, X) \cap X$. Geodesic neighborhoods can be interpreted as a discrete equivalent of the notion of open sets (see [37]).

Using these geodesic neighborhoods, the topological numbers relative to the point $\mathbf{x}$ and the set X can be computed:

Definition 2.6 Topological numbers

$$\begin{aligned} T_6(\mathbf{x}, X) &= \#C_6[N_6^2(\mathbf{x}, X)] \\ T_6^+(\mathbf{x}, X) &= \#C_6[N_6^3(\mathbf{x}, X)] \\ T_{18}(\mathbf{x}, X) &= \#C_{18}[N_{18}^2(\mathbf{x}, X)] \\ T_{26}(\mathbf{x}, X) &= \#C_{26}[N_{26}^1(\mathbf{x}, X)] \end{aligned}$$

The topological numbers are the number of connected components within certain geodesic neighborhoods. These numbers have been introduced by Bertrand in [7] as an effective way to characterize the topology type of a given voxel. The values of $T_n(\mathbf{x}, X)$ and $T_{\overline{n}}(\mathbf{x}, \overline{X})$ characterize isolated, interior and border points as well as different kinds of junctions (see tab. 2.1). Their efficient computation, which only involves the 26-neighborhood, is described

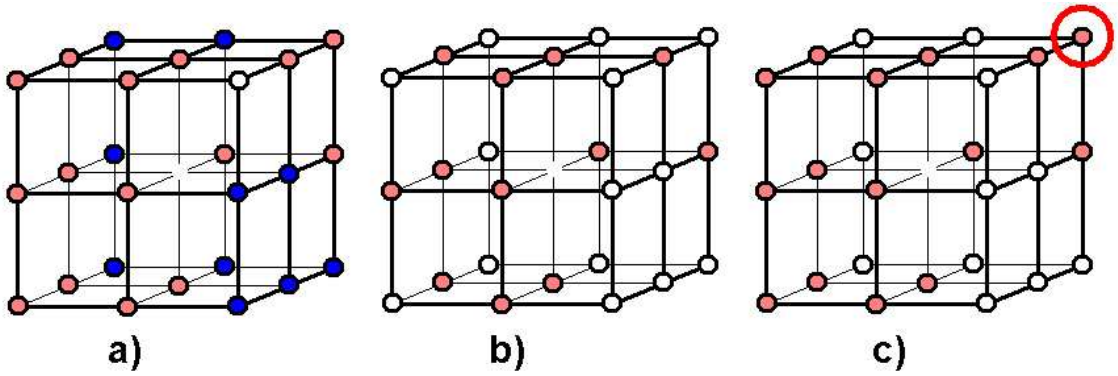

Figure 2-3: a) The 26-neighborhood of a digital object X. Red points correspond to the foreground object X, while blue points correspond to the background object $\overline{X}$. b) The geodesic neighborhood $N_6^2(\mathbf{x}, X)$ of the point $\mathbf{x}$. c) The geodesic neighborhood $N_6^3(\mathbf{x}, X)$ of the same point $\mathbf{x}$. Note the crucial difference between both neighborhoods circled in red. The circled voxel explains why the topological numbers T_6 and T_{6^+} are different; we have $T_6(\mathbf{x}, X) = 2$ and $T_{6^+}(\mathbf{x}, X) = 1$. Consequently, the point $\mathbf{x}$ will be simple for the connectivity 6^+ but not for the connectivity 6.

in [8]. Figure 2-4 provides several examples of digital configurations with their associated topological numbers. $N_n(\mathbf{x}, X)$ denotes $N_n^k(\mathbf{x}, X)$, where k equals 2,3,2, and 1, when n is 6,6^+, 18, and 26 respectively. Using this notation, we have $T_n(\mathbf{x}, X) = \#C_n[N_n(\mathbf{x}, X)]$.

Table 2.1: Voxel topology types and topological numbers

$T_n(\mathbf{x}, X)$	$T_{\overline{n}}(\mathbf{x}, X)$	Topology Type of $\mathbf{x} \in X$
$T_n = 0$		isolated point
	$T_{\overline{n}} = 0$	interior point
$T_n = 1$	$T_{\overline{n}} = 1$	border point (simple point)
$T_n = 2$	$T_{\overline{n}} = 1$	curve point
$T_n > 2$	$T_{\overline{n}} = 1$	curves junction
$T_n = 1$	$T_{\overline{n}} = 2$	surface point
$T_n = 1$	$T_{\overline{n}} > 2$	surfaces junction
$T_n > 1$	$T_{\overline{n}} > 1$	surface(s)-curve(s) junction

D - Simple Point and Homotopic Deformation

Definition 2.7 Simple point *A point of a binary object is* simple *if it can be added or removed without changing the topology of both the object and the background, i.e. without changing the number of connected components, cavities and handles of both X and $\overline{X}$.*

A simple point is easily characterized by the two topological numbers $T_n(\mathbf{x}, X)$ and $T_{\overline{n}}(\mathbf{x}, \overline{X})$: a point is simple if and only if $T_n(\mathbf{x}, X) = T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1$.

The definition of a discrete homotopy follows from the concept of simple point.

Definition 2.8 Homotopic deformation *We define a homotopic deformation of an object X as a sequence of deletions or additions of simple points.*

Finally, two objects X and Y share the same homotopy type if there exists a sequence of transformations $X_0...X_k$ and a sequence of points $\mathbf{x}_1...\mathbf{x}_k$, such that $X_0 = X$ and $X_{i-1} = X_i \bigcup \{\mathbf{x}_i\}$ or $X_{i-1} = X_i \backslash \{\mathbf{x}_i\}$ and the point $\mathbf{x}_i$ is simple relative to X_i for $i = 1, ..., k$.

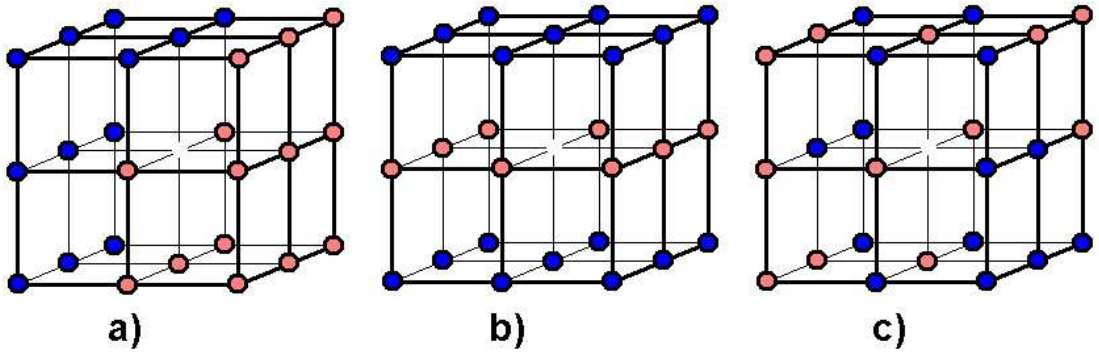

Figure 2-4: Examples of topological numbers for different digital configurations. The red points correspond to the foreground object X. a) A simple point: $T_n = T_{\overline{n}} = 1$. b) A non-simple point, $T_n = 1$, $T_{\overline{n}} = 2$. c) In this complex configuration, the point $\mathbf{x}$ will be simple or not depending on the chosen connectivity. The point $\mathbf{x}$ is simple only when the chosen connectivity is $n = 6^+$ (implying that $n = 18$); in this case, we obviously have: $T_n = 1$, $T_{\overline{n}} = 1$. For all other choices of connectivity, the point is non-simple, but the reasons differ. For $n = 6$ (and $\overline{n} = 26$), we have $T_n = 4$, $T_{\overline{n}} = 1$. In the other two cases ($n = 18$ and $n = 26$), we have $T_n = 1$ but $T_{\overline{n}} = 2$.

One final important comment needs to be made. Arbitrary digital homotopic deformations might not be able to deform any initial object into another one that shares the same homotopy type. To explicate this point, let's consider two spherical objects, X_1 and X_2 with $\chi(X_1) = \chi(X_2) = 2$, the first one being strictly included in the second one ($X_1 \subset X_2$ and $X_1 \neq X_2$). We then consider the set $S_{X_1 \rightarrow X_2}$ of homotopic dilatations from X_1 into X_2 (i.e. successive additions of simple points $\mathbf{x} \in X_2$). Our comment implies that not all deformations of $S_{X_1 \rightarrow X_2}$ will be able to deform the object X_1 into the object X_2. Some deformations $d \in S_{X_1 \rightarrow X_2}$ might generate some digital objects $X_d = d(X_1)$ with $X_d \subset X_2$ and $X_d \neq X_2$, such that every point $\mathbf{x} \in X_2 \setminus X_d$ is non-simple.

This limitation is little known and has been, overtly or covertly, ignored in most (if not all) of the literature. Most often, this limitation can be disregarded as only highly random homotopic deformations will exhibit such behaviors. However, images that contain a lot of noise might seriously limit the applicability of digital homotopic deformations.

2.3.2 Surfaces in Discrete Imaging

We now turn to the transposition of continuous topological concepts to discrete surface representations. There are essentially two ways of representing a surface in discrete imaging. Surfaces can be either represented explicitly, by using parameterized models, or implicitly as the level set of some function defined in the 3D embedding space. Both types of representations have advantages and disadvantages, and have been extensively used for the purpose of medical image segmentation [104, 61, 18, 16, 39]. Particularly, the theory of active contours (or deformable models) constitutes a general technique of matching a "deformable" surface onto an image by means of energy minimization. This popular and powerful image

segmentation method is presented in the next section.

A - Explicit Representations

An explicit representation models a surface by a set of vertices, edges, and faces, associated with a chosen parameterization of each face. The set of vertices, edges, and faces composes the polyhedral representation of the surface. The parameterization of the faces determines the exact geometry of the surface. For instance, tessellations correspond to linear parameterizations of each face, while splines use higher-order approximations. Triangulations are a special kind of tessellation, in which each face is a triangle.

The topological invariance of the Euler-characteristic in Thm. 2.3 implies that explicit models unambiguously encode the intrinsic topology of the surfaces. Given a specific polyhedral decomposition, the Euler-characteristic of the surface is easily computed using the formula $\chi = v - e + f$. For instance, a connected surface with an Euler-characteristic $\chi = 0$ is homeomorphic to a torus, and possesses a single handle. However, the Euler-characteristic does not provide any information on the location of the handle.

While the intrinsic topology of the surface is directly encoded in the polyhedral decomposition, there is no guarantee that the surface representation is not self-intersecting. As previously mentioned, the topological equivalence defining the intrinsic topology of a geometric entity ignores the embedding space. Consequently, additional precaution must be taken in order to ensure that the discretization of a surface does not generate self-intersections. Figure 2-5 illustrates this point with a simple 2D closed curve. Depending on the resolution of the discretization of the continuous curve, self-intersections may appear in the embedding space, even though the Euler-characteristic of the polyhedral decomposition does not change ($\chi = v - e = 0$). The self-intersection problem is important when the surfaces are iteratively deformed in order to match a targeted structure. We will discuss this point when we present the theory of active contours in the next section.

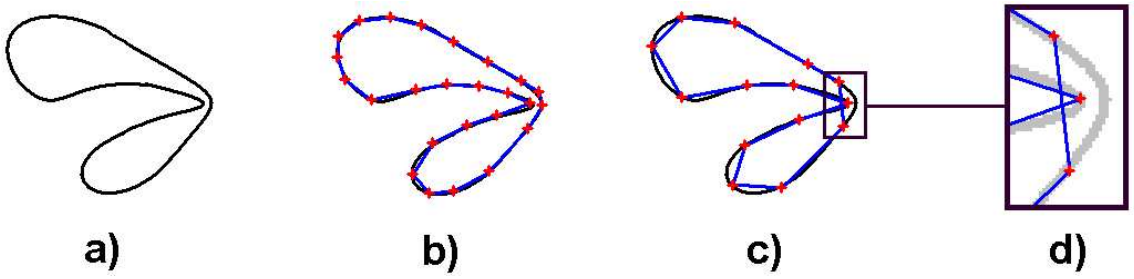

Figure 2-5: a) a simple closed curve with the topology of a circle. b) One example of a polyhedral decomposition of the curve using 25 vertices and edges. The corresponding Euler-characteristic $\chi = v - e = 0$ is the one of a circle. c) Another discretization of the same curve using 14 edges and vertices. Note that the Euler-characteristic is still the one of a circle $\chi = 0$, even though the discrete representation of the curve self-intersects in the 2D embedding space. d) A close-up of the self-intersecting representation.

Finally, we note that explicit representations can approximate surfaces at any level of precision, by using more refined meshes. Contrary to the theory of digital topology that constitutes a discrete approximation of the continuous space, and is therefore limited by the resolution of the 3D digital images, explicit representations can approximate accurately any surface by using high-resolution meshes.

B - Implicit Representations

Implicit models encode the surface of interest Γ as the level set of a higher-dimensional function ϕ defined in the embedding space. The function ϕ, defined on a 3D voxel grid, is usually the signed distance function of the surface with the contour being the zero level set of ϕ: $\Gamma = \phi^{-1}(0)$.

This type of representation has several advantages. First, no explicit representation and no parameterization are required. In the theory of active contours, this will prove to be a huge advantage as implicit representations can naturally change topology during the deformation of the model. Self-intersections, which are costly to prevent in parametric deformable models, are avoided and topological changes are automated. Also, many fundamental properties of the surface Γ, such as its normal or its curvature, are easily computed from the level set function ϕ.

However, these models can only represent manifolds of codimension one without borders, such as closed curve in $\mathbb{R}^2$ or closed surfaces in $\mathbb{R}^3$. For the purpose of segmenting anatomical structures, the use of such representations is not a limitation. Another - more subtle - drawback of implicit representations is that, even though level sets achieve sub-voxel accuracy, the exact location of the contour depends on the image resolution. For instance, in the case of two self-touching banks of a sulcus, the finite image resolution and the topological constraint necessitate some voxels to be labeled as outside voxels (ideally, these voxels should be the ones belonging to CSF), therefore imposing a constraint on the location of the surface. Other limitations of the implicit representations are presented in the next section.

So far, one has not specified how implicit representations can ensure that the topology of the encoded surface is the correct one. Since implicit representations make use of the underlying 3D voxel grid (through a signed distance function) to encode the contour of interest, digital topology can be used to specify the topology of the contour. The foreground object X is simply defined as the set of negative grid points (i.e. $X = \{\mathbf{x} \in \mathbb{R}^3 \mid \phi(\mathbf{x}) \leq 0\}$), and the background object $\overline{X}$ as the set of strictly positive grid points (i.e. $\overline{X} = \{\mathbf{x} \in \mathbb{R}^3 \mid \phi(\mathbf{x}) > 0\}$). Then, given a choice of compatible connectivities, the topology of the contour is determined unambiguously.

Finally, we need to explain how we can generate an explicit representation from an implicit surface model. This leads us to the theory of isocontour extraction.

2.3.3 From Images to Surfaces: Isocontour Extraction

In the previous section, we described the manner in which topology can be adapted to the two most common data structures used in medical imaging. The ability to go from one representation to the next arises as a difficulty. As described in the following sections, it is possible to generate triangulations from 3D binary digital segmentations, such that the resulting topology of the surfaces is consistent with the choice of digital topology. However, we note that it is not always possible to produce a digital binary volume, whose topology is similar to that of a given triangulation: digital topology constitutes a discrete approximation of the continuous space at a finite resolution, while triangulations approximate continuous surfaces at any level of precision.

A - The Original Marching Cubes Algorithm

The marching cubes (MC) algorithm was first introduced by Lorensen and Cline in 1987 [59] as a way to generate a polygonal decomposition from a scalar field sampled on a rectilinear grid. Given an isovalue, the MC algorithm extracts quickly a representation of the isosurface of the scalar field. Formally, an isosurface is defined as a surface that connects all the points of a 3D space that have the same associated function value; this function value is called the isovalue.

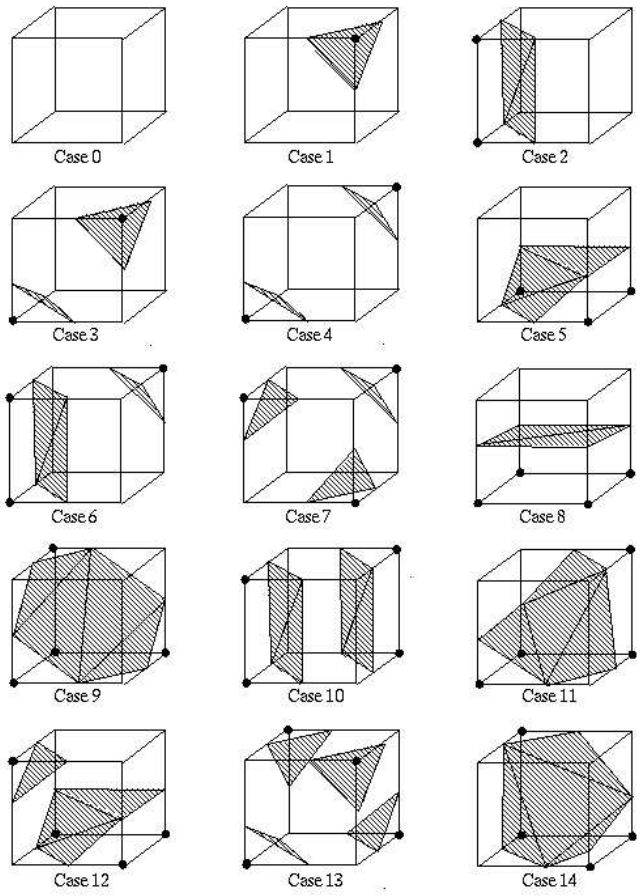

Figure 2-6: The 15 representative cases of the marching cubes algorithm. This figure was taken from [46].

The MC algorithm first partitions the data into a set of cubic (or rectilinear) cells, the cell vertices being the grid points. Based on the relative polarity of their scalar value (above or below the isovalue), each vertex is assigned a binary label, which indicates whether the grid point is inside or outside the isosurface. Then each cubic cell is processed sequentially. Patches that approximate the isosurface are produced within each cube, and the polygon patches are naturally joined together to form the final isosurface representation.

The tiling generated for each cubic cell is based on the assumption that there is exactly one surface intersection at a cube edge that connects oppositely labeled vertices, and there is no intersection if an edge connects two vertices of the same polarity. Also, no isolated components can be fully contained inside a cubical cell. Basically, these assumptions state

that the resolution of the isosurface is that of the rectangular grid. Each vertex in a cubic cell has one of two possible labels, resulting in only $2^8 = 256$ ways an isosurface can intersect a cube. Using rotational and complementarity invariance properties, the 256 cases can be reduced to 15 major cases shown in Fig. 2-6. The MC algorithm extracts a triangulated surface whose vertices lie on the edges of the cubic lattice, generating 0 to 4 triangles per cell. The exact location of each vertex is determined by linear interpolation from the values of the cubic lattice.

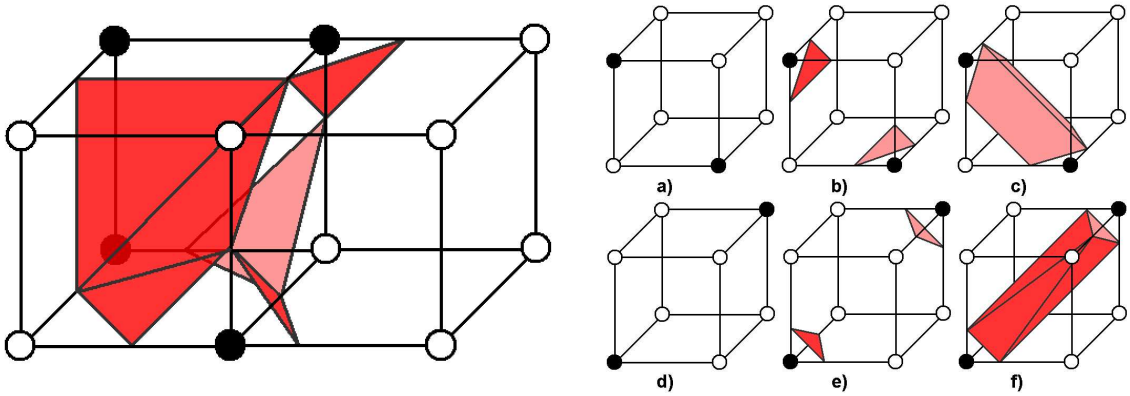

Figure 2-7: Left: An incorrect tiling occurring on an ambiguous face in between case 12 and case 3 with the 256-lookup table. Right: a) An ambiguous face. b) One possible tiling for the 6- or 6^+-connectivity. c) One possible tiling for the 18- or 26-connectivity. d) An ambiguous cube. e) One possible tiling for the 6-, 6^+ or 18-connectivity. f) One possible tiling for the 26-connectivity. The original MC algorithm, introduced by Lorensen and Cline, generate tilings corresponding to b) and e).

This method is quite fast (one pass on the cubic grid is necessary to generate a triangulation), but the resulting triangulation might not reflect the topology of the underlying digital grid and may produce tiling inconsistencies. The left figure in Fig. 2-7 shows an ambiguous configuration that produces an inconsistent tiling. These inconsistent tilings are related to the so-called ambiguous face and ambiguous cube configuration (Fig. 2-7-right-a,d). The major differences between different MC algorithm implementations lie in how they choose between the two possible tilings for the ambiguous face and cube cases. Extensive discussion of isocontour extraction algorithms can be found in the thesis of Han [47].

B - Connectivity-Consistent Marching Cubes Algorithm

The original marching cubes formulation is not able to generate topologically-correct isosurfaces, since the resulting tessellations may contain tiling and topological inconsistencies. In [46], the authors proposed a connectivity-consistent marching cubes algorithm by building a specialized case table for each type of digital topology. In this MC algorithm, the tiling for the ambiguous cases is determined by the choice of connectivity used for the underlying 3-dimensional cubic lattice. The coordinates of surface intersections are still computed using linear interpolation, but the resolution of ambiguous faces and cubes depends on the pre-defined digital connectivity rule. In particular, Fig. 2-7c,e correspond to the chosen tilings when the black points are assumed to be 18-connected, and Fig. 2-7c,f illustrate the selected tilings when the black points are assumed to be 26-connected. Figure 2-8 illustrates that the mesh generated by this algorithm depends on the chosen connectivity pair. In this

example, four different meshes are obtained from the same level set function.

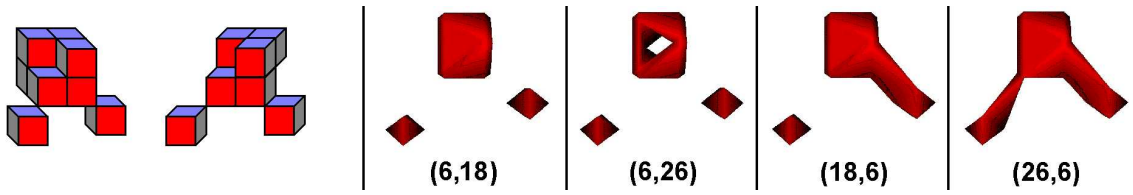

Figure 2-8: Mesh extracted from the same level set function by the topology-consistent marching cubes algorithm, when using different connectivity pairs. The voxel representations on the right correspond to two front views, one front left view and one front right view, of the binarized level set function (relative to the same isovalue used in the topologically-consistent marching cubes).

2.4 State of the Art in Segmentation under Topological Constraints

The segmentation of anatomical structures under topological constraints has been an intensive area of research over the past years. Many segmentation algorithms are able to identify and locate precisely these structures, although without constraining the topology. Those include active contours [64, 65, 18, 104, 106, 103] and digital techniques [74, 79, 100, 29, 100, 107]. The resulting segmentations often contain several topological defects, such as disconnected components, cavities or handles, which do not correspond to the true anatomy of the structures of interest.

Methods for producing topologically-correct segmentations can be broadly divided into two categories. A first set of approaches directly incorporates topological constraints into the segmentation process. These methods deform iteratively a model carrying the desired topology onto the desired anatomical structure while preserving the model topology. These "intuitive" methods have been intensively used for the purpose of medical imaging segmentations. Yet, the model evolution is often difficult to control and the accuracy and correctness of the final segmentation difficult to guarantee.

Recently, new approaches have been developed to correct retrospectively the topology of an already segmented image. These techniques, which do not impose any topological constraints during the segmentation process, can focus on obtaining more accurate models. However, while these methods guarantee that the final segmentations have a spherical topology, the accuracy of the topological corrections might not be one which that a trained operator would achieve. In particular, these methods fail to integrate statistical and geometrical information into the topology correction process, and the automatic extraction of accurate and topologically-correct segmentations is still a challenging problem.

2.4.1 Topologically-Constrained Segmentations

The topology-enforcing techniques proceed by deforming iteratively a model of known topology onto a targeted structure, while preserving its topology. Several techniques have been used for the segmentation of anatomical structures, with the topological constraint taking different forms depending on the chosen method.

A - Active Contours

Active contours, also known in the literature as snakes, active surfaces, deformable models/contours/surfaces, represent a class of popular and powerful medical image segmentation methods due to their ability to combine low-level image information with high-level prior knowledge about object shapes [64, 103]. They constitute a general technique of matching a deformable model onto an image by means of energy minimization. A deformable model is a curve, a surface, or a higher-dimensional geometric object, that deforms within an input image subject to both internal and external forces and external constraints. Since their introduction by Kass et al. in [54], active contours have been applied in many computer vision research areas (image segmentation [12, 104, 61, 18, 16, 39], region tracking [68, 73], shape from stereo [25, 51, 24, 40], shape from shading [35, 105], and shape from point clouds [5, 93, 108], etc.). Comprehensive studies can be found in [64, 103, 65].

This methodology has several remarkable features. Due to its high versatility in terms of the choice of a shape representation and the design of the evolution equation, active contours can be applied to various types of input data (n-dimensional images, unstructured point sets, ...) and to a large range of problems. Also, the use of a continuous geometric formulation has many benefits. It leads to a neat mathematical framework; it provides subpixel accuracy; it correctly models the continuity of shape, and allows one to incorporate some regularity assumptions or some complex prior information about the target shape, thereby yielding a good robustness to noisy and incomplete data.

In this dissertation, we focus on 2D surfaces evolving in 3D images. In 3D, an active surface is represented by a family of closed surfaces $\Gamma : t \in \mathbb{R}^+ \to \Gamma(t)$, where $\forall t \in \mathbb{R}^+$, $\Gamma(t)$ is a surface in $\mathbb{R}^3$ that can be parameterized $\Gamma(t) : \mathbf{p} \in \mathbb{R}^2 \to \mathbf{\Gamma}(\mathbf{p}, t) = [x(\mathbf{p}, t), y(\mathbf{p}, t), z(\mathbf{p}, t)] \in \mathbb{R}^3$. The variable $t \in \mathbb{R}^+$ parameterizes the family of surfaces Γ and $\mathbf{p} \in \mathbb{R}^2$ parameterizes each surface $\Gamma(t)$ at a given time t. The geometric shape of the evolving contour is fully determined by the normal component of the driving force, while the tangential component only affects the parameterization. Consequently, the evolution equation can be always written as:

$$\forall t \in \mathbb{R}^+ \; \forall \mathbf{p} \in \mathbb{R}^2 \; \frac{\partial \mathbf{\Gamma}(\mathbf{p}, t)}{\partial t} = F(\mathbf{\Gamma}(\mathbf{p}, t), t)\mathbf{n}(\mathbf{\Gamma}(\mathbf{p}, t), t),$$

where $F(\mathbf{\Gamma}(\mathbf{p}, t), t)$ is a scalar function evaluated at location $\mathbf{\Gamma}(\mathbf{p}, t)$ and $\mathbf{n}(\mathbf{\Gamma}(\mathbf{p}, t), t)$ represents the outward normal to the closed contour $\Gamma(t)$ at location $\mathbf{\Gamma}(\mathbf{p}, t)$. The vector field $\mathbf{v}(\mathbf{x}, t) = F(\mathbf{x}, t)\mathbf{n}(\mathbf{x}, t)$ is called the *velocity field*, and is designed to drive the evolving contour towards the desired boundary.

As we have already seen in Sect. 2.3.2, surfaces can either be represented explicitly, by using parameterized models, or implicitly as the level set of a 3D function defined on the embedding space. In the theory of active contours, this leads to two different implementations. The former encodes the manifold of interest with an explicit representation using a Lagrangian formulation, while the latter represents implicitly the contour as the level set of a function defined on higher-dimensional manifold in an Eulerian formulation [72]. We refer to the dissertation of Pons for an in-depth discussion of these concepts [77].

A.1 - Explicit representations

Explicit representations correspond to the type of parameterized representations proposed in the original snake model of Kass, Witkin and Terzopoulos [54]. Other types of parameterized representations are found in the literature: finite elements [14, 66], B-spline [82],

Fourier harmonics [90], superquadrics [94, 4, 5], ...

Parameterized models maintain an explicit representation of the contour and preserve the *intrinsic* topology of the initial contour. However, the preservation of the whole topology requires also the prevention of self-intersections, which proves to be computationally intensive and requires elaborate methods to detect and prevent surface intersection during the evolution. To illustrate this point, let's consider an active contour modeled by a tessellation. A straight-forward implementation of surface self-intersections has a computational complexity of f^2, where f is the number of faces in the tessellation. Although the use of a coarsely discretized spatial lookup table will reduce the number of faces to be tested, resulting in a linear time complexity, this process is time-consuming and drastically slows down the active surface evolution [16, 61].

In addition, the preservation of the initial topology is often a strong limitation to most explicit models, since explicit representations cannot change their topology during the evolution in order to fit the data topology. This is one important reason why the level set representation has received much interest in the deformable models literature. Several attempts have been made to overcome this problem. McInerney and Terzopoulos [65, 67] have introduced the concept of *T-snakes* and *T-surfaces*, which are some topology adaptative deformable curves and triangulations. However, their approach is limited to a specific type of motion, where the model inflates or deflates only. Lachaud and Montanvert [57] propose a method based on the concept of δ-triangulation. Their approach is computationally intensive. A length parameter δ is used to control the sampling of the triangulation and to detect self-intersections, by monitoring the distance between pairs of neighbor and non-neighbor vertices. Delingette and Montagnat [20, 21] propose to modify the topology of an evolving simplex mesh with some elementary topological operators, but their approach needs manual interaction in 3D. Thus, a fully automatic and efficient handling of topology changes using explicit models remains an open issue.

A.2 - Implicit representations

Geometric active contours (i.e. level sets), which have been introduced by Caselles et al. [12], offer many advantages over explicit representations. In addition to their ease of implementation, level sets do not require any parameterization of the evolving contour. Self-intersections, which are costly to prevent in explicit deformable models, are naturally avoided and topological changes are automated. Also, many fundamental properties of the active contours, such as its normal or its curvature, are easily computed from the level set function. Last but not least, the theory of viscosity solutions [15] provides robust numerical schemes and strong mathematical results to deal with the evolution of the contour.

However, the level set formulation has several disadvantages that limit its applicability. First, the level set formulation can only represent manifolds of codimension one without borders, such as closed curve in $\mathbb{R}^2$ or closed surfaces in $\mathbb{R}^3$. Also, the representation of the contour of interest using the level set of a function defined on higher-dimensional manifolds leads to implementations that are much more computationally expensive than explicit approaches. Some methods have been proposed to alleviate this problem, such as the narrow band method [1]. In this methodology, the function encoding the active contour is updated only in a small neighborhood (i.e. a narrow band) of the active contour, and the neighborhood is updated iteratively throughout the level set evolution.

Another shortcoming of the level set formulation is the point-wise correspondence prob-

lem. The implicit formulation and the absence of parameterization result in the loss of the point-wise correspondence during the evolution. Some recent work proposed a method, based on a system of coupled Eulerian partial differential equations, to overcome this limitation [76, 77].

The ability to automatically change the topology of the active contour is often presented as an advantage of level set methods over explicit deformable models. However, this behavior is not desirable in some applications. This is typically the case in biomedical image segmentation, where the topology of the target shape is prescribed by anatomical knowledge. In order to overcome this problem, a topology-preserving variant of the level set method has been proposed [46]. Their method is based on the theory of digital topology and uses the underlying embedding space to constraint the topology of the interior of the level set. However, the strict topology preservation necessitates an initialization of the active contour that is close to its final configuration in order to avoid topological barriers that can easily generate large geometrical errors. In the case of complex structures, like the cortical surface, such initialization proves to be extremely difficult. For this purpose, Han et al. have designed a complex algorithm to correct retrospectively the topology of a binary segmentation, thereby providing retrospectively an accurate initialization of the geometric active contours [44].

In the next two paragraphs, we present in more details the level set method. The standard level set formulation is presented first, followed by the topology-preserving method of [46].

A.2.a - The Standard Level Set Formulation

The level set method models the evolution of an active contour $\Gamma : t \in \mathbb{R}^+ \rightarrow \Gamma(t)$, where $\Gamma(t)$ is a closed and embedded hypersurface in $\mathbb{R}^n$, by the level set of a function defined on $\mathbb{R}^n$. Although many functions could be chosen to represent the active contour Γ, the signed distance function is preferred for its stability in numerical computations.

In more details, the moving contour Γ is represented by a level set function $\phi : \mathbb{R}^n \times \mathbb{R}^+ \rightarrow \mathbb{R}$ such that:

$$\begin{cases} \phi(\mathbf{x}, t) < 0 & \text{if} \quad \mathbf{x} \text{ is inside } \Gamma(\text{t}) \\ \phi(\mathbf{x}, t) = 0 & \text{if} \quad \mathbf{x} \in \Gamma(t) \\ \phi(\mathbf{x}, t) > 0 & \text{if} \quad \mathbf{x} \text{ is outside } \Gamma(\text{t}) \end{cases}$$

A deformation of Γ under the velocity field $\mathbf{v}$:

$$\frac{\partial \mathbf{\Gamma}(\mathbf{x}, t)}{\partial t} = \mathbf{v}(\mathbf{x}, t)$$

corresponds to the level set formulation:

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + \mathbf{v}(\mathbf{x}, t).\nabla \phi(\mathbf{x}, t) = 0.$$

One possible implementation of the level set evolution using an explicit time step is described in Alg. 1.

A number of points need to be explicated. The above procedure updates all grid points in the image, when only the points close to the current active contour need to be updated. Some methods have been proposed to alleviate this problem, such as the narrow band

method [1]. The grid points are updated only in a small neighborhood of the active contour.

Also, in the level set formulation, the stability of the numerical evolution depends crucially on the chosen numerical schemes. The gradients and other derivatives require adapted numerical schemes, which depend on the type of velocity field or the type of partial differential equation to be updated.

Algorithm 1 Standard level sets

```
for all iterations do
  for all grid points x do
    Compute the new value of the level set function at (t + Δt)
    φ(x, t + Δt) = φ(x, t) + Δt v(x, t).∇φ(x, t)
    if necessary then
      Reinitialize the distance function φ.
```

Finally, we note that the distance function ϕ needs to be regularly reinitialized during the evolution of the active contour. Some methods have been proposed to avoid this costly reinitialization step [41]. We refer the reader to the dissertation of Pons for more details [77] on the theory of level set.

A.2.b - Topology-Preserving Level Sets

The ability to automatically handle topology changes is a long-acknowledged advantage of the level set method over explicit deformable models, but may not be desirable in some applications where some prior knowledge of the target topology is available. This is typically the case in biomedical image segmentation, where the topology of the organs and their mutual topological relations is prescribed by anatomical knowledge.

In order to overcome this problem, a topology-preserving variant of the level set method has been proposed [46]. The level set function is iteratively updated with a modified procedure based on the concept of *simple point*, borrowed from digital topology (see Sect. 2.3.1); the final mesh is obtained by a topology-consistent marching cubes algorithm (see Sect. 2.3.3). This method ensures that the resulting mesh has the same topology as the user-defined initial level set.

Algorithm 2 Topology-preserving level sets. Han et al. [46]

```
for all iterations do
  for all grid points do
    Compute the new value of the level set function
    if the sign does not change then
      Accept the new value
    else {sign change}
      Compute the topological numbers
      if the point is simple then
        Accept the new value
      else
        Discard the new value
        Set instead a small value of the adequate sign
```

Han et al. [46] have used the concept of simple point to design a topology-preserving

variant of the level set framework. The binary object of interest is the interior of the contour, i.e. the domain where the level set function is strictly negative: $X = \{x \mid \Phi(x) < 0\}$. The digital topology of X is preserved during the evolution by means of a modified update procedure detailed in Algorithm 2. This approach prevents *digital* non-simple grid points from changing sign, thereby retaining the initial digital topology throughout the level set evolution.

For this method to be useful, it must be complemented with a topology-consistent isocontour extraction algorithm. Standard marching squares or marching cubes algorithm [60] do not generate topologically-consistent tessellations. In order to alleviate this problem, Han et al. have designed a modified connectivity-consistent marching contour algorithm, by building a specialized case table for each type of digital topology (see Sect. 2.3.3). Using the topology-preserving level set algorithm and the topology-consistent marching contour algorithm in conjunction, with the same digital topology $(n, \overline{n})$ throughout the process, guarantees that the output mesh is topologically equivalent to the user-defined initial level set.

A.3 - Limitations of active contour methods

All these methods have the advantage of allowing the user to specify the proper topology and not allowing it to change. Unfortunately, the energy functionals driving the deformation are highly non-convex and the attainment of the desired final surface most often requires an initialization of the model that is close to its final configuration. Furthermore, these methods often fail to represent accurately deep folds in the surface. In addition, local topological constraints can easily lead to large geometric inaccuracies in the final cortical representation that are difficult to correct retrospectively. In the case of cortical segmentation, this can occur, for instance, when an erroneous segmentation results in a bridge connecting two banks of a sulcus. In order to maintain the correct topology, the surface must "drape" over the incorrectly classified region. Finally, one more subtle drawback of implicit representations is that even though level sets achieve sub-voxel accuracy, the exact location of the contour depends on the image resolution. For instance, in the case of two self-touching banks of a sulcus, the finite image resolution and the topological constraint necessitate some voxels to be labeled as outside voxels, therefore imposing a constraint on the location of the surface. To solve this problem, Han et al. [45] have implemented a moving grid algorithm, which aims at optimally deforming the underlying 3D grid for accurate implicit representations.

B - Digital Homotopic Deformation

Similarly to active contour models, digital approaches [62, 78, 10, 6] deform an initial region with a known given topology (typically a single voxel carrying a spherical topology), by addition/deletion of points, minimizing a global energy functional while preserving the correct digital topology. Regions are grown or shrunk by adding points that will not change the region topology. Most of these methods are based on the theory of digital topology and the notion of simple point that we have reviewed in Sect. 2.3.1.

The limitations of these methods are the same as those of active contours. The final segmentation is strongly influenced by the order in which the region is deformed, and also by local topological constraints which potentially lead to large geometrical errors. Particularly,

the locations of the final cuts (i.e. the locations of the non-simple points) strongly depend on the order in which the points are added to the topologically constrained region. Also, these methods are limited by the resolution of the image and might not be able to represent deep folds in the structure. Finally, we note that the theory of digital topology does not provide a consistent framework for multi-labeled images and is therefore limited to binary images. Some approaches [78, 6] ignore this issue by assigning the same connectivity rule to all structures of interest - in which case, the connectivity has to be chosen equal to (6,26). However, in this *inconsistent* framework, it is not possible to guarantee all voxels to be assigned to a specific structure. Also, the detection of topological changes, i.e. the location of non-simple points, is more challenging, since topological numbers cannot be computed in a consistent manner.

C - Segmentation by Registration

Finally, some approaches have been proposed to match a template with a given topology onto a specified MRI image [13, 53, 6]. These methods have the strong advantage of being able to enforce complex topology in the segmentation process, and to encode the spatial relationships that exist in between structures [6]. Yet, the design of elaborate templates that include several structures with the correct topology is challenging.

Similarly to other topologically-constrained methods, the topological preservation might lead to large geometrical errors in the final segmentation. Moreover, the projection of the template topology onto a given image is a non-trivial task [53].

D - Summary of the Limitations of Topologically-Constrained Segmentations

Approaches that directly integrate the topological constraint into the segmentation process have the advantage of allowing the user to explicitly specify the topology of the final segmentation. In the case of segmentation by registration, full brain models containing several structures can be matched onto a targeted image.

However, these methods are highly sensitive with regard to their initialization, and accurate final configurations most often require an initialization of the models that is close to its final configuration. One of the main reasons is that the energy functionals driving the deformation are most often highly non-convex and the evolution easily trapped in local minima. The design of more elaborate energy functionals generally offers a solution to this problem. Another reason is that topologically constrained evolution often leads to large geometric errors, due to the topological constraint and the presence of topological barriers. This is the case for methods that aim at segmenting the cortex starting from one single object located deep inside the cortical surface. Large topological barriers are often generated during the template deformation leading to inaccurate final segmentations. This is mostly a result of the presence of noise in the image and of the fact that topologically constrained segmentation prevents the formation of cavities (easy to detect and suppress) as well as the formation of handles.

Finally, we note that digital methods, as well as implicit representations that use the 3D embedding space to encode the surface of interest, are constrained by the finite resolution of the 3D grid. This might be problematic for the segmentation of the pial surface (the surface separating gray matter from cerebrospinal fluid), as opposite banks of a sulcus are often in tangential contact.

Topologically constrained segmentation methods and their limitations is the focus of Chapters 3 and 4. In Chapter 3, using the theory of digital topology, we introduce a new concept that extends and generalizes the restrictive notion of simple point. Simple points preserve the topology of a digital object during a deformation. However, the generation of cavities or disconnected components as well as the formation of handles is prevented. The extension of the concept of simple point to multi-label images, which we call multisimple point, ensures that no topological defects are generated while splitting or merging th e components of the object.Based on this concept, in Chapter 4, we then introduce a new active contour framework that is more flexible than topology-preserving level sets. This algorithm introduces a fine degree of control over topological changes in level-set-based image segmentation.

2.4.2 Retrospective Topology Correction

Recently, new approaches have been developed to correct retrospectively the topology of an already-segmented image. These techniques, which do not impose any topological constraints on the segmentation process, can focus on attaining more accurate models. Many segmentation techniques, using local intensity, prior probabilities, and/or geometric information regardless of topology, will be able to generate accurate cortical surfaces, with few topological inconsistencies.

These methods can be divided into two main classes: volume-based methods that work directly on the volume lattice and correct the topology by addition/deletion of voxels [89, 56, 44], and surface-based methods that aim at modifying the tessellation by locating and cutting handles [28, 43].

A - Volume-based Approaches

Most volume-based approaches have been specifically designed to correct the topology of the cortical surface.

In pioneering work, Shattuck and Leahy proposed a method to remove all the handles from a binary white matter segmentation of the brain [89]. They examine the connectivity of the segmentation to detect topological defects and minimally correct them by changing as few voxels as possible. One drawback of their approach is that the "cuts", which are necessary to correct the topological defects, can only be oriented along the Cartesian axes and give rise to "unnatural" topological corrections. Their method is based on the theory of digital topology but is limited to the 6-connectivity and has not been generalized for any other connectivity rule.

Inspired by their work, Han et al. developed an algorithm to correct the topology of a binary object under any digital connectivity [44]. They detect handles by graph analysis, using successive foreground and background morphological openings to iteratively break the potential topological defects at the smallest scales. Contrary to the approach of Shattuck and Leahy, "cuts" are not forced to be oriented along cardinal axes. However, topological corrections at a specific scale depend on the choice of filter, either foreground or background morphological filter, which fails to evaluate simultaneously the effect of two complementary solutions (i.e. cutting the handle or filling the corresponding hole) on the corrected segmentation.

Kriegeskorte and Goeble use a region growing method prioritized by the distance-to-surface of the voxels in order to force the cuts to be located at the thinnest part of each topological defect [56]. The same process is applied to the inverse object, offering an alternative solution to each cut. An empirical cost is then assigned to each solution and the final decision is the one minimizing the global cost function. It is important to note the limitations of methods that rely on distance ordering maps. For instance, in 2D, the distance from the object boundary can always help find the thinnest part of a non-uniform, ring-shaped object. Unfortunately, this same effect does not hold in 3D if the ring-shaped object is actually a flat 3D object. Yet, in the case of cortical segmentations with "few" flat regions, the distance ordering provides a good solution for the location of topological defects at the thinnest parts of the volume.

B - Surface-based Approaches

Approaches of the other type operate directly on the triangulated surface mesh.

Guskov and Wood [43] locate the topological defects present in the tessellation by simulating wavefront propagation on the tessellation. A selected vertex is used to initialize a region growing algorithm, which detects loops (i.e. topological defects) in the triangulation where wavefronts meet. Topological corrections are obtained through the use of opening operators on the triangle mesh, resulting in a fast method that depends on the initially selected vertex.

In a similar work, Jaume [50] identifies minimal loops in the volume by wavefront propagation. This method assumes that the initial triangulation was generated through the use of a topologically-consistent algorithm (see sect 2.3.3). The minimal loops are then used to identify non-simple voxels in the volume, which are consequently deleted. Again, this approach orients the "cuts" along the Cartesian axes and generates "unnatural" topological corrections.

Fischl et al. [28] proposed an automated procedure to locate topological defects by homeomorphically mapping the initial triangulation onto a sphere. Topological defects are identified as regions in which the homeomorphic mapping is broken and a greedy algorithm is used to retessellate incorrect patches, constraining the topology on the sphere $\mathcal{S}$ while preserving geometric accuracy by a maximum likelihood optimization. In this approach, all possible edges in a defective region are ordered using some measure; each edge is sequentially added to the existing tessellation if and only if it does not intersect any of the existing or previously added edges.

Although this approach can attain reasonable surfaces in many cases, it is worth noting that the information necessary to evaluate the "goodness" of an edge does not exist in isolation, but only as a function of the tessellation of which the edge is a part. This is a critical point, as it implies that a greedy algorithm cannot in general achieve geometrically accurate surfaces, as the necessary information does not exist at the time the edge ordering is constructed. Another considerable subtlety needs to be noted. Every vertex in the original defect, even those present due to segmentation inaccuracies, will be present in the final retessellation, resulting in extremely jagged patches that only a strong smoothing would correct. As a consequence, the final configuration will approximately correspond to an average of all vertex positions in the original configuration. Finally, we note that, even though the final intrinsic topology will be the correct one (the one of a sphere, corresponding to an Euler number $\mathcal{X} = 2$), the proposed method does not guarantee that the final surface will not self-intersect.

C - Summary of the Limitations of Retrospective Topology Correction Algorithms

Most of these methods assume that the topological defects in the segmentation are located at the thinnest parts of the volume and aim at correcting the topology by minimally modifying the volume or tessellation [89, 44, 43, 50]. Although they will often lead to accurate results, due to the accuracy of initial segmentations, topological corrections may not be optimal: additional information, such as the expected local curvature or the local intensity distribution, may lead to different corrections (i.e. hopefully comparable to the ones a trained operator would make).

While these methods can be effective, they cannot be used to correct the topology of arbitrary segmentations, as they make assumptions on the topology of the initial input image. Most frequently, fully-connected volumes are assumed and cavities are supposed to be removed as a preprocessing step. In the case of cortical segmentations, the largest connected component will most often approximate accurately the final cortical surface, since the topological errors present in cortical segmentations are essentially handles. While T1-weighted MRI images[3] have relatively good contrast in the gray/white matter areas, the highly folded nature of the cortical surface associated with the finite resolution of medical images often generates incorrect handles in segmentations produced by methods that do not constrain the topology. Therefore, the topology correction amounts to locating and correcting the handles present in the segmentation. On the other hand, the situation is different for subcortical segmentations. Subcortical segmentations are difficult to locate if solely based on intensity information. The modification of a small number of voxels is usually sufficient to correct their topology, but one should not assume that handles are the only topological defects. Due to the presence of imaging artifact, anatomical variability, varying contrast properties, and poor registration, no assumptions can be made on the topology of the initial segmentation. Especially, the selection of the largest connected component might exclude some large structures that will not be recovered by any (existing) topology correction methods.

We focus on the retrospective topology correction of already-segmented structures in Chapters 5 and 6. We propose two methods for both types of data structures commonly used in medical image segmentation, 3-dimensional volumes or surfaces. Each approach integrate additional (e.g. geometric, statistical) information to accurately locate and correct potential topological defects. A fully automated volume-based method to correct the topology of any binary volume volumetric segmentation under any digital connectivity is introduced in Chapter 5, while Chapter 6 focuses on the cortical reconstruction problem.

[3] T1-weighted MRI imaging is the most common image modality used for cortical segmentation

2.5 Conclusion

In this chapter, we covered the notions of topology that are of central importance to this dissertation.

The essential notions of homeomorphism and homotopy, which are necessary to characterize the topology type of a geometric object, have been presented. We have clearly distinguished the *intrinsic* topology of an object from its homotopy type. Also, we have emphasized the connections linking topology and differential geometry that gave to definition of the crucial notion of Euler-characteristic of a surface. The Euler-characteristic of a surface is a topological invariant that characterizes the *intrinsic* topology of an object.

The adaptation of the continuous concepts of topology onto a discrete framework that is practical to the segmentation of medical images proves to be challenging. However, we have shown that topologically-consistent frameworks can be constructed by replacing the notion of continuity by the weaker notion of connectivity. The theory of digital topology is of central importance. In particular, we have introduced the important notions of simple point and topological numbers, and defined the discrete equivalent of homotopic deformations based on the notion of simple point. Finally, we have presented isocontour extraction techniques and topology-preserving level sets.

The essential notions presented in this chapter were:

- **Intrinsic topology.**
  The intrinsic topology is defined by homeomorphic deformations.

- **Topology and homotopy type.**
  The topology of an object is characterized by homotopic transformations.

- **Euler-characteristic and genus.**
  The Euler-characteristic of a compact connected surface $\mathcal{C}$ is the number $\chi(\mathcal{C})$ defined by:
  $$\chi = v - e + f \ ,$$
  where v, e and f denote respectively the number of vertices, edges and faces of any polyhedral decomposition $\mathcal{D}$ of the surface $\mathcal{C}$. The Euler-characteristic of a surface is a topological invariant, defining the intrinsic topology of a surface. The Euler-characteristic is related to another topologically invariant quantity, the genus g. We have the relation $\chi = 2(K - g)$, where K is the number of connected components of the surface $\mathcal{C}$.

- **Simple point, topological numbers, and homotopic deformation.**
  A simple point is a digital point that can be added or deleted from a binary object without changing the topology type of the object. Simple points are efficiently characterized by two topological numbers that can be computed locally from the 3D neighborhood of the point. Digital homotopic deformations are characterized as sequences of addition or deletion of simple points.

Section II
Topologically-Constrained Segmentation

Multisimple Points
Genus Preserving Level Sets

Chapter 3

Multisimple Points

In this chapter, we introduce the digital concept of multisimple point. While simple points constrain the topology of a digital binary object to remain the same, multisimple points preserve only the number of handles in a digital volume. We introduce two extended topological numbers and derive necessary and sufficient conditions for a point to be multisimple. Using this criterion, digital deformations that preserve the number of handles are designed.

3.1 The Need for a New Digital Concept

In this chapter and the following one, we focus on the limitations of segmentation techniques that directly integrate some topological constraints into the segmentation process. As mentioned in Sect. 1.4 and Sect. 2.4.1.D, the strict topology preservation is too restrictive for most applications. We propose to tackle these problems using the theory of digital topology. This chapter consists of our theoretical contributions in digital topology while the next one describes a new segmentation methodology.

Digital topology provides an elegant theory in which 3-dimensional images are assigned a precise topological meaning. In this framework, homotopic deformations of binary objects consist of iterative deletions or additions of simple points, i.e. points that do not change the digital topology of the initial objects. This set of homotopic deformations defines the classes of topological equivalence over the space of digital images. For instance, any object that can be homotopically deformed into a sphere is said to have the topology of a sphere. Homotopic deformations have been used extensively in image segmentation for the purpose of projecting the topology of a given template onto a structure.

While being able to control the topology of a digital object is an attractive feature, forcing it to remain identical through an homotopic evolution imposes a strong constraint. In particular, homotopic deformations prevent the generation of cavities and disconnected components as well as the formation of handles. On the one hand, preventing the creation of handles that are difficult to correct retrospectively [89, 56, 44, 28, 43] is advantageous. On the other hand, changes of the number of connected components are less problematic, because the different components can be easily identified at post-processing time using standard region growing algorithms [36]. This is also the case for cavities, which can be interpreted as background $\overline{n}$-connected components.

In this chapter, we introduce the concept of *multisimple* point that generalizes the notion of simple point. We introduce two new topological numbers, the *extended topological*

numbers, which, used in conjunction to the topological numbers [7], allow us to distinguish different levels of topological changes. Furthermore, the multisimple point concept defines new sets of transformations that generalize the restrictive homotopic transformations. For clarity, the proofs are reported in Appendix A.

3.2 Limitations of the Concept of Simple Points and of Topological Numbers

3.2.1 Simple Point and Topological Numbers

A simple point can be added or deleted from a binary object without changing its topology type. In [7], Bertrand proposed an elegant way to characterize the topology type of a given voxel by introducing two topological numbers $T_n(\mathbf{x}, X)$ and $T_{\overline{n}}(\mathbf{x}, \overline{X})$ computed from the adapted topological neighborhood $N_n^{\cdot}(\mathbf{x}, X)$ and $N_{\overline{n}}^{\cdot}(\mathbf{x}, \overline{X})$ respectively. Simple points are characterized by the necessary and sufficient condition $T_n(\mathbf{x}, X) = T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1$. Section 2.3.1 describes the main notions of digital topology.

Deletion or addition of non-simple points change the topology of a binary object. Non-simple points have many different topological types. These types correspond to interior or isolated points, and to different kinds of junctions. The values taken by the topological numbers $T_n(\mathbf{x}, X)$ and $T_{\overline{n}}(\mathbf{x}, \overline{X})$ can help characterize different topological types. For instance, an isolated point is characterized unambiguously by $T_n(\mathbf{x}, X) = 0$, and an interior point by $T_{\overline{n}}(\mathbf{x}, \overline{X}) = 0$. Other values of the topological numbers characterize different types of junctions, as listed in the table 2.1 and illustrated in Fig. 2-4 in Sect. 2.3.1.

3.2.2 Limitations

While it could be tempting to use the topological numbers to help characterize the different topological types of non-simple points, this approach is bound to fail. Topological numbers are locally computed from geodesic neighborhoods and do not carry any information on the global connectivity of the neighboring objects. Figure 3-1 illustrates this limitation with a simple example. The two circled voxels are characterized by the same pair of topological numbers, $T_n(\mathbf{x}, X) = 2$ and $T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1$, even though the deletion of one would disconnect the binary object in two components, while the deletion of the other would destroy one handle.

Being able to disconnect and merge components without changing the number of handles is our objective. This would result in more flexible deformations, less sensitive to initialization and unexpected artifacts in the images to be segmented. Yet, the concept of topological numbers is too restrictive and necessitate the introduction of more elaborate topological criteria.

3.3 Towards a First Characterization of Multisimple Points

Our main objective is to provide a method to merge and split connected components, without generating handles. As we noted, the topological numbers of a point $\mathbf{x}$ are locally computed and do not carry any information on the global connectivity of the neighboring connected components of $\mathbf{x}$ (they measure the number of connected components in the sets $N_n^{\cdot}(\mathbf{x}, X)$ and $N_{\overline{n}}^{\cdot}(\mathbf{x}, \overline{X})$). In order to integrate information on the global connectivity, we consider the set $C_n(\mathbf{x}, X)$ of n-connected components of $X \setminus \{\mathbf{x}\}$ that are n-adjacent to $\mathbf{x}$.

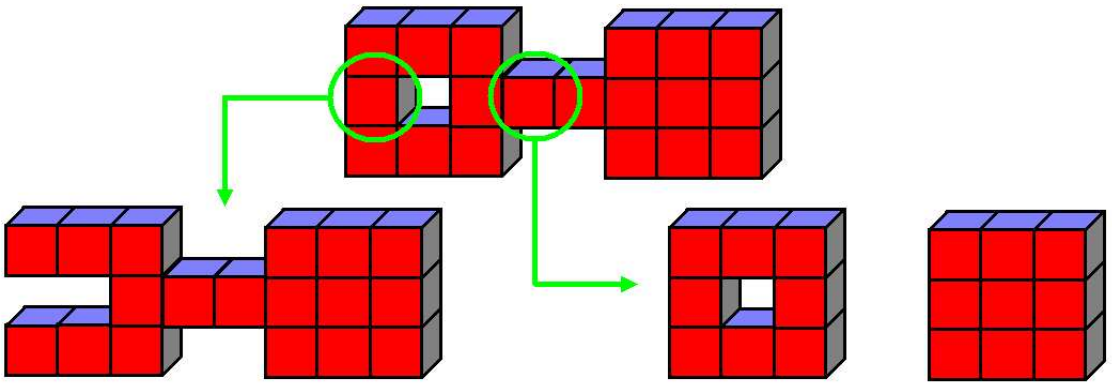

Figure 3-1: The two circled voxels are non-simple. Even though they possess different topology type, they are characterized by the same topological numbers, $T_n(\mathbf{x}, X) = 2$, $T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1$. The removal of the right voxel would remove a handle in the volume, while the removal of the left voxel would disconnect the object into two disconnected components.

3.3.1 Merging and Splitting Connected Components

We say that a point is *multisimple relative to an object* X if and only if it can be added or removed without changing the number of handles and cavities of the object. Contrary to the case of simple points, the addition of a multisimple point may merge several connected components, and its removal may split a component into several parts.

We consider the set $C_n(\mathbf{x}, X)$. If $|C_n(\mathbf{x}, X)| = 1$, the addition or the removal of $\mathbf{x}$ might respectively create or remove a handle and this would be the case *if and only if* the point $\mathbf{x}$ is non-simple. On the other hand, if the cardinality of the set $C_n(\mathbf{x}, X)$ is strictly greater than one, the addition or the removal of $\mathbf{x}$ involves a merge or a split respectively. For each component $C \in C_n(\mathbf{x}, X)$, taken separately, no handle is generated or removed if and only if the point $\mathbf{x}$ is simple relative to the single component C.

Therefore, we define a point to be *multisimple relative to* X if and only if it is simple relative to each component in $C_n(\mathbf{x}, X)$ taken separately, resulting in an elementary criterion using topological numbers.

Definition 3.1 Multisimple point x relative to X
A point $\mathbf{x} \in X$ *is said to be multisimple relative to* X *if and only if*

$$\begin{cases} T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1 \\ \forall\ C \in C_n(\mathbf{x}, X),\ T_n(\mathbf{x}, C) = T_{\overline{n}}(\mathbf{x}, \overline{C}) = 1 \end{cases} \tag{3.1}$$

When merging or splitting connected components by adding or removing a multisimple point, the total genus (i.e. the total number of handles) of the different components is preserved. For example, a torus and a sphere merge into a torus. A double torus split into two tori or into a double torus and a sphere. We note that, under this condition, an isolated point is a multisimple point: components can appear or disappear. The proof is provided in Appendix A.

3.3.2 Generation of Cavities

The concept of multisimple point evaluates if the addition or removal of a point changes the number of handles or cavities of an object. In particular, components may split into several connected components without introducing any handles or cavities. Using the foreground/background duality (Sect.2.2.4), we interpret the split of a foreground component as the generation of one or several background cavities.

By duality, we propose a criterion for the generation of cavities, that prevents the generation of handles and of disconnected components. We introduce the concept of *multisimple point relative to* $\overline{X}$.

Definition 3.2 Multisimple point x relative to $\overline{X}$
A point $\mathbf{x}$ *is said to be multisimple relative to* $\overline{X}$ *if and only if:*

$$\left\{ \begin{array}{rcl} T_n(\mathbf{x}, X) & = & 1 \\ \forall\ C \in C_{\overline{n}}(\mathbf{x}, \overline{X}),\ T_n(\mathbf{x}, \overline{C}) & = & T_{\overline{n}}(\mathbf{x}, C) = 1 \end{array} \right. \tag{3.2}$$

Similarly, we note that, under this condition, an interior point is a multisimple point, which allows cavities to be created or to disappear.

3.3.3 A Sufficient Condition for Multisimple Points

Based on the previous definitions, Def. 3.1 and Def. 3.2, we extend the concept of multisimple point to both foreground and background components.

Definition 3.3 Multisimple simple point x relative to X or $\overline{X}$
A point $\mathbf{x}$ *is said to be multisimple relative to* X *or* $\overline{X}$ *if and only if:*

$$\left\{ \begin{array}{rcl} T_{\overline{n}}(\mathbf{x}, \overline{X}) & = & 1 \\ \forall\ C \in C_n(\mathbf{x}, X),\ T_n(\mathbf{x}, C) & = & T_{\overline{n}}(\mathbf{x}, \overline{C}) = 1 \end{array} \right.$$

or

$$\left\{ \begin{array}{rcl} T_n(\mathbf{x}, X) & = & 1 \\ \forall\ C \in C_{\overline{n}}(\mathbf{x}, \overline{X}),\ T_n(\mathbf{x}, \overline{C}) & = & T_{\overline{n}}(\mathbf{x}, \overline{C}) = 1 \end{array} \right.$$

The addition or removal of a multisimple point $\mathbf{x}$ for X and $\overline{X}$ does not create or remove any handles in the volume, but allow for components to merge, split, appear or disappear.

3.4 Characterization of Multisimple Points

3.4.1 Extended Topological Numbers

Definition 3.3 provides a characterization of multisimple points that is sufficient. Any point $\mathbf{x}$ that verifies the criterion of Def. 3.3 can be removed or added to a digital object without generating or closing any handles in the volume. Yet, this characterization is not a necessary condition. Some voxels can be deleted or added to a digital object without introducing or destroying any handles in the volume, even though they would not verify the characterization of Def. 3.3. For instance, consider an empty cube C and a line segment L, such that the intersection $L \cap C$ is exactly one single voxel $\mathbf{x}$ that is an extremity of L (Fig. 3-2). We focus on the object $X = L \cup C$. We note that $\mathbf{x}$ is not a multisimple point

in the sense of Def. 3.3, since we have $T_n(\mathbf{x}, X) = 2$ and $T_{\overline{n}}(\mathbf{x}, \overline{X}) = 2$ under the topology $n = 6$ or $n = 6+$ (assuming that the sampling of these objects was well done as in Fig. 3-2). Yet, $\mathbf{x}$ can be removed from X and the number of handles preserved.

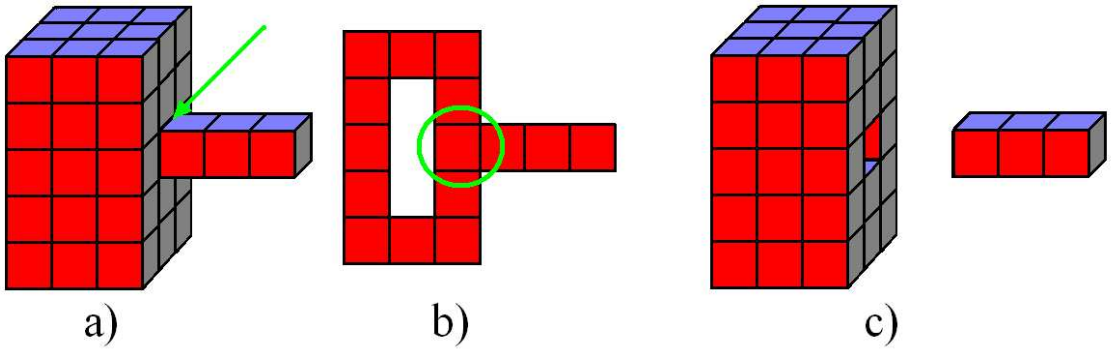

Figure 3-2: a-b) Under the 6 or 6+-connectivity, the circled voxel does not verify the condition of Def. 3.3, since $T_{\overline{n}}(\mathbf{x}, \overline{X}) = 2$. c) However, this voxel can be deleted from the digital object without introducing any handle in the volume. Therefore, definition 3.3 does not provide a necessary condition.

As we mentioned in Sect. 3.3, if the cardinality of the set $C_n(\mathbf{x}, X)$ is strictly greater than one, the addition or removal of a point $\mathbf{x}$ involves a merge or a split respectively. In order to capture these connectivity changes, we define two numbers, $T_n^+(\mathbf{x}, X)$ and $T_{\overline{n}}^+(\mathbf{x}, \overline{X})$, which we call *extended topological numbers*:

Definition 3.4 Extended Topological Number T_n^+
The extended topological number of a point $\mathbf{x}$ *relative to an object* X *is:*

$$T_n^+(\mathbf{x}, X) = |C_n(\mathbf{x}, X)|.$$

The condition $T_n^+(\mathbf{x}, X) > 1$ implies that the deletion of the point $\mathbf{x}$ involves a split in the digital object X. However, this condition does not guarantee that the genus of X (i.e. the number of handles) does not change. Before providing necessary and sufficient conditions of multisimple points, we introduce some useful properties of the extended topological numbers.

3.4.2 Properties of the Extended Topological Numbers

In this section, we derive a few properties of the extended topological numbers T_n^+ and illustrate their relation to the topological numbers T_n.

We first note that the extended topological numbers are always less than or equal to the topological numbers. Extended topological numbers characterize potential merges or splits only, while topological numbers control any kind of topological changes.

Property 3.1

$$T_n^+(\mathbf{x}, X) \leq T_n(\mathbf{x}, X)$$

In addition to splits and merges, the topological numbers characterize the potential formations of handles. For a given connected component $C \in C_n(\mathbf{x}, X)$, the condition

$T_n(\mathbf{x}, C) > 1$ implies that the addition of the point $\mathbf{x}$ to X generates, at least, one handle in the connected component C. Therefore, the following property holds:

Property 3.2

$$\begin{array}{rcl} T_n^+(\mathbf{x}, X) < T_n(\mathbf{x}, X) & \Leftrightarrow & \exists C \in C_n(\mathbf{x}, X) \text{ such that } T_n(\mathbf{x}, C) > 1 \\ & & \text{or} \\ T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X) & \Leftrightarrow & \forall C \in C_n(\mathbf{x}, X)\ T_n(\mathbf{x}, C) = 1 \end{array}$$

Finally, we note that we have the following equivalence:

Property 3.3

$$T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1 \Leftrightarrow \forall C \in C_n(\mathbf{x}, X)\ T_{\overline{n}}(\mathbf{x}, \overline{C}) = 1$$

Properties 3.2 and 3.3 illustrate the link between extended topological numbers and multisimple points as defined in Def. 3.1. Using these properties, the definition of multisimple point can be reformulated as:

Definition 3.5 Multisimple point x relative to X
A point $\mathbf{x} \in X$ is said to be multisimple relative to X if and only if:

$$\left\{ \begin{array}{rcl} T_{\overline{n}}(\mathbf{x}, \overline{X}) & = & 1 \\ T_n^+(\mathbf{x}, X) & = & T_n(\mathbf{x}, X) \end{array} \right. \tag{3.3}$$

3.4.3 Necessary and Sufficient Conditions

To better understand the meaning of the extended topological numbers, we assume that the digital object X is composed of one single *connected* component. We focus on a point $\mathbf{x} \in X$ such that the set $C_n(\mathbf{x}, X)$ is composed of one single component C. This is equivalent to choosing a point $\mathbf{x} \in X$ such that $X = C \bigcup \{\mathbf{x}\}$ and $T_n^+(\mathbf{x}, X) = 1$. We first note that the removal of $\mathbf{x}$ does not disconnect the volume X, but might introduce an $\overline{n}$-handle or remove an n-handle in the volume. The removal of an n-handle occurs if and only if we have $T_n(C) > 1$, which, using Pro. 3.2, is equivalent to the relation $T_n^+(\mathbf{x}, X) < T_n(\mathbf{x}, X)$. Similarly, an $\overline{n}$-handle is generated if and only if we have $T_{\overline{n}}(C) > 1$, i.e $T_{\overline{n}}^+(\mathbf{x}, X) < T_{\overline{n}}(\mathbf{x}, X)$. Therefore, we see that the preservation of the number of handles is guaranteed by the two relations: $T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X)$ and $T_{\overline{n}}^+(\mathbf{x}, X) = T_{\overline{n}}(\mathbf{x}, X)$.

Since all components in $C_n(, X)$ are disconnected components, we see that a point is multisimple if and only if $T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X)$ and $T_{\overline{n}}^+(\mathbf{x}, X) = T_{\overline{n}}(\mathbf{x}, X)$. Consequently, we define a point to be multisimple for $(X, \overline{X})$ by:

Definition 3.6 Multisimple point x for $(X, \overline{X})$
A point $\mathbf{x}$ is a multisimple for $(X, \overline{X})$ if and only if:

$$\left\{ \begin{array}{rcl} T_n^+(\mathbf{x}, X) & = & T_n(\mathbf{x}, X) \\ T_{\overline{n}}^+(\mathbf{x}, X) & = & T_{\overline{n}}(\mathbf{x}, X) \end{array} \right.$$

This condition ensures that no handles are generated in the background or foreground objects. This condition is also necessary, since the generation or the deletion of a handle implies that one of the topological numbers is larger than its corresponding extended topological number.

3.5 Implementation Issues

Given a digital image I, constituted of a binary object X composed of set of n-connected components C_n^i (i.e. $X = \bigcup_i C_n^i$), and of the corresponding object $\overline{X}$ constituted of $\overline{n}$-components $C_{\overline{n}}^i$ (i.e. $\overline{X} = \bigcup_i C_{\overline{n}}^i$), we show how to efficiently implement the concept of multisimple point. We describe a method to adequately modify the different connected components of X and $\overline{X}$ under topological control (i.e. using the concept of multisimple point), so that the topology of each component is known at all time. The only requirement is that the initial topology of the different connected components of X and $\overline{X}$, C_n^i and $C_{\overline{n}}^i$, must be known.

In this process, we create and maintain a map L of labels over the 3D image $I = X \bigcup \overline{X}$ encoding for the different connected components of X and $\overline{X}$. Each n- and $\overline{n}$-connected component is assigned a different label that is saved into the map L. Some care must be taken in order to ensure that the map of labels is correctly updated, so that every label in the map L corresponds to one specific component with a known topology. In this section, we refer to the object X as the foreground object and to $\overline{X}$ as the background object.

3.5.1 Update of the Encoding Map of Connected Components

In order to provide a consistent update scheme for the encoding map L, we consider the deletion and the addition of a multisimple point $\mathbf{x} \in X \bigcup \overline{X}$.

The deletion of a point $\mathbf{x}$ from an object X under the digital topology (n,$\overline{n}$) might involve the split of several n-components or the merge of several $\overline{n}$-components. Information on how to update the label map is extracted from the topological numbers T_n, T_n^+, $T_{\overline{n}}$, and $T_{\overline{n}}^+$. The deletion of a multisimple point generates a merge of $\overline{n}$-components if and only if $T_{\overline{n}}^+ > 1$, and a split of n-components if and only if $T_n^+ > 1$. Algorithm 3 describes how to consistently update the labels of the different connected components. Similarly, the addition of a point $\mathbf{x}$ in X should be interpreted as the deletion of a background point. Algorithm 4 describes the corresponding update scheme.

3.5.2 Computational Complexity

The map L of labels encodes the different connected components of X and $\overline{X}$. Algorithms 3 and 4 describe how to consistently update the labels of the different connected components. When a point is deleted or added to an object X, the simple point condition, more restrictive, is checked first, because it is computationally cheaper. If the point is non-simple, then $C_n(\mathbf{x}, X)$ and $C_{\overline{n}}(\mathbf{x}, \overline{X})$ are computed in order to check the multisimple criterion.

Interestingly, if $\mathbf{x}$ is part of the background object and is a candidate for addition, $C_n(\mathbf{x}, X)$ can be deduced directly from the map L. $C_n(\mathbf{x}, X)$ is indexed by the set of n-adjacent labels of $\mathbf{x}$.

If $\mathbf{x}$ is a candidate for removal, the complete set $C_n(\mathbf{x}, X)$ must be computed. The removal of the point $\mathbf{x}$ could disconnect the object X or remove a handle, therefore changing the global connectivity of X. Information on the resulting connectivity is not present yet in the set of neighbors $N_n^*(\mathbf{x})$, and the set $C_n(\mathbf{x}, X)$ must be computed using standard region growing algorithms [36]. Figure 3-1 illustrates this situation. The left circled voxel has one single neighboring component while the right circled voxel is adjacent to two components. This information can not be deduced from the set $N_n^*(\mathbf{x})$.

However, when dealing with components that do not possess any handles, the most common situation in practice, the computation of $C_n(\mathbf{x}, X)$ involves *local* computations only.

Algorithm 3 Deletion of a point $\mathbf{x} \in X$ under the topology $(n,\overline{n})$

```
Compute T_n(x, X) and T_n̄(x, X̄)
if isolated point {T_n(x, X) = 0} then
   Deletion of a n-component
   Delete the point x
   L(x) ← the only label n̄-adjacent to x
   continue
if interior point {T_n̄(x, X) = 0} then
   Generation of a n̄-cavity
   Delete the point x
   L(x) ← new label
   continue
if the point is simple {T_n = T_n̄ = 1} then
   Delete the point x
   L(x) ← the only label n̄-adjacent to x
else {non-simple}
   Compute the extended topological numbers T_n^+(x, X) and T_n̄^+(x, X̄)
   if the point is multisimple {T_n^+ = T_n and T_n̄^+ = T_n̄} then
      if T_n̄ > 1 then
         Merge of n̄-components
         Merge the n̄-components of C_n̄(x, X̄)
      if T_n > 1 then
         Split of n-components
         Split the n-component that x belongs to
      Delete the point x
      L(x) ← the only label n̄-adjacent to x
   else {non-multisimple}
      Prevent the deletion
```

Since no handles are present in the initial volume, the geodesic neighborhood $N_n^{\cdot}(\mathbf{x}, X)$ contains the necessary information on the connectivity of the adjacent components of $C_n(\mathbf{x}, X)$. The set of adjacent neighboring connected components $C_n(\mathbf{x}, X)$ can be replaced by the set $C_n(\mathbf{x}, N_n^{\cdot}(\mathbf{x}, X))$.

Merging

The merging of connected components into one single component can be done easily using the label map L. If a point $\mathbf{x}$ is part of the background object and is a candidate for addition, $C_n(\mathbf{x}, X)$ can be deduced directly from the map L. The merging step assigns the same label to each neighboring connected components.

Splitting

One must be more careful when splitting components (i.e. split of a foreground component into several components or generation of cavities). This step requires the assignment of some unused labels to the new connected components that must be previously identified. This can be done in linear time using standard region growing algorithms.

Algorithm 4 Addition of a point $\mathbf{x} \in X$ under the topology $(n,\overline{n})$

```
Compute T_n(x, X) and T_n̄(x, X̄)
if isolated point {T_n(x, X) = 0} then
    Creation of a n-component
    Add the point x
    L(x) ← new label
    continue
if interior point {T_n̄(x, X) = 0} then
    Deletion of a n̄-cavity
    Add the point x
    L(x) ← the only label n-adjacent to x
    continue
if the point is simple {T_n = T_n̄ = 1} then
    Delete the point x
    L(x) ← the only label n-adjacent to x
else {non-simple}
    Compute the extended topological numbers T_n^+(x, X) and T_n̄^+(x, X̄)
    if the point is multisimple {T_n^+ = T_n and T_n̄^+ = T_n̄} then
        if T_n̄ > 1 then
            Split of n̄-components
            Split the n̄-components of C_n̄(x, X̄)
        if T_n > 1 then
            Merge of n-components
            Merge the n-component that x belongs to
        Add the point x
        L(x) ← the only label n-adjacent to x
    else {non-multisimple}
        Prevent the deletion
```

3.6 Beyond Homotopic Deformations

A homotopic deformation of a digital object X is a digital transformation that strictly preserves the digital topology of X. Homotopic deformations are realized as sequences of additions and deletions of simple points. The concepts introduced in this chapter allow us to define new sets of transformations, which extend the restrictive notions of homotopic deformations. Depending on the criterion, we define 3 new sets of transformations.

- Based on Def. 3.1, we define a set of transformations that preserve the number of cavities and handles in an object, but allow the different components of the digital object to merge, split, appear or disappear.
- Definition 3.2 ensures that the number of foreground components does not vary, and that handles are not generated, resulting in a set of transformations that allow the number of cavities to vary only.
- The concept of *multisimple point for (X,$\overline{X}$)*, Def. 3.6, prevents handles from disappearing or being created. The resulting deformations allow components to merge and split, appear or disappear, without generating handles during the deformation. We call this set of transformations the set of *genus-preserving deformations*.

We note that the concepts introduced in this chapter can be used to characterize unambiguously the topological type of any voxel. Consequently, more elaborate transformations can be designed that behave exactly as desired by the user.

3.7 Contributions of this Chapter

Digital deformations involving simple points (i.e. homotopic deformations) preserve strictly the topology of a digital object. During the deformation, no handles, cavities, or disconnected components are generated nor deleted. This concept leads to a large class of powerful segmentation algorithms under strict topology preservation. However, the strict preservation is a strong restriction that limits the applicability of these methods. In this chapter, we introduced the concept of multisimple point, which extends and generalizes the concept of simple point. Under this new criterion, deformations that preserve the number of handles but allow the number of components to vary can be designed. This will lead to new segmentation algorithms that are much more flexible than previous methods. This is the subject of the next chapter.

The content of this chapter is a joint work with Gilles Bertrand, from the laboratory A^2SI (Laboratoire Algorithmique et Architecture des Systèmes Informatique) at the ESIEE (Ecole Supérieure d'Ingénieur en Electronique et Electrotechnique). Part of this work has previously appeared in a technical report from the CSAIL [87].

Chapter 4

Genus-Preserving Level Sets

In this chapter, we present a method to exert a topological control over a level set evolution. Level set methods offer several advantages over parametric active contours, in particular automated topological changes. In some applications, where some a priori knowledge of the target topology is available, topological changes may not be desirable. This is typically the case in biomedical image segmentation, where the topology of the target shape is prescribed by anatomical knowledge. However, topologically constrained evolutions often generate topological barriers that create large geometric inconsistencies. We introduce a topologically-controlled level set framework that greatly alleviates this problem. Unlike existing work, our method allows connected components to merge, split or vanish under some specific conditions that ensure no topological defects are generated. We demonstrate the strength of our method in a wide range of numerical experiments and illustrate its performance on the segmentation of cortical surfaces and blood vessels.

4.1 Beyond Digital Deformations

In the previous chapter, we introduced the concept of multisimple point. The introduction of this digital concept was motivated by the limitations of topologically constrained segmentation techniques, particularly their sensitivity to different initializations and to the presence of noise or unexpected structures in the images. Multisimple points extend and generalize the restrictive concept of simple point. Based on this concept, we defined new digital transformations that are more flexible than homotopic deformations.

Digital deformations are rarely used in medical image segmentation - much less frequently than active contour methods. The main reason is the difficulty to integrate curvature or shape information into the segmentation process using solely digital methods. Consequently, digital deformations are more sensitive to noise than active contour methods.

Nevertheless, the concepts introduced in the previous chapter can be used to design a flexible active contour framework. Similar to the approach described in [46], which imported the concept of simple point into the level set framework to design a topologically-preserving level set framework, we integrate the concept of multisimple point into the level set framework. Therefore, we combine the advantages of level set methods - particularly, the integration of curvature information into the segmentation process - with the subtle topological control offered by multisimple points.

4.2 Active Contour Segmentation

Active contours constitute a general technique of matching a deformable model onto an image by means of energy minimization (Sect. 2.4.1). Since their introduction by Kass et al. in [54], deformable models have benefited many computer vision research areas. Particularly, numerous algorithms based on the theory of deformable models have been proposed for the purpose of medical image segmentation [61, 18, 104, 46, 106].

Depending on the implementation, there are essentially two ways of representing an active contour. Parametric approaches encode the manifold of interest with an explicit representation using a Lagrangian formulation, while geometric active contours are represented implicitly as level sets of functions defined on higher-dimensional manifolds in an Eulerian formulation [72]. Geometric active contours, which have been introduced by Caselles et al. [12], offer many advantages over parametric approaches. In addition to their ease of implementation, level sets do not require any parameterization of the evolving contour. Self-intersections, which are costly to prevent in parametric deformable models, are naturally avoided and topological changes are automated. Also, many fundamental properties of the active contours, such as the normal or the curvature, are easily computed from the level set function.

The ability to automatically change topology is often presented as an advantage of the level set method over explicit deformable models. However, this behavior is not desirable in some applications. This is typically the case in biomedical image segmentation, where the topology of the target shape is prescribed by anatomical knowledge. In order to overcome this problem, a topology-preserving variant of the level set method has been proposed [46]. The level set function is iteratively updated with a modified procedure based on the concept of *simple point* from digital topology [7]; the final mesh is obtained with a modified topology-consistent marching cubes algorithm. This method ensures that the resulting mesh has the same topology as the user-defined initial level set. We refer to Sect. 2.4.1 for a more detailed description.

While such topological preservation is desired in some applications, it is often too restrictive. Because the different components of the contour are not allowed to merge or to split up, the number of connected components remains constant throughout the evolution. This number must be known by the user *a priori* and the initial contour must be designed accordingly. Also, the sensitivity to initial conditions, which already limits the applicability and efficiency of active contour methods, is considerably increased. The initial contour must both have the same topology as the target shape and be close enough to the final configuration, otherwise the evolution is likely to be trapped in topological dead-ends including large geometric inconsistencies (Fig. 4-1-b and Fig. 4-2-b).

Although being able to control the topology of an active contour is an attractive feature, forcing it to remain identical through an evolution constitutes a strong constraint. In this chapter, we propose a method to exert a more subtle topological control on a level set evolution. Some *a priori* knowledge of the target topology can be integrated without requiring that the topology be known exactly. Our method greatly alleviates the sensitivity to initial conditions by allowing connected components to merge, split or vanish without introducing any topological defects (such as handles). For example, an initial contour with a spherical topology may split into several pieces, go through one or several mergings, and finally produce a certain number of contours, all of which are topologically equivalent to a sphere. A subset of these components may then be selected by the user as the desired output (typically the largest component if one spherical contour is needed, the others being

caused by noise).

Our approach is based on the concept of multisimple point that we have introduced in the previous chapter (see Chapter 3). This criterion ensures that no topological defects are generated while splitting or merging the components of the object. The resulting algorithm fills the gap between the original level set framework and topology-preserving level set methods. Some experiments presented at the end of this chapter illustrate some potential applications that could greatly benefit from our approach.

4.3 Genus-Preserving Level Sets

The simple point condition is an efficient way to detect topological changes during a level set evolution. However, in many applications, the topology-preserving level set method of Han et al. is too restrictive.

The primary concern is topological defects such as handles, which are difficult to retrospectively correct [89, 56, 44, 28, 43]. On the other hand, changes in the number of connected components during the evolution are less problematic. Different connected components are easily identified using standard region growing algorithms. A subset of them may be selected by the user as the final output, typically the largest one if a single component is needed, the others being imputable to noise in the input data.

The concept of multisimple point that we previously defined allows distinct connected components to merge and split while ensuring that no additional handle is generated in the object. For example, an initial contour with spherical topology may split into several pieces, go through one or several mergings, and finally produce a specific number of surfaces, all of which are topologically equivalent to a sphere.

4.3.1 Multisimple points

The different values of T_n and $T_{\overline{n}}$ characterize the topology type of a given point $\mathbf{x}$, providing important information with regard to its *local* connectivity to the object X. In particular, isolated points are characterized by $T_n = 0$ and $T_{\overline{n}} = 1$, while different junctions by $T_n > 1$ and $T_{\overline{n}} = 1$.

This additional information was exploited in Chapter 3 to carefully design a multi-label digital framework, which allows different connected components to split, merge or vanish under topology control. We defined a point to be *multisimple* if and only if it can be added or removed without changing the number of handles of the object. Contrary to the case of simple points, the addition of a multisimple point may merge several connected components, and its removal may split a component into several parts.

We introduced two extended topological numbers, noted $T_n^+(\mathbf{x}, X)$ and $T_{\overline{n}}^+(\mathbf{x}, \overline{X})$, which used in conjunction to the topological numbers provide a characterization of multisimple point. A point is said to be multisimple if and only if we have:

$$\begin{cases} T_n^+(\mathbf{x}, X) &= T_n(\mathbf{x}, X) \\ T_{\overline{n}}^+(\mathbf{x}, X) &= T_{\overline{n}}(\mathbf{x}, X) \end{cases}$$

When merging or splitting connected components by adding or removing a multisimple point, the total genus (i.e. the total number of handles) of the different components is preserved. For example, a torus and a sphere merge into a torus. A double torus may split

into two tori or into a double torus and a sphere. We note that, under this condition, an isolated and interior points are multisimple points, which allows components to disappear[1].

4.3.2 Genus-Preserving Level Sets

With the concept of multisimple point in hand, we are now ready to describe our new level set framework. Similarly to the approach of Han et al. [46], we exploit the binary nature of the level set function ϕ that partitions the underlying digital image into strictly negative inside points and positive outside points. During the evolution, we maintain a map L of labels coding for the different connected components of X and $\overline{X}$. Each connected component of X is represented by a set of connected negative points, and is assigned a specific label in the label map L. Similarly, connected components of $\overline{X}$ constitute sets of connected positive points, and are assigned distinct labels in L. The label map L is updated concurrently with the level set function ϕ.

The update procedure for each grid point at each iteration is concisely described in Alg. 5. The update scheme for the label map is the same as the one described in Chapter 3. For clarity, we explain the main concepts. During the evolution, the simple point condition, more restrictive, is checked first, because it is computationally cheaper. If the point is non-simple, then $C_n(\mathbf{x}, X)$ and $C_{\overline{n}}(\mathbf{x}, \overline{X})$ are computed in order to check the multisimple criterion.

Algorithm 5 Level Sets Under Topology Control With Handle Preservation

```
Compute the new value of the level set function
if the sign does not change then
  Accept the new value
else {sign change}
  Compute the topological numbers
  if the point is simple then
    Accept the new value
    Update L(x)
  else {non-simple point}
    if the point is multisimple then
      Accept the new value
      Update L(x)
    else
      Discard the new value
      Set instead a small value of the adequate sign
```

If $\mathbf{x}$ is part of the background (resp. foreground) object and is a candidate for addition, $C_n(\mathbf{x}, X)$ (resp. $C_{\overline{n}}(\mathbf{x}, \overline{X})$) can be deduced directly from the map L. If $\mathbf{x}$ is a candidate for removal in X (resp. in $\overline{X}$), the complete set $C_n(\mathbf{x}, X)$ (resp. $C_{\overline{n}}(\mathbf{x}, \overline{X})$) must be computed. However, when dealing with components that do not possess any handles, the most common situation in practice, the computation of $C_n(\mathbf{x}, X)$ and $C_{\overline{n}}(\mathbf{x}, \overline{X})$ only involves local computations.

As noted in Chapter 3, the map of labels L needs to be carefully updated. The more complex case is the removal of a multisimple point involving a split. In this case, some unused labels must be assigned to the new connected components. Algorithm 6 describes in

[1] No components can be spontaneously created in a level set evolution.

detail the update procedure for the label map. Note that components can only be generated through the splitting of an already existing component, as level set evolutions do not allow for the spontaneous generation of new disconnected fronts.

Algorithm 6 Update Scheme for the Evolution of Level Sets Under Topology Control

```
Compute the new value of the level set function
if the sign does not change then
  Accept the new value
else {sign change}
  Compute the topological numbers
  if the point is simple then
    Accept the new value
    if negative new value then
      L(x) ← the only foreground label n-adjacent to x
    else
      L(x) ← the only background label n̄-adjacent to x
  else {non-simple point}
    Compute the extended topological numbers T_n^+(x, X) and T_n̄^+(x, X̄)
    if the point is multisimple {T_n^+ = T_n and T_n̄^+ = T_n̄} then
      if negative new value then
        if T_n̄ > 1 then
          Splitting of n̄-component
          Split the n̄-component of C_n̄(x, X̄)
        if T_n > 1 then
          Merging of n-components
          Merge the n-component that x belongs to
        Add the point x
        L(x) ← the only label n-adjacent to x
      else {positive new value}
        if T_n̄ > 1 then
          Merging of n̄-components
          Merge the n̄-components of C_n̄(x, X̄)
        if T_n > 1 then
          Splitting of n-component
          Split the n-component that x belongs to
        Delete the point x
        L(x) ← the only label n̄-adjacent to x
    else {non-multisimple point}
      Discard the new value
      Set instead a small value of the adequate sign
```

The resulting procedure is an efficient level set method that prevents handles from being created during the evolution, allowing the number of connected components (including cavities) to vary. We insist on the fact that digital topology does not provide a consistent framework for multi-label images. However, by ensuring that no components of the same object X or $\overline{X}$ are adjacent, topological inconsistencies are avoided.

4.3.3 Implementation Issues

We consider the evolution of an active contour Γ under the velocity field $\mathbf{v}$. The level set formulation is the following:

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + \mathbf{v}(\mathbf{x}, t).\nabla \phi(\mathbf{x}, t) = 0,$$

where the isocontour $\phi^{-1}(0)$ represents the evolving contour. As described in Chapter 2 (see Sect. 2.4.1), the implementation of the level set method is computationally expensive. In order to increase the computational speed of geometric deformable models, a narrow band method is usually adopted [1]. Only the grid points that are in a small neighborhood of the active contour are updated during the level set evolution.

During a level set evolution, most points of the narrow-band do not change sign and do not imply a potential change of topology. The simple point condition, which only involves local calculations, is computationally cheap and leads to an efficient algorithm (see [8] and [46]).

Similarly, when the initial level set components do not possess any handles, multisimple points, which constitute a direct extension of the concept of simple points, only require local computations. This situation is the most common in practice, since one is most often interested in segmenting structures that have a spherical topology.

The merging of connected components into one single component can easily be done using the label map L. If a point $\mathbf{x}$ is part of the background object and is a candidate for addition, $C_n(\mathbf{x}, X)$ can be deduced directly from the label map and the geodesic neighborhood $L \bigcap N_n^{\cdot}(\mathbf{x}, X)$. The merging step simply amounts to assigning the same label to each neighboring connected components.

One must be more careful when splitting components (i.e. split of a foreground component into several components or generation of cavities). During the evolution, some components may need to be split into several components, which requires the assignment of some unused labels to the new connected components that must be previously identified. This can be done using standard region growing algorithms.

Also, an update of the narrow band might generate a series of *useless and computationally expensive* splits depending on the ordering of the points in the narrow band. This would be the case, for instance, when a one-voxel wide region is iteratively broken into several pieces. In order to avoid the useless generation of connected components, we prevent components from splitting during the first pass in the narrow band by refusing sign changes of candidate voxels for a split. They are assigned a small value of the appropriate sign and an indicative flag is used to signal a potential component split. After the first pass, the multisimple condition is re-evaluated for each candidate voxels and the components are retrospectively updated. Finally, we note that, in order to avoid useless multiple splits of the same component into several pieces, every candidate voxel that is adjacent to an updated voxel has its flag deleted once a split has happened.

4.3.4 Variations on Topologically-Controlled Level Sets

The proposed framework can be modified to allow more specific topological control during the level set evolution. The multisimple condition introduced in Chapter 3 can be used to distinguish different topological changes. The splitting of a component or the merging of several components correspond to the condition:

$$\left\{ \begin{array}{rcl} T_n^+(\mathbf{x}, X) & = & T_n(\mathbf{x}, X) \\ T_{\overline{n}}^-(\mathbf{x}, X) & = & 1 \end{array} \right. , \tag{4.1}$$

while the generation of a cavity of the merging of several background components are characterized by:

$$\left\{ \begin{array}{rcl} T_n(\mathbf{x}, X) & = & 1 \\ T_{\overline{n}}^+(\mathbf{x}, X) & = & T_{\overline{n}}^-(\mathbf{x}, X) \end{array} \right. . \tag{4.2}$$

We note that the condition $T_n = 1$ implies $T_n^+ = 1$, which proves that the previous criteria characterize multisimple point.

Using these specific criteria, one can design more elaborate level set frameworks that allow some specific topological changes only. For instance, using the criterion 4.1, one can design a level set evolution that allow foreground components to split or merge only. On the other hand, cavities can be controlled using the criterion 4.2.

4.4 Experiments and Applications

In this section, we show the interest of using the genus-preserving level set method for image segmentation. We present some experiments illustrating the performance of our approach and introduce some potential applications. We first apply our level set framework to two synthetic examples to demonstrate its ability to handle disconnected components and cavities. Next, two real data segmentation tasks are presented: the generation of cortical surfaces from MRI images and the extraction of blood vessels from MRA images.

In the following, we have used a simplistic velocity field, which is a combination of an intensity-based term, $(I - I_{thres})$, and a mean curvature term, H:

$$\mathbf{v}(\mathbf{x}, t) = [\alpha(I(\mathbf{x}) - I_{thres}) - H(\mathbf{x}, t)]\mathbf{n}(\mathbf{x}, t),$$

where I denotes the scalar image to be segmented, I_{thres} is a suitable intensity threshold, which separates the object from the background, $\mathbf{n}(\mathbf{x}, t))$ is the outward normal to the isosurface of the actice contour at location $\mathbf{x}$, and α is a weighting parameter. The

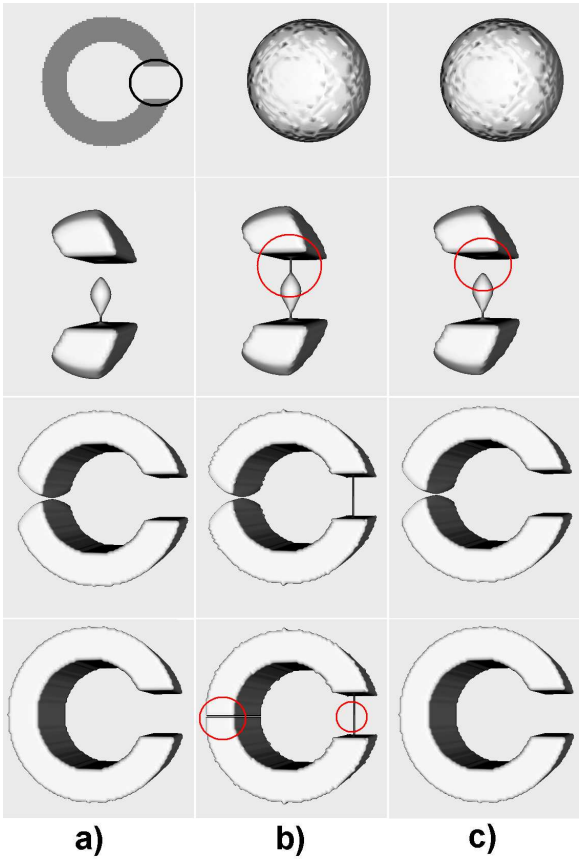

Figure 4-1: Segmentation of a 'C' shape using a spherical initialization. The top row shows cuts of the 3D shape locating the initial contour. a) Traditional level sets. b) Topology-preserving level sets. c) Genus-preserving level sets. Differences of behavior are circled in the images. In this case, our method behaves exactly like the standard level set method. Topology-preserving level sets are trapped in a deadlock.

corresponding level set evolution equation is:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = [-\alpha(I - I_{thres}) + \frac{1}{n-1}\mathrm{div}(\frac{\nabla \phi}{|\nabla \phi|})]\ |\nabla \phi|.$$

More complex images would require more elaborate evolution laws. However, the choice of a particular segmentation method is not the issue here. We rather focus on the improvements brought by our approach, as regards to the management of the topology, relative to the standard level set method and to the topology-preserving method of [46].

4.4.1 Synthetic data

Experiment 1: Segmentation of a 'C' shape

First, we consider the segmentation of a simple 'C' shape under two different initializations (Fig. 4-1 and Fig. 4-2). Our method, columns c, is compared to the original level set formulation, columns a, and the topology-preserving model introduced by Han et al. [46], columns b. The differences of behavior are circled in the images. Two different initializa-

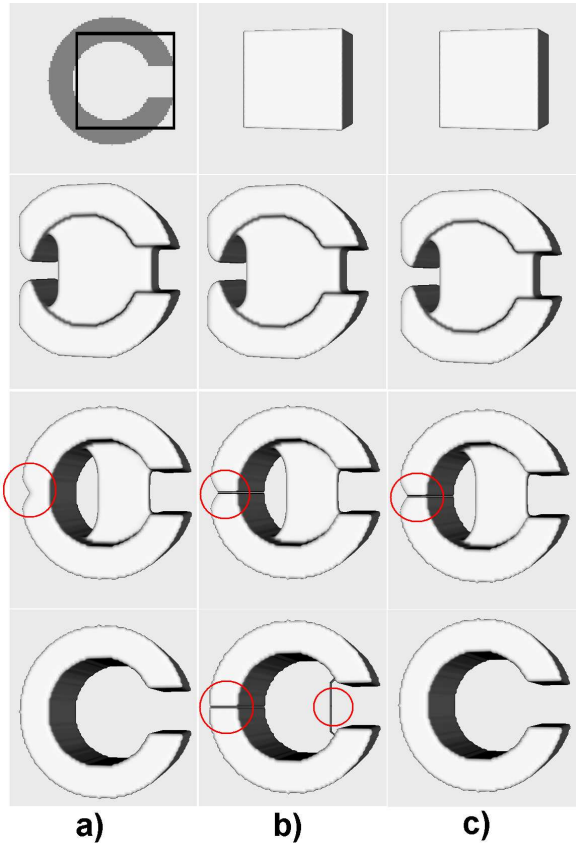

Figure 4-2: Segmentation of a 'C' shape using a rectangular initialization. The top row shows cuts of the 3D shape locating the initial component. a) Traditional level sets. b) Topology-preserving level sets. c) Genus-preserving level sets. Differences of behavior are circled in the images. Our method is able to achieve a correct segmentation without generating a toroïdal topology during the evolution.

tions (a little sphere in Fig. 4-1 and a larger box in Fig. 4-2) were used to test the sensitivity of each method to initial conditions.

In these simple examples, both standard level sets and genus-preserving level sets yield the expected result. With the first initialization (Fig. 4-1-a,c) the two methods behave in exactly the same way, because no handle is generated during the evolution. During the evolution, three distinct components are generated, one of which vanishes, while the two other components merge, closing the 'C' shape. With the second initialization (Fig. 4-2-a,c) they behave differently. Standard level sets temporarily generate a toroïdal topology (row 3, column a), whereas our method prevents the formation of the handle (row 3, column c) by delaying a merging until a split in another part of the object occurs.

In contrast, topology-preserving level sets yield poor results. For the two different initializations, they get trapped in a topological deadlock. Although the final surface has the correct topology, it has large geometric errors (row 4, column b): a filament linking the two ends of the 'C' and a separating membrane at the middle of the 'C'. These topological barriers, generated during the evolution, are difficult to correct retrospectively.

The behavior of our approach corresponds to a trade-off between standard level sets and topology-preserving level sets. Compared to the former, the formation or closing of handles is prevented. Compared to the latter, the ability to change topology under certain condition greatly alleviates the sensitivity to initial conditions.

Experiment 2: Formation of cavities

The second experiment, shown in Fig.4-3, illustrates the ability of our approach to generate cavities during the evolution. The object to be segmented is a synthetic cube, containing

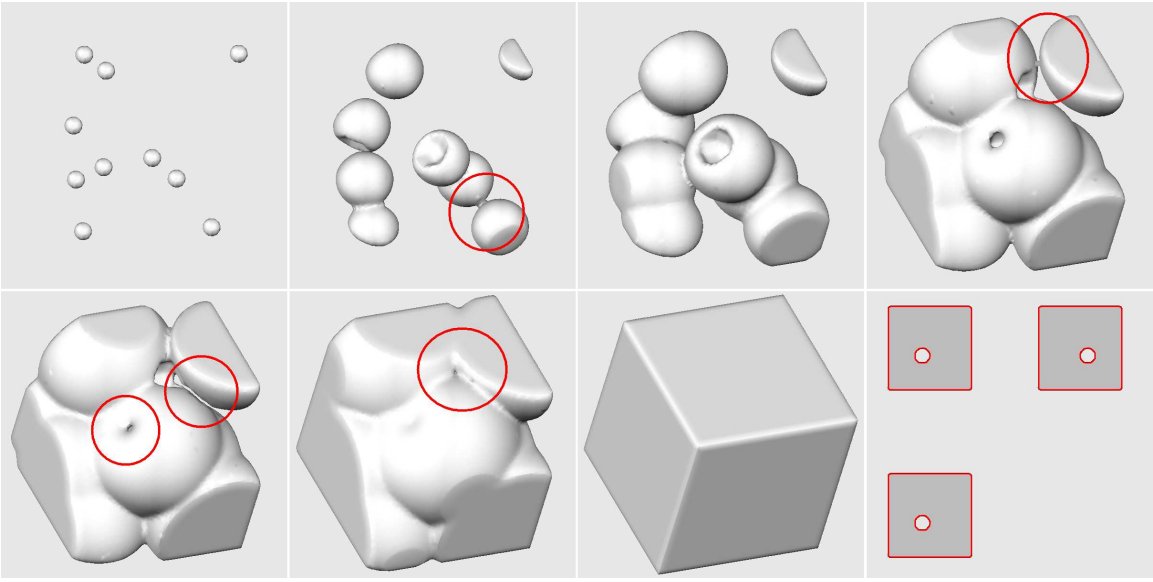

Figure 4-3: Segmentation of a cube containing 3 cavities. 10 initial seed points were randomly selected. Note how components split, merge and disappear during the evolution, and how the active contour encloses cavities.

3 large cavities. 10 seed points, randomly selected inside and outside the volume, were used to initialize the level set evolution reported in Fig.4-3. During the evolution, components merge, vanish and produce cavities, generating a final accurate representation constituted of 3 spherical cavities and the main object. We note that all the components are easily extracted, since they carry distinct labels that are iteratively updated during the evolution.

4.4.2 Real data

Two segmentation tasks are presented that illustrate the potential benefits of our novel level set framework : the segmentation of cortical surfaces from MRI and the extraction of blood vessels from MRA data sets.

Experiment 3: Cortical segmentation

Excluding pathological cases, the cortex, which corresponds to a highly-folded thin sheet of gray matter, has the topology of a sphere. The extraction of accurate and topologically-correct cortical representations is still an active research area. In this example, the cortical surface is initialized with 55 spherical components, automatically selected in a greedy manner, such that every selected point is located at a minimal distance of $10mm$ from the previous ones (Fig.5-3). Topology-preserving level sets could not handle such an initialization, since the number of components would remain constant throughout the evolution. As a consequence, only one initial seed could be used, leading to a slower segmentation process and potentially to topological deadlocks. Standard level sets yield a final cortical surface with 18 handles. On the other hand, using our method, the components progressively merge together and enclose cavities, resulting in a final surface composed of 6 spherical components: the cortical surface and 5 small cavities.

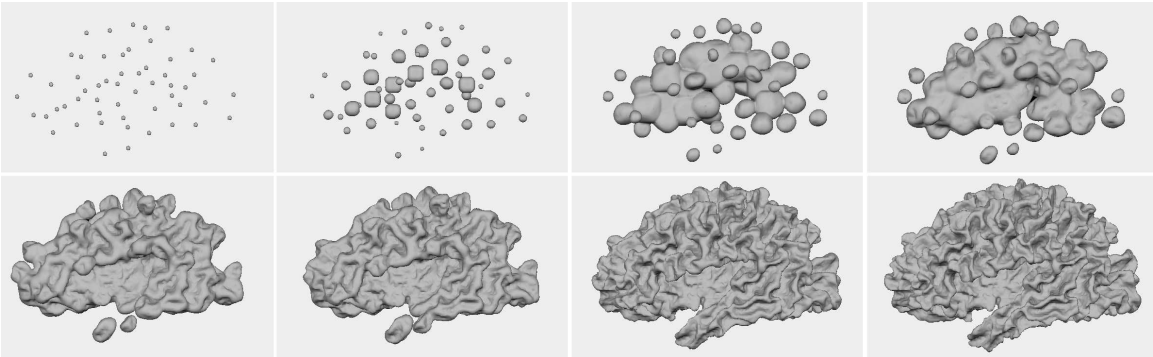

Figure 4-4: Segmentation of the cortex from an anatomical MRI. The initial level set was constituted of 55 connected components. The final surface has a spherical topology, corresponding to an Euler number of 2. The same level set evolution without topological control results in a final surface with 18 topological defects (Euler number of $\chi = -34$)

.

Experiment 4: Segmentation of blood vessels

Finally, we show how our method could be applied to the segmentation of blood vessels from Magnetic Resonance Angiography. Because these vessels do not split and merge, their topology is the one of several distinct components with no handles (i.e. each component has the topology of a sphere). While traditional level sets produce segmentations that could include topological defects, topologically constrained level sets would result in a slow and laborious segmentation. Since the simultaneous evolution of several components, which cannot be merged together, can easily be trapped in topological dead-ends, each component would need to be iteratively initialized, when the evolution of the previous one has terminated. Moreover, when using an initialization with a bounding box, topology-preserving level sets yield a final surface with many geometrical inconsistencies due to topological barriers, displayed in Fig. 4-6.

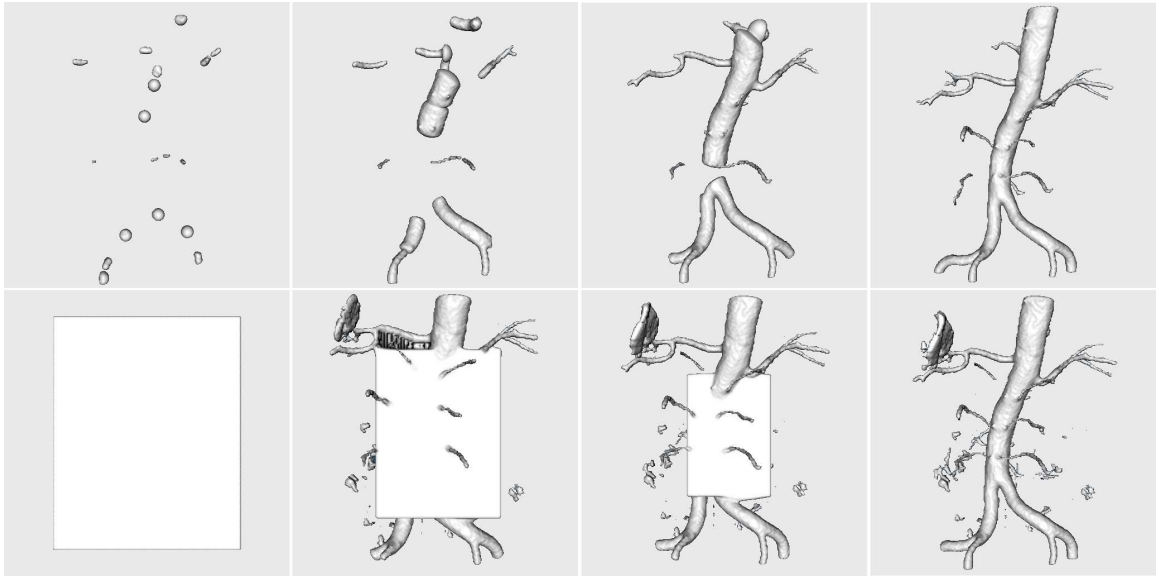

Figure 4-5: Segmentations of blood vessels in a 3D angiography under two different initializations. Top row: 20 seed points were selected to initialize the active contour, which generates 3 components. Bottom row: An enclosing contour is used to initialize the level set. After 9 merges and 99 splits, the final segmentation is constituted of 91 components, 53 of which were due to random noise.

On the other hand, our method offers the possibility to concurrently evolve multiple components that can merge, split and vanish. The initial contour can be initialized by a set of seed points, manually or automatically selected, or by a single enclosing component, without affecting much the final representation.

Figure 4-5 shows the segmentation of an angiography under two different initializations. In a first experiment (top row), 20 seed points were automatically selected at the brightest locations in the MRA. The level set evolution iteratively merged most of the components, generating a final segmentation with 3 spherical components. In the second experiment (bottom row), one single global component, enclosing most of the object of interest, was used to initialize the active contour. During the evolution, 9 components merged and 99

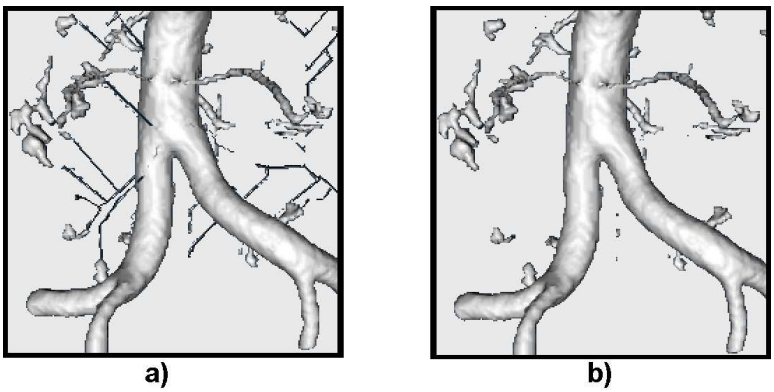

Figure 4-6: Segmentations of blood vessels from MRA produced by a) a topologically constrained evolution [46], starting from a bounding box b) our genus-preserving level set framework. The image contains several artifacts, which are mostly due to noise. As a consequence, several disconnected components are present in the final segmentation produced by our method. Topologically constrained segmentation fails to segment out these disconnected components and produce an incorrect segmentation.

split producing a final segmentation composed of 91 components, 53 of which were single voxel components due to random noise in the imaging process.

4.5 Contributions of this Chapter

In this chapter, we introduced a new level set framework that offers control over the topology of the level set components during the evolution. Contrary to previous approaches that either do not constrain the topology or enforce a hard topological constraint, our method exerts a subtle control over the topology of each component to prevent the formation of topological defects, such as handles (or cavities depending on the application and the choice of active contour model). Distinct components can merge, split or disappear during the evolution, but no handles (or cavities) are generated. In particular, a contour composed solely of spherical components will only produce spherical components throughout the evolution. In this case, the most common situation in practice, all computations are local and the multisimple point checking can be done efficiently. The only computational complexity comes from the generation of new components, as new labels need to be assigned to each.

While the original level set model does not provide any topological control, topology-preserving level sets impose too restrictive of a constraint. Our framework establishes a trade-off in between the two models. Compared to the former, the formation of new handles and the closing of existing handles are prevented. Compared to the latter, the ability to change topology under certain conditions greatly alleviates the sensitivity to initial conditions. Our framework offers a subtle topological control that alleviates most problems of topologically-constrained methods (i.e. sensitivity to initialization and noise, simultaneous evolution of multiple components and speed of convergence). The experiments presented in this chapter illustrate some applications that could potentially benefit from our approach.

Finally, we also note that the proposed framework can be adapted to allow different levels of topological control during the level set evolution. Particularly, the two criteria 4.1 and 4.2 can be used to distinguish different types of voxels, such as the ones leading to a split or a merge of components from the ones generating or destroying cavities, among others.

The content of this chapter is a joint work with Jean-Philippe Pons, from the Odyssée Project at the I.N.R.I.A. (Institut National de Recherche en Informatique et en Automatique) in Sophia-Antipolis, France. Part of this work has previously appeared in technical report from the CSAIL [87], a special issue of *NeuroImage* [26], and was presented at a workshop of the *International Conference on Computer Vision* [88].

Section III
Retrospective Topology Correction

Topology Correction of 3D Digital Images
Topology Correction of Surfaces

Chapter 5

A general digital framework for the topology correction of binary images

We propose a method for automatically correcting the spherical topology of any binary segmentation under any digital connectivity. A multiple region growing process, concurrently acting on the foreground and the background, divides the segmentation into connected components and successive maximum a posteriori decisions guarantee convergence to the correct spherical topology. In contrast to existing procedures that assume specific initial segmentation (e.g. full connectivity, no cavities, etc) and are designed for a particular task (e.g. cortical representation), no assumption is made of the initial image. Our method, applied to subcortical segmentations allows us to accurately correct the topology of fourteen deep nuclei in a few minutes; the topology correction of each separate hemisphere white matter is achieved in approximately 30 minutes.

5.1 Introduction

In this chapter, we focus on the retrospective correction of the topology of 3-dimensional digital segmentations. Many segmentation algorithms produce 3-dimensional segmentations without constraining the topology. Only a few approaches, reviewed in Chapter 2, have been proposed to correct the topology of binary segmentations.

While these methods can be effective, they cannot be used to correct the topology of arbitrary segmentations. They make assumptions on the topology of the initial input image and assume fully-connected volumes without cavities. However, for most segmentations, because of the presence of imaging artifacts, anatomical variability, varying contrast properties, and poor registration, no assumption can be made on the initial segmentation.

Also, most of these methods don't use any geometric or statistical information; they aim at correcting the topology by minimally modifying the volume or tessellation, i.e. with the smallest amount of voxel changes. Although such an approach will often lead to accurate results, due to the accuracy of initial segmentations, topological corrections might not be optimal.

In this chapter, we introduce a fully automated volume-based method to correct the topology of any binary volumetric segmentation under any digital connectivity. The novelty of our approach comes from the fact that any initial segmentation, containing disconnected regions, handles, and cavities, will be corrected. A multiple region growing process allows

us to simultaneously work on different parts of the volume and to incorporate statistical information. At each step of our iterative topological correction, minimum cost decisions are taken and convergence is guaranteed. In the following sections, we discuss the main assumptions of the algorithm and present some results on subcortical and cortical data. An application of the proposed method for multiple subcortical segmentations concludes this chapter.

5.2 Methods

Our method is phrased within the theory of digital topology which establishes an elegant and efficient topological framework over the set of digital images. We correct the topology of any binary segmented volume S_o under a set of compatible digital connectivities $(n,\overline{n})$. Compatible connectivities $(n,\overline{n})$, which are necessary to avoid topological paradoxes, represent the connectivities used for the foreground object F and the background object $B = \overline{F}$ respectively. We refer to Sect. 2.3.1 for more details on digital topology.

5.2.1 Notations

In the next sections, we will need the following definitions and notations:
Connected Components: we denote by $C_n(X)$ the set of n-connected components in the digital object X. The set of n-adjacent n-connected components to a point $\mathbf{x}$ in a digital object X is still denoted $C_n(\mathbf{x}, X)$.
Residual and body labels: during the algorithm, different connected components are generated, and voxels are assigned different labels. Body labels characterize voxels belonging to a body component with a known topology, and residual labels characterize voxels belonging to a component with an unknown topology.
Seed point: a residual point of X that is simple or isolated *relative to the body label* points of X. Under this definition, changing the residual label of a seed point to body will not introduce any topological defects into the body component segmentation of X.
Multisimple point: a residual point $\mathbf{x} \in X$ that can be added to any of its adjacent body components ($\in C_n(\mathbf{x}, X)$) without introducing any topological defects. This concept was introduced in Chapter 3. The merging of a multisimple point into one of the adjacent body components, associated with the merging of the other adjacent body components into the first one, will not change the topology of the new component. We will make important use of this concept, as it allows us to work concurrently on different parts of the object, but still be able to control the whole topology of the resulting segmentation.

5.2.2 Overview of the Approach: Bayesian Interpretation

Our topology correction proceeds in two consecutive steps. First, the topological defects are accurately located using a multiple region growing process integrating statistical and geometrical information. Then, each defect is iteratively corrected by maximizing a fitness function that assesses the goodness of each topological correction.

Our topology correction can be phrased within the framework of Bayesian parameter estimation theory: we search for the topologically-correct *maximum a posteriori* (MAP) estimate of the segmentation S given an observed (potentially multi-spectral) image I.

According to this approach, one can relate $p(S|I)$, the probability of a segmentation S given the observed image I, to $p(I|S)$, the probability of the image occurring given a certain

segmentation, together with $p(S)$, the prior probability of the segmentation:

$$p(S|I) \propto p(I|S)p(S). \tag{5.1}$$

In our Bayesian interpretation, the fitness function, assessing the goodness of each topological correction, is the posterior probability $P(S|I)$. Different models for $P(S|I)$ are possible. In the following, we introduce one model that captures the volumetric information present in digital segmentation. More elaborate models (e.g. models integrating curvature information) are possible, and will be presented in Sect. 5.6.

$S(\mathbf{x})$ denotes the label of the voxel $\mathbf{x}$, i.e. foreground or background. Assuming that the noise at each voxel is independent from all other voxels in the image, we can rewrite $p(I|S)$ as the product of the distribution at each voxel v over the voxel grid V:

$$p(I|S) = \prod_{\mathbf{x}\in V} p(I(\mathbf{x})|S(\mathbf{x})). \tag{5.2}$$

Finally, we assume that the labels $S(\mathbf{x})$, *for all* $\mathbf{x} \in V$, are independently distributed, which allow us to rewrite $p(S)$ as the product of the distribution at each voxel over V:

$$p(S) = \prod_{\mathbf{x}\in V} p(S(\mathbf{x})). \tag{5.3}$$

The last hypothesis might appear overly simplistic, as voxels are certainly not topologically independent. However, we note that the topological independence of neighboring voxels is intrinsically related to the image resolution. One voxel that would generate a handle in a binary segmentation at a specific resolution would not produce a topological defect at a higher resolution. Therefore, the latter assumption on the independence of each voxel should simply be interpreted as a way to integrate prior information into the segmentation process. Some voxels, due to their location relative to the overall image, are more likely to be labeled as foreground than others.

Using Eq. 5.2 and 5.3, the probability distribution of the segmentation given the observed image can be rewritten

$$p(S|I) \propto \prod_{\mathbf{x}\in V} p(I(\mathbf{x})|S(\mathbf{x}))p(S(\mathbf{x})). \tag{5.4}$$

We assume that we can compute the probability $p(I(\mathbf{x})|S(\mathbf{x}))p(S(\mathbf{x})) = p(I(\mathbf{x}), S(\mathbf{x}))$ for each voxel $\mathbf{x}$: we use $p_s(\mathbf{x}, S(\mathbf{x}))$, $p_s(\mathbf{x})$ or p_s to denote this joint probability. Consequently, we assign to each voxel a cost: $c(\mathbf{x}) = ln(p_s(\mathbf{x}, S(\mathbf{x}))/p_x(\mathbf{x}, \overline{S}(\mathbf{x}))$, which represents the variation in the probability of the MAP estimate, when changing the voxel label from $S(\mathbf{x})$ to $\overline{S}(\mathbf{x})$ (Fig. 5-3-b).[1]

Directly computing the MAP estimate of S in Eq. 5.1 under the spherical topological constraint is computationally intractable. Instead, we employ an iterative method on the initial segmentation S_o, which makes minimal decisions at each step. A block diagram of the algorithm is shown in Fig. 5-1. The algorithm proceeds as follow: first, a multiple homotopic region growing process segments each object (foreground and background) into

[1]Most segmentation procedures incorporate statistical information to accurately locate specific structures and the probabilities p_s can be computed *a priori*. However, this external information might not be available, and a different cost must be assigned to each voxel: we will discuss alternative cost options in Sect. 5.3.2.

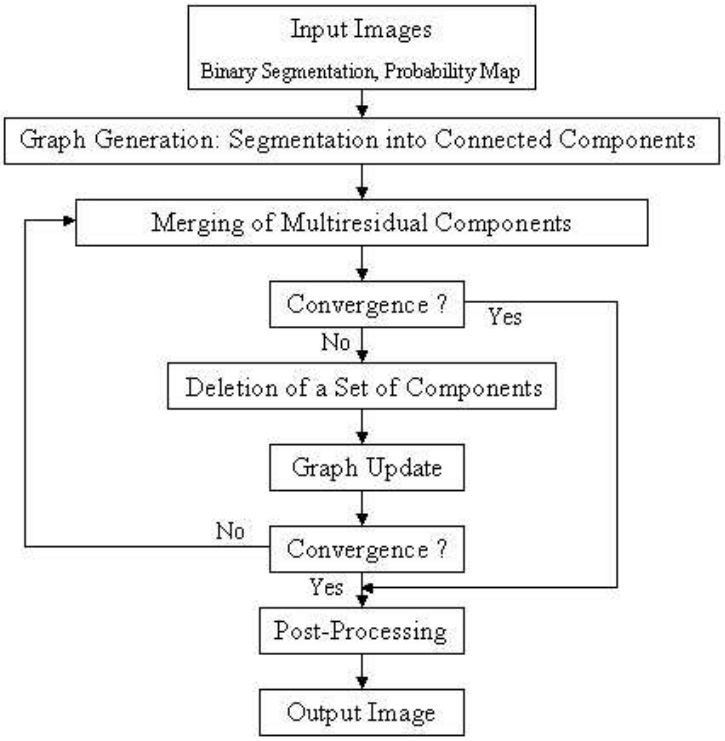

Figure 5-1: Diagram of the digital topology correction algorithm

a set of connected components. This set is composed of body components and of residual components, the latter corresponding to "links" connecting body components (Sect. 5.2.3). Schematically, the topology correction amounts to finding the components that should be removed (i.e. should be added to the inverse object), and the ones that should be kept. The whole process is iterative: a cost is assigned to each component and the algorithm modifies the segmentation into connected components (Sect. 5.2.4). At each step, the set of connected components is updated, and convergence is guaranteed by monotone increase of a threshold T_{thres} (Sect. 5.2.4). Figure 5-2 illustrates the simple concepts behind the algorithm and we will often refer to this figure to exemplify the explanations. For clarity, discussion of the main assumptions of the algorithm will be elucidated in the next section.

5.2.3 Location of Topological Defects: Connectivity Graph Generation

In order to correct the topology of a given binary image, one must first locate the set of topological defects present in the segmentation. Existing techniques for correcting the topology of binary segmentations assume that the topological defects are located at the thinnest parts of the volume and attempt to correct them by modifying a minimal number of voxels. Due to the accuracy of the initial segmentation, this assumption often leads to adequate corrections. However, additional information, such as statistical (i.e. local intensity distributions) and/or geometrical information (i.e. curvature), might improve the location of the topological defects and their correction.

Additional information is directly integrated in a multiple homotopic region growing process. Similarly to existing methods, we assume that topological defects are more likely to

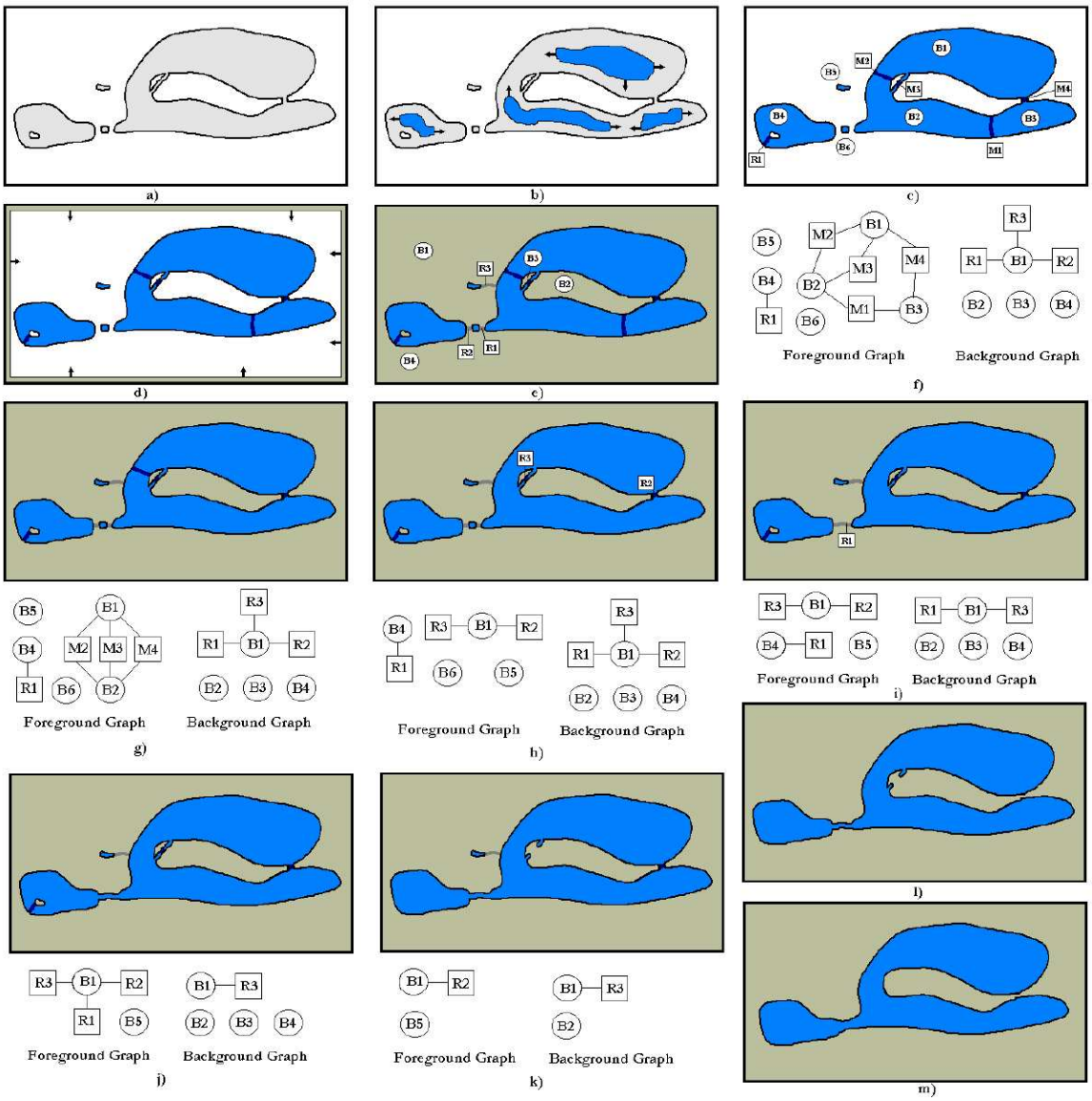

Figure 5-2: Illustration of the algorithm principles. An initial binary digital image (a) is segmented into a set of connected components (e) by applying a multiple region growing process on the foreground object (b,c) and the background object (d,e). These two sets of connected components (foreground and background) constitute two connectivity graphs (f). The topology correction iteratively modifies the graphs until final convergence (g-l). A final homotopic deformation leads to the final binary segmentation (m). Detailed steps: g) Addition of the foreground component f-M1 associated with the merging of f-B2 into f-B1. h) addition of f-M2. Note that f-M3 and f-M4 becomes residual component f-R3 and f-R2 respectively. i) Deletion of f-B6. b-R1 and b-R2 are merged together into b-R1. j) Deletion of b-R1, which leads to the merging of f-B4 into f-B4. k) Deletion of f-R3 and merging of b-B3 into b-B2. Deletion of b-B4 which causes f-R1 to vanish. l) Deletion of b-R3 of f-R2, and final convergence.

be located at the thinnest parts of the volume; however, we observe that topology corrections often involve voxels with low probability. Therefore, the prioritization driving the expansion process is based on the following *empirical* function f_p:

$$f_p(\mathbf{x}, S) = \lambda p_s(\mathbf{x}, S) + (1 - \lambda) f_d(\mathbf{x}, S), \tag{5.5}$$

which combines information from the posterior probability p_s and from the distance-to-surface map. The distance-to-surface map of the object X represents the distance $d(\mathbf{x}, X)$ from one point $\mathbf{x} \in X$ to its surface, i.e. its distance to the closest point(s) of $\overline{X}$. The function f_d is simply linearly built as $f_d(\mathbf{x}) = \frac{d(\mathbf{x},X)}{d_{max}}$ with $d_{max} = \max_{\mathbf{x} \in X} d(\mathbf{x}, X)$. Away from the boundary of the object, the homotopic expansion process is mostly driven by the distance-to-surface map, whereas statistical information becomes prominent in regions close to the borders. This function will locate residual components at thin parts of the volume, which have low probability (Fig. 5-3-c). We will discuss this prioritization in detail in Sect. 5.3.1. For each voxel $\mathbf{x}$, the value $f_p(\mathbf{x}, S(\mathbf{x}))$ can be interpreted as a confidence value, which reflects our confidence in the segmentation $S(\mathbf{x})$ of the voxel $\mathbf{x}$.

Our multiple region growing process segments each object (foreground and background) into a set of connected components. This set is composed of body[2] components, which carry a known spherical topology, and of residual components of unknown topology. Residual components correspond to "links" connecting body components and can be interpreted as topological defects. In order to generate this segmentation, every point is initially assigned a residual label and body components are slowly expanded outward to incorporate new simple points.

The multiple region growing process makes use of two threshold, T_s and T_e with $T_s > T_e$. T_s is used to locate the seed points and generate body components. A new body component will be created from a seed point $\mathbf{x}$ only if its confidence value $f_p(\mathbf{x})$ is above T_s, avoiding over-segmentation. T_e pauses the homotopic expansion of a body component when the values f_p of the adjacent simple points fall below T_e.

As we will discuss further in Sect. 5.3.1, homotopic deformations that are driven by probability maps are highly sensitive to noise: inaccurate probability maps can introduce some incorrect topological constraints in the homotopic expansion, eventually leading to large geometric errors. The introduction of T_s and T_e, associated with the empirical probability f_p, minimizes the "noise" effects on the segmentation and prevents over-segmentation into connected components. It also reduces the influence of topological errors caused by incorrect segmentations. Incorrect segmentations are often the results of a few misclassified voxels that appear brighter or darker than expected relative to their tissue classes. The use of two thresholds tends to single out these misclassified regions, which often generates small disconnected body components.

A - Foreground Object

We remind the reader that a seed point will not introduce any topological defect into any of the body components, allowing us to start growing a new body component without introducing any topological artifact such as handles or holes.

The region growing process proceeds as follow. Every voxel of the foreground object is first assigned a residual label, except the one with the largest confidence value $f_p(\mathbf{x}, F)$ that

[2]For consistency with previous work, we use the same notations as the ones proposed by Han et al. in [44]

is assigned a body label. This seed point creates the first body component, which is then iteratively expanded by adding adjacent simple points y, prioritized by their confidence value: adjacent voxels are checked in decreasing order (the ones with the largest values first) and added if they are simple. This homotopic expansion pauses when the adjacent voxel confidence values fall below T_e or stops when no residual voxel can be added to this body component without changing its topology. Then, the algorithm tries to grow another component by searching for the next seed point with the highest confidence value above T_s. We keep generating and growing new body components until no new seed point, with $f_p > T_s$, is found. Then, the constraint $f_p > T_e$ is relaxed and the multiple region growing process resumes. All the previously created components are homotopically expanded until no new simple point is found. Figure 5-2-b shows an example of the resuming of the multiple region process.

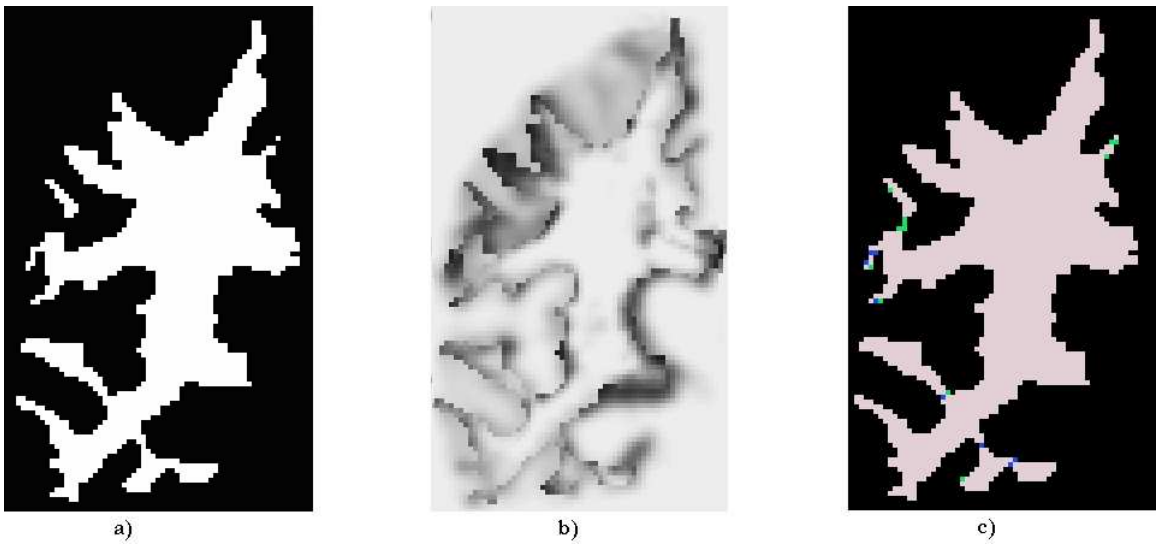

Figure 5-3: a) An horizontal view of a white matter binary segmentation. b) The corresponding cost map $c(\mathbf{x}) = \log \frac{p_s(\mathbf{x},S(\mathbf{x}))}{p_x(\mathbf{x},\overline{S}(\mathbf{x}))}$. Bright voxels have large costs. c) The segmentation into connected components : the single black component is the first body background component and has the topology of a hollow sphere. The pink components are foreground body components and the green and blue components are background and foreground residual components respectively. Notice that residual components tend to be located at low probability locations, i.e. low cost regions (dark regions in b), and at the thinnest parts of the volume.

Eventually, the second constraint $f_p > T_s$, is relaxed and a final region growing process segments the remaining residual voxels into body components (in Fig. 5-2-c, this leads to the creation of components $B5$ and $B6$).

Finally, the remaining residual voxels are segmented into residual connected components (Sect. 5.2.3).

B - Background Object

The same multiple region growing process is applied to the inverse object. However, since we are working on the background object, which is supposed to surround completely the foreground object, the first body component contains the set of voxels located at the border of the image (Fig 5-2-b). Therefore, the topology of the first background component will

be the one of a hollow sphere. Then the algorithm proceeds as previously described (Fig. 5-2-d,e).

C - Residual Segmentation

The last step of the segmentation into connected components is the segmentation of the remaining residual voxels, into residual connected components. A residual voxel can be adjacent to one single body component or to several body components. Residual voxels that are adjacent to one single component are non-simple: if they were simple they would have been merged into the adjacent body component during the previous multiple region growing process. Components constituted of these voxels are called residual components, and components with residual voxels adjacent to several body components are called multi-residual components. Residual components represent "pure" topological defects, as they cannot be merged into the body label segmentation without introducing topological defects. On the other hand, multi-residual components can either be constituted of multisimple points or non-multisimple points. Their introduction allows us to consider more possibilities when correcting the topology. Finally, we note that the topology of the residual components is not known.

The whole segmentation of the foreground object and the background object constitutes two non-fully connected graphs, which interact with each other (Fig. 5-2-f). Finally, a cost is assigned to each component: the cost of each component C is simply defined as sum of the cost of each voxel constituting the component:

$$cost(C) = \sum_{\mathbf{x} \in C} c(\mathbf{x}) = \sum_{\mathbf{x} \in C} \log \frac{p_s(\mathbf{x}, S(\mathbf{x}))}{p_s(\mathbf{x}, \overline{S}(\mathbf{x}))}. \tag{5.6}$$

The cost of one component reflects the variation in $\log[p(S|I)]$ (under our set of assumptions), the logarithm of the probability of the segmentation S given the observed image I, if this component were deleted from X by adding its voxels to the inverse object $\overline{X}$. More complex models, incorporating curvature information, could be considered. These more elaborate models would result in more complex fitness function. We will briefly discuss in Sect. 5.6 the potential integration of curvature information into our framework.

5.2.4 Correction of Topology Defects: Graph Analysis

Once all topological defects have been located (as a set of residual components), we exploit the topological foreground/background duality to iteratively correct them. Under the digital topology framework, every foreground topological defect has a corresponding background topological defect and vise-versa. For example, to a foreground disconnected component corresponds a background cavity and reciprocally. This implies that the correction of a topological defect can be realized by applying a transformation either to the foreground object or the background object (this is one of the main reasons why a pair of compatible connectivities is required in digital topology). This approach has already been used by several authors [44, 56], but in a less general framework than the one we develop here. The goal is to successively decrease the number of residual and body components, until one single body component per object remains: a foreground component with a spherical topology and a background component with the topology of a hollow sphere.

The algorithm works on the background and the foreground simultaneously and proceeds iteratively: multi-residual components are considered first and merged into neighboring

components if their constituting points are multisimple. Then, the lowest cost component $C \in C_n(X)$ is identified and deleted from the object X it belongs to. At each iteration, the algorithm updates the segmentation into connected components (i.e. the connectivity graphs) by resuming the region growing process. We now explain, in detail, the few steps of the topology correction.

A - Addition of Multi-Residual Components:

We have introduced the concept of multi-residual components to enlarge the panel of potential solutions and ensure that residual components are located at meaningful locations (Fig. 5-3-c). Multisimple points allow us to locally modify the component segmentation: body connected components might be fused together, but the topology is preserved. Given a connectivity graph, multi-residual components constitute connections between body components.

A multi-residual component is said to be multisimple if at least one of its constituting voxels are multisimple. At each iteration, multisimple components are merged into body components. The multi-residual component with the largest cost is first chosen and added to the body component segmentation if it is multisimple: the component and its neighboring body components are merged into one single body component and the graph is updated.

Algorithm 7 Addition of a Multi-Residual Component $M \in C_n(X)$

Search among multisimple points in M, the search is prioritized by the confidence values given by f_p.
if no multisimple point **then**
  Stop.
else {multisimple point}
  Merge this point into the largest adjacent body component B, and merge the other adjacent body components into this component.
  Try to add the remaining points of M to the newly formed body component B, the points that cannot be merged into B form new residual connected components.
  Update the cost of the modified components and the status of multi-residual components.

Then, the next largest cost multi-residual component is considered for merging. For each merging, the status of the remaining multi-residual components is checked. The algorithm stops when all components have been added or the remaining ones are not multisimple. Figure 5-2 illustrates, with a simple example, this concept: the multi-residual components $M2$, $M3$ and $M4$ of Fig. 5-2-g corresponds to three potential connections between $B1$ and $B2$. The merging of $M1$ into $B1$, associated with the merging of $B2$ into $B1$ change the multi-residual status of the components $M3$ and $M4$ to residual $R3$ and $R2$ respectively in Fig. 5-2-h.

B - Deletion of the Lowest Cost Component

After the analysis of multi-residual components, we check if the algorithm has converged: if one single component per object remains, the algorithm has corrected the topology of the initial segmentation. If not, spherical topology is not achieved yet and consecutive decisions have to be taken. Assuming a decomposition into connected components, the

algorithm identifies the lowest cost component among body, residual and non-multisimple multi-residual components. This component is deleted from $C_n(X)$, meaning that each voxel of this component $C \in C_n(X)$ has its label changed to residual of the inverse object $\overline{X}$, and the connectivity graphs are updated:

Algorithm 8 Deletion of the Lowest Cost Component $C \in C_n(X)$

1. Deletion of C: $C \rightarrow \overline{C}$. Voxel labels are changed voxel to residual of $\overline{X}$.
2. Search among seed points in $\overline{C}$.

if no seed point **then**
  Go to 6.
else {seed point **x**}
  if **x** is isolated **then**
    3. Generate a new body component
  else {**x** is simple}
    4. Merge this point into its neighboring body component
  5. Update the cost of the modified components. Go to 2.
6. Segment the remaining residual voxels into residual and multi-residual components.

We note fact that local decisions do not imply large geometrical errors. For instance, in Fig. 5-2-h,i, the deletion of the foreground body component $B6$ does not lead to the removal of any large component.

C - Convergence

Stated as previously, the algorithm is not guaranteed to converge: the deletion of a component might lead to the creation of even lower cost components in the inverse object. Therefore, we use a threshold T_{thres} that is monotonically increased at each iteration.

Algorithm 9 Convergence

0. Set $T_{thres} = -\infty$.
1. Update the multi-residual graphs by iteratively applying Alg. 7 to each multi-residual components.

if Check convergence: X or $\overline{X}$ has one single component **then**
  Stop.

3. Find the set of lowest cost components $\{C_i\} \in C_n(X)$ such that: $\exists C_j \in \{C_i\}$ *s.t.* $cost(C_j) > T_{thres}$; set $T_{thres} \leftarrow cost(C_j)$.
4. Delete the components: $\forall \mathbf{x} \in \cup C_i$, set $\mathbf{x} \rightarrow \overline{X}$

if X has one single component **then**
  Stop.

5. Apply Alg. 8 to the inverse object $\overline{X}$ and go to 1.

Similarly to the approach of Han et al. in [44], this algorithm can be modified to force corrections to be made on one single object exclusively (foreground or background): it suffices to constrain the search for lowest cost components to the inverse object.

5.2.5 Post-processing

Finally, once the correct topology has been achieved, the algorithm homotopically maximizes the posterior probability $p(S|I)$, by looking for simple points that should be deleted

(with a negative cost). Every simple point $\mathbf{x}$ of the segmentation S with a negative cost is added to its inverse object $\overline{S}(\mathbf{x})$.

Finally, a topologically-consistent tessellation of the topologically-correct digital volume is generated using a connectivity-consistent marching cubes algorithm (Sect. 2.3.3).

5.3 Discussion of the Algorithm

Before presenting some results, we discuss the main concepts and assumptions of the algorithm.

5.3.1 Multiple Region Growing Process

The multiple region growing process, segmenting the initial objects (F and B) into connected components, is driven by a prioritization map f_p, which aims at locating the defects at significant locations. This is a key point of the algorithm and a difficult task. We have decided to use a prioritization map which combines some statistical information about the segmentation and distance-to-surface information through a mixing parameter λ. As previously stated, our assumption is that most of the topological defects are more likely to be located at at the thinnest parts of the volume and that topology correction often involves few voxels.

Similarly to the approach of Kriegeskorte and Goeble [56], a value of $\lambda = 0$ will locate the topological defects at the thinnest parts of the volume. The approach proposed by Han et al. in [44] is based on the same assumption: a structuring element that is progressively dilated locates topological defects at thin parts of the volume.

On the other hand, using only statistical information ($\lambda = 1$) should locate the defects at locations with low posterior probabilities. However, we note that homotopic deformation are highly sensitive to noise and inaccurate probability maps easily lead to topological defects being incorrectly located. Some incorrectly segmented voxels might have a large joint probability p_s (e.g. image artifacts, partial voluming effect, intensity inhomogeneities, etc). This could lead an expanding region to pass through these voxels and locate residual components at incorrect locations. For this reason, we combine the two informations through the use of the mixing parameter $\lambda = 0.5$. This method tends to decrease the noise issues and will still locate the defects at low probability location. More elaborate prioritization maps, introducing local anisotropy into the growing process, are possible. Another approach, based on level-sets, is discussed in Sect. 5.6.

Also, the multiple region growing process necessitates the definition of two thresholds T_e and T_s. The use of thresholds, to concurrently grow multiple components and pause the expansion process, minimizes the effects of noise on the segmentation. The first threshold T_s will prevent the creation of a new component from an isolated seed point, when its confidence value f_p is not large enough: only seed points with a large confidence value lead to the creation of a new body component, avoiding over-segmentation. The second threshold pauses the expansion of a body component, when the values $f_p(\mathbf{x})$ of adjacent simple points $\mathbf{x}$ become too low. The introduction of the threshold T_e prevents a region from expanding over and passing through "unlikely" voxels to reach regions with higher confidence values, which would certainly locates some of the topological defects at incorrect locations. T_s and T_e are defined per object. T_s corresponds to the value for which 25% of the voxels have a higher confidence value. T_e corresponds to the value for which 25% have a lower confidence value.

Finally, we note that the initial seed points, generating the body components, could be interpreted as landmarks and initialized with the help an atlas at significant locations.

5.3.2 Cost Function

During the topology correction, each voxel is assumed to carry a cost of being modified. We have taken a Bayesian approach, modeling each voxel label as independent of its neighbors and assigning a cost $c(\mathbf{x}) = ln(p_s(\mathbf{x}, X)/p_x(\mathbf{x}, \overline{X})$ to each voxel. This cost represents the variation in the logarithm of the posterior probability $p(S|I)$ of the segmentation S given the observed image I and constitutes an efficient way to make iterative minimal decisions.

However, the joint probability p_s might not be available to the user. Alternative cost options are possible. Without any more information than an initial digital binary segmentation, the user might minimize the number of modified voxels at each step, therefore assigning a constant positive cost to each voxel. At each iteration, the algorithm will delete the smallest components, i.e. the ones with the least number of voxels. Or, using a monotonously increasing function of the distance-to-surface map, such as the previously defined function f_d, one could weigh deep voxels more than others, making them less likely to be deleted. Finally, a sophisticated approach that we briefly discuss in Sect. 5.6 could incorporate curvature information.

5.4 Results

Our goal, when implementing this algorithm, was to develop a fully automated method that is able to correct the topology of digital binary segmentations, without any assumptions on the initial segmentations. In order to validate the proposed algorithm, we have applied our method to 26 brain segmentations, manually and automatically labeled. The Whole Brain Segmentation algorithm proposed by Fischl et al. in [29] was used to generate automatic brain segmentations and to compute the joint probabilities $p_s(\mathbf{x}) = p(I(\mathbf{x}), S(\mathbf{x}))$ for all voxels $\mathbf{x}$. The pair of compatible digital connectivities that we use for all the experiments reported in this chapter is (6,18). In our experience, the choice of connectivity does not significantly affect the results. We present some experiments on subcortical segmentations and white matter segmentations.

5.4.1 Description of the Data Set

The data set used in this study was acquired using a Siemens Vision system in 1994/1995 with the following parameters: TR: 9.70 ms; TE: 4.00 ms; TI: 621.00 ms; flip angle: 10.00j; 1.25 sections (resampled to 1-mm isotropic). Data comes from studies reported in Buckner et al. [11] and Logan et al. [58] and also later subjects imaged using the same anatomic protocol[3]. This data set consists of 6 Young Normal Control, 14 non-demented and 6 demented adults.

5.4.2 Subcortical Segmentations

Each of the 26 subcortical segmentations is composed of $k = 14$ nuclei: left and right ventricle, putamen, pallidum, amygdala, hippocampus, thalamus, and caudate nucleus (Fig. 5-4). We apply our algorithm independently on each subcortical nucleus. Topology correction of an individual structure takes a few seconds on a current machine.

Most subcortical segmentations have few topological defects. Results show that addition and deletion of very few voxels is necessary to correct the topology of each structure. Manual segmentations are corrected by changing the labels of approximately 0.05% of the total number of voxels. Automatic segmentations require of the order of 0.1% of labels to be changed. A typical example is given in Fig. 5-4, which shows the segmentation of the right pallidum before topology correction (Fig.5-4-a) and after topology correction (Fig.5-4-b).

[3]We thank Randy Buckner and the Washington University Alzheimer's Disease Research Center for providing the data set.

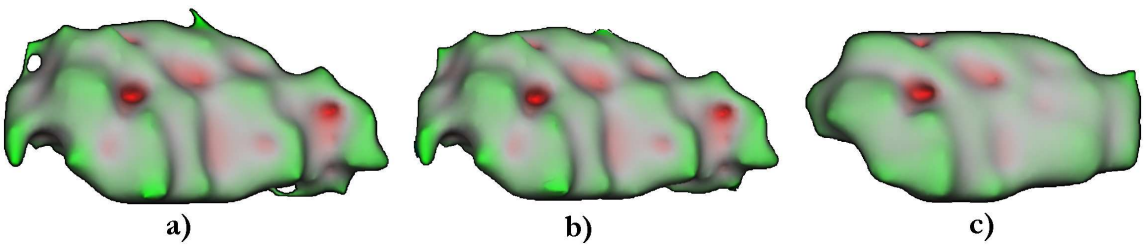

Figure 5-4: Results of the topology correction subcortical structures. a) Initial segmentation of the right Pallidum ($\chi = -4$, $g = 3$). b) Topologically corrected right Pallidum. c) The same structure after the homotopic deformation with 14 nucleus.

In this example, the initial segmentation contains 3 handles (Euler characteristic of the tessellation $\chi = -4$, genus $g = 1 - \chi/2 = 3$); the final surface has the correct spherical topology ($\chi = 2$ and $g = 0$).

Applying the topology correction independently on each nucleus, some voxels are assigned more than one label. Results show that this problem concerns less than 0.01% of the voxels. In our experiments, every multi-labeled voxel had, at most, one label that corresponded to a non-simple configuration. Consequently, a final modification of these multi-labeled voxels by deleting the "simple" labels generates valid subcortical segmentations (i.e. with one label per voxel at most), which carries the correct topology. When these conditions hold, more sophisticated methods can be applied, as the one proposed in Sect. 5.5. Nevertheless, we are aware of the limitations of this approach, as we cannot guarantee (in theory) that no voxel has more than one non-simple labels.

5.4.3 Cortical Segmentation

The correction of the topology of the cortical surface is a much more challenging task. Its highly convoluted nature often produces numerous topological defects that interact with each other, and that are difficult to precisely locate and correct. We have applied our method to 26 brains in order to generate white matter segmentations with a correct spherical topology. Before applying the algorithm, we merge the ventricles into the white matter segmentation in order to avoid topological defects to be introduced in this area. We note that the medical structures caudate, putamen, and pallidum nucleus are considered to be part of the white matter segmentation. We apply the algorithm on each hemisphere separately.

A - Convergence

We first look at the convergence of our algorithm. A binary segmentation S_o of a hemisphere of white matter contains on the order of 10^5 voxels ($100 \times 100 \times 100$ image domain). The multiple region growing process typically segments the binary volume S_o in approximately 50 body components and a few hundred residual components. During the first iteration of the algorithm most of the multi-residual components are merged into the body

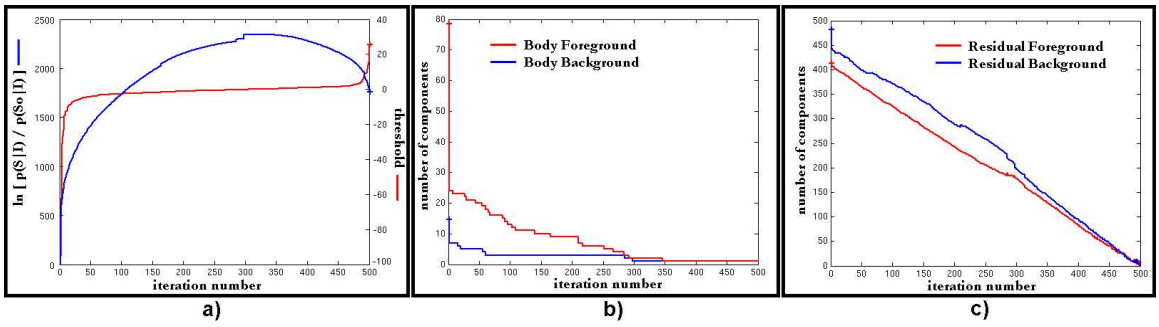

Figure 5-5: a) The threshold increase and the evolution of the probability $p(S|I)$. After a number of iterations, the threshold becomes positive, because there is no obvious decision anymore. Consequently, the probability $p(S|I)$ decreases. b) Number of body components as a function of the iteration number. Note the strong decrease of body components after one iteration, mostly due to the merging of multisimple components. c) Number of residual components in function of the iteration number.

segmentation, leading to a sharp decrease of the number of body components. After the first iteration, residual components are iteratively deleted, which corresponds to a quasi-constant decrease in the number of body and residual components. Figure 5-5 illustrates this decrease on a typical example, whose initial surface is shown in Fig. 5-6. After the first step, approximately 500 residual components remain; the algorithm also converges in approximately 500 iterations (Fig. 5-5-c).

The threshold T_{thres}, which is necessary to guarantee convergence, monotonously increases, reaching a positive value, after which the posterior probability $p(S|I)$ tends to decrease (Fig. 5-5-a). During the iterative topology correction, the algorithm often reaches a point, after which there is no obvious decision anymore. This reflects the fact that medical images often contain artifacts that perturb the topology of the segmentations. Incorrectly classified voxels represent ambiguous voxels with low confidence values. Modification of these voxels often correspond to a decrease in the probability $p(S|I)$. In the final steps of the algorithm, the algorithm enforces topology correction by choosing for the "best" decisions (i.e. the ones that decrease the least the posterior probability $p(S|I)$).

B - Results on Cortical Segmentations

Similarly to subcortical segmentation results, approximately 0.1% of the total number of white matter voxels (this approximately corresponds to 100 voxels) need to have their label changed to achieve topology correction. Most of the computation time is taken by the multiple region growing process and a white matter segmentation is corrected in approximately 30 minutes. Most of the modified voxels are located at the periphery of the white matter volume, since cortical segmentations contain few cavities or disconnected components.

However, in some cases, the algorithm generated a few incorrect topological corrections, mostly in the regions of the temporal pole. Due to the partial volume effect, the temporal region contain numerous ambiguous voxels. Digital approaches, which fail to include additional information, such as curvature information, into the topology correction process, are sensitive to the presence of image artifacts. One potential solution to alleviate this problem is proposed in Sect. 5.6.

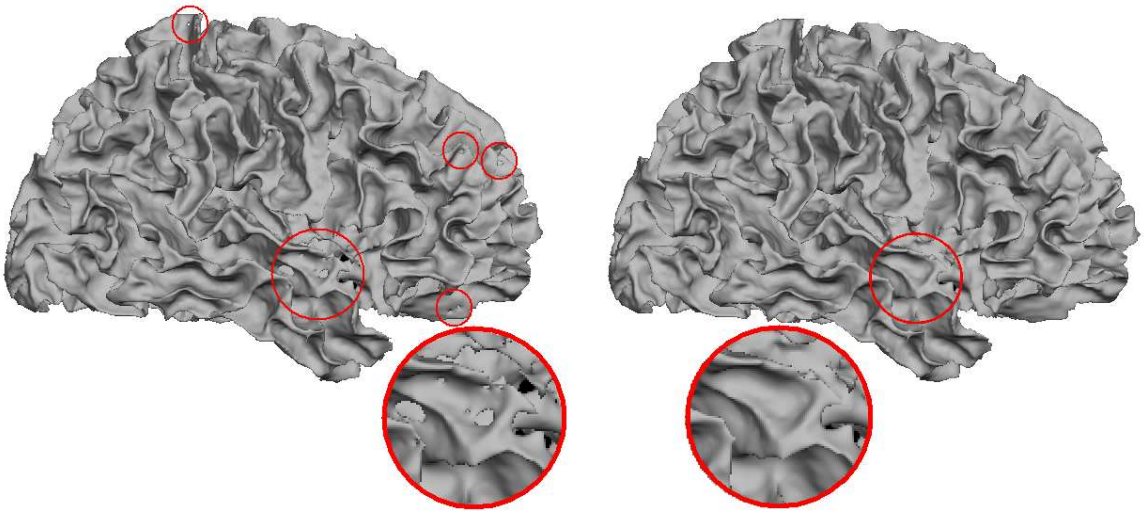

Figure 5-6: Initial Surface and Final Surface. The initial surface, generated under $(n, \overline{n}) = (6, 18)$, has an Euler characteristic of $\chi = -236$. Some topological defects are circled in red.

5.5 Extension: Homotopic Markov Random Field

In the previous section, we have presented some results about the topology correction of subcortical structures. In this section, we describe a method to integrate curvature information into the segmentation process based on a variant of Markov Random Fields. We take an approach similar to the one presented in [29] and we refer to this work for a complete description of the Bayesian framework. We only present the main concepts of our approach.

Once the topology of each structure has been independently corrected, additional information can be incorporated into the segmentation process of the image, such as curvature or prior information about the spatial distribution of the labels. The segmentation of an image is modeled by a Markov Random Field [29]. Briefly, segmentation W represents a complete segmentation into k different labels: each structure s is assigned a different label W_s, such that $\forall \mathbf{x} \in s, W(\mathbf{x}) = W_s$, and we search for the MAP estimate of $p(W|I)$. We still assume that the noise at each voxel is independent from noise at all other voxels in the image, but the voxel labels are not supposed to be independently distributed anymore. We assume that the spatial distribution of labels can be well approximated by an anisotropic non-stationary Markov random field, which allows us to encode prior information about the relationship between labels as a function of location within the brain (i.e. non-stationary), as well as with local direction (i.e. anisotropic). Under this assumption, the prior probability of the full segmentation can be expressed as:

$$p(W) \propto \prod_{\mathbf{x}\ inV} p(W(\mathbf{x})) \prod_{i=1}^{6} p(W(\mathbf{x}_i)|W(\mathbf{x}), \mathbf{x}_i), \tag{5.7}$$

where the 6 $\mathbf{x}_i$ represent the 6 voxels in the positive and negative cardinal directions at each voxel location $\mathbf{x}$. Finally, we incorporate curvature information into the Bayesian framework, as simple points are border points, for which the curvature can be easily computed $curv(\mathbf{x})$. A 3-by-3-by-3 neighborhood is sufficient to compute the curvature of the label $W(\mathbf{x})$ at voxel location $\mathbf{x}$. We use a topologically-consistent marching cube algorithm to tessellate each cube (Sect. 2.3.3).

The MAP estimate is the one maximizing the posterior probability:

$$p(W|I) \propto \prod_{\mathbf{x}\in V} p(I(\mathbf{x})|W(\mathbf{x}))p(W(\mathbf{x}) \prod_{i=1}^{6} p(W(\mathbf{x}_i)|W(\mathbf{x}), \mathbf{x}_i) \prod_{border\ \mathbf{x}} p_c(curv(\mathbf{x})) \tag{5.8}$$

In order to apply this framework to the topologically correct subcortical segmentations, topology changes are avoided by working with homotopic deformations: only simple points are considered in a modified version of the iterated conditional modes (ICM) algorithm proposed by Besag (1986). At each iteration, only simple points are updated in a random order. We note that a point must be checked for being simple if one of its neighbors has been updated.

Figure 5-7 shows the results of the topologically constrained subcortical segmentations applied to 14 structures. Most curvature artifacts, due to noise in the observed image, are removed from the initial segmentations, leading to more faithful surfaces. Each final surface carries the correct spherical topology, contrary to the initial ones, which had a total of 9 topological defects ($\chi = 10$).

The agreement between the automated and manual labelings is comparable to that obtained by comparing the labelings of different experts, although further testing on a

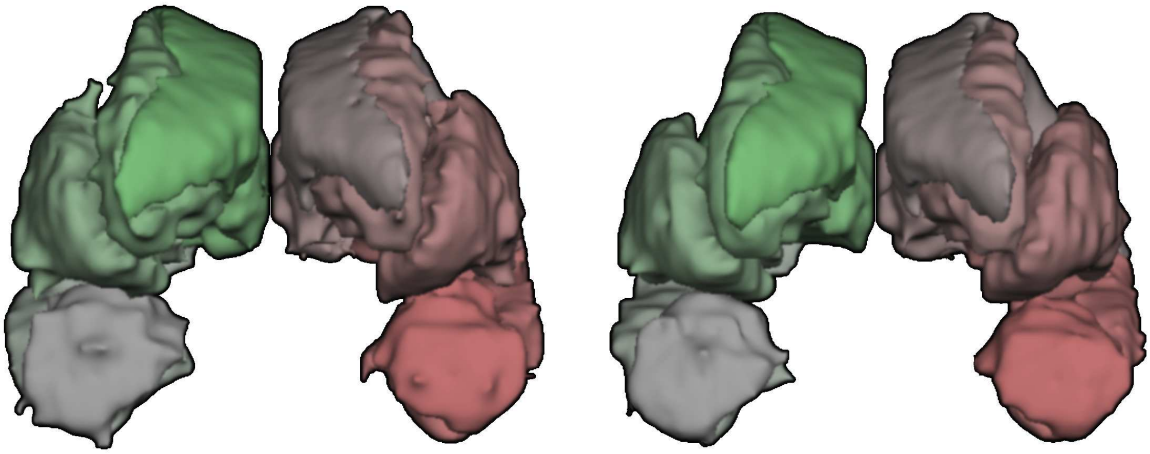

Figure 5-7: Results of the topology correction subcortical structures. The whole subcortical segmentations with 14 nucleus (left and right ventricle, putamen, pallidum, amygdala, hippocampus, thalamus, caudate nucleus) before topology correction (left) and after topology correction followed by an homotopic deformation (right). The initial surfaces had a total of 9 topological defects ($\chi = 10$).

broader database is required.

5.6 Future Work

We have presented a new algorithm, correcting the topology of digital binary segmentation. Our topology correction was phrased within the Bayesian theory under a set of simplistic assumptions. More elaborate approaches can easily be integrated into our general framework.

We note that probability maps are often corrupted by noise and homotopic deformations are known to be highly noise sensitive. In future work, we propose to overcome this limitation by integrating level-set active contours into the region growing process. Level-sets constitute an efficient way to implicitly encode a surface using the embedding space (i.e. the 3-dimensional grid). It is therefore possible to constrain the topology of a level-set active contour by using the digital topology of the underlying grid. We propose to generalize our region growing process using level set deformation under topological control (Chapter 4), in which each digital component is represented by a level-set component. For each object S (foreground and background), we propose to deform its level-set representation ϕ_S subject to the following velocity field:

$$\mathbf{v}(\mathbf{x}, t) = [f_p(\mathbf{x}, S) - \epsilon H(\mathbf{x}, t)]\mathbf{n}(\mathbf{x}, t), \tag{5.9}$$

where H denotes the mean curvature of the contour at location $\mathbf{x}$. The corresponding level set evolution is:

$$\frac{\partial \phi_S(\mathbf{x}, t)}{\partial t} = [-f_p(\mathbf{x}, S) + \frac{\epsilon}{n-1}\mathrm{div}(\frac{\nabla \phi_S}{|\nabla \phi_S|})]\ |\nabla \phi_S| \tag{5.10}$$

The second term of this equation corresponds to a mean curvature motion, which enforces a smoothness constraint into the deformation process. Therefore, curvature information is naturally integrated into the segmentation process, establishing an efficient way to alleviate noise sensitivity. Using this framework, the fitness function, measuring the goodness of

topological correction, could be written as:

$$f(S) = \int_{R^3} log \frac{p_s(\mathbf{x}, S(\mathbf{x}))}{p_s(\mathbf{x}, \overline{S}(\mathbf{x}))} d^3\mathbf{x} + \alpha \int_{surface} H(s)\ d^2 s, \tag{5.11}$$

where α is an empirical coefficient, and $H(s)$ represents the mean curvature of the surface (i.e. the boundary of the segmentation) at location s on the surface. This fitness function would favor configurations with relatively few misclassified voxels and with a resulting smooth surface. The empirical constant α establish a tradeoff between geometric information (e.g. curvature) and statistical information (e.g. misclassified voxels).

Applied to the topology correction of white matter segmentations, this approach would certainly limit the effect of early wrong decisions during the graph analysis. The graphs are constituted of approximately 50 body components and several hundred residual components.When obvious decisions no longer exist, the algorithm enforces topology correction by using the monotonously increasing threshold to select the next "best" component. An early wrong decision may lead to large geometric inaccuracies that are difficult to correct retrospectively. However, we describe in the next chapter another surface-based approach that is more adapted to the topology correction of cortical representations.

5.7 Contributions of this Chapter

In this chapter, we have presented a new algorithm that corrects the topology of digital binary segmentation. No assumption is made on the topology of the initial segmentation, and spherical topology is achieved under any choice of digital topology. A Bayesian framework allows us to integrate statistical information into the topology correction. Our algorithm can enforce exclusively background or foreground corrections. Applied to subcortical segmentations, the topology of fourteen deep nuclei is corrected in a few minutes, and white matter topology correction is achieved in about 30 minutes.

To our knowledge, this approach is the first one that has been proposed to integrate statistical information into the topology correction. Our method, nested in the theory of Bayesian parameter estimation, selects maximum *a posteriori* topological solutions based on the available information present in the image (i.e. intensity).

In addition, we have suggested two ways of integrating geometric information into our Bayesian framework. We have introduced a homotopic Markov Random Field segmentation that incorporates curvature information using the border points of the segmentation. Finally, we have proposed to generalize our framework using genus-preserving level sets, thereby integrating curvature information directly into the segmentation process.

Part of this work has been presented at the conference *Medical Image Computing and Computer-Assisted Intervention* [85].

Chapter 6

A Genetic Algorithm for the Topology Correction of Cortical Surfaces

In this chapter, we focus on the retrospective topology correction of surfaces. We propose a technique to accurately correct the spherical topology of cortical surfaces. We construct a mapping from the original surface onto the sphere to detect topological defects as minimal non-homeomorphic regions. A genetic algorithm corrects each defect by finding the maximum-a-posteriori retessellation in a Bayesian framework. During the genetic search, incorrect vertices are iteratively identified and eliminated, while the optimal retessellation is constructed. We address the mapping dependency problem by generating several configurations corresponding to different spatial optimal retessellations. Applied to synthetic and real data, our method generates optimal topological corrections with only a few iterations.

6.1 The Cortical Reconstruction Problem

In this chapter, we focus on the generation of accurate representations of the cortical surface under the spherical topological constraints. Although we constrain the final topology of the cortical sheet to be that of a sphere, the proposed approach extends to any local planar topology.

6.1.1 Cortical Anatomy

The human cerebral cortex is a highly folded ribbon of gray matter that lies inside the cerebrospinal fluid and outside the white matter of the brain. Locally, its intrinsic "unfolded" structure is that of a two-dimensional (2-D) sheet, which is several millimeters thick. The analysis of cortical data is greatly facilitated by the use of accurate 2-D models of the cortical sheet [17, 99], which alleviates most drawbacks of the three-dimensional embedding space (such as the underestimation of true cortical distances or the overestimation of cortical thicknesses).

In the absence of pathology, each cortical hemisphere is a simply-connected 2-D sheet of neurons that carries the simple topology of a sphere. There has been extensive research dedicated to the extraction of accurate and topologically-correct models of the brain surface that allows for the establishment of a global 2-D coordinate system onto the cortical brain

surface. However, because of its highly convoluted nature that results in most of its surface being buried within folds, noise, imaging artifacts, partial voluming effects and intensity inhomogeneities, the automatic extraction of accurate and topologically correct cortical surfaces is still a challenging problem.

6.1.2 Limitations of Previous Approaches

Methods for producing accurate cortical segmentations under topological constraint have been reviewed in Chapter 2.

Most methods that have been proposed [43, 89, 44, 50] assume that the topological defects in the segmentation are located at the thinnest parts of the volume and aim at correcting the topology by minimally modifying the volume or tessellation. These methods, which rely on the accuracy of the initial segmentations, often produce valid cortical representations, even though the topological corrections may not be optimal: additional information, such as the expected local curvature or the local intensity distribution, may lead to different corrections, i.e. hopefully comparable to the ones a trained operator would make.

Only a few techniques have been proposed to integrate additional information into the topology correction process [56, 28]. However, for each topological defect, these methods fail to produce more than two potential solutions. In the specific case of a handle, the two potential solutions usually correspond to either cutting the handle or filling the corresponding hole. However, the exact location of these potential solutions is most often determined based on some criteria that ignore the underlying MRI intensity profile and/or local curvature and the resulting corrections can never be optimized relative to these parameters.

6.1.3 Approach

For a given topological defect, the MRI intensity profile contains important information regarding the location and position of the potential topological correction. The resulting corrected surface should be located at the border of the white and gray matter, with white matter tissue being inside the surface and gray matter outside. Also, the smoothness of the corrected defect should match the smoothness of the rest of the cortical surface. This information should be used to guide the generation of optimal topological corrections.

In order to make full use of the available information, we propose a technique that directly extends the approach taken by Fischl et al. in [28], addressing most of its limitations. In their previous work, Fischl et al. proposed an automated procedure to locate topological defects by homeomorphically mapping the initial triangulation onto a sphere. Topological defects are identified as regions in which the homeomorphic mapping is broken and a greedy algorithm is then used to retessellate incorrect patches. The main limitations of their approach are the following:

1) Even though the final intrinsic topology will be the correct one (i.e. that of a sphere) the method does not guarantee that the final surface will not self-intersect.

2) Every vertex present in the original topologically incorrect surface will be present in the final retessellation, resulting in extremely jagged patches.

3) The information necessary to evaluate the "goodness" of an edge does not exist in isolation, but only as a function of the tessellation of which the edge is a part. This implies that a greedy algorithm cannot in general achieve geometrically accurate surfaces, as the necessary information does not exist at the time that the edge ordering is constructed.

In order to extend the greedy retessellation developed in [28], we propose to take a somewhat different approach, and evaluate the goodness of fit of the entire retessellation, not of individual edges. We introduce a genetic algorithm to explore the space of possible surface retessellations and to select an *optimal* configuration. During the search, incorrect vertices are iteratively identified and eliminated from the tessellation. Our method proceeds as follow:

1) Generate a mapping from the original cortical surface onto the sphere that is maximally homeomorphic. Each topological defect is identified as a set of overlapping triangles.

2) Discard the tessellation in each defect and generate an optimal retessellation using a genetic algorithm to search the space of potential retessellations.

In addition, we note that the space of potential retessellations is dependent on the initial mapping. We address this problem by generating a set of well-chosen distinct mappings. The resulting method is a completely self-contained topology correction algorithm, which determines optimal topologically correct solution based on the MRI intensity profile and the expected local curvature.

6.2 Identification of Topological Defects

We identify the presence of topological defects in the surface by computing its Euler-characteristic. In the presence of topological defects[1], we generate a mapping from the cortical surface $\mathcal{C}$ onto the sphere $\mathcal{S}$ that is maximally homeomorphic and identify each defect as a set of overlapping faces. This step is identical to the approach developed by Fischl et al. in [28]. Briefly, the identification of topological defects begins with the inflation and projection of the cortical surface $\mathcal{C}$ onto a sphere $\mathcal{S}$. Next, we generate a maximally homeomorphic mapping $\mathcal{M} : \mathcal{C} \rightarrow \mathcal{S}$ by minimizing an energy functional that directly penalizes regions in which the determinant of the Jacobian matrix of $\mathcal{M}$ becomes zero or negative; these regions are non-homeomorphic regions (Sect. 2.2.2). Finally, the topological defects are identified by regions, where the homeomorphism is broken (i.e. regions with negative determinant or, equivalently, regions with overlapping faces). The following detailed steps were taken from [28].

6.2.1 Initialization of the Mapping: Spherical Inflation

The initial mapping of the cortical surface to that of a sphere could be accomplished by simply projecting each point of the cortical surface to the closest point on the sphere. Doing so would result in large regions of the initial mapping being non-homeomorphic.

Instead, we use a simple procedure to unfold and smooth the folded cortical surface so that it approaches that of a sphere whose origin is the centroid of the initial surface. The algorithm consists in iteratively updating the position of each vertex based on a smoothness force $\mathbf{F}_S$, and a radial spherical force $\mathbf{F}_R$:

$$\mathbf{x}_k(t+1) = \mathbf{x}_k(t) + \mathbf{F}_S(t) + \lambda_R \mathbf{F}_R(t) \tag{6.1}$$

[1]The Euler number of a surface is a topological invariant (Sect. 2.2.3). For a tessellation, it can be easily computed as: $\chi = \#vertices - \#edges + \#faces$

where $\mathbf{x}_k$ is the position of the k^{th} vertex at iteration number t and the smoothness force $\mathbf{F}_S$ is given by:

$$\mathbf{F}_S = \frac{1}{N_k} \sum_{j \in N_k} (\mathbf{x}_j - \mathbf{x}_k) - \frac{1}{V} \sum_{i}^{V} \sum_{j \in N_j} (\mathbf{n}_i \mathbf{n}_j).(\mathbf{x}_j - \mathbf{x}_i) \tag{6.2}$$

where:

N_k is the set of vertices neighboring the k^{th} vertex;

V is the number of vertices in the tessellation;

$\mathbf{n}_k$ and $\mathbf{n}_k$ are the surface normals at location k and its transpose, respectively.

The smoothness term $\mathbf{F}_S$ moves each vertex in the direction of the centroid of its neighbors, while projecting out the average inwards movement this creates over the entire surface. The radial term simply drives each vertex toward the surface of a sphere with the desired radius R:

$$\mathbf{F}_R = (\mathbf{R_k} - \mathbf{x_k}) \tag{6.3}$$

where $\mathbf{R}_k$ is the radial projection of x_k onto the sphere with radius R.

We use an R on the order of $100mm$ as this results in a sphere with about the same total surface area as an average cortex, and a λ_R of 0.25 to allow sufficient smoothing to take place during the spherical inflation. Once the inflation has converged, the surface is projected so that it lies precisely on the surface of a sphere of radius R.

6.2.2 Quasi-Homeomorphic Mapping

Once the initial spherical configuration $\mathcal{M}_0$ has been established, we generate a mapping $\mathcal{M}$ that is maximally homeomorphic, which we term a *quasi-homeomorphic* mapping. In generating the mapping $\mathcal{M}$, we are only concerned with its topological properties, that is, we wish $\mathcal{M}_1$ to be as close to a homeomorphism as possible. A mapping is a homeomorphism if the determinant of its Jacobian matrix is non-singular, and the mapping itself is continuous. This is of course the multidimensional analog of monoticity. To construct the mapping, we minimize an energy functional that directly penalizes regions in which the determinant becomes zero or negative, thus encouraging positive definiteness. Note that this is the only term in the energy functional - no preservation of metric properties is needed.

The Energy Functional

More specifically, noting that the Jacobian yields a measure of the deformation of an oriented area element under the mapping $\mathcal{M}$, the energy functional $E_{\mathcal{M}}$ limits the penalization of compression primarily to negative semi-definite regions. If the initial area on the folded surface of the i^{th} face is A_i^0, and the area on the spherical surface $\mathcal{S}$ at time t of the numerical integration is A_i^t, then the energy functional is given by:

$$E_{\mathcal{M}} = \sum_{i=0}^{F} \log(\frac{1 + e^{kR_i}}{k}) - R_i \ , \ R_i = \frac{A_i^t}{A_i^0} \ , \frac{\partial E_{\mathcal{M}}}{\partial A_i^t} = \frac{-e^{-kR_i}}{A_i^0(1 + e^{-kR_i})} \tag{6.4}$$

The logarithmic nonlinearity limits the penalization of compression primarily to negative semi-definite regions, as can be seen in the plot in Fig. 6-1-a. R_i is an approximation of the Jacobian of the transformation $\mathcal{M}$ (Sect. 2.2.2). The extent to which highly compressed

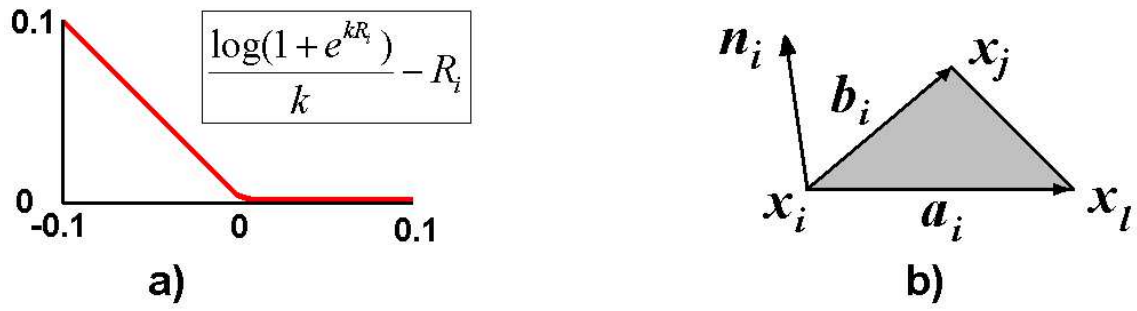

Figure 6-1: a) Non-linearity of the energy functional $E_{\mathcal{M}}$ b) Triangle properties

positive definite regions are penalized is determined by k. In practice, we used a value for k of 100.

Numerical Implementation

In order to complete the definition of the topology term of the energy functional, we consider the i^{th} triangle in the surface tessellation, with edges $\mathbf{a}_i$ and $\mathbf{b}_i$ connecting the vertex $\mathbf{x}_i$ to two of its neighbors $\mathbf{x}_l$ and $\mathbf{x}_j$ respectively. In the spherical representation, the normal vector field can be given a consistent orientation on the surface[2] using the embedding space, and A_i becomes an *oriented* area, which may take on negative values indicating folds in the surface. The normal vector is chosen as pointing outward on the surface of the sphere $\mathbf{n}_i = \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|}$ (the sphere is centered at the origin).

Using the chain rule, the directional derivative of E_M with respect to the position of the k^{th} vertex:

$$\frac{\partial E_{\mathcal{M}}}{\partial \mathbf{x}_k} = \frac{\partial E_{\mathcal{M}}}{\partial A_i^t} \frac{\partial A_i^t}{\partial \mathbf{x}_k} \tag{6.5}$$

The first factor is given by Eq. 6.6. The second is the change in the area of the i^{th} triangle caused by moving the k^{th} vertex, which can be computed from the prior description of the metric properties of the tessellation using the chain rule as:

$$\frac{\partial A_i^t}{\partial \mathbf{x}_k} = \frac{\partial A_i^t}{\partial \mathbf{a}_i} \frac{\partial \mathbf{a}_i}{\partial \mathbf{x}_k} + \frac{\partial A_i^t}{\partial \mathbf{b}_i} \frac{\partial \mathbf{b}_i}{\partial \mathbf{x}_k} \tag{6.6}$$

with

$$\frac{\partial A_i^t}{\partial \mathbf{x}_k} = \mathbf{b}_i \wedge \mathbf{n}_i \ , \frac{\partial A_i^t}{\partial \mathbf{x}_k} = \mathbf{n}_i \wedge \mathbf{a}_i \ . \tag{6.7}$$

The partials of the change in the legs with respect to a change in the vertex position are

[2]This is always possible except in pathological cases such as the Mobius strip that are said to be non-orientable.

dependent on what position the vertex in question occupies in a given triangle

$$\frac{\partial \mathbf{a}_i}{\partial \mathbf{x}_k} = \begin{cases} [-1,-1,-1]^T & : \quad k = i \\ [1,1,1]^T & : \quad k = l \\ [0,0,0]^T & : \quad otherwise \end{cases} \tag{6.8}$$

$$\frac{\partial \mathbf{b}_i}{\partial \mathbf{x}_k} = \begin{cases} [-1,-1,-1]^T & : \quad k = i \\ [1,1,1]^T & : \quad k = j \\ [0,0,0]^T & : \quad otherwise \end{cases} \tag{6.9}$$

6.2.3 Identification of Topological defects

The resulting mapping $\mathcal{M}$ - from the initial tessellation $\mathcal{C}$ to the sphere $\mathcal{S}$ - is maximally homeomorphic. The surface is examined for regions of non invertibility, as these are areas where the current tessellation must be discarded and a new one constructed in order to ensure the proper topology. Multivalued regions, containing overlapping triangles, constitute topological defects where the homeomorphic mapping is broken. $\mathcal{M}$ associates at each vertex v of the initial cortical surface $\mathcal{C}$ a vertex $v_{\mathcal{S}} = \mathcal{M}(v)$ on the sphere $\mathcal{S}$. Vertices with spherical coordinates that intersect a set of overlapping triangles are marked as defective and topological defects are identified as connected sets of defective vertices.

6.3 Optimal Topology Correction using a Genetic Algorithm

6.3.1 Definition of the Retessellation Problem

Once a topological defect has been identified, its tessellation is discarded. The retessellation problem can then be stated as follows.

Given a set of defective vertices, each of which has been assigned a spherical location by the quasi-homeomorphic mapping $\mathcal{M}$, find the vertices that should be kept in the defect and the set of edges connecting them, so that an energy functional, measuring the goodness of the retessellation, is maximized.

Topological inconsistencies, which are resulting from mislabeled voxels in the segmentation process, generate tessellations that include incorrect vertices. These vertices should be identified and discarded from the final solution. A potential topological correction of the defect corresponds to the generation of a new tessellation such that no edge intersection occurs in the spherical surface.

Many such tessellations exist, and one would like to select an optimal solution that maximizes the goodness of fit of the retessellation. Before describing our method, we diverge slightly from the main topic in order to evaluate the size of the considered space.

Evaluation of the size of the space of potential retessellations

In order to evaluate the size of the space of potential retessellations, we consider one single defect, constituted of n_b bordering vertices and n_v inside vertices. For this defect, the number of potential edges is $N = (n_v + n_b)((n_v + n_b) - 1)/2$. The Euler-characteristic of the retessellated patch is equal to: $\chi = (n_v - n_e + n_f) = 1$, where n_e and n_f are the

number of edges and faces inside the defect respectively. For a topologically correct spherical surface, every face has exactly 3 edges and every edge is bordering 2 faces exactly. This implies that we have the following relation: $3(n_f + n_b) = 2(n_e + 2n_b)$ or $n_f = 2/3n_e + n_b/3$. Therefore, the number of added edges in the final topologically correct retessellation exactly is : $n_e = (n_b - 3) + 3n_v$. Therefore, we can approximate the size of the space of potential retessellation by $C_N^{n_e} = C^{(n_b-3)+3n_v}_{(n_v+n_b)((n_v+n_b)-1)/2}$. However, we note a set of added edges imposes constraints on the set of other potential added edges, and that this constraint actually decreases the size of the space. Nevertheless, the edge ordering will be used to naturally discard vertices in the retessellation. This implies that the space of potential retessellations depends on the ordering of the edges, leading to a space of extremely large dimensionality.

We evaluate the fitness of a corrected region with the maximum-a-posteriori estimate of the retessellation, given geometric information about the observed surface, and the underlying MRI values. The numerical technique we propose to explore in the maximization of the fitness function is a genetic algorithm [3, 19, 34, 80, 102, 101] or GA (for a good introduction see [69]). The GA is an appropriate choice for this type of problem as the space to be searched is potentially quite large (the defects can contain upwards of 300,000 candidate edges), and there is no easy way to compute gradient information. More importantly, we define a set of genetic operations used to propagate information from one generation to the next that correspond to "relevant" surface operations.

6.3.2 A Genetic Algorithm for the Surface Retessellation

Genetic Algorithms were developed by John Holland in the 1960s as a means of importing the mechanisms of natural adaptation into computer algorithms and numerical optimization [49], with much subsequent theoretical work [38]. In genetic algorithms, a candidate solution to a problem is typically called a chromosome, and the evolutionary viability of each chromosome is given by a fitness function. Typically, genetic algorithms are defined by different operators: Selection, Crossover and Mutation.

- Selection: the selection of chromosomes from a population for reproduction, usually based on their fitness.
- Crossover: the generation of a new chromosome by combining parts of two "parent" chromosomes, roughly patterned after the biological process of recombination of two haploid, or single-chromosome, individuals.
- Mutation: the random change of parts of a chromosome (typically with relatively low probability).

In the next paragraphs, we explain the role of these operators in detail and specify how their definition is meaningfully tailored to the current problem.

A - Representation and Retessellation

Perhaps the most important decision in the construction of a GA is the choice of representation for the underlying problem. Here we have a number of constraints that must be satisfied that lead to the representation we use. These essentially amount to the requirement that every potential edge be represented exactly once in an ordering for the retessellation.

This guarantees that the retessellation will result in the proper topology [28]. Thus the representation we choose is an edge ordering, represented by a permutation of N integers. The retessellation procedure then simply involves adding edges in the order specified by the permutation.

Such a procedure will generate retessellated patches that include all vertices present in the defect, resulting in irregular jagged surfaces. In order to alleviate this problem, we directly encode the vertex selection into the representation. Given an edge ordering, we iteratively construct the corresponding tessellation and discard isolated vertices that are located inside formed triangles. During the retessellation, every time an newly added edge generates a triangle, inside vertices that are not connected are simply discarded. This way, edges added first in the retessellation will force their bordering vertices to be included in the final retessellation. The edges added last, which most often generate the surface irregularities, will consequently be discarded.

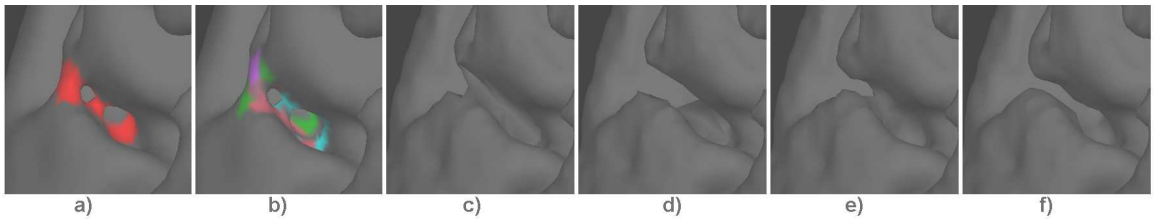

Figure 6-2: a) Example of a topological defect containing 2 handles and constituted of 183 defective vertices. b) Result of the clustering of the non-intersecting edges into 5 segments. c-e) These candidate retessellations represent different configurations of the initial population generated using the edge clustering. f) The optimal solution generated by our genetic approach in 15 generations after 4 mutations and 8 crossovers.

B - Selection of the Initial Population

The selection of the initial population is particularly significant for the considered problem. The space to be searched is potentially quite large and the selection of a "good" initial population can drastically improve convergence of the algorithm. Topological defects are constituted of sets of overlapping triangles. The intersecting edges on the sphere $\mathcal{S}$ correspond to different topological paths in the original cortical surface $\mathcal{C}$. In order to generate an initial population with a large variance (i.e. composed of individuals with large *shape* differences) we first group the non-overlapping edges into different clusters. Using the spherical quasi-homeomorphic mapping M, intersecting edges are iteratively segmented into different clusters. Next, these clusters are used to select the initial population of chromosomes. We say that a chromosome is generated from a cluster C_i, if the first edges (in the ordering) constituting this chromosome comes from C_i. Consequently, chromosomes generated from different clusters will have different shapes, hopefully leading to an initial population with a large variance. Figure 6-2 provides a few examples of initially selected chromosomes in the case of a simple topological defect.

C - Mutations and Crossovers

The two most important operations used in GAs are mutation and crossover. Mutation involves the random modification of a part of the code of an "individual" in the population and crossover the exchange of a part of the code of an "individual" with another one in the

population. We define these operations in order to accommodate the nature of the current problem. Intersecting edges represent choice between different surface configurations. In the following section, we note I_i the set of edges intersecting the edge e_i: $I_i = \{e_j | \text{int}(e_i, e_j) = 1\}$, where $\text{int}(e_i, e_j)$ is the intersection operator, and returns 1 if edge e_i intersects edge e_j, and 0 otherwise.

C.1 - Mutations: the mutation operation, described in Alg. 10, corresponds to a random swap operation of intersecting edges in the ordered representation. This procedure will allow the selective exploration of the different retessellations represented by different members of I_i, thus reducing the size of the effective search space.

Algorithm 10 Mutation Operator

for all edges in the ordering $e_i \in \mathcal{O}$ **do**
  Draw a random number r from $U_{\mathbb{R}}(0, 1)$ the uniform distribution on the real numbers between 0 and 1
  if $r \leq p_{mut}$ **then**
    Draw a random number k from $U_{\mathbb{N}}(1, |I_i|)$ the uniform distribution on the natural numbers between 1 and $|I_i|$
    Exchange the positions of e_i and e_j where e_j is the k^{th} entry in the set I_i.

C.2 - Crossovers: the crossover operator we define is the random combination of permutations (see Alg. 11). Some care must be taken here to insure that every edge is represented exactly one time. Towards that end, the crossover operator will add a random number of edges from each parent retessellation, only if that edge has not been added. The crossover operator will randomly select one of the permutations to draw from first, then copy a random number of edges from it to the "offspring" retessellation. For each edge, we draw a random number r from $U_{\mathbb{R}}(0, 1)$, and stop copying edges if $r < 1/2$. Next, a random number of edges will be copied from the second parent, if they are not already represented in the offspring. This procedure will continue until every edge is represented.

Algorithm 11 Crossover Operator

Two parent orderings $\mathcal{O}_1$ and $\mathcal{O}_2$ indexed by two integers $k_1 = k_2 = 0$; $\alpha = 1$
Draw a random number r from $U_{\mathbb{R}}(0, 1)$
if $r \leq p_{mut}$ **then**
  Set $\alpha = 2$
repeat
  Set $e = e_{k_\alpha} \in \mathcal{O}_\alpha$ and set $k_\alpha = k_\alpha + 1$
  if $e \in \mathcal{O}_{dst}$ **then**
    add edge into offspring chromosome: $2 \rightarrow \mathcal{O}_{dst}$
  Draw a random number r from $U_{\mathbb{R}}(0, 1)$
  if $r \leq p_{mut}$ **then**
    Set $\alpha = (\alpha + 1) mod[2]$
until all edges are represented in $\mathcal{O}_{dst}$: $|\mathcal{O}_{dst}| = |\mathcal{O}_\alpha|$

It is important to note that the previously defined genetic operations carry meaningful geometric operations. Mutation, which randomly swaps the ordering of intersecting edges,

corresponds to local jumps from one configuration to another one. The crossover operation naturally combines different parts of the code from the two candidate tessellations, generating a configuration that often expresses distinct local surface properties of both parents. In addition, since the edge ordering naturally encodes which vertices are discarded (the vertices included last being discarded), the crossover operation, which iteratively combines two edge orderings, most often generates offspring chromosomes that preserve the best geometric characteristics of the parents (most likely, the same vertices will be discarded).

D - Fitness and Likelihood Functions

We use some prior knowledge about the cortex to define the fitness function. A cortical surface is a smooth manifold $\mathcal{C}$ that partitions the embedding space into an inside part, composed of white matter, and an outside part, composed of gray matter. We characterize the goodness of a retessellation by measuring two of its properties:

- the smoothness of the resulting surface,
- the MRI values I inside and outside the surface.

Formally, the posterior probability of the i^{th} retessellation T_i is given by:

$$p(T_i|\mathcal{C}, I) \propto p(I|\mathcal{C}, T_i)p(T_i|\mathcal{C}). \tag{6.10}$$

The likelihood term $p(I|\mathcal{C}, T_i)$ encodes information about the MRI intensities inside and outside the surface. Each retessellated patch, being topologically correct, separates the underlying MRI volume into two distinct components[3], an inside part $\mathcal{C}^-$ and an outside part $\mathcal{C}^+$. An acceptable candidate solution should generate a space partition with most of its inside and outside voxels corresponding to white and gray matter voxels respectively. In order to estimate the likelihood $p(I|\mathcal{C}, T_i)$, we assume that the noise is spatially independent. This probability can be rewritten:

$$p(I|\mathcal{C}, T_i) = \underbrace{\prod_{\mathbf{x}\in\mathcal{C}^-} p_w(I(\mathbf{x})|\mathcal{C}, T_i) \prod_{\mathbf{x}\in\mathcal{C}^+} p_g(I(\mathbf{x})|\mathcal{C}, T_i)}_{\text{volume-based information}} \underbrace{\prod_{v=1}^{V_i} p(g_i(v), w_i(v)|\mathcal{C}, T_i)}_{\text{surface-based information}}, \tag{6.11}$$

$p_w(I(x)|\mathcal{C}, T_i)$ and $p_g(I(x)|\mathcal{C}, T_i)$ are the likelihood of intensity values at location x in the volume inside and outside the tessellation respectively, $p(g_i(v), w_i(v)|\mathcal{C}, T_i)$ is the joint likelihood of intensity values inside and outside the tessellation at vertex v in tessellation T_i.

Geometric information can be incorporated via $p(T_i|\mathcal{C})$, which represents priors on the possible retessellation. For example, $p(T_i|\mathcal{C})$ could have the form:

$$p(T_i|\mathcal{C}) = \prod_{v=1}^{V_i} p(\kappa_1(v), \kappa_2(v)|\mathcal{C}), \tag{6.12}$$

where κ_1 and κ_2 are the two principal curvatures of the surface, computed at vertex $\mathbf{v}$.

Given that the vast majority of the surface is in general not defective, we fortunately have ample amounts of data with which to estimate the correct forms of the distributions

[3]We use the angle weighted pseudo-normal algorithm to compute the signed distance of the tessellation [2]. The voxel grid is partitioned into inside negative values and outside positive values

$p(T_i|\mathcal{C})$, $p_g(I(\mathbf{x})|\mathcal{C}, T_i)$, $p_w(I(\mathbf{x})|\mathcal{C}, T_i)$ and $p(g_i, w_i|\mathcal{C}, T_i)$. In particular, the single tissue distributions $p_g(I(\mathbf{x})|\mathcal{C}, T_i)$ and $p_w(I(\mathbf{x})|\mathcal{C}, T_i)$ are locally estimated around each topological defect in a region that excludes the defect itself (we exclude all voxels that intersect one of the N potential edges). This makes the resulting procedure completely adaptive and self-contained, in the sense that no assumptions need to be made about the contrast of the underlying MRI image(s), and no training or parametric forms are required for $p(T_i|\mathcal{C})$. An example of the estimation of $p(g_i, w_i|\mathcal{C}, T_i)$ and $p(T_i|\mathcal{C})$ is given in Fig. 6-3. Image b) shows the joint distribution of gray and white matter given the surface computed using the non-defective portion of the gray/white boundary representation of a single subject. Note the diagonal character of the distribution, indicating that the intensities are mutually dependent - brighter white matter typically means brighter gray matter due to factors such as bias fields induced by RF inhomogeneities and coil sensitivity profiles, as well as intrinsic tissue variability. One possible form of the priors on the tessellation is given in Fig. 6-3-c, which shows the joint distribution of the two principal curvatures κ_1 (green) and κ_2 (red) computed over the non defective portion of a single surface. It is important to note in this context that all these distributions can only be applied after a candidate retessellation has been completed, as the gray/white joint density requires surface normals, gray and white intensity distributions necessitate the underlying MRI volume to be partitioned in two separate components and the principal curvatures require the calculation of the second fundamental form, all of which are properties of the surface, not of individual edges.

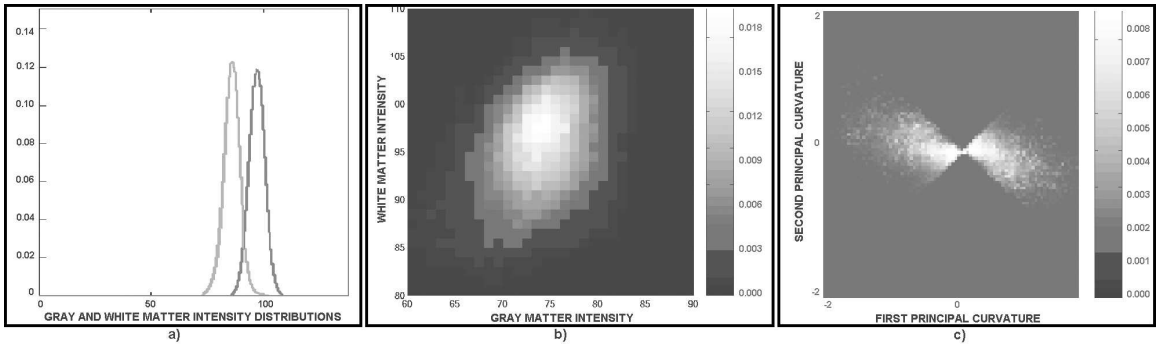

Figure 6-3: a) Example of the gray and white matter distributions estimated locally from a given a topological defect. b) Joint distribution of gray and white matter given the surface computed using the non-defective portion of the gray/white boundary representation of a single subject. The gray and white matter intensity are two correlated variables, as indicated by the diagonal structure of the joint distribution. c) Joint distribution of two principal curvatures of the surface.

E - Optimization Using Active Contour Patches

During the genetic search, candidate patches T_i are selected based on their fitness value $p(T_i|\mathcal{C}, I)$. Due to the spherical topological constraint, each patch defines a valid manifold that can be treated as an active contour with fixed boundaries. Each patch is locally deformed in order to maximize the posterior probability $p(T_i|\mathcal{C}, I)$. Instead of deriving the exact *Euler-Lagrange* equation of the active contour T_i for the energy functional $p(T_i|\mathcal{C}, I)$[4], we use an approximation procedure. We note that the fitness function of a chromosome measures the smoothness of the resulting surface and the MRI intensity profile inside and

[4]The *Euler-Lagrange* equation is computationally unstable as it contains third-order derivatives.

outside the surface. We simply update the position of each interior vertex $\mathbf{x}_k$ of the candidate tessellation based on a smoothness force $\mathbf{F}_S$ and an MRI intensity-based force $\mathbf{F}_M$:

$$\mathbf{x}_k(t+1) = \mathbf{x}_k(t) + \mathbf{F}_S(t) + \lambda_M \mathbf{F}_M(t). \tag{6.13}$$

The smoothness force is the same one as the one defined in Eq. 6.2. The intrinsic curvature-based force enforces a smoothness constraint on the deformed active contours and tends to minimize the prior term $p(T_i|\mathcal{C})$. The MRI intensity-based force $\mathbf{F}_M$ is designed to drive the active contour towards the true boundary separating the gray from the white matter:

$$\mathbf{F}_M = [T_v - I(\mathbf{x}_k)]\nabla I(\mathbf{x}_k), \tag{6.14}$$

where the targeted value T_v is computed from the gray and white matter distributions. The mean intensity and variance of the gray and white matter intensities are estimated from the respective distributions p_g and p_w, denoted by μ^g, σ^g, μ^w and σ^w, and the local threshold T_v is computed based on the Mahalanobis distance:

$$T_v = \frac{\mu^w \sigma^g + \mu^g \sigma^w}{\sigma^w + \sigma^g}. \tag{6.15}$$

At each iteration, we measure the exact fitness function $p(T_i|\mathcal{C}, I)$ of the active contour and stop the deformation when the fitness function is maximized. The constant λ_M is empirically set to 0.5.

F - Iterative Elimination of Vertices

During the genetic search, some vertices will be consistently discarded from the best patches. These vertices, which are the ones that were erroneously kept in the initial cortical tessellation, should be identified and eliminated from the final tessellation. To this end, we introduce in our genetic search, an elimination operator, which selectively eliminates the worst vertices from the defect. The elimination step operates as follows: after every few iterations, we eliminate the vertices that were consistently discarded from the best candidate patches.

The proposed approach is implemented with the following parameters. The initial population size is chosen depending on the number of defective vertices. The retessellation process is quadratic in the number of vertices contained within the convex hull of each defect. Typical defect contains on the order of 100 vertices for a population size of 20 candidate retessellations. At each step of the genetic search, a new population is generated from selected chromosomes based on their fitness. Given a population of individuals, the top one third is selected to form the elite group. These chromosomes are kept for the next generations. The worst individuals, corresponding to the bottom one third, are replaced with mutated copies of the best. Finally, the remaining ones are generated from crossover operations from parents iteratively chosen from the elite population. The mutation rate p_{mut} is experimentally chosen to be 10%. The algorithm stops when no new best candidate has been found for the past 10 generations. For a large topological defect of size 100 vertices, the algorithm usually converges in less than 50 generations, which corresponds to a computational time of approximately 10 minutes on a 1-G-Hz Pentium IV. Typical topolog-

ical defects contain less than 50 vertices, and are usually corrected in a couple of minutes. An optimal configuration is usually the result of approximately 30 genetic operations, 80% of which are crossovers and 20% mutations. The elimination operator is applied every 5 generations. The number of discarded vertices depends on the topological defect. In some cases, more than 40% will be eliminated.

6.4 The Mapping Dependency Problem

6.4.1 Definition of the Mapping Dependency Problem

Given a quasi-homeomorphic mapping from the initial cortical surface onto the sphere, the genetic algorithm that we introduced generates optimal solutions. However, we note that the space to be searched is dependent on the initial spherical location of the defective vertices. The spherical location of the defective vertices reduces the space of potential retessellations by constraining the relative connectivity of some vertices. Particularly, some tiling configurations might not be attainable, as these configurations might lead to overlapping faces on the sphere. This implies that, in some cases, the genetic search might not be able to produce desired solutions.

For most defects, "quasi-optimal" configurations are achieved during the genetic search. This is essentially due to the small size of the defects, which often takes the form of one single handle of size smaller than $5mm$. We use the term "quasi-optimal" to indicate the fact that the resulting solution might not be the best solution, but still provides an excellent approximation that does not lead to any detectable errors (visually and in terms of the fitness function). For such small defects, the initial spherical projection does not span the entire space of potential configurations, but the initial mapping produces "quasi-optimal" configurations that have similar fitness values to true optimal configurations. Also, we note that the active contour model used for each generated patch greatly reduces the impact of the spherical mapping onto the final solution.

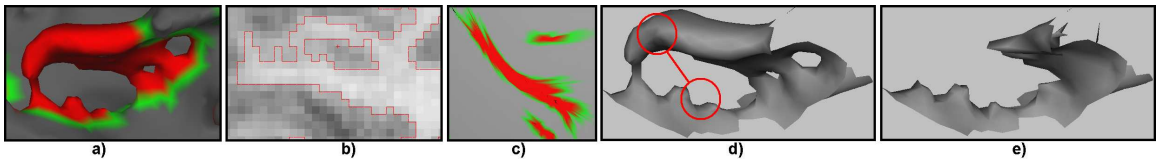

Figure 6-4: a) Original defect: red and green vertices represent inside and border vertices respectively. b) One sagital view of the defect c) Corresponding spherical projection. d) Original defect. The vertices in the circled regions have the same location on the sphere. e) Incorrect solution generated by the genetic algorithm using the spherical mapping. This solution corresponds to the best candidate within the space of potential retessellation constrained by the initial spherical mapping.

However, in rare cases (only with very large defects in our experience), the mapping dependency problem is important. Figure 6-4 illustrates this problem on a real data example. The defect consists of 343 vertices and an average size $20mm$. The solution generated by the genetic algorithm is presented in Fig 6-4-e . The circled vertices in Fig 6-4-d have the same spherical location in the defect. Therefore, no candidate solutions could be generated to include these vertices all at once. To address this problem, we propose to generate several well-chosen mappings corresponding to different optimal retessellations, and to simply select the best candidate retessellation as the final solution. We estimate the size of each defect $\mathcal{D}$ by computing the geodesic distance (onto the cortical representation) of each interior vertex to the border of the defect $\partial\mathcal{D}$. The size $s_{\mathcal{D}}$ of each defect is simply estimated by $s_{\mathcal{D}} = 2 \max_{v \in \mathcal{D}} d(v, \partial\mathcal{D})$. If this size is greater than $10mm$, we apply the procedure described in the next sections to generate several mappings.

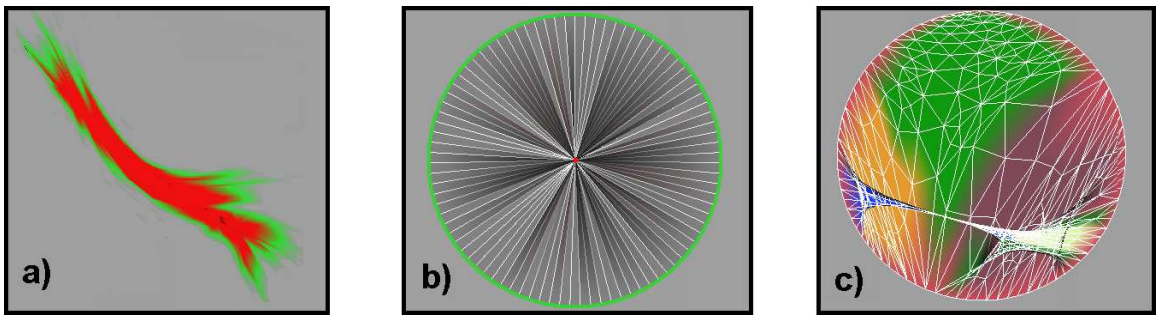

Figure 6-5: a) Location of a single defect onto the sphere. b) The same defect projected onto the 2D plane using the 2D unit sphere. The border vertices are regularly mapped onto the unit circle $\mathcal{U}(0,1)$ and the interior vertices are positioned at the center of the circle. c) Positions of the inside vertices in one possible mapping generated using the procedure described Sect. 6.4.3.

6.4.2 From the Sphere to the Plane

The initial quasi-homeomorphic mapping $\mathcal{M}$ could be used to generate several mappings $\mathcal{M}_i$ leading to different configurations. While the border vertices of a defect are fixed on the sphere and prevented from moving, we wish to update the position of the inside vertices, therefore generating several different quasi-homeomorphic mappings. However, as shown in the example provided in Fig. 6-5-a, the spherical location of the defects may take the form of complex shape, which are rarely convex. For these complex shapes, the modification of the position of the interior vertices proves to be difficult, as the topology of each defect has to be preserved. Specifically, this means that interior vertices must be prevented from crossing the closed contour formed by the border vertices[5]. Generating several mappings (i.e. perturbing the positions of the inside vertices under the non-crossing constraint) within complex non-convex shape is extremely difficult.

In addressing this problem, we use the fact that the border of each defect defines a closed curve, which can be projected onto the 2D plane. The retessellation problem can then be transposed from the sphere onto the 2D plane. Given a specific defect, we project its border vertices regularly along a unit circle $\mathcal{U}(0,1)$ in the 2D plane. The interior vertices are then initially positioned at the center of the circle (Fig. 6-5-b). Using the 2D plane and the unit circle $\mathcal{U}$ to generate several mappings give rise to several advantages. First, checking for intersecting edges is easier on the plane than on the sphere. Next, the unit circle $\mathcal{U}$ has a convex shape. Modifying the position of the interior vertices, while forcing them to stay inside the convex shape $\mathcal{U}$ is much easier than using the original spherical mapping. The convexity of the unit circle and Thm. 6.1 presented below motivate the relaxation procedure that we have adopted to generate several quasi-homeomorphic configurations (described in the next section).

6.4.3 Generating Different Mappings

In order to generate different mappings, we first cluster proximal interior vertices in the original cortical surface into a set of p groups by using a modified k-means algorithm. First, p vertices are randomly selected among the interior vertices. These vertices constitute the initial means $\mathbf{x}_i$ of each group C_i. The remaining vertices are then assigned to the closest

[5]This is a consequence of the Jordan curve theorem, which states that any simple closed curve partitions the plan into an inside region and an outside region

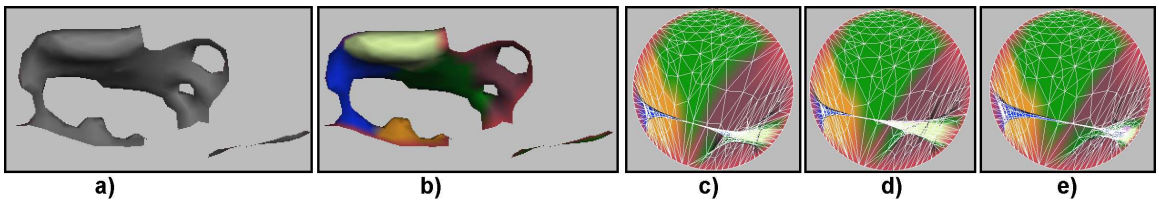

Figure 6-6: a) Original defect. b) Clustering of proximal vertices using a k-means algorithm with 10 clusters. c-e) Different Mappings generated using a relaxation procedure that ignores one cluster at a time. Figure e) represents the resulting mapping when the blue cluster in figure b) is ignored.

(using geodesic distances) group. Then, the most "central" vertex of each cluster (i.e. the one minimizing the geodesic average of each cluster $\sum_{v \in C_i} d(\mathbf{x}_i, v)$) is chosen as the new cluster representant, and we iterate the procedure until convergence. Figure 6-6-b shows the result of the clustering process into 10 clusters. The number of clusters is based on the size of the defect and empirically set to $p = [\frac{s_D}{2}]$.

Next, we generate p mappings in the 2D plane, by applying p iterative relaxation procedures. Given a chosen cluster C_i, the positions of the interior vertices are iteratively updated to be at the average of their neighboring vertex positions, excluding the vertices that belong to the chosen cluster C_i. The motivation for this procedure is based on the following theorem (Tutte [97], Floater [33], Richter-Gebert [81]):

Theorem 6.1 Topology of a planar graph
Given a planar 3-connected graph with a boundary fixed to a convex shape in $\mathbb{R}^2$*, the positions of the interior vertices form a planar triangulation (i.e. none of the triangles overlap) if and only if each vertex position is some convex combination of its neighbor's positions.*

Theorem 6.1 implies that the method of barycentric coordinates generates all possible valid embeddings of the graph in the plane, given the (convex) positions of the boundary. This theorem has been vastly used for the purpose of parameterizing 3D meshes [42]. In general, the method of barycentric coordinates can be formulated as the solution of a 2D vector Laplace equation on the interior vertices, an equation which can be numerically solved using a relaxation procedure. This implies that given a valid 2D manifold with a planar topology, the relaxation procedure that we use will converge towards a planar triangulation, therefore eliminating overlapping faces.

Once the p mappings have been generated, we simply apply the genetic algorithm to each configuration and select the best solution (the one with the best fitness).

6.5 Implementation Issues

The approach proposed in this chapter requires the frequent evaluation of binary edge-edge intersection operators. During the correction of a typical topological defect, containing approximately 50 defective vertices, the proposed method evaluates more than 10^7 edge-edge intersection tests. By the intrinsic nature of the quasi-homeomorphic mapping, which aims at minimizing regions with negative areas (i.e. negative Jacobian), topological defects correspond to extremely dense regions, with vertices potentially being as close as $10^{-5}mm$. At this scale, floating rounding errors are frequent and lead to "catastrophic" results: a non-detected intersection often leads to topologically inconsistent retessellations with incorrect Euler-numbers (i.e. surfaces with the incorrect topology).

In order to address this problem, we replace all float computations with *exact* calculations. To do so, we approximate the spherical location $\mathbf{x}$ of each vertex v with rational numbers $(\frac{p_x}{q_x}, \frac{p_y}{q_y}, \frac{p_z}{q_z})$, where the accuracy of the approximation can be taken as high as desired[6]. We note that we cannot ensure that each vertex lies exactly onto the sphere (i.e. $\|(\frac{p_x}{q_x}, \frac{p_y}{q_y}, \frac{p_z}{q_z})\| \neq R$), but this point is unimportant, since only the direction of the vector $\overrightarrow{\mathbf{0x}}$ matters.

Once every spherical vertex is approximated by a set of rational coordinates, the edge-edge intersection can be evaluated using only exact computations. Consequently, the result of a test becomes a robust binary value that is no longer sensitive to rounding errors.

More specifically, the intersection test consists of the successive evaluation of simple cross-product and dot-product operations. To illustrate this point, we consider four spherical vertices v^i, where $i \in \{1, 2, 3, 4\}$. Each vertex v^i has some spherical coordinates $\mathbf{x}^i = (\frac{p_x^i}{q_x^i}, \frac{p_y^i}{q_y^i}, \frac{p_z^i}{q_z^i})$ respectively. The edge $\overset{\frown}{\mathbf{x}^1\mathbf{x}^2}$ is intersecting the edge[7] $\overset{\frown}{\mathbf{x}^3\mathbf{x}^4}$ if and only if:
1) the two points $\mathbf{x}^3$ and $\mathbf{x}^4$ are located on both sides of the virtual spherical geodesic going through the two points $\mathbf{x}^1$ and $\mathbf{x}^2$,
2) reciprocally, if the two points $\mathbf{x}^1$ and $\mathbf{x}^2$ lies on both sides of the geodesics going through the points $\mathbf{x}^3$ and $\mathbf{x}^4$.

In evaluating these conditions, we denote by $\mathbf{t}^{ij}$ the vector $\overrightarrow{\mathbf{x}^i\mathbf{x}^j}$, and by $\mathbf{n}^{ij}$ the vector $\frac{\mathbf{x}^i+\mathbf{x}^j}{2}$. We note that, since the sphere is assumed to be centered at location $\mathbf{0} = (0, 0, 0)$, the vector $\mathbf{n}^{ij}$ is also the normal at location $\frac{\mathbf{x}^i+\mathbf{x}^j}{2}$. We have:

$$\mathbf{t}_k^{ij} = (\mathbf{x}_k^j - \mathbf{x}_k^i) \text{ and } \mathbf{n}_k^{ij} = \frac{\mathbf{x}_k^j + \mathbf{x}_k^i}{2}. \tag{6.16}$$

The first condition is then equivalent to:

$$[(\mathbf{n}^{12} \wedge \mathbf{t}^{12}) \cdot \overrightarrow{\mathbf{n}^{12}\mathbf{x}^3}] \times [(\mathbf{n}^{12} \wedge \mathbf{t}^{12}) \cdot \overrightarrow{\mathbf{n}^{12}\mathbf{x}^4}] < 0, \tag{6.17}$$

and the second one to:

$$[(\mathbf{n}^{12} \wedge \mathbf{t}^{34}) \cdot \overrightarrow{\mathbf{n}^{34}\mathbf{x}^1}] \times [(\mathbf{n}^{34} \wedge \mathbf{t}^{34}) \cdot \overrightarrow{\mathbf{n}^{12}\mathbf{x}^2}] < 0. \tag{6.18}$$

The approximation of the original spherical coordinates with a set of rational coordinates needs to be done only once at the end of the spherical mapping and before the detection of the topological defects. Also, we note that the same approach can be used for planar defects, where each vertex location in the 2D plane is approximated by a set of rational coordinates.

[6] The set of the rational numbers $\mathbb{Q}$ is dense in the set of the real numbers $\mathbb{R}$.

[7] Two vertices define a unique shortest geodesic onto the sphere. For clarity, we call this geodesic an edge.

6.6 Results and Discussion

Before reporting results of the proposed approach on synthetic and real datasets, we measure the goodness of our method relative to a random search algorithm. This is to verify that our approach actually improves the speed of convergence and that the genetic operations allow the generation of superior candidate retessellations.

6.6.1 Genetic versus Random Search

We compared our approach with a random search algorithm, in which random permutations of the edge ordering were iteratively generated. The graphs in Fig. 6-7 illustrate the strength of our approach on a real data example. The topological defect is shown in Fig. 6-2-a. For each method, the first candidate tessellation corresponded to the solution generated by the greedy approach proposed in [28] with its vertices added last being discarded (Sect. 6.3.2.A). Compared to a random search, the genetic search converges much faster (at least, second order magnitude). The genetic algorithm boosts the overall fitness of the population by keeping the best representations at each generation and producing new candidates using the elite population. In a few generations composed of a small number of chromosomes (20 chromosomes per generation in this example), the genetic search is able to produce new optimal retessellations (Fig. 6-2-f).

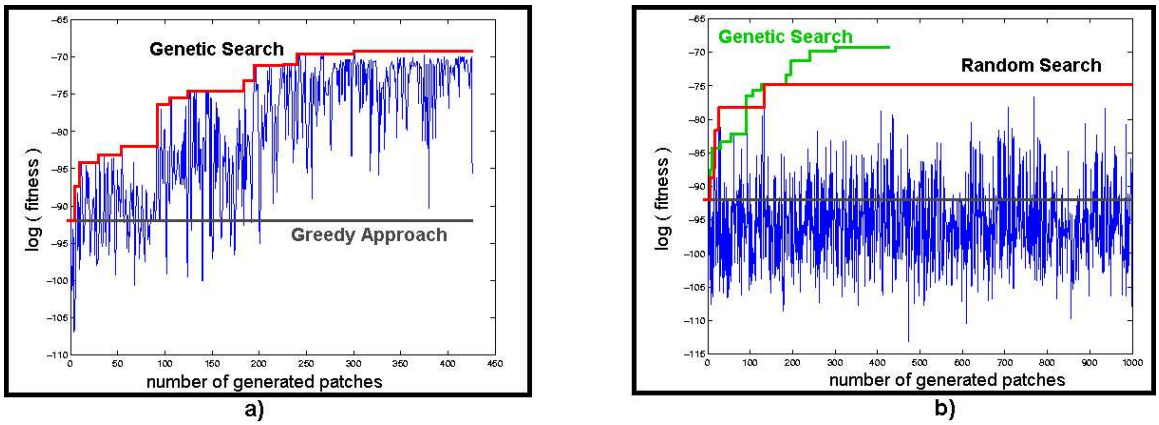

Figure 6-7: a) Evolution of the log of the fitness function during the genetic search. b) Evolution of the log fitness function during a random search. Note how the genetic search iteratively improves the average fitness of each generated chromosome, which, as a consequence, will be able to generate new optimal chromosomes. On the other hand, random retessellation rarely generates new optimal patches. In this defect, which was constituted of 183 vertices, even after 50000 random draw, the fitness function of the best randomly generated chromosome was still 5 order of magnitude below the best GA chromosome (generated as the 300^{th} offspring during the 15^{th} generation).

6.6.2 Application to Synthetic Data and Real Data

A - Synthetic Data

In order to validate the proposed method, we first generated surfaces containing simple topological defects (handles, holes). These data were used to explore the performance of the algorithm in terms of typical topological defects. The underlying MRI volumes were generated by adding white noise to the expected tissue intensities : gray and white intensity values were drawn from Gaussian distributions $G(\mu_g = 90, \sigma_g = 5.0)$ and $G(\mu_w = 110, \sigma_w = 5.0)$ respectively. Figure 6-8, top row, illustrates the behavior of the algorithm with regard to different MRI volumes, when the same topological defect has to be corrected (left: a simple handle). We note that traditional active contour models could not have generated the same results due to the amount of noise in the images and the presence of large local minima in the energy functional.

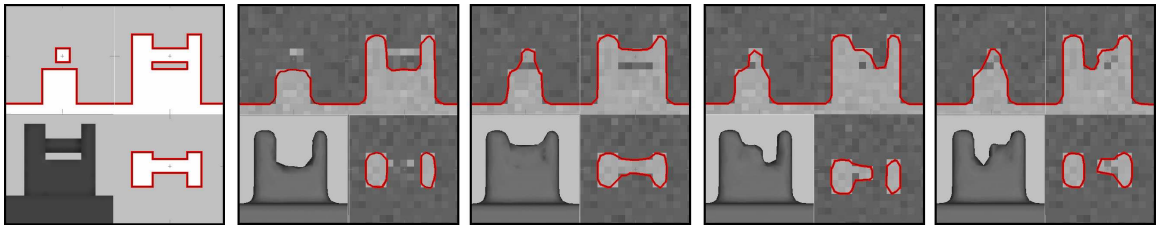

Figure 6-8: Results of our proposed approach on different phantom examples. The same topological defect (left: a small handle constituted of about 100 vertices) is corrected using different underlying MRI volumes. In each case, our approach generated an optimal configuration corresponding to the expected solution.

B - Real Data

We have applied our proposed approach to 43 real images. The dataset is composed of MRI volumes of different qualities, from different populations. Results were evaluated by experts to assess the correctness of the final corrections.

B.1 - Description of the Data Set

Validation data came from several data sets. They were a mix of pulse sequence (SPGR, MP-RAGE), scanner types (Siemens 1.5T, GE 1.5T) and pathology (normal control, schizophrenia and Alzeihmer's).

Seventeen scans were acquired in 2000/2001 using a Siemens Sonata system with the following parameters: TR: 7.25 ms; TE: 3.22 ms; TI: 600.00 ms; flip angle: 7.00j; 1.3-mm sections (resampled to 1 mm isotropic). This data set consists of 8 young (YNC), 7 elderly normal controls (ENC), and 2 Alzeihmer's (AD).

The second data set was acquired using a Siemens Vision system in 1994/1995 with the following parameters: TR: 9.70 ms; TE: 4.00 ms; TI: 621.00 ms; flip angle: 10.00j; 1.25 sections (resampled to 1-mm isotropic). Data comes from studies reported in Buckner et al. [11] and Logan et al. [58] and also later subjects imaged using the same anatomic protocol[8]. This data set consists of 6 Young Normal Control, 14 non-demented and 6 demented adults.

[8]We thank Randy Buckner and the Washington University Alzheimer's Disease Research Center for

B.2 - Discussion of the Results

The algorithm was able to generate correct solutions that the initial greedy approach [28] failed to produce. Methods that do not integrate statistical and geometric information will often fail to produce solutions comparable to the ones a trained operator would make. This is illustrated in Fig. 6-9, where valid solutions do not always correspond to minimal corrections (i.e. cutting the handle in the two examples of Fig. 6-9). Only general approaches that integrate additional information can lead to correct solutions. In addition, to our knowledge, our approach is the only one that has been proposed to explore the space of potential solutions in order to select the best correction to a topological defect.

To evaluate the quality of the corrections, we compute the average Hausdorff distance for each defect between automatically corrected surfaces (using our method) and manually corrected surfaces produced by a trained operator. The average Hausdorff distance is less than $0.2mm$.

An average cortical surface contains on the order of 50 topological defects, most of which are relatively small: most defects contain less than 50 vertices, and are corrected in a couple of minutes. Larger defects, with more than 100 vertices, correspond to a computation time of approximately 10 minutes. We note that the retessellation process is quadratic in the number of vertices contained within the convex hull of each defect. Consequently, a full brain is corrected in approximately 2 hours on a 1-GHz Pentium IV machine. More importantly, we note that the whole process could be parallelized, since each defect is independent of the other. Consequently, a full brain could be corrected in approximately 10 minutes.

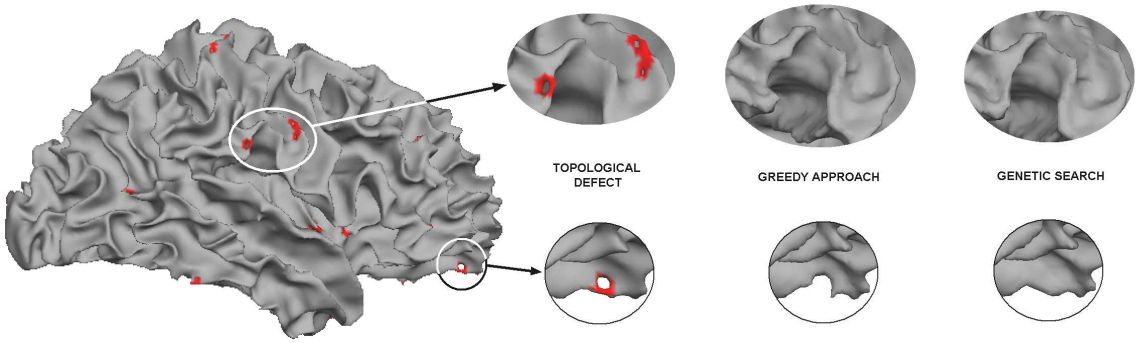

Figure 6-9: Topology correction of a cortical representation. The initial surface was constituted of 30 defects (Euler number $\chi = -58$). Compared to the greedy approach of Fischl et al. [28], which failed to find the correct solutions in many defects, our approach was able to generate valid solutions. This is illustrated on two examples, in which valid topological solutions do not correspond to minimal corrections.

Moreover, we note that the proposed method does not directly prevent the final surface from self-intersecting. Self-intersecting configurations typically have low fitness values and are naturally discarded during the genetic search. The self-intersecting constraint could be directly integrated into the retessellation process, but would drastically slow down the proposed approach. In our experience, final, corrected representations rarely intersect (less than one in ten thousand faces, which corresponds to approximately 1 defect per brain). To ensure that the solution generates a valid manifold, we check retrospectively that the final retessellation does not self-intersect. In the case of self-intersection, we re-apply the

providing the data set.

genetic algorithm with the additional constraint of generating only valid candidate patches. Self-intersecting patches are identified and discarded from the population.

Our method has been applied to 43 real images, i.e. 86 brain hemispheres that each had on average 50 defects. Only one single defect, the one used as an example throughout this section, has been identified to be inaccurately corrected, and necessitated the generation of several mappings. This seems to imply that, in most cases (i.e. in more than 99.9% of the cases) the genetic algorithm used in conjunction to the original spherical mapping produces correct topological corrections.

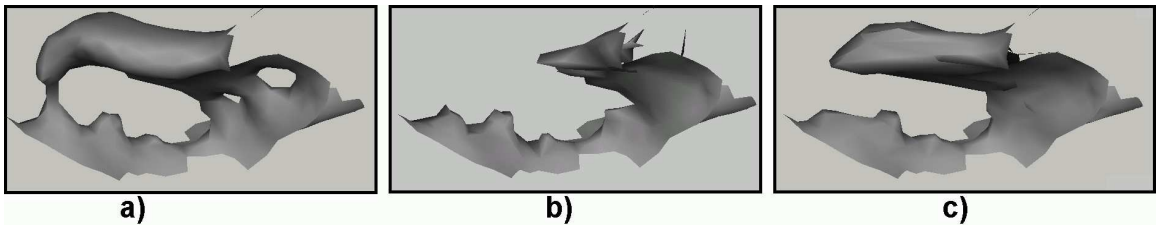

Figure 6-10: a) Original defect. b) Solution generated by the genetic algorithm from the initial quasi-homeomorphic mapping. c) Best solution generated from the mapping in Fig. 6-6-e.

In the case of an incorrect topological correction, the method proposed in Sect. 6.4 provides a simple solution by generating a few optimal retessellations corresponding to different mappings. Figure 6-10 illustrates the final solution that was generated using the mapping shown in Fig. 6-6-e.

Finally, we note that Thm. 6.1 opens new research directions for the direct integration of the mapping problem into the topology correction process. Applied to different (potentially random) relaxation procedures, vertices, whose position is not some convex combination of its neighbor's positions, could be identified and eliminated from the retessellation process. Future research would require the investigation of relaxation procedures that would limit the number of "non-convex" vertices, and we note that this is deeply related to finding how and where to "cut" handles in the defect.

6.7 Contributions of this Chapter

We have proposed an automated method to accurately correct the topology of cortical representations. Our approach integrates statistical and geometric information to select the optimal correction for each defect. In particular, we have developed a genetic algorithm that is specifically adapted to the retessellation problem. Iterative genetic operations generate candidate tessellations that are selected for reproduction based on their goodness of fit. The fitness of a retessellation is measured by the smoothness of the resulting surface and the local MRI intensity profile inside and outside the surface. The resulting procedure is completely adaptative and self-contained. During the search, defective vertices are identified and discarded while the optimal retessellation is constructed.

Given a quasi-homeomorphic mapping from the initial cortical surface onto the sphere, our method will be able to generate optimal solutions. For each defect, the space to be searched (i.e. the edge ordering) is dependent on the spherical location of the defective vertices. Some configurations of the quasi-homeomorphic mapping could lead to optimal but incorrect retessellations. In order to address this limitation, we have proposed to generate several quasi-homeomorphic mappings producing different spatial optimal retessellations. The final chosen retessellation is the one achieving the best fitness function.

To our knowledge, this approach is the only one that has been proposed to explore the space of potential solutions in order to optimally select the best correction to a topological defect. Some care was taken in order to ensure that no floating rounding errors occurs during the topology correction.

Finally, we note that the proposed approach is not restricted to spherical topologies, and that it can be used to correct the planar topology of any set of vertices.

A preliminary version of this work was presented at the conference *Information Processing in Medical Imaging* [86].

Chapter 7

Conclusion

This dissertation concerns the accurate segmentation of medical images under topological constraints. We have made a number of contribution to advance several aspects of the field of medical image segmentation and offer new research perspectives. On the theoretical level, we have introduced the digital concept of a *multisimple point* and derived necessary and sufficient characterizations. On the methodological level, we have developed a novel active contour framework for the evolution of level sets under topology control, the *genus preserving level sets*. Also, we have phrased the topology correction of segmentations into a Bayesian framework that naturally integrates statistical and geometrical information into the topology correction process. On the application level, we have proposed two algorithms for the retrospective topological correction of digital 3D images and 2D cortical surfaces. To our knowledge, no techniques had been previously introduced to naturally integrate additional information into the topology correction process, to explore the whole space of potential topological corrections, and to produce optimal solutions with respect to the underlying MRI intensity profile and the expected curvature in a rigorous manner.

The concept of *multisimple point* extends the notion of simple point that is often too restrictive for most applications. Using this criterion, new sets of digital deformations have been proposed to generalize the restrictive notion of homotopic deformation.

This concept has been used to design new segmentation algorithms that are much more flexible than previous methods. First, we have improved the control of topology changes with the level set method. We developed a new active contour framework for the evolution of level sets with preservation of the genus: the *genus-preserving level sets*. Our method offers a subtle topological control over the topology of the level sets, and constitutes a trade-off between traditional level sets and topology-preserving level sets.

Also, the concept of multisimple point has been used to develop a method for automatically correcting the spherical topology of any 3D binary segmentation under any digital connectivity. In contrast to existing procedures that assume specific initial segmentation (e.g. full connectivity, no cavities, etc) and are designed for a particular task (e.g. cortical representation), no assumption is made on the initial image, and spherical topology is achieved under any choice of digital connectivity. Also, our topology correction algorithm is nested in the theory of Bayesian parameter estimation, which allows the integration of statistical information into the topology correction.

Finally, we have introduced a genetic algorithm for the correction of the topology of cortical surfaces. Unlike existing approaches, our method is able to generate several potential

topological corrections and to select the maximum-a-posteriori retessellation in a Bayesian framework. Our approach integrates statistical, geometrical and shape information into the correction process, providing optimal solutions with regard to the MRI intensity profile and the expected curvature. The resulting procedure is completely adaptative and self-contained.

The methods developed in this dissertation have been validated using synthetic and real data. Some experiments on synthetic images and real MR images have demonstrated the advantages of the genus-preserving method and have illustrated some potential applications that could greatly benefit from our approach. Our algorithm for the topology correction of 3D binary images has been successfully applied to subcortical segmentations and white matter segmentations. Applied to synthetic and real data, our genetic algorithm generated optimal topological corrections with only a few iterations.

Finally, some potential directions for future research have been highlighted in each chapter.

The concept of multisimple point can benefit several research areas that rely on digital theory: image segmentation, computer graphics, digital image processing, and so on. For instance, this concept could be used to design new thinning algorithms that would preserve some specific characteristics of a digital object. Also, multisimple points could help characterize some invariant properties of digital objects, such as the Betti numbers.

In Chapter 4, we introduce a genus-preserving level set framework, and we propose some potential applications that could benefit from our approach. In addition, this level set framework could be integrated into our digital topology correction method described in Chapter 5. The use of curvature information in the segmentation process would greatly reduce the sensitivity to noise and would improve the location and correction of the topological defects. This approach would certainly reduce the impact of early wrong decisions during the graph analysis.

Also, it is in our interest to integrate the generation of several random mapping configurations directly into the retessellation process, ensuring search over the whole space of potential retessellation. Particularly, Thm. 6.1 opens new research directions for the direct integration of the mapping problem into the topology correction process. Applied to different (potentially random) relaxation procedures, vertices, whose position is not some convex combination of its neighbor's positions, could be identified and eliminated from the retessellation process.

In conclusion, in this dissertation, we have improved the theoretical tools applicable to the segmentation of images under topological constraints, proposed novel methodologies for image segmentation, and developed well-founded algorithms to achieve accurate segmentation of medical images under topological constraints. Additionally, we have presented the reliability and applicability of these methods as compared to existing techniques in the field.

Appendices

Appendix A

Proofs

Multisimple point x relative to X
A point $\mathbf{x} \in X$ is said to be multisimple relative to X if and only if

$$\begin{cases} T_{\overline{n}}(\mathbf{x}, \overline{X}) & = & 1 \\ \forall\ C \in C_n(\mathbf{x}, X),\ T_n(\mathbf{x}, C) & = & T_{\overline{n}}(\mathbf{x}, \overline{C}) = 1 \end{cases} \tag{A.1}$$

Proof
In order to verify that the concept of multisimple point does not introduce any topological defects in the volume, we need to show that the addition or deletion of a multisimple point $\mathbf{x}$ *does not introduce any holes or cavities in any of the connected components* $C \in C_n(\mathbf{x}, X)$ *or* $\overline{X}$*, where* $X = \{\mathbf{x}\} \bigcup_i C_i$.
A simple point $\mathbf{x} \in X$*, characterized by* $T_n(\mathbf{x}, X) = T_{\overline{n}}(\mathbf{x}, \overline{X}) = 1$*, can be removed without changing the topology of the image. This implies a one-to-one correspondence between the connected components, the holes of* X *and* $\overline{X}$ *and the connected components, the holes of* $X \setminus \{\mathbf{x}\}$ *and* $\overline{X} \bigcup \{\mathbf{x}\}$*, the* n*-connectivity being used for* X *and the* $\overline{n}$*-connectivity being used for* $\overline{X}$*. Using this property, the proof can easily completed as follow.*

- **Holes**
  Since the connected components $C_i \in C_n(\mathbf{x}, X)$ *are not adjacent, any simple closed path in* $X = \{\mathbf{x}\} \bigcup_i C_i$ *is strictly contained in* $C_j \bigcup \{\mathbf{x}\}$ *for one of the connected components* C_j*. The multisimple criterion, which ensures that the point* $\mathbf{x}$ *is simple relative to the component* C_j*, guarantees that the addition or deletion of* $\mathbf{x}$ *does not create any* n*-holes in the process. Similarly, any simple closed* $\overline{n}$*-path in the background component* $\overline{X} \bigcup \{\mathbf{x}\}$ *is contained in* $\overline{C}$ *for any of the components* $C \in C_n(\mathbf{x}, X)$*. Therefore, Eq. A.1 ensures that no* $\overline{n}$*-holes are created in* $\overline{X}$*.*

- **Cavities**
  The point $\mathbf{x}$ *is simple relative to each component* $C \in C_n(\mathbf{x}, X)$*. As a consequence, its addition or deletion does not introduce any cavities in any of the components* C *or its complement* $\overline{C}$*. Finally, we note that no cavities other than the ones formed by the* n*-connected components of* $C_n(\mathbf{x}, X)$ *are generated in* $\overline{X}$*. If this were the case, since the components of* $C_n(\mathbf{x}, X)$ *are not adjacent, this would mean that the cavity would have been created in one of the components* $C \in C_n(\mathbf{x}, X)$*, which would contradict what we have just proven.*

Multisimple point x relative to $\overline{X}$

A point $\mathbf{x}$ is said to be multisimple relative to $\overline{X}$ if and only if:

$$\left\{ \begin{array}{rcl} T_n(\mathbf{x}, X) & = & 1 \\ \forall\ C \in C_{\overline{n}}(\mathbf{x}, \overline{X}),\ \ T_n(\mathbf{x}, \overline{C}) & = & T_{\overline{n}}(\mathbf{x}, C) = 1 \end{array} \right. \tag{A.2}$$

Proof

The proof is the same as the one of Eq. A.1, where every digital topological notion is replaced by its dual notion: $X \rightarrow \overline{X}$, $n \rightarrow \overline{n}$,...

$$T_n^+(\mathbf{x}, X) \leq T_n(\mathbf{x}, X) \tag{A.3}$$

$$T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X) \ \Leftrightarrow \ \ \forall\ C \ \ \in C_n(\mathbf{x}, X) \ \ \ T_n(\mathbf{x}, C) = 1 \tag{A.4}$$

Proof

Topological numbers characterize potential merges or splits, while topological numbers control any kind of topological changes. In addition to splits and merges, the topological numbers record the potential formations of handles. For a given connected component $C \in C_n(\mathbf{x}, X)$, $T_n(\mathbf{x}, C) > 1$ implies that the addition of the point $\mathbf{x}$ to X results in the generation of, at least, one handle in the connected component C.

First, we have $\forall C_i \in C_n(\mathbf{x}, X)$, $T_n(\mathbf{x}, C_i) \geq 1$. We note that we cannot have $T_n(\mathbf{x}, C_i) = 0$, because C_i is adjacent to $\mathbf{x}$.
Also, we note that, since the components $C_i \in C_n(\mathbf{x}, X)$ are not adjacent (i.e. $\forall (i, j)$ s.t. $i \neq j \ \ C_i \bigcap C_j = \{0\}$), we have the following set equality:

$$N_n^{\cdot}(\mathbf{x}, X) = \bigcup_i N_n^{\cdot}(\mathbf{x}, C_i),$$

where the union is disjoint. Therefore, we have the following equality:

$$T_n(\mathbf{x}, X) = |N_n^{\cdot}(\mathbf{x}, X)| = \sum_i |N_n^{\cdot}(\mathbf{x}, C_i)|.$$

Finally, we can derive the above equations:

$$T_n^+(\mathbf{x}, X) = |C_n(\mathbf{x}, X)| \leq \sum_i |N_n^{\cdot}(\mathbf{x}, C_i)| = |N_n^{\cdot}(\mathbf{x}, X)| = T_n(\mathbf{x}, X),$$

since $\forall i \ |N_n^{\cdot}(\mathbf{x}, C_i)| \geq 1$. We have the equality $T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X)$ if and only all the components C_i of $C_n(\mathbf{x}, X)$ verify $|N_n^{\cdot}(\mathbf{x}, C_i)| = 1$ (i.e. $T_n(\mathbf{x}, C_i) = 1$). We note that we

also have : $T_n^+(\mathbf{x}, X) < T_n(\mathbf{x}, X) \Leftrightarrow \exists C \in C_n(\mathbf{x}, X)$ such that $\mathrm{T_n}(\mathbf{x}, \mathrm{C}) > 1$.

Multisimple point x for (X,$\overline{X}$)
A point $\mathbf{x}$ is a multisimple for (X,$\overline{X}$) if and only if

$$\begin{cases} T_n^+(\mathbf{x}, X) &= T_n(\mathbf{x}, X) \\ T_{\overline{n}}^+(\mathbf{x}, X) &= T_{\overline{n}}(\mathbf{x}, X) \end{cases} \tag{A.5}$$

Proof

Given a connected component C_i of $C_n(\mathbf{x}, X)$, the addition of the point $\mathbf{x}$ to the digital object C_i will not generate an n-handle if and only if $T_n(\mathbf{x}, C_i) = 1$. Since all components of $C_n(\mathbf{x}, X)$ are non-adjacent, no n-handle is generated in X if and only if we have $T_n(\mathbf{x}, C_i) = 1$ for all components C_i. Using Eq. A.4, this is equivalent to the condition $T_n^+(\mathbf{x}, X) = T_n(\mathbf{x}, X)$. Similarly, we have that the deletion of $\mathbf{x}$ from X will not generate any $\overline{n}$-handle in $\overline{X}$ if and only if we have $T_{\overline{n}}^+(\mathbf{x}, \overline{X}) = T_{\overline{n}}(\mathbf{x}, \overline{X})$. Consequently, the deletion or addition of $\mathbf{x}$ will not generate or delete any handle in the volume (i.e. $\mathbf{x}$ is a multisimple point) if and only if Eq. A.5 is verified.

Appendix B

Publications of the Author

- **Journal Articles**

1. B. Fischl, A. van der Krouwe, C. Destrieux, E. Halgren, F. Ségonne, D.H. Salat, E. Busa, L.J. Seidman, J. Goldstein, D. Kennedy, V. Caviness, N. Makris, B. Rosen, A.M. Dale, "*Automatically Parcellating the Human Cerebral Cortex*," Cerebral Cortex, 2004, 14:11-22
2. L. M. Angelone, A. Potthast, F. Ségonne, S. Iwaki, J. Belliveau, G. Bonmassar, "*Metallic Electrodes and Leads in Simulation EEG-MRI: Specific Absoption Rate (SNR) Simulation Studies*," Bioelectromagnetics 25 (4), May 2004.
3. F. Ségonne, A.M. Dale, E. Busa, M. Glessner, D. Salat, H. K. Hahn, and B. Fischl. "*A Hybrid Approach to the Skull Stripping Problem in MRI*," NeuroImage, vol. 22, pp. 1160-1075, 2004.
5. B. Fischl, D.H. Salat, A. van der Kouwe, N. Makris, F. Ségonne, A.M. Dale,: "*Sequence-Independant Segmentation of Magnetic Resonance Images*," NeuroImage, 23(supplement 1):S46-S55, 2004. Note: Special Issue: Mathematics in Brain Imaging. Edited by P.M. Thompson, M.I. Miller, T. Ratnanather, R.A. Poldrack and T.E. Nichols.
6. Olivier Faugeras, Geoffray Adde, Guillaume Charpiat, Christophe Chefd'Hotel, Maureen Clerc, Thomas Deneux, Rachid Deriche, Gerardo Hermosillo, Renaud Keriven, Pierre Kornprobst, Jan Kybic, Christophe Lenglet, Lucero Lopez-Perez, Tho Papadopoulo, Jean-Philippe Pons, Florent Ségonne, Bertrand Thirion, David Tschumperl, Thierry Viville, Nicolas Wotawa, "*Variational, geometric and statistical methods for modeling brain anatomy and function*," NeuroImage, 23(supplement 1):S46-S55, 2004. Note: Special Issue: Mathematics in Brain Imaging. Edited by P.M. Thompson, M.I. Miller, T. Ratnanather, R.A. Poldrack and T.E. Nichols.
7. C. Fennema-Notestine, I.B. Ozyurt, C.P. Clark, S. Morris, A. Bischoff-Grethe, M.W. Bondi, T.L. Jernigan, B. Fischl, F. Ségonne, D.W. Shattuck, R.M. Leahy, D.E. Rex, A.W. Toga, K.H. Zou, and G.G. Brown, "*Quantitative Evaluation of Automated Skull-Stripping Methods Applied to Contemporary and Legacy Images: Effects of Diagnosis, Bias Correction, and Slice Location*," Human Brain Mapping. 2005 Jun 28.
8. R. Desikan, F. Ségonne et al. "*A Computer Generated Labeling System for Subdividing the Human Cerebral Cortex on MRI Scans into Gyral Based Regions of Interest*," submitted to NeuroImage.

- **Conference Articles**

1. F. Ségonne, E. Grimson, and B. Fischl, "*Topological Correction of Subcortical Segmentation*," accepted at MICCAI, Montreal Nov. 15-18 2003.
2. F. Ségonne, E. Grimson, and B. Fischl, "*A Genetic Algorithm for the Topology Correction of Cortical Surfaces*," accepted at I.P.M.I 2005.
3. F. Ségonne, J.P. Pons, E. Grimson and B. Fischl, "*A Novel Level Set Framework for the Segmentation of Medical Images Under Topology Control*," in-review.

- **Technical Reports**

1. F. Ségonne, J.P. Pons, B. Fischl, and E. Grimson, " *A Novel Active Contour Framework. Muti-Component Level Set Evolution Under Topology Control*," CSAIL memo AIM-2005-020.

- **Thesis**

1. F. Ségonne "*Unsupervised Skull Stripping in MRI*," Masters Thesis, co-supervised by O. Faugeras and B. Fischl, MIT AI Lab, June 2002.

- **Abstracts**

1. F. Ségonne et al.: "*A Hybrid Approach to the Skull Stripping Problem in MRI*," HBM, June 2001.
2. F. Ségonne et al.: "*Skull-Stripping: A Comparison of Three Automated Algorithms*," HBM, June 2-6, 2002.
3. D. H. Salat et al.: "*Age-related changes in T1 relaxation times across the surface of the cortex*," HBM, June 2-6, 2002.
4. G. Bonmassar, L. Angelone, F. Ségonne, A. Potthast, B. Fischl, L. Wald, and J. Belliveau, "*SAR Computations in a Realistic and High-Resolution Model of the Head with EEG Electrodes in Place*," Presented at the 8th International COnference on Functional Mapping of the Hman Brain, June 2-6, 2002, Sendai, JAPAN. Available on CD-Rom in NeuroImage, Vol. 16, No.2.
5. L. M. Angelone, A. Potthast, S. Iwaki, F. Ségonne, L. Wald, B. Fischl, J. W. Belliveau, G. Bonmassar, " *Effects of EEG electrodes and RF coils in simultaneous EEG - MRI recording, with a HR model*," ISMRM Eleventh Scientific Meeting and Exhibition, Toronto, Canada.
6. L.M. Angelone, A. Potthast, S. Iwaki, F. Ségonne, L. Wald, B. Fischl, J.W. Belliveau, G. Bonmassar. (2003). "*Specific Absorption Rate changes in simultaneous EEG - MRI recording*," Presented at the XVII MGH Scientific Advisory Committee Symposium, February 5 th , 2003, Boston, USA.
7. L.M. Angelone, A. Potthast., S. Iwaki, F. Ségonne, J.W. Belliveau, G. Bonmassar, "*Simulation study suggests unsafe levels of Specific Absorption Rate during simultaneous EEG-MRI recording*," Presented at the 57th Annual Meeting American Epilepsy Society, December 5 - 10, 2003 Boston, MA Available on Epilepsia, Vol. 44, Issue s9, p. 255.

8. P. Yu, F. Ségonne, X. Han, B. Fischl, "*Shape Analysis of Neuroanatomical Structures Based on Spherical Wavelets*," HBM 2005, Toronto.

Bibliography

[1] D. Adalsteinsson and J.A. Sethian. A Fast Level Set Method for Propagating Interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.

[2] J.A. Baerentzen and H. Aanaes. Signed distance computation using the angle weighted pseudonormal. *IEEE, Transaction on Visualization and Computer Graphics*, 11(3):243–253, 2005.

[3] J.E. Baker. Adaptive selection methods for genetic algorithms. In *First Int'l Conf. on Genetic Algorithms and their Applications*, Erlbaum, 1985.

[4] E. Bardinet, L.D. Cohen, and N. Ayache. Tracking and motion analysis of the left ventricle with deformable superquadrics. *Medical Image Analysis*, 1(2):129–149, 1996.

[5] E. Bardinet, L.D. Cohen, and N. Ayache. A parametric deformable model to fit unstructured 3D data. *Computer Vision and Image Understanding*, 71(1):39–54, 1998.

[6] P.L. Bazin and Pham D.L. Topology preserving tissue classification with fast marching and topology templates. *IPMI*, pages 234–245, 2005.

[7] G. Bertrand. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Pattern Recognition Letters*, 15(10):1003–1011, 1994.

[8] G. Bertrand. A boolean characterization of three-dimensional simple points. *Pattern Recognition Letters*, 17:115–124, 1996.

[9] J.C. Bezdek and L.P. Hall, L.O.and Clarke. Review of mr image segmentation techniques using pattern recognition. *Medical Physics*, 20:1033–1048, 1993.

[10] S. Bischoff and L. Kobbelt. Isosurface reconstruction with topology control. *Pacific Graphics Proceedings*, pages 246–255, 2002.

[11] R.L. Buckner, A.Z. Snyder, A.L. Sanders, M.E. Raichle, and J.C. Morris. Functional brain imaging of young, nondemented, and demented older adults. *J. Cogn. Neurosci.*, 12(2):24–34, 2000.

[12] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *The International Journal of Computer Vision*, 22(1):61–79, 1997.

[13] G.E. Christense, R.D. Rabbitt, and M.I. Miller. 3d brain mapping using a deformable neuroanatomy. *Phys. Med. Biol.*, 39:609–618, 1994.

[14] L.D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *pami*, 15(11):1131–1147, November 1993.

[15] M.G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton–Jacobi equations. *Trans. AMS*, 277:1–43, 1983.

[16] A.D. Dale, B. Fischl, and Sereno M.I. Cortical surface-based analysis i: Segmentation and surface reconstruction. *NeuroImage*, 9:179–194, 1999.

[17] Anders M. Dale and Martin I. Sereno. Improved localization of cortical activity by combining eeg and meg with mri cortical surface reconstruction: A linear approach. *Journal of Cognitive Neuroscience*, 5(2):162–176, 1993.

[18] C. Davatzikos and R.N. Bryan. Using a deformable surface model to obtain a shape representation of the cortex. *IEEE TMI*, 15:758–795, 1996.

[19] K.E. De Jong. *An analysis of the behavior of a class of genetic adaptive systems.* University of Michigan Press, Ann Arbor, 1975.

[20] H. Delingette. General object reconstruction based on simplex meshes. *The International Journal of Computer Vision*, 32(2):111–146, 1999.

[21] H. Delingette and J. Montagnat. Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding*, 83(2):140–171, 2001.

[22] R. Desikan, F. Ségonne, and etal. A computer generated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. *Human Brain Mapping*, 2005.

[23] H.A. Drury, D.C. Van Essen, C.H. Anderson, C.W. Lee, T.A. Coogan, and J.W. Lewis. Computerized mappings of the cerebral cortex: A multiresolution flattening method and a surface-based coordinate system. *J. Cogn. Neurosci*, 8(1):1–28, 1996.

[24] Y. Duan, L. Yang, H. Qin, and D. Samaras. Shape reconstruction from 3D and 2D data using PDE-based deformable surfaces. In *European Conference on Computer Vision*, volume 3, pages 238–251, 2004.

[25] O. Faugeras and R. Keriven. Variational principles, surface evolution, PDE's, level set methods and the stereo problem. *IEEE Transactions on Image Processing*, 7(3):336–344, 1998.

[26] Olivier Faugeras, Geoffray Adde, Guillaume Charpiat, Christophe Chefd'Hotel, Maureen Clerc, Thomas Deneux, Rachid Deriche, Gerardo Hermosillo, Renaud Keriven, Pierre Kornprobst, Jan Kybic, Christophe Lenglet, Lucero Lopez-Perez, Théo Papadopoulo, Jean-Philippe Pons, Florent Ségonne, Bertrand Thirion, David Tschumperlé, Thierry Viéville, and Nicolas Wotawa. Variational, geometric, and statistical methods for modeling brain anatomy and function. *Neuroimage*, 23S1:S46–S55, 2004. Special issue: Mathematics in Brain Imaging - Edited by P.M. Thompson, M.I. Miller, T. Ratnanather, R.A. Poldrack and T.E. Nichols.

[27] B. Fischl and A.M. Dale. Measuring the thickness of the human cerebral cortex from magnetic resonnace images. *Proceedings of the National Academy of Sciences*, 97:11044–11049, 2000.

[28] B. Fischl, A. Liu, and A.M. Dale. Automated manifold surgery: Constructing geometrically accurate and topologically correct models of the human cerebral cortex. *IEEE TMI*, 20:70–80, 2001.

[29] B. Fischl, D.H. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. Van der Kouwe, R. Killinay, D. Kennedy, S. Klaveness, A. Montillo, N. Makris, B. Rosen, and A.M. Dale. Whole brain segmentation: Automated labeling of neuroanatomical strucutres in the human brain. *Neuron*, 33:341–355, 2002.

[30] B. Fischl, M.I. Sereno, and A.M. Dale. Cortical surface-based analysis II : Inflation, flattening, and a surface-based coordinate system. *Neuroimage*, 9(2):195–207, 1999.

[31] B. Fischl, M.I. Sereno, R.B.H. Tootell, and A.M. Dale. High-resolution inter-subject averaging and a coordinate system for the cortical surface. *Human Brain Mapping*, 8:272–284, 1999.

[32] B. Fischl, A. Van der Kouwe, C. Destrieux, E. Halgren, F. Ségonne, D.H. Salat, E. Busa, L.J. Seidman, J. Goldstein, D. Kennedy, V. Caviness, N. Makris, B. Rosen, and A.M. Dale. Automatically parcellating the human cerebral cortex. *Cerebral Cortex*, 14:11–22, 2004.

[33] M.S. Floater. One-to-one piecewise linear mappings over triangulations. *Mathematics of Computation*, 2:685–696, 2003.

[34] S. Forest. Emergent comutation: Self-organizing, collective, and cooperative phenomena in natural and artificial computing networks. *Physica D*, 42:1–11, 1990.

[35] Pascal Fua and Yves G. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *The International Journal of Computer Vision*, 16(1):35–56, September 1995.

[36] J.P. Gambotto and O. Monga. a parallel and hierarchical algorithm for region growing. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*. IEEE, June 1985. San Fransisco, CA.

[37] T.W. Gamelin and R.E. Greene. *Introduction to Topology: Second Edition*. 1999.

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

[39] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Cortex segmentation: A fast variational geometric approach. *IEEE TMI*, 21(2):1544–1551, 2002.

[40] B. Goldlücke and M. Magnor. Space-time isosurface evolution for temporally coherent 3D reconstruction. In *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 350–355, 2004.

[41] J. Gomes and O. Faugeras. Reconciling distance functions and level sets. *Journal of Visual Communication and Image Representation*, 11:209–223, 2000.

[42] C Gotsman, X Gu, and A. Sheffer. Fundamentals of spherical parameterization for 3d meshes. *ACM Transactions on Graphics*, 22, 2003.

[43] I. Guskov and Z. Wood. Topological noise removal. *Graphics I proceedings*, pages 19–26, 2001.

[44] X. Han, C. Xu, U. Braga-Neto, and J.L. Prince. Topology correction in brain cortex segmentation using a multiscale, graph-based approach. *IEEE TMI*, 21(2):109–121, 2001.

[45] X. Han, C. Xu, and J.L. Prince. A 2d moving grid geometric deformable model. *IEEE Conf. on Comp. Vis. Patt. Recog.*, pages 153–160, 2003.

[46] X. Han, C. Xu, and J.L. Prince. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):755–768, 2003.

[47] Xiao Han. *Anatomically Consistent Segmentation of Medical Imagery Using a Level Set Method and Digital Topology*. PhD thesis, Baltimore, Maryland, October 2003.

[48] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[49] J. H. Holland. *Adaptation in Naturel and Artificial Systems*. PhD thesis, university of Michigan, 1975.

[50] S. Jaume. *Topology Simplification Algorithm for the Segmentation of Medical Images*. PhD thesis, University of Louvain (Belgium), Feb 2004.

[51] H. Jin, S. Soatto, and A.J. Yezzi. Multi-view stereo beyond Lambert. In *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 171–178, 2003.

[52] T. Kapur, E.L. Grimson, W.M. III Wells, and R. Kikinis. Segmentation of brain tissue from magnetic resonance images. *Medical Image Analysis*, 1(2):109–127, 1996.

[53] B. Karaç and C. Davatzikos. Topology preservation and regularity in estimated deformation fields. *IPMI*, pages 426–437, 2003.

[54] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *First International Conference on Computer Vision*, pages 259–268, London, June 1987.

[55] R. Kikinis and etal. Temporal love sulco-gyral pattern anomalies in schizophrenia: An in vivo mr three-dimensional surface rendering study. *Neuroscience Letters*, 182:7–12, 1994.

[56] N. Kriegeskorte and R. Goeble. An efficient algorithm for topologically segmentation of the cortical sheet in anatomical mr volumes. *NeuroImage*, 14:329–346, 2001.

[57] J.-O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine 3D surface extraction. *Medical Image Analysis*, 3(2):187–207, 1999.

[58] J.M. Logan, A.L. Sanders, A.Z. Snyder, J.C. Morris, and R.L. Buckner. Under-recruitment and nonselective recruitment: dissociable neural mechanisms associated with agin. *Neuron*, 33:827–840, 2002.

[59] W.E. Lorensen and H.E. Cline. Marching cubes: A high-resolution 3D surface reconstruction algorithm. *ACM Computer Graphics*, 21(4):163–170, 1987.

[60] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In M.C. Stone, editor, *Proceedings of the SIGGRAPH*, pages 163–169, Anaheim, CA, July 1987. in Computer Graphics, Volume 21, Number 4.

[61] D. MacDonald, N. Kabani, D. Avis, and A.C. Evens. Automated 3d extraction of inner and outer surfaces of cerebral cortex from mri. *NeuroImage*, 12:340–356, 2000.

[62] 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 of Mathematical Imaging and Vision*, 5:297–318, 1995.

[63] MathWorld. The web's most extensive mathematical resource.

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

[65] T. McInerney and D. Terzopolos. Deformable models in medical image analysis: A survey, 1999 update. *Handbook of Medical Image Processing*, 1999.

[66] T. McInerney and D. Terzopoulos. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis. *Computerized Medical Imaging and Graphics*, 19(1):69–83, 1995.

[67] Tim McInerney and Demetri Terzopoulos. T-snakes: Topology adaptive snakes. *Medical Image Analysis*, 4:73–91, 2000.

[68] D.N. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):580–591, 1993.

[69] M. Mitchell. *An introduction to genetic algorithms*. MIT Press, Cambridge, 1996.

[70] J. Munkres. *Topology: A First Course*. 1975.

[71] Barrett O'Neill. *Elementary Differential Geometry*. June 1997.

[72] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.

[73] N. Paragios and R. Deriche. Geodesic active regions and level set methods for motion estimation and tracking. *Computer Vision and Image Understanding*, 97(3):259–282, 2005.

[74] D.L. Pham and J.L. Prince. An automated technique for statistical characterization of brain tissues in magnetic resonance imaging. *International Journal of Pattern Recognition and Artificial Intelligence*, 11(8):1189–1211, 1997.

[75] D.L. Pham, C. Xu, and J.L. Prince. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2:315–337, 2000.

[76] J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras. How to deal with point correspondences and tangential velocities in the level set framework. In *International Conference on Computer Vision*, volume 2, pages 894–899, 2003.

[77] Jean-Philippe Pons. *Methodological and Applied Contributions to the Deformable Models Framework*. PhD dissertation, Ecole Nationale des Ponts et Chaussées, November 18 2005.

[78] F. Poupon, J.-F. Mangin, D. Hasboun, C. Poupon, I. Magnin, and V. Frouin. Multi-object deformable templates dedicated to the segmentation of brain deep structures. *LNCS*, 1496:1134–1143, 1998.

[79] J.C. Rajapakse. Statistical approach to segmentation of simgle-channel cerebral mr images. *IEEE TMI*, 16(2), April 1997.

[80] G. Rawlins. *Foundations of Genetic Algorithms*. Morgan Kaufmann, 1991.

[81] Richter-Gebert. Realization spaces of polyutopes. *Lecture Notes in Math - Springer*, 1643, 1996.

[82] P. Saint-Marc, H. Rom, and G.G. Medioni. B-spline contour representation and symmetry detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1191–1197, 1993.

[83] D.H. Salat, R.L. Buckner, A.Z. Snyder, D.N. Greve, R.S. Desikan, E. Busa, J.C. Morris, A.M. Dale, and B. Fischl. Thinning of the cerebral cortex in aging. *Cerebral Cortex*, 14(7):721–730, July 2004.

[84] F. Ségonne, A.M. Dale, E. Busa, M. Glessner, D. Salat, H.K. Hahn, and B. Fischl. A hybrid approach to the skull stripping problem in MRI. *NeuroImage*, 22(3):1060–1075, 2004.

[85] F. Ségonne, E. Grimson, and B. Fischl. Topological correction of subcortical segmentation. In *Proceedings of Medical Image Computing and Computer-Assited Intervention*, volume 2879, pages 695–702, 2003.

[86] F. Ségonne, E. Grimson, and B. Fischl. A genetic algorithm for the topology correction of cortical surfaces. In *Proceedings of Information Processing in Medical Imaging, LNCS*, volume 3565, pages 393–405, 2005.

[87] F. Ségonne, J.-P. Pons, F. Fischl, and E. Grimson. A novel active contour framework: Multi-component level set evolution under topology control. AI memos AIM-2005-020, CSAIL, MIT, 2005.

[88] F. Ségonne, J.-P. Pons, E. Grimson, and B. Fischl. A novel level set framework for the segmentation of medical images under topology control. In *Workshop on Computer Vision for Biomedical Image Applications: Current Techniques and Future Trends*, 2005. to appear.

[89] D.W. Shattuck and R.M. Leahy. Automated graph based analysis and correction of cortical volume topology. *IEEE TMI*, 20(11):1167–1177, 2001.

[90] L.H. Staib and J.S. Duncan. Boundary finding with parametrically deformable models. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 14(11):1061–1075, November 1992.

[91] J.L. Tanabe, D. Amend, N. Schuff, V. DiSclafani, F. Ezekiel, D. Norman, G. Fein, and M.W. Wiener. Tissue segmentation of the brain in alzeihmer's disease. *J. Neuroradiol.*, 18:115–123, 1997.

[92] X. Tao, X. Han, M.E. Rettmann, J.L. Prince, and C. Davatzikos. Statistical study on cortical sulci of human brains. *Proceedings of Inf. Proc. in Med. Imag.*, pages 37–49, 2001.

[93] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D.J. Kriegman. Parameterized families of polynomials for bounded algebraic curve and surface fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):287–303, March 1994.

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

[95] P.M. Thompson, D. MacDonald, M.S. Mega, C.J. Holmes, A.C. Evans, and A.W. Toga. Detection and mapping of abnormal brain structure with a probabilistic atlas of cortical surfaces. *J. Comput. Assist. Tomogr.*, 21(4):567–581, 1998.

[96] P.M. Thompson, J. Moussai, S. Zohoori, A. Goldkorn, A.A. Khan, M.S. Mega, G.W. Small, J.L. Cummings, and A.W. Toga. Cortical variability and asymmetry in normal aging and alzheimer's disease. *Cerebral Cortex*, 8(6):492–509, 1998.

[97] W.T. Tutte. How to draw a graph. *Proc. London Math. Soc.*, 13(3):743–768, 1963.

[98] M. Vaillant and C. Davatzikos. Hierarchical matching of cortical features for deformable brain image registration. *Proceedings of Inf. Proc. in Med. Imag.*, pages 182–195, 1999.

[99] D.C Van Essen and H. Drury. Structural and functional analyses of human cerebral cortex using a surface-based atlas. *Journal of Neuroscience*, 17(18):7079–7102, 1997.

[100] W.M. III Wells and et al. 1995. *Proceedings of First International conference on computer vision, virtual reality and robotics in medicine, CVRMed'95, Nice, France*, Adaptative Segmentation of MRI data.

[101] L.D. Whitley. An executable model of a simple genetic algorithm. *in Foundations of Genetic Algorithms 2*, 1993.

[102] L.D. Whitley and M.D. Vose. *Foundations of Genetic Algorithms 3*. Morgan Kaufmann, 1995.

[103] C. Xu, D.L. Pham, and J.L. Prince. Medical image segmentation using deformable models. *Handbook of Medical Imaging - Medical Image Processing and Analysis*, 2:129–174, 2000.

[104] C. Xu, D.L. Pham, M.E. Rettmann, D.N. Yu, and J.L. Prince. Reconstruction of the human cerevral cortex from magnetic resonance images. *IEEE TMI*, 18:467–480, 1999.

[105] A.J. Yezzi and S. Soatto. Deformotion: Deforming motion, shape average and the joint registration and approximation of structures in images. *The International Journal of Computer Vision*, 53(2):153–167, 2003.

[106] X. Zeng, L.H. Staib, R.T. Schultz, and J.S. Duncan. Segmentation and measurement of the cortex from 3d mr images using coupled surfaces propagation. *IEEE TMI*, 18:100–111, 1999.

[107] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain MR images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1), January 2001.

[108] H. Zhao, S. Osher, B. Merriman, and M. Kang. Implicit and non-parametric shape reconstruction from unorganized points using a variational level set method. *Computer Vision and Image Understanding*, 80(3):295–314, 2000.