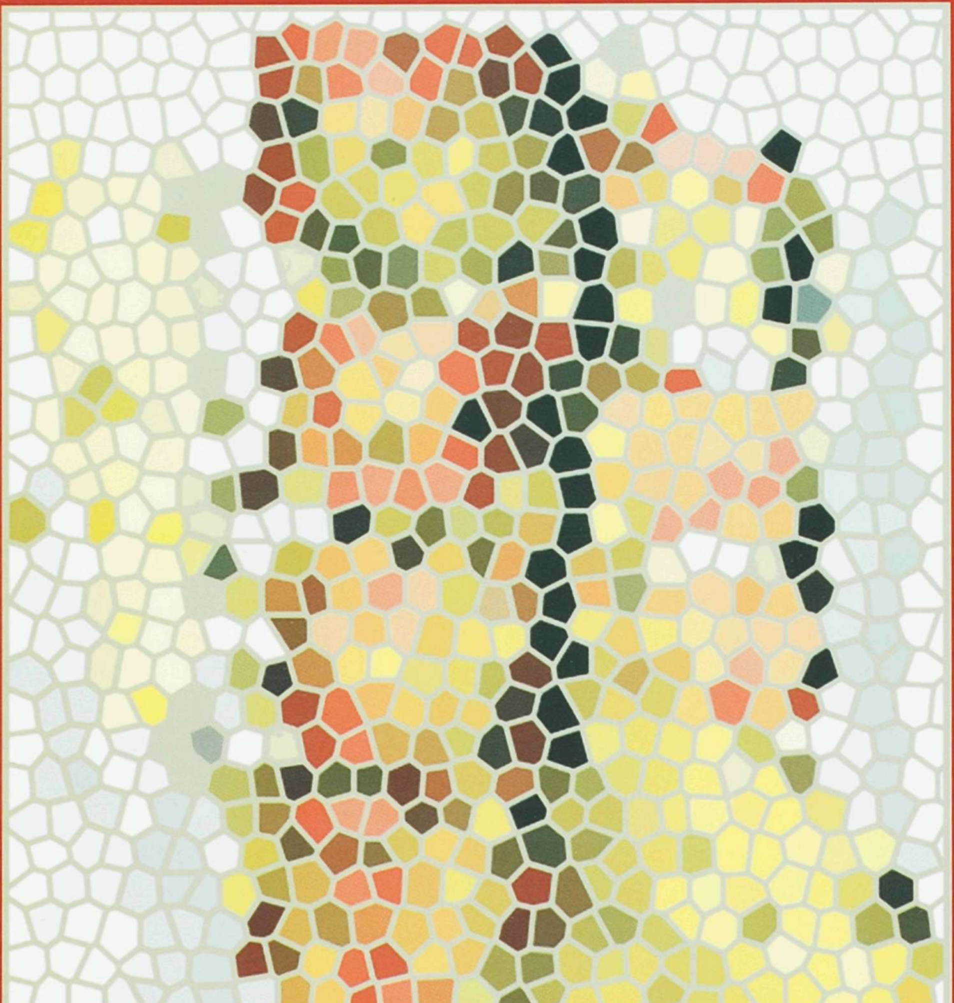
Strings and necklaces: on learning and browsing medical image segementations

Ghebreab, S.

Strings and Necklaces

On learning and Browsing Medical Image Segmentations



Sennay Ghebreab

Strings and Necklaces

On Learning and Browsing Medical Image Segmentations

Sennay Ghebreab

This book is typeset by the author using LaTeX 2$_\varepsilon$. The main body of the text is set using Times font ©Adobe Systems Incorporated. All graphics and images are included in the text in encapsulated Postscript format ^{TM} Adobe Systems Incorporated.

The graphic on the cover symbolically represents the complex task a computer faces when trying to find relevant structures in digital images. For the observer it is difficult, if not impossible, to decide without prior knowledge whether a structure is present in the graphic and to decide what that structure is. The graphic is a mosaic of the segmentation of the lumbar spine illustrated in chapter 6 and on the cover flops (in color). This information aids the observer in recognizing the various structures such as the vertebrae and the aorta. In a similar manner prior information may assist automatic segmentation methods in finding relevant structures in digital images.

ISBN 90-9015953-3

Strings and Necklaces
On Learning and Browsing Medical Image Segmentations

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam,
op gezag van de Rector Magnificus prof. mr. P.F. van der Heijden
ten overstaan van een door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen in de Aula der Universiteit
op woensdag 3 juli 2002 te 11:00 uur

door

Sennay Ghebreab

geboren te Addis Abeba, Ethiopië.

Promotiecommissie

Promotor: Prof. dr ir A. W. M. Smeulders

Overige leden: Prof. dr J. S. Duncan
Prof. dr K. J. F. Gaemers
Prof. dr ir C. A. Grimbergen
Prof. dr J. van der Lei
Prof. dr ir M. A. Viergever

Faculteit: Faculteit der Natuurwetenschappen, Wiskunde en Informatica

Research Group
Intelligent Sensory Information Systems

The research described in this thesis was carried out at the Intelligent Sensory Information Systems Group (University of Amsterdam). The research took place in cooperation with the Image Science Institute (Utrecht University) and Philips Medical Systems (EasyVision Advanced Development) within the framework of the Innovation Oriented Research pogramm (IOP Beeldverwerking, project number IBV 96012). Part of the research was conducted at the Image Processing and Analysis Group (Yale University). The project was financially supported by the Dutch Ministry of Economic Affairs.

Contents

Chapter 1

Introduction

In every day life, humans are accustomed to utilizing high-level visual concepts like objects, people and places to help navigate through the daily abundance of information reaching the eye. While these concepts come naturally to a human observer, they pose a significant difficulty to computer systems that are attempting to automatically grasp the content of image data for a wide variety of tasks, ranging from tracking moving cars in video data to matching engineering drawings to image data. In all these applications, computer systems require the ability to capture the projections of coherent entities amidst an overwhelming quantity of image data, recognize objects from them and draw semantically meaningful conclusions. This prerequisite has generated a vivid interest in the development of qualified imaging technology. In particular in the medical community, where the number of imaging modalities is still quickly growing, much attention is devoted to thoroughly exploring and exploiting the immense capabilities of imaging technology.

In today's medical imaging science many examples are found demonstrating the ever increasing interest in imaging technology. At the National Institute of Health (United States), the online WebMIRS tool is under development to facilitate the dissemination of biomedical database information across a wide-area network, allowing database access to text and associated images without requiring any software beyond a web browser. The VOXEL-MAN system, developed at the University Hospital Eppendorf in Hamburg (Germany), is being augmented for free-form volume-sculpting operations for specification of free form regions in three-dimensional images. At the Surgical Planning Laboratory, a teaching affiliate of Harvard Medical School (United States), interactive anatomy atlases are being developed for three-dimensional visualization and exploration of anatomical structures and pathology to facilitate preoperative and intraoperative planning. The Center for Advanced Instructional Media at Yale University (United States) aims at developing innovative computer-based educational teaching cases not only for the benefit of medical student and inhouse staff but also for the medical community at large. These are only few of the manifestations of the advent of imaging technology in medicine. A continued uprise of medical imaging is expected to have a rigorous impact on the quality of clinical health care at large.

1.1 Image Segmentation

Images serve as tools of medical assistance or as educational tools only when extraction of medically important morphological information is possible. To extract, this information it is often necessary to identify the elements in the image data belonging to the same anatomical structure, a process referred to as *image segmentation.* The segmentation problem is one of the most urgent topics in digital imaging, faced wherever there is a desire for high-level reasoning about the image content. Going back to the basics, the general segmentation problem is to divide an image into parts that have a different physical basis. This problem is transposed into one of finding elements in the image that have a one-to-one correspondence with real world entities when the image has the same dimensionality as that entity. The purpose of segmentation in medical imaging is to partition the image into areas that directly relate to anatomical structures or pathological conditions therein. Image segmentation preludes many processing and analysis tasks, hence plays an inevitable and pivotal role in automatic medical image understanding.

However, in spite of considerable research efforts in the past decades, generic image segmentation is one of the unsolved grand challenges in image understanding in general and medical imaging in particular. Segmentation of medical imagery is still defying due to the quality of medical images and complexity of their content. Medical images often contain anatomical structures that are not clearly captured by the imaging sensor or that are easily confused with similar adjacent structures. Consequently, in an image with insufficient intensity contrast, with interfering anatomical structures or with unexpected pathology, an image segmentation method is bound to encounter serious problems when it is not specified in full detail what is sought for in the image. The permanent quest for segmentation tools that can handle such complex images has led to numerous methods solving the segmentation problem for specific organs, imaging sensors or medical problems. However, there is still little progress in the development of generic and robust methods.

1.2 Research Directions

In an endeavor to develop generic segmentation tools the medical imaging community has devoted considerable attention to model-based segmentation methods, aiming at extracting boundary elements from the image data belonging to the same anatomical structure, then integrating these elements into a coherent and consistent model of that structure. The interest in model-based methods stems from the fact that they exploit bottom-up constraints derived from the image together with top-down a priori knowledge about location, size and shape of structures [79]. Deformable models have proven to be effective where traditional low level segmentation techniques, which only consider local information, fail due to incorrect assumptions during the integration of boundary elements. However, a number of difficult problems still need to be addressed for enhancement of deformable model methods. In this thesis we concentrate on some key challenges for deformable model methods in particular and medical image analysis in general:

- Automatic versus interactive segmentation. Interactive methodology is likely to remain dominant in practice for some time to come [79] and hence requires a structured approach rather than an ad-hoc solution.
- Statistical versus geometrical models. Much attention has been devoted to geometrical anatomical models subjected to physically plausible deformation laws, while the often complex variation in shape and gray-level appearance may be more appropriately captured by statistical models [20].
- Normal versus abnormal data. Many problems require the analysis of not only normal but also abnormal anatomical structures, however, often image analysis algorithm development ignores the latter [33].
- Landmark versus non-landmark data. Anatomical data is often used without particular attention to biomedical homology, while points reliably located by anatomical and geometrical criteria [10] may facilitate the understanding of anatomical structures.

The aim of the work described in this thesis is development of generic and robust methods for image segmentation and browsing. Focusing on medical imaging problems, we strive to learn a visual system what constitutes an anatomical structure and how its constituents can be exploited for segmentation and browsing. We conceive of an anatomical structure in an image as an entity that is defined by a manifold of features along its continuous boundary. In this context, statistical analysis of a collection of continuous multi-dimensional features allows to define the important characteristics of a population of normals or abnormals. When a visual learning systems is capable of doing this, one can view the problem of medical image segmentation as one of learning the appearance of the structure once and employing a learned model for extracting, weighting and selecting multivariate continuous features of an unknown image to determine the plausibility of a boundary being recorded. The problem of medical image browsing can be comprehended as one of repeatedly endorsing better feature weightings on the basis of novel image segmentations for the purpose of precise specification of the browsing intention. This thesis explores the potential of using and adopting these prospects.

1.3 Thesis Outline

Traditional image segmentation approaches are inequitably based on the assumption that medical images are self-contained, i.e. they contain most of the information necessary for identification of anatomical structures. However, clinicians employ a large purview of related knowledge on such fields as anatomy, pathology, physiology and radiology in order to arrive at a reasonable image interpretation. It is therefore natural to strive for an integration of user knowledge with the power of computation to arrive at interactive segmentation methods. In chapter 2, we review human-computer interaction in image segmentation with the goal of determining promising approaches to interactive segmentation.

In chapter 3, we elaborate on one promising interactive segmentation approach, namely deformable models. Deformable model methods have been used intensively for image segmentation thanks to their ability to combine a priori shape information with visual information contained in an image of interest. Also their intuitive interaction mechanism has contributed to their popularity. However, the incapacity of deformable models to handle sophisticated structures that require the definition of multiple boundary features has limited their application for medical imaging problems. For this reason, chapter 3 explores new ways of tackling the image segmentation problem by statistical deformable models. The question that is raised is how multiple boundary features such as edge gradient and contour curvature, can be exploited for learning a continuous variational image segmentation model.

In chapter 4, we describe a method for image retrieval. Retrieval of images from large repositories is commonly done on the basis of textual descriptions associated with the images. To exploit the rich visual information contained in the images themselves, content-based image retrieval methods have recently emerged. The problem with contemporary content-based retrieval methods is their premise that information extracted from a single example image suffices to determine the user's retrieval intention. Apart from this, information from the entire image is used to define a specific structure contained in a much broader, less relevant, quantity of image data. Chapter 4 concentrates on the problem of how to learn the user's retrieval intention and recover images best matching his/her concept from large collections of images, commencing from a single example.

Chapter 5 turns the focus back to the problem of segmentation, now in the context of three-dimensional images. Three-dimensional images often contain huge amounts of information, easily distracting automatic segmentation methods. To reduce distraction by unrelevant image data, salient image information is frequently exploited, where saliency is associated with an a priori specified feature rather than with a feature that is automatically selected on optimality. When dealing with sophisticated anatomical structures from which many salient points are likely to emanate, it is more natural to consider multiple features and to determine the best one automatically on a geometrical basis. Chapter 5 tackles the problem of how to conveniently define saliency and how to befittingly use it for (interactive) segmentation of volumetric image data in the context of deformable models.

In chapter 6, we take on the problem of segmenting spinal images. Segmentation of spinal images is an essential task for the analysis of spinal morphology. This is difficult to achieve fully automatically due to the articulated structure of vertebrae and their dense surroundings with ribs and other organs. Current approaches mainly focus on landmark-based segmentation of two-dimensional views of the spine, followed by a three dimensional reconstruction from the two-dimensional segmentations. The problem with these approaches is that they are less suitable for capturing truly intricate three-dimensional characteristics. For this reason, in chapter 6 we attempt to construct a three-dimensional integral spine model for landmark-based segmentation by combining techniques described in the preceding chapters.

Chapter 2

Review of Methods for the Interactive Segmentation of Medical Images

In recent years considerable attention has been devoted to develop automatic methods for image segmentation, which is a prerequisite for many multidimensional image analysis tasks. Despite this, fully automatic segmentation often fails, producing incorrect results. Most current computer-based segmentation techniques therefore still need significant user input to specify a region of interest, to initialize or control the segmentation process, or to perform subsequent adjustment of results. In this chapter, we depart from the view that, as user intervention is inevitable, interaction has to be approached in a structured manner and integral to computation. Following [89] we review interactive segmentation in literature with the goal of determining promising approaches to interactive segmentation. We first characterize, review and examine human-computer interaction in terms of user input, interpretation of user input and purpose thereof. We then focus on deformable models, as they turn out to be a promising platform for interactive segmentation. We characterize, review and examine some well-known methods in terms of their boundary model, objective function and optimization. We conclude this chapter with research questions that need to be addressed to develop generic interactive segmentation methods that optimize user input.

2.1 Introduction

Accurate image segmentation is one of the key problems in computer vision. The segmentation problem is to divide an image into parts that have a one to one correspondence with objects in the real world. The segmentation problem arises wherever there is a desire for high-level reasoning about the image. In content-based image retrieval, for example, images containing objects may be fetched on the basis of a boundary sketch provided by the user. In order to accomplish this task, objects in images need to be segmented and results matched with the sketch. In computer-aided diagnosis, segmentation is required for such tasks as visualization, registration and measurement of anatomical structures. To measure the deformation of the spine, for example, vertebral structures in CT or MRI images need to be outlined. Shape features are then computed from the resulting delineations.

A large number of image segmentation methods is found in literature. Where the purpose of all methods is identification of objects, different approaches are taken to solve the problem. Image segmentation methods can roughly be classified into methods based on boundary detection and methods based on similarity. The concept of segmenting an image by boundary detection is based on abrupt changes in gray-level values in the image, due to the fact that the two sides of the boundary have a different physical basis in the real world. Areas of interest within this category are detection of isolated points and detection of lines and edges in an image. The alternative approach is based on region growing. This concept of segmenting an image is based on similarity of the gray-level values in the image under the assumption that the gray values of an entity in the real world must have some property in common. Region growing techniques can be used to extract a single region from the image or to solve the full segmentation problem. Both approaches have received much attention to improve upon their shortcomings.

The large number of segmentation methods to date reflects the complexity of the image segmentation problem. Especially in the field of medical imaging, a wide variety of segmentation methods have been proposed to solve the segmentation problem. This is mainly due to the fact that objects in medical images are complex in shape and image appearance, and vary considerably across subjects. Complexity in appearance also refers to the fact that distinct objects may have very similar gray-level appearance in an image due to characteristic of the image modality, while the same objects can have different gray-level appearance in different regions of the image due to distortions caused by the imaging equipment. These properties of digital medical images have been a roadblock for the emergence of entirely automatic segmentation methods.

This explains why in most segmentation methods the user still plays a decisive role. The user's responsibility may be the initialization of the segmentation process, intervening during the process or adjusting the result. The minimal contribution of the user in medical image processing is judging the end result. However, despite the fact that human-computer interaction is required in almost all segmentation methods the emphasis in literature lies on description of the computational part, deprecating human-computer interaction. Few methods can be found which squarely describe interaction and computation. The question that arises is: *how to integrate interaction* *and computation in such a manner as to produce accurate, repeatable and efficient image segmentation*? These requirements are of particular importance for the segmentation of medical images [89].

In this chapter segmentation methods in literature are examined. In section 2 relevant interactive segmentation methods in literature will be reviewed and compared from a human-computer interaction perspective. In section 3, the emphasis lies on deformable model methods. A review will be given of promising deformable models found in literature. We conclude this chapter with section 4 were research questions are posed to be addressed in the remainder of this thesis.

2.2 Interactive Segmentation

An attempt to a better understanding of human-computer interaction, particularly in the segmentation of medical images, is found in [89]. In the reference, a comprehensive assessment of interaction in contemporary segmentation methods is given. A general scheme is provided for interactive segmentation method consisting of a user, a computational part, an interactive part, and a user interface. The focus is on the computational part, which is responsible for generating segmentation results on the basis of parameters and the interactive part which mediates between the user and the computational method.

As in the reference, we consider human-computer interaction as a process wherein the human operator conveys his/her knowledge to the computational method (see figure 2.1). The human operator easily recognizes objects in the image where the computational method sometimes fails to do so. In this context, the role of the human operator should be to guide the computational method. In this context the question is posed of: *how to minimize and ease interaction for the purpose of steering image segmentation*?

2.2.1 Characterization of Interactive Segmentation Methods

To characterize human-computer interaction, the interactive part of segmentation methods is captured in the following aspects: the type of user-input, the interpretation of user input and the purpose of user input. We follow [89] closely.

Type of User-Input

The type of user input considers the form in which the user conveys his/her knowledge to the computational method. As discussed in [89] three types of user input are common in interactive segmentation methods.

In the first type the user provides the computational method with *parameter values*. This is done in, for instance [88] where weights indicating relative importance of object properties are set and in [110] where the user provides a threshold. This type of interaction requires knowledge of the functioning of the computational part, with interaction potentially becoming inefficient.

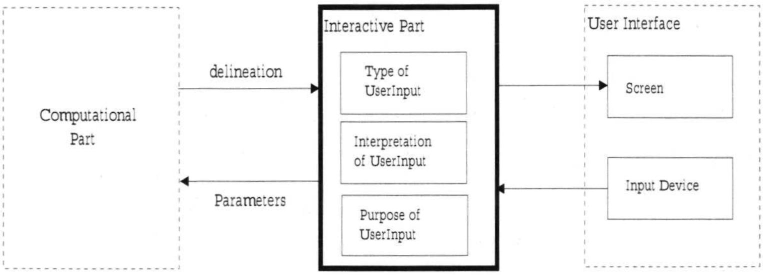


Figure 2.1: The interactive part of the human-computer interaction process in image segmentation. Main aspects are: type of user input, interpretation of user input and purpose of user input.

Visual input by indicating positions in the image grid is another type of interaction input. The positioning of seeds in the image, e.g. in [110], the indication of regions corresponding with objects of interest, e.g. in [77], and the repositioning of graphical models in the image, e.g. in [88], are different forms of visual input. This type of input is simple and efficient for the user if it concerns roughly indicating points, lines or regions. Otherwise it is cumbersome and time-consuming.

Interaction to select from a *pre-defined menu* where entries refer to parameter values for the computational method is also an re-occurring type of interaction [89]. For instance, the user may be allowed to select a graphical model from a set of pre-defined ones as in e.g. [54]. This type of input is simple for the user and the most efficient type of interaction. The drawback is that often a sophisticated computational method is required to translate user input to a language understood by the computational method.

Input Type	**Accuracy**	**Repeatability**	**Efficiency**
parameter values	-/+	-/+	-
visual input	+	-	-/+
menu selection	-	+	+

Table 2.1: Recapitulation of user input type in the human-computer interaction process in image segmentation.

To achieve accurate, repeatable and efficient segmentation, it is natural to select visual input on the image grid over parameter setting and menu-based input for several reasons. In the first place, visual input enables the user to convey his/her knowledge in the same domain in which the image is defined, allowing the input of information on the same level of abstraction as the image information. This is expected to support and facilitate efficient user input. In the second place, visual input allows user input close to the object of interest, increasing the accuracy of interaction. Lastly, visual input may be realized by a superficial glance at the screen and a few mouse clicks. Point-and-click type of user input leads to efficient interaction. Table 2.1 summarizes types of user input.

Interpretation of User Input

The interpretation of user input considers how the data provided by the user is translated to a form suitable for the computational method. Two approaches have been identified in [89].

In its most simple form ***user input is directly used*** by the computational method. An example of a method which directly interprets user input is found in [94]. The drawback, in this case, is that the user is required to know the functioning of the computational method to a certain extent. Moreover, due to the direct impact on the computational method repeatability is difficult to reach.

The alternative is to use ***user input indirectly.*** A mapping function is then required to transform user input into low-level parameters recognized by the computational method. For instance, rough indications of positions in the image grid may be used as a starting point for the search of object regions and boundaries as in e.g. [39], [81], [38], [77]). Or the user provides an initial delineation which is subsequently refined, e.g. in [88], [54].

Interpretation of User Input	Accuracy	Repeatability	Efficiency
direct interpretation	-/+	-	-/+
indirect interpretation	-/+	+	+

Table 2.2: Recapitulation of interpretation of user-input in the human-computer interaction process in image segmentation.

For medical image segmentation indirect interpretation of user input is preferred for accurate, repeatable and efficient segmentation for the following reasons. In the first place, indirect interpretation enables the input of information on a level of abstraction which is consistent with that of the user's. High-level user input is expected to support the user in performing interaction. In the second place, indirect interpretation of user leads to repeatable results. This is due to the fact that user input is processed ultimately by the computational method, which is less subjective than the user. Finally, indirect interpretation enables learning from user input, making reduction of future user input possible. Table 2.2 summarizes types of user input interpretation.

Purpose of User Input

The purpose of interaction considers what it is the user tries to achieve with his/her input. Several interactive scenarios have been distinguished in [89].

The role of the user may be to *judge* whether the computational method has produced an acceptable result. This role is present in all segmentation methods since the user is always in a position to reject and re-actuate a segmentation process. This is particularly relevant in medical applications where accuracy is important.

A user may also *correct* the result of the computational method manually as is done in [94]. Manual correction may guarantee user satisfaction, but leads to subjective results because the user has the last saying in the process. Manual correction does not favor repeatability.

Setting parameters for the computational method is the most occurring approach to interactive segmentation. It is found in various forms. For example, in [88], the user provides spatial parameters to bring a graphical model closer to the true boundary, while in [110] the user provides thresholds for a better correspondence of an image region with a real world object. Parameter setting produces repeatable results and allows learning based on user-input.

Another purpose of interaction is to *compose* results by combining primitive results produced by the computational method, e.g. in [105]. The accuracy of such methods may not obey the precision needed due to the fact that results depend on limited set primitives.

The user may also act with the purpose of ***building*** a segmentation process. For instance, in [67] the user determines the type and sequence of low-level image processing operations to produce results. The drawback with this kind of interaction is that the user is required to have knowledge of low-level image processing operations, even in cases where visual languages aid the user in the construction of a segmentation process.

Purpose of User Input	**Accuracy**	**Repeatability**	**Efficiency**
judge	+	-	-/+
correct	+	-	-/+
parameter setting	+	+	-/+
compose	-/+	+	-/+
build	-/+	+	-/+

Table 2.3: Recapitulation of purpose of interaction in the human-computer interaction process in image segmentation.

We prefer to conceive of parameter setting as the best mechanism to convey intention in user interaction. In the first place, parameter setting puts the user in control of the entire segmentation process, guaranteeing accurate results, while at the same time the computational method produces results efficiently. In the second place, parameter setting leads to repeatable results, because the computational method ultimately produces the results. Finally, parameter setting is efficient because it enables individual treatment of parameters. This is important because interaction is generally confined to parts of objects which form an obstacle, requiring only specific parameters to be corrected. Table 2.3 summarizes purpose of human-computer interaction.

2.2.2 Review of Interactive Segmentation Methods

A variety of interactive segmentation methods in literature combine the knowledge of the human operator with the strength of a computational method into one segmentation process. A limited number lets the user provide visual input which is translated to set parameters of the computational method. In the following we review some of these methods.

Interactive Boundary Detection Methods

The *piecewise deformable model* presented in [88] is a generic and locally controllable deformable model. The idea behind this method is to treat image segmentation as an optimization problem, where an objective function is used to reward certain boundaries in the image. The piecewise deformable model consists of a combination of boundary models and deformable splines, each with a particular segmentation task. A boundary model segment encodes expected shape and image features locally. A spline segment describes a boundary piece in the image in terms of shape and gray-level appearance. An objective function which measures the difference between sampled and expected features is optimized by deforming the spline segments in the image. The optimum is expected when the spline segments lie on the boundary of interest. The spline segments are initialized in the image by the user. The user also helps spline segments over local obstacles. This has only local consequence to ensure that correctly positioned spline segments in the image remain unaffected. The piecewise deformable model adopts human-computer interaction to initialize the model and to steer it by ***dragging*** with the mouse any of the spline pieces to better locations, from which the deformation continues.

In [54], a ***shape-based model*** is presented that uses prior knowledge of an object's shape to guide the search for boundaries. The shape-based model is defined by a long axis, which forms the midpoint of a collection of radial contours. A radial contour stores constraints describing the relative positions of discrete points on both sides of a boundary, creating an uncertainty interval. During an automated search-and-propagate process, boundaries are searched along the initial radial contour with a one-dimensional boundary detector, the radial lengths are computed and the uncertainty intervals are reduced to zero. The first computed radial length is propagated to adjacent slices to find complete surfaces. Given a shape model the user specifies at least a single radial length by selecting two points in the image which usually correspond to extremal points of the shape. The user may select more landmark points in the image to enhance the initialization. The shape-based model adopts human-computer interaction for bootstrapping by ***pointing and clicking*** landmark points in the image and to correct errors made during the search-and-propagate process by ***dragging*** contours in image slices.

In [39], two paradigms, referred to as *live wire* and *live lane*, are introduced for image segmentation. In these approaches pixels are considered as vertices of a graph where oriented boundaries are its arcs. To each oriented boundary a set of features is assigned which is converted to a single cost value. In both approaches, the problem of finding the best boundary segment between two points specified on the true boundary is then translated to finding the minimum cost-path between the two points. In live wire, the user first selects an initial point on the boundary. For any subsequent point indicated by the cursor, an optimal path from the initial point to the current point is found and displayed in real time as if the user has a live wire at hand which is moved by moving the cursor. If the cursor goes close to the boundary, the live wire snaps onto the boundary and follows it. If the live wire describes a boundary segment appropriately, the user deposits the cursor to fix the live wire and provide a new starting point. In live lane, the user selects only the initial point. Subsequent points are selected automatically as the cursor is moved within a lane surrounding the boundary whose width changes as a function of the speed and acceleration of cursor motion. Live-wire segments are generated and displayed in real time between successive points. The user gets the feeling that the live wire snaps onto the boundary as he/she roughly mark in the vicinity of the boundary. In short, the live-wire and live-lane adopt human-computer interaction to actively, but approximately, ***track*** with the mouse boundaries in the image which minimize a cost function.

Mortensen et al. [81] present an interactive tool called ***intelligent scissors***. Intelligent scissors allow objects to be extracted using simple gesture motions with a mouse. When the gestured mouse position comes in proximity to an object boundary, a live-wire boundary snaps to it, and wraps around the object of interest. Live-wire boundary detection formulates boundary detection as an optimal path search in a weighted graph. Optimal graph searching provides mathematically piece-wise optimal boundaries while greatly reducing sensitivity to local noise or other intervening structures. Robustness is further enhanced with on-the-fly training which causes the boundary to adhere to the specific type of boundary currently being followed, rather than simply the strongest edge in the neighborhood. Boundary cooling automatically freezes unchanging segments and automates input of additional seed points, allowing the user to be much more free with the gesture path, thereby increasing the efficiency and finesse with which boundaries can be extracted. The intelligent scissor adopts human-computer interaction to track boundaries in the image by simple ***gesture motions*** with the mouse.

Segmentation Method	**Object model**	**Interaction target**	**Interaction type**
Piecewise model	splines	object boundary	dragging
Shape-based model	radial contours	object boundary	pointing and clicking
			dragging
Live-wire/live-lane	polylines	object boundary	tracking
Intelligent scissors	polylines	object boundary	gesture motions

Table 2.4: Recapitulation of interaction in boundary-based image segmentation methods.

Table 2.4 summarizes interaction in boundary-based image segmentation methods.

Interactive Region Growing Methods

Sivewright et al. [110] present an interactive *seeded region growing method* that assists the clinical user in outlining relevant volumes. Starting from a seed representing an initial interior, statistical tests are performed for adjacent image positions in a manner equivalent to specifying an upper and lower value in thresholding methods. Image positions that pas the test are added to the interior of the region. This process is repeated until regions are constructed consisting of connected areas or volumes having similar pixel intensity values. Region growth is extended into adjacent slices to building up a three-dimensional volume. Direct 3D volume growth from a seed voxel is also possible. Results are returned as a set of interior and exterior regions boundaries. Threshold parameters may be determined automatically from a sample region or may be set by interaction. Region or volume growing is initiated by pointing and clicking with the mouse in the image. The seeded region growing method adopts human-computer interaction to *point and click* single points in the image and to grow homogeneous regions from there.

The *watershed method* described in [80] comes from the framework of mathematical morphology and may also be conceived of as a region growing method. In this method an image is seen as a topographical landscape by interpreting the strength of each edge as an altitude. The watershed transformation is the flooding process of this landscape from a number of sources. The water that falls on this landscape will follow the steepest slope until it reaches a minimum. The catchment basins that are formed in this manner represent regions of similar intensity in the image. The watershed lines, dams in the flooding interpretation, that divide adjacent basins are edges that separate the regions. In practice, many irrelevant basins are constructed leading to significant over-segmentation. If prior to the watershed transformation it is known which minima correspond to the desired object and which to the background, markers can be used to pierce only those points in the topographical surface that will lead to a significantly better segmentation. Markers are set by clicking with the mouse at the image locations corresponding to the object and background. Hence, the watershed method adopts human-computer interaction to decide on existence and location of markers by *pointing and clicking* in the image and to form connected regions from these within boundary conditions.

The *active paintbrush method* [77] computes primitive regions by a watershed like procedure applied on the gradient magnitude of the image intensity. The watershed boundaries coincide with boundaries in the image. To achieve fewer and larger primitive regions neighboring regions are merged based on intensity similarity, which is formulated as a global optimization problem using the minimum description length principle. This results in a hierarchical description of the image as a tree of merged segmentation primitives where boundaries of increasing granularity are preserved. The computational method then searches primitive region tree on the basis of user input and presents the user the outline of the largest region in the tree that satisfies these constraints. Selecting the regions from the hierarchical region tree that make up the object of interest is done by interaction. The interaction consists of the user dragging the mouse pointer over the object in the screen, whereby all regions hit by the mouse are selected and displayed in the image. Alternatively, the user may specify a region of interest around the object to be segmented or indicate points in the image that do or do not belong to the object of interest. The active paintbrush adopts human-computer interaction to indicate image locations by sweeping the mouse over it and to group small entities corresponding to the locations into complete objects by merging them.

In the *magic crayon tool* described in [6] an image is presented by a hierarchy which is built using multi-scale and differential geometric methods. The nodes in the hierarchy represent small, primitive regions in the image. The regions are constructed using a ridge flow method, which identifies ridges in the image on the basis of local maxima in the intensity graph. A region of points is assigned to each ridge, each point in this region being the initial value of a flow line on the intensity graph that terminates at the ridge. Each node in the hierarchy represents primitive regions of the image on a selected scale. Small regions at one scale are merged with other regions at a larger scale, i.e. small regions are linked to a parent node at the next level in the hierarchy. The aim is to select a set of subtrees within the hierarchy so that the subtrees corresponding with primitive regions correctly define the object of interest. In conclusion, the magic crayon tool adopts human-computer interaction to indicate, by sweeping with the mouse, subtrees within a hierarchy representing the image and to merge these subtrees into complete objects.

Segmentation Method	**Object Model**	**Interaction Target**	**Interaction Type**
Region growing	-	point single object	point and click
Watershed method	-	points on multiple objects	point and click
Active paintbrush	-	object region	sweeping
Magic crayon	-	graph area	sweeping

Table 2.5: Recapitulation of interaction in region-based image segmentation methods.

Table 2.5 summarizes interaction in region-based image segmentation methods.

2.2.3 Examination of Interactive Segmentation

The reviewed methods adopt human computer interaction to guide the computational method in the form of letting the user provide visual input which is translated to parameters suitable for the computational method. In accordance with our aim of minimizing and confining user input, next, the question is addressed of *how to use interaction in such a manner that the role of the user is consolidated to delivering visual input by a few mouse clicks to guarantee accurate, repeatable and efficient segmentation*? We approach this question in the context of the reviewed methods and examine which of them is most suitable for interactive segmentation.

Visual Input

Visual input is targeted at the *object boundary* or at the *object region.*

In the first case, no object model is available and input is performed on the basis of the visual evidence in the image only, as is the case in the live wire and live lane methods [39] and the intelligent scissors [81]. Alternatively input is performed on the basis of the correspondence between the visual evidence and the visual model depicting a possible boundary solution. This is the case in the piecewise deformable model [88] and the shape-based model [54] where respectively splines and radial contours model possible object boundaries throughout the segmentation process.

In the second case, no object model is present and input is performed on the basis of the visual evidence only. The input is targeted at points within one region such as in the seeded region growing method [110] or within multiple regions such as in the watershed method [80]. The input may also be targeted at whole regions such as in the active paintbrush [77].

We choose for visual input on the basis of the correspondence between the visual evidence and a visual model. A visual model, preferably of the entire boundary, provides the user continuously with visual feedback of plausible solutions in the form of an object model defined by spatial parameters. The user only needs to recognize the correspondence between the boundary in the image and the object model at a glance and provide visual information with a few mouse clicks where he/she thinks necessary. The type of user input where an object model is already at hand and only needs to be related with the object boundary in the image is efficient, in particular when it concerns the segmentation of large data sets.

Indirect Interpretation of User Input

The purpose of indirect interpretation of user-input is either high-level user input or reduction of future user intervention.

High-level human-computer interaction hides technical aspects of the computational method from the user. This is the case with the live wire and live lane [39], the intelligent scissors [81], the magic crayon [6] and the active paintbrush [77], where rough indications in the image suffice for region growing or boundary detection. In the piecewise deformable model [88] and shape-based model [54] higher level user-input is realized in the form of bootstrapping an a prior model of the boundary in the image or dragging parts of it to new locations to refine the result.

In the second place, indirect interpretation of user-input also allows *learning from user-input* by progressively renewing the knowledge contained in the method. In the live wire [39] and intelligent scissors [81], the image properties of stable boundary parts are used to dynamically adjust parameters of the computational method.

To achieve consolidated user input where visual input is delivered by a few mouse clicks, it is natural to support both high-level human-computer interaction and learning on the basis of user input. High-level human-computer makes visual input possible, while learning on the basis of user input reduces user intervention and increases repeatability.

Parameter Setting

Setting parameters is done with the purpose of initialization or steering.

Initialization is the process of bootstrapping a segmentation process. It is done by positioning an initial model in the image as done with the piecewise deformable model [88] and shape-based model [54]. Alternatively it is done by placing seeds and markers in the image as done in seeded region growing [110], the watershed method [80], the live wire and live lane [39] and intelligent scissors [81].

Steering is the process of guiding the computational method through the segmentation process. It is done either by active participation from the part of the user, as done in the live wire and live lane [39], intelligent scissors [81], the active paintbrush [77] and the magic crayon [38]. It may also be done by passively guiding the computational method as done in [88] where the user occasionally drags spline segments to correct locations and in [54] where the user edits contours to correct errors in the shape-based model [54].

Initialization by bootstrapping an initial model close to the true boundary supports consolidated user input where visual input is delivered by a few mouse clicks. This significantly increases efficiency and accuracy. In addition, the combination of having a plausible solution in the form of a model and a user who at a glance recognizes the corresponding boundary in the image reduces steering to a few and simple mouse clicks in the image. In this type of human-computer interaction the user remains in control of the entire segmentation process in the sense that he/she can always intervene in case of unsatisfactory progress of the process he/she initialized.

2.2.4 Discussion

The deformable model approach seems most promising for the purpose of interactive image segmentation for the following reasons. Deformable models combine interaction and computation elegantly. They support visual input by the user in the form of spatial parameters in the image grid, both to accurately initialize the complete boundary model close to the true boundary and to locally steer the computational method to produce better results efficiently. The user continuously receives visual feedback in the form of curves or surfaces defined by the spatial parameters and remains in control of the process in the sense that he/she can always intervene in case of unsatisfactory results. Hence, of the reviewed papers, the ***piecewise deformable model*** [88] is most suited for development of interactive segmentation methods.

Moreover, most of the reviewed segmentation methods are mainly directed towards the segmentation of two-dimensional images, where interaction is a depreciated but well developed concept. When dealing with three-dimensional images many concepts from two-dimensional interactive segmentation can be directly inherited, however, some aspects require a completely different approach. For example, while the rough indication of object regions in two-dimensions is straightforward and may lead to accurate interaction, in three dimensional images this task is far from trivial and efficient, reducing accuracy and possibly downplaying methods that adopt this type of user-input. Volume segmentation with help of three-dimensional deformable models has however been studied [33] more intensively providing a relatively stable platform for the development of three-dimensional interactive segmentation methods.

In short, we conclude that the piecewise deformable model in particular and deformable models in general have properties suitable for interactive segmentation strategies to solve multi-dimensional segmentation problems. We take over the idea behind deformable models and employ them as our main platform for further investigation.

2.3 Deformable Model Methods

Deformable models are a top-down approach to image segmentation, needed for segmentation problems that are difficult, if not impossible, to tackle using classical approaches. They challenge the widely held view that vision tasks are bottom-up processes; features are extracted from an image and higher level processes group or interpolate to find plausible object boundaries. Deformable models start with an a prior boundary model of what an object boundary should look like and refine the model on the basis of image and shape features extracted from a new image to find a mathematically optimal boundary. As a result, they are more robust than non-model based methods, which make little use of a priori knowledge.

The role of the computational method is to find the optimal boundary in the image in an automatic fashion. In our interactive segmentation intention, the question that arises is: *how to use the computational method in combination with human-computer interaction in such a way as to produce accurate and reproducible results in an efficient manner*? We will address this question in the context of deformable models.

2.3.1 Characterization of Deformable Model Methods

The essence of most deformable model methods can be captured in terms of the following aspects. What prior boundary information is specified? How this is represented geometrically or otherwise? How boundaries in the image are qualified in terms of image and shape features? How the optimization deforms the model from an initial configuration to find an optimal boundary in the image? The main components of the computational part of deformable models (see figure 2.2) correspond with these aspects: the boundary model, the objective function and the optimization. Much of the current research in deformable models is focussed on generalizing boundary models to cope with more than one segmentation problem, reformulating objective functions to obtain more manageable feature landscapes, and overcoming convergence and stability problems encountered during the optimization process.

The deformable model which has attracted most attention is the *snake* described in [61]. Snakes represent a special case of the general multi-dimensional deformable model theory [79]. In the remainder of this section we discuss the formulation of snakes in its simplest form and use it as the reference model for all papers to be reviewed in the remainder of this chapter.

The *boundary model* of the snake specifies the expected shape of the object and

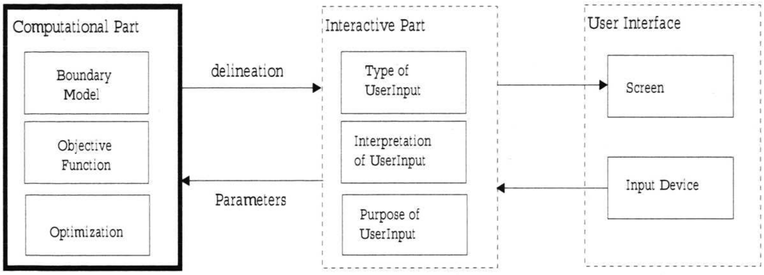


Figure 2.2: Deformable models in the human-computer interaction process in image segmentation. Main aspects are: boundary model, objective function and optimization.

its size and location in the $\mathcal{D}$-dimensional image $I(\mathbf{x}), I : \mathbf{x} \in \Re^{\mathcal{D}} \rightarrow \Re$. For the two-dimensional case it is represented geometrically by the contour

$$\mathbf{v}(u) = (x(u), y(u)) \tag{2.1}$$

where x and y are coordinates in the image and $u \in [0...1]$ is in the parameter domain.

To attract the snake to object boundaries a local image objective is defined whose local minima coincide with intensity discontinuities. The snake is attracted to edges by means of the information derived from the image gradient $\nabla(G_\sigma * I)(\mathbf{x})$, where ∇ is the gradient operator and $G_\sigma * I$ denotes the image convolved with a Gaussian smoothing filter whose characteristic width σ controls the spatial extent. The local image objective is defined from the gray-level data as follows

$$E_g(\mathbf{v}(u)) = -|\nabla(G * I(\mathbf{v}(u)))|^2. \tag{2.2}$$

The shape of potential objects is described on the basis of features extracted from the contour $\mathbf{v}(u)$ itself. To guarantee physically feasible shapes a local shape objective is defined, which keeps this contour from stretching or contracting along its length and another one to keep it from bending. Stretching is measured in terms of the first derivative of the snake's shape while bending is measured in terms of the second derivative. Assuming arc-length parameterization the local shape objective is defined as

$$E_s(\mathbf{v}(u)) = |\frac{\partial \mathbf{v}}{\partial u}|^2 + |\frac{\partial^2 \mathbf{v}}{\partial u^2}|^2. \tag{2.3}$$

The global *objective function* qualifies a sampled boundary in the image on the basis of the object properties expressed in terms of the above image and shape information. The fit quality is the weighted combination of the local image objective value and the local shape objective value, integrated along the entire contour, with the application dependent weight ω_g and ω_s indicating the relative importance of each feature type

$$Q(\mathbf{v}) = \oint_u (\omega_s E_s(\mathbf{v}(u)) + \omega_g E_g(\mathbf{v}(u)))du. \tag{2.4}$$

An initial position for the boundary model is specified to bootstrap the search for the optimal boundary. Initialization is done either by a higher-level mechanism, or by other processing. In most cases this is the user. The user also has the opportunity to control the snake behavior by means of an external energy (not defined here), that allows points of attraction and repulsion or other user defined interactions. Initialization leads to a preliminary result $\mathbf{v}_b(u)$ which needs to be refined to obtain the final outcome $\mathbf{v}_f(u)$. The result after refinement is defined such that

$$\mathbf{v}_f = \underset{\mathbf{v}_b}{\text{argmax}} \quad Q(\mathbf{v}_b) \tag{2.5}$$

Optimization aims at finding the optimal boundary in the image, starting from $\mathbf{v}_b(u)$. The best snake position is defined as the solution of a variational problem, requiring the minimization of the objective function. In accordance with the calculus of variation, the model which minimizes equation 2.4 must satisfy the Euler-Lagrange equation [61]

$$-\frac{\partial}{\partial u}(\frac{\partial \mathbf{v}}{\partial u}) + \frac{\partial^2}{\partial u^2}(\frac{\partial^2 \mathbf{v}}{\partial u^2}) + \nabla E_g(\mathbf{v}(u,t)) = 0 \tag{2.6}$$

The corresponding Euler equations, which give the necessary conditions for this minimizer, comprise a force balance equation. By introducing a temporal parameter t, the force balance equation can be made dynamic. When the dynamic equation reaches its steady state, a solution to the static problem has been found. The resulting contour is then assumed to be the boundary of the target object.

Analysis of snakes has revealed some common drawbacks. In general the problem is that the two terms 2.2 and 2.3 are defined for the entire boundary. As a consequence the objective function makes the snake always move towards objects with an entirely homogeneous boundary where the intensity changes are highest and the shape is smooth, even if the object of interest does not comply to that description. Many reformulations have been devised in an effort to tackle this and other problems.

2.3.2 Review of Deformable Model Methods

A review of some of deformable model methods is found in [59]. In the reference, a classification is made of deformable model methods in literature on the basis of the boundary representation adopted by the methods. A similar approach is found in [48] where deformable models are reviewed from computer graphics point of view. A more general review is found in [79]. McIrney and Terzopoulos discuss several aspects of deformable models for medical images. A comprehensive assessment is given without providing a particular classification of deformable models.

In the following sections, we will order several deformable model methods in literature on the basis of their important contributions.

Boundary Model

Yuille [131] proposes a method for detecting and describing features of objects using deformable templates. The object of interest is described by a detailed parameterized template. An objective function is defined which links edges, peaks, and valleys in the image intensity to corresponding properties of the template. The template then interacts dynamically with the image by altering its parameter values to minimize the energy function, thereby deforming itself to find the best fit. The final parameter values are used as descriptors for the object. The template needs to be hand-crafted completely for other segmentation problems. In the reference, the outer boundary of the eye is defined by two parabolic structures and so on until a q-dimensional feature vector is derived to capture the characteristics of the eye. The objective function is a function of this parameter vector. The priori boundary information in this method is specified in terms of the geometric primitives, the mutual relations between these and the gray-level appearance of the boundary. The deformable template explicitly models prior information about boundary shapes and their respective gray-level appearance with help of geometric primitives.

Duncan et al. [114] propose a more general shape-based deformable model. They apply flexible constraints in the form of a probabilistic parametric deformable model, to the problem of segmenting natural 2-D objects whose diversity and irregularity of shape make them poorly represented in terms of fixed features or form. Probability distributions on the parameters of the representation bias the model to a particular overall shape while allowing for deformations. Boundary finding is formulated as an optimization problem using a maximum a posteriori objective function. Object boundaries are represented geometrically by the continuous function

$$\mathbf{v}^*(u) = \sum_{k=1}^{\inf} p_k \phi(u), p_k = \int_a^b \mathbf{v}(u)\phi_k(u)du \tag{2.7}$$

where p are the coefficients which are projections of the function onto the k Fourier basis functions ϕ and (a, b) is the interval on which the shape is defined. The Fourier parameterization is chosen because of the geometric interpretation in terms of ellipses and because it is invariant to rotation, scale and translation. The priori information in this method is expressed in the contour representation: the truncation of the sum limits the number of parameter and smoothes the contour. The number of basis functions is a tradeoff between the desired accuracy, conciseness and degree of smoothing. The parametric deformable model captures prior information about global object shape with help of Fourier descriptors, requiring objects to be smooth in order to be represented and detected by the model.

Terzopoulos introduces deformable superquadrics in [120] which are a class of dynamic models that can deform both locally and globally, this way combining global properties of a conventional superellipsoid with the local degrees of freedom of a spline. The model's global deformational degrees of freedom capture gross shape features, while local deformation parameters reconstruct the details of complex shapes that the global abstraction misses. The equations of motion which govern the behavior of deformable superquadrics make them responsive to externally applied forces. They fit models to visual data by transforming the data into forces and simulating the equations of motion through time t to adjust the translational, rotational, and deformational degrees of freedom of the model. Object boundaries are represented geometrically by

$$\mathbf{v}^*(u,t) = \mathbf{c}(u,t) + \mathbf{R}\mathbf{p}(u,t) \tag{2.8}$$

where $\mathbf{c}(u,t)$ is the origin of an inertial frame ϕ, $\mathbf{R}$ is the rotation matrix which gives the orientation of ϕ and $\mathbf{p}(u,t)$ denotes the position of a point on the model. $\mathbf{p}(u,t)$ is expressed as the sum of a reference shape $\mathbf{s}(u,t)$ and a displacement function $\mathbf{d}(u,t)$: $\mathbf{p} = \mathbf{s} + \mathbf{d}$. Global boundary shape is specified in terms of $\mathbf{s}$ which is a superquadric ellipsoid, while $\mathbf{d}$ specifies local boundary shape. The main contribution of the reference is the simultaneous modeling of prior information about global and local object shape with help of superquadrics and splines.

In [90], Olstad and Torp propose a grammatical encoding to represent boundary shape and the associated signatures in the underlying images. The variability encountered in an object's shape is addressed with the energy minimization procedure which is embedded in the grammatical framework. They propose an algorithmic solution that combines a non-deterministic version of the Knuth-Morris-Pratt algorithm for string matching [102] with a time-delayed discrete dynamic programming algorithm for energy minimization. Object boundaries are represented by polygonal shapes, existing of straight lines connected at a N ordered vertices $\mathbf{v}_n, n = 1, ..., N$ with corresponding terminals α_n with which prior boundary information is associated. The encoding of a priory boundary information in a grammatical model taking the form of $\alpha_n = l * cs*$. This expression generates sentences like $ll \cdots llcss \cdots ss$ where the number of l terminals and s terminals are arbitrary. l penalizes all deviations from a straight line, c checks that the point is a corner with a negative bend and s is a smoothness model with a hard constraint on the convexity. The boundary model captures prior information about object shape in an explicit way using grammatical encodings, making it generic enough to express a wide variety of shapes.

Cootes et al. [70] describe a compact parameterized model of facial appearance which takes into account many sources of variability. The model represents both shape and gray-level appearance, and is created by performing a statistical analysis over a training set of face images. A robust multi resolution search algorithm is used to fit the model to faces in new images. This allows the main facial features to be located, and a set of shape, and gray-level appearance parameters to be recovered. Object boundaries are represented geometrically by the point distribution model with cardinality N

$$\mathbf{v} = \{\mathbf{v}_1, ..., \mathbf{v}_N\}. \tag{2.9}$$

The point distributions in a training set are used to construct a model of the boundary denoted by $\bar{\mathbf{v}} + \mathbf{P}\mathbf{b}$, where $\mathbf{P}$ is a matrix of unit eigenvectors of the covariance of deviations, and $\mathbf{b}$ is a vector of eigenvector weights. By modifying $\mathbf{b}$, new instances of the shape model are created. The elements of $\mathbf{b}$ determine the deviation of the model from the prior mean boundary defined by points $\bar{\mathbf{v}}$. Having found the shape using the point distribution model, the face is deformed into a normalized frame, in which a model of the intensities of the shape-free face is used to interpret the image. The deformation of the face is done on the basis of landmarks similar to the thin-plate spline technique of Bookstein [9]. The statistical model learns information about objects from a training set, allowing to consider natural variations in shape and gray-level appearance when looking for objects in new images.

The piecewise deformable model presented in [88] also models shape and gray-level appearance. As described in the previous section the piecewise deformable model is a combination of pieces of splines representing boundary segments in the image. Prior boundary information is specified in terms of basic characteristics of the contour and the expected value of local image and shape feature values. An objective function which measures the difference between sampled and expected features is optimized by deforming these spline segments in the image. The optimum is expected when the spline segments is on the boundary of interest. Object boundaries are represented geometrically by B-splines

$$\mathbf{v}^*(u) = [\sum_{n=1}^{N} v_n^1 B_n(u), \sum_{n=1}^{N} v_n^2 B_n(u)] \tag{2.10}$$

where N is the number of control points, (v_n^1, v_n^2) is the two-dimensional position of the nth control point, and $B_n(t)$ is the weight defined by the value of the corresponding weight function at path parameter u. B-splines can provide a very compact representation for curves, requiring less control points, than for polygonal representation for comparable accuracy. Apart from this the use of B-splines also allows for better representations of highly curved shapes. The piecewise model captures boundary information about local shape as well as gray-level appearance piecewise with help of splines, allowing to deal with heterogeneous boundaries.

Boundary model	**Boundary representation**	**Prior boundary information**
Deformable template	parabolic	global shape and gray-level
Parametric deformable model	elliptic	global shape
Deformable superquadric	quadric	global and local shape
Grammatical model	polygon	local shape
Point distribution model	point set	global shape and gray-level
Piecewise deformable model	spline	local shape and gray-level

Table 2.6: Recapitulation of boundary representation and a priori information adopted by deformable model methods.

Table 2.6 summarizes the characteristics of the examined boundary models.

Objective Function

In [129], segmentation is achieved by tuning the parameters of the geometrical model in such a way that the boundary template locates and describes the object in the image in an optimal way, taking into account directional information. That is, in contrast to other methods, in the reference the optimality of the solution is based on an objective function that matches directional image information with directional shape information. This is expressed by an image objective function based on directional gradient information derived from Gaussian smoothed derivatives of the image defined as

$$E_g^*(\mathbf{v}(u)) = \nabla I(\mathbf{v}(u)) \cdot \mathbf{v}'_\perp(u) \tag{2.11}$$

where the dot product measures the correspondence of the direction of the normal of the contour with the direction of the image gradient is. The shape objective is similar to equation 2.3. It is based on the contour curvature, but made independent of spatial scale by multiplying the curvature with the length of the contour. The proposed objective function locates an object boundary even in the case of a conflicting object positioned close to the object of interest.

Cohen [17] presents an image objective function that significantly increases the capture range range of the snake method. The local image objective is an increasing continuous function of the distance to the closest point in the set of already detected contour points. Using this potential field, the contour can detect features from further away. Apart from this, weak image evidence will only be ignored if there is a better edge in the vicinity. The local image objective is defined as

$$E_g^*(\mathbf{v}(u)) = D(I(\mathbf{v}(u))) \tag{2.12}$$

where D denotes the Euclidean or Chamfer distance map. The objective function does not contain a shape term since the contour is represented by a B-spline which has an inherent regularization part. Hence, the main contribution of the reference is that the proposed objective function increases the capture range of snakes, thereby making it more robust to local minima and initialization.

Xu et al. [130] present a new image objective function which they call gradient vector flow. It is computed as a diffusion of the gradient vectors of gray-level or binary edge map derived from the image. It differs fundamentally from the traditional image objective in that it cannot be written as the negative gradient of a potential function, and the corresponding snake is formulated directly from a force balance condition rather than a variational formulation. To obtain the corresponding dynamic snake equation the image term in equation 2.2 is replaced by

$$-\nabla E_g^*(\mathbf{v}(u)) = \mathbf{r}(\mathbf{x}) \tag{2.13}$$

where $\mathbf{r} : \Re^n \rightarrow \Re^n$ is the gradient vector flow field that minimizes $\int_{\Re^n} |\nabla \mathbf{r}|^2 + |\nabla f|^2 |\mathbf{r} - \nabla f|^2 d\mathbf{x}$ where ∇f is an edge map computed in the traditional manner. The shape objective is identical to equation 2.3. This formulation retains the nice properties of the above mentioned distance based-based image feature, while coping well with concavities in the object boundary. The proposed objective function increases the capture range of snakes and attracts snakes to boundary concavities, reducing the sensitivity to the initial configuration and local minima in the energy landscape.

In [107], Schnabel et al. present a hierarchical multi-scale shape descriptor based on snakes. In contrast to the original snake method the proposed method considers object boundaries at various levels of detail using multi-scale differential invariants.

Starting at a course scale for close initialization, image features are extracted and the snake is optimized at decreasing scales. Additionally, distance information is used to attract the snake from large distance. The local image objective function is represented as a function of scale σ

$$E_g^*(\mathbf{v}(u), \sigma) = |\nabla I(\mathbf{v}(u), \sigma)|^2 + D(I(\mathbf{v}(u)), \sigma). \tag{2.14}$$

The authors also introduce a new shape objective. They propose to minimize the deviation of the isophote image intensity curvature from the contour curvature, instead of minimizing the curvature along the contour. The shape objective is also represented as a function of scale

$$E_s^*(\mathbf{v}(u)) = |\frac{\partial^2 \mathbf{v}}{\partial u^2}|^2 - C(\mathbf{v}(u), \sigma) \tag{2.15}$$

where C denotes the isophote curvature of the image intensity at scale σ. In conclusion, the proposed objective is designed to attract the snake to multi-resolution object boundaries with points of high curvature.

Staib et al. [113] propose a gradient-based deformable model finding approach that integrates region information. Their approach uses Green,s theorem to derive the boundary of a homogeneous region-classified area in the image and integrates this with gray-level gradient-based deformable model to combine the perceptual notions of edge information with homogeneous region information. The image objective function is formulated as

$$E_g^*(u) = |\nabla I(\mathbf{v}(u), \sigma)|^2 + \int_A \Delta I(\mathbf{v}(u)) dA \tag{2.16}$$

where ΔI denotes the image indicating homogeneous regions and A is the area bounded by the contour $\mathbf{v}(u)$. The shape objective is expressed as multi-variate Gaussian prior obtained from previous outlines, which is used to constrain the optimization within the a posteriori framework. The proposed objective function makes deformable models more robust to noise and poor initialization.

Segmentation method	**Boundary properties**	**Boundary configuration**
Piecewise DM	curvature + gradient + laplacian + ..	heterogenous
DM in gradient vector field	curvature + gradient vector field	homogeneous
Balloon	curvature + distance map	homogeneous
GFV snake	curvature + gradient vector flow	homogeneous
Scale-space DM	isoph. curv. + gradient scale space	homogeneous
Boundary and region DM	curvature + gradient magnitude/region	homogeneous

Table 2.7: Recapitulation of objective functions in terms of boundary properties and configurations as adopted by deformable model methods.

Table 2.7 summarizes the conclusions with respect to the definition of objective functions as proposed in the reviewed papers.

Optimization

Amir et al. [1] pointed out several problems with the variational approach to energy minimization. In their work, they showed that the convergence properties of the original snake can not be predicted within the variational framework because variational approaches do not guarantee global optimality of the solution and because they require estimates of higher order derivatives of the image data, which tend to be unstable with noisy data. They also noted that the variational approaches only allow constraints which are additive and differentiable, making it impossible to embed strict hard constraints into snake's energy functional. They proposed a new formulation for the snake using a dynamic programming method. The snake is discretized to a set of contour points $\mathbf{v} = (\mathbf{v}_1 ... \mathbf{v}_n)$ and the objective function is minimized by combining solutions to sub-problems. With the image term omitted for simplicity this reduces to finding

$$S_n(\mathbf{v}_{n+1}, \mathbf{v}_n) = \min_{\mathbf{v}_{n-1}} S_{n-1}(\mathbf{v}_n, \mathbf{v}_{n-1}) + \alpha(|\mathbf{v}_n - \mathbf{v}_{n-1}|) + \beta|\mathbf{v}_{n+1} - 2\mathbf{v}_n + \mathbf{v}_{n-1}|^2 \tag{2.17}$$

where S_n is the optimal value function which is obtained by performing a minimization over the discrete point $\mathbf{v}_n$. A discrete dynamic programming algorithm solves subproblems just once and then saves its answer in a table, thereby avoiding the work of recomputing the answer every time the sub-sub-problem is encountered. Convergence of the energy minimization is guaranteed without ensuring that the global minimum is eventually found. Hence, the contribution of the proposed optimization approach is improvement of the optimality, numerical stability and convergence of snakes, while at the same time allowing hard constraints to be satisfied.

Williams and Shah's greedy algorithm for active contours [128] performs an efficient local neighborhood search which is more speedy and less memory consuming than the varational approach and dynamic programming. The greedy algorithms performs a search where update of contour elements is explicit and allows the simple inclusion of hard constraints into the model. A neighborhood is defined around each element. For each element in turn the overall energy change caused by a move to each candidate position is calculated. The position that minimizes the resultant contour energy is chosen as the new position of that element. As each move is chosen so that the overall energy of the contour cannot increase, convergence is assured. The snake is discretized to a set of contour points $\mathbf{v} = (\mathbf{v}_1 ... \mathbf{v}_n)$ and is evolved according to

$$\mathbf{v}_n = \alpha(d - |\mathbf{v}_n - \mathbf{v}_{n-1}|) + \beta(|\mathbf{v}_{n-1} - 2\mathbf{v}_n + \mathbf{v}_{n-1}|^2) + \gamma(E_g) \tag{2.18}$$

where d is the average distance between point which is updated after each iteration. The objective function is computed for $\mathbf{v}_n$ and each of its neighbors. The location having the smallest value is chosen as the new position for $\mathbf{v}_n$. $\mathbf{v}_{n-1}$ has already been moved to its new position during the current iteration, while the location of $\mathbf{v}_{n+1}$ has not yet been moved. In the next iteration $\mathbf{v}_{n+1}$ is optimized in the same manner with new d and new weights α, β, γ. This process continues until a local optimum is found for all points. The greedy optimization algorithm is primarily designed to increase the speed up convergence of snakes while allowing hard constraints to be satisfied.

Cohen introduces the balloon method in [16]. He presents a model of deformation which solves some of the problems encountered with the snake method. To obtain more stable results, the definition of the external forces derived from the gradient of the image is modified to avoid instability due to discretization of the evolution problem to obtain more stable results. A new image objective is also introduced that changes the evolving behavior of the contour in that it pushes the contour to edges like a balloon. The contour passes over edges and is stopped only if the edge is strong, thereby avoiding local minima. Contour finding is performed only in a given area. To enforce this behavior an additional energy term is added that is a negative scalar multiple of the area inside the contour. The equation of motion becomes

$$\nabla E_f(u) = k_1 \mathbf{n}(u) - k \frac{\nabla E_g}{||\nabla E_g||} \tag{2.19}$$

where $\mathbf{n}$ is the normal vector to the contour at point $\mathbf{v}(u)$ and k is the amplitude of this force. Changing the sign of k or the orientation of the curve causes the contour to deflate instead of inflate. In this manner the contour expands and is attracted and stopped by edges. If the edge is to weak the contour passes due to the pressure force. The proposed optimization approach improves the convergence of snakes even in case of poor initialization.

In [51] a dual active contour is presented. In contrast to other methods which approach image boundaries from one side, the dual active contour uses two contours: one contour expands from inside the target feature, the other contracts from the outside. The two contours are interlinked to provide a balanced technique with an ability to reject weak local energy minima. The internal energy of the contour is reformulated to be scale invariant, and allows a relative assessment to reject poor local minima. The dual active contour allows any additional local shape information to be integrated within the minimization process. This shape models local control over the contour's equilibrium. The problem of determining parameters is simplified by reducing the parameters to a single regularization parameter which is consistent with the original paradigm. The snake is discretized to a set of contour points $\mathbf{v} = (\mathbf{v}_1 ... \mathbf{v}_n)$. Evolution of this contour reduces to

$$\mathbf{v}_n^{t+1} = \mathbf{v}_n^t + \frac{1}{2}(\lambda \frac{\mathbf{e}_n}{h} + (1-\lambda)\mathbf{F}_n) + g(t)\frac{\mathbf{u}_n - \mathbf{v}_n^t}{|\mathbf{u}_n - \mathbf{v}_n^t|} \tag{2.20}$$

where $\mathbf{F}_n$ is the force derived from the image objective, $\mathbf{u}$ is the other contour and $g(t)$ is the strength of the adaptive driving force which is modified by the minimization algorithm. Once both contours have found equilibrium, elements of the two are compared to see if the same minimum energy point has been reached. When this is not the case, the contour with large energy is perturbed until its energy further decreases and finds equilibrium once again. The proposed optimization strategy relieves the user from the problem of initialization by reducing the search space to an area bounded by two contours. Apart from this, the local minima are also dealt with.

In [84] a snake-based approach is proposed which allows a user to specify only the distant end points of the curve he/she wishes to delineate without having to supply an almost complete polygonal approximation. This is achieved by using the image information around the end points to provide boundary conditions and by introducing an optimization schedule that allows a snake to take image information into account first only near its extremities and then, progressively, toward its center. In effect, the snakes are clamped onto the image contour in a manner reminiscent of a ziplock being closed. Neueschwander et al. reformulate $K \cdot V = F_v$, which is equation 2.2 after discretization, in order to perform the optimization gradually starting from the head and tail of the contour. The equation which governs the contour evolution is formulated as

$$(K^* + \gamma^{[t]}\mathbf{I}) \cdot \mathbf{V}^{*[t]} = \gamma^{[t]}\mathbf{V}^{*[t-1]} + 1|_{\gamma*[t]} \cdot F^*_{\mathbf{V}^{*[t-1]}} \tag{2.21}$$

where $\gamma^{[t]}$ is a viscosity term, the superscript $[t]$ denotes the iteration step, V^* stands either for x or y and $F^*_{\mathbf{V}^{*[t-1]}}$ is the driving image force. The snake can be used to alleviate the often repetitive task practitioners face when segmenting images by eliminating the need to sketch a feature of interest in its entirety, that is, to perform a painstaking, almost complete, manual segmentation. The ziplock optimization approach simplifies the initialization process and yields better convergence properties.

Segmentation method	**Initial estimate**	**Optimization strategy**
Dynamic active contour	contour far from boundary	global
Greedy active contour	contour near boundary	local
Balloon	contour inside/outside boundary	oriented
Dual active contour	two contours closing boundary	confined
Ziplock snake	points on boundary	constrained

Table 2.8: Recapitulation of optimization in terms of initial estimate and optimization strategy as adopted by deformable model methods.

Table 2.8 recapitulates the conclusions with respect to the optimization strategies adopted by the reviewed methods.

2.3.3 Examination of Deformable Model Methods

In this section we briefly examine the deformable model approach, primarily on the basis of the reviewed methods.

Boundary Model

The boundary model is concerned with expressing qualities of boundaries as observed in the image. Two aspects are considered: the type of a priori information that is encoded onto the boundary model and the type of geometrical representation for the boundary model. When aiming at employing all available information a boundary model is chosen on the basis of its capacity to capture (priori) boundary information about shape, gray-level appearance and the variations therein.

Prior boundary information can range from very general formulations, such as "smooth object" to very specific where detailed information about the expected shape and gray-level appearance of the boundary is captured for every boundary point.

One type of boundary model specifies information about *local shape* of an object, its size and location in the image. This is the case in the snake model [61] where the boundary model is not biased towards a specific shape. Rather it tries to adhere to a geometric constraint that accidently may describe the shape of the object of interest. The generality of such constraints is appealing. However, it also means that complex shapes are hard to deal with as they cannot not be expressed geometrically or analytically.

Another type of boundary model captures *global shape* information about an object boundary. This is the case in the parametrically deformable model [114] where the prior information is encoded in the boundary representation. The boundary model can not take any form, rather is restricted to a certain shape range. The shape parameters, in this case, define a specific geometric structure. They bias the boundary model towards this structure. This guarantees almost always a feasible shape as the outcome of the segmentation. That does not imply the shape is located at the right place for each specific image, nor does it imply the shape is a proper answer to the data configuration. In addition, it also reduces the generality of the method to the extent that many complex objects cannot be approached by this approach.

Another frequently recurring type of boundary model captures *local shape and image information.* This is the case in the piecewise deformable model [88]. Besides information about the local shape of the object the boundary model also specifies expected gray-level appearance along the boundary. Boundary information is captured piecewise so as to capture the variety of information along the contour of heterogeneous boundaries. As a consequence the range of objects it can handle is large. The drawback is that often a detailed configuration of the expected boundary needs to be constructed manually prior to segmentation. Many parameters need to be estimated making the boundary model instable.

Finally several boundary models capture *global shape and gray-level appearance.* The deformable eye template of Yuille [131] and the facial shape and appearance model of Cootes et al. [70] are an example of this. In the eye template a geometric primitive represents the shape of a boundary. Each such primitive is associated with an expected gray-level appearance. The face model takes a similar approach but specifies global shape and grey-level appearance separately. This is desirable as such global models allow interaction between parts of the model that are far away from each other. However when considering interactive segmentation, this may also be viewed as a drawback as local adjustments by the user will always lead to modifications at parts where this is not wanted.

Boundary representation deals with the geometric description of boundaries. We follow [59] in making a distinction between free-form boundary models and parametric boundary models.

Free form boundary models can represent any arbitrary shape as long as some general regularization constraint such as continuity or smoothness are satisfied. The snake method and the spline-based piecewise deformable model have a free form boundary model. The boundary model can take any form and is usually hard to bias towards specific non-circular geometrical structures.

Parametric deformable models are capable of encoding specific characteristic shape and shape variation. The shape can either be characterized by a geometrical or analytic formula, templates [131], or using a prototype and variation modes such as in the Fourier model in [114] and the point distribution model in [70]. Prototype-based models are generally more flexible since prior probability distribution is used to constrain the model to vary within a set of allowed shapes.

We opt for specifying both shape and gray-level information in the boundary model because this is more informative and hence leads to more accurate and robust results. Apart from this, we suggest that boundary models should have a geometrical representation which allows free forms that are confined to feasible forms by shape and image criteria encoded in the objective function. This permits to handle a wide range of structures. In addition, it facilitates local steering by the user. The piecewise deformable model [88], the grammatical model [90] and appearance model of Cootes et al. [70] are most appropriate to this end.

Objective Function

The objective function is concerned with providing a quality to boundaries in the image. Two aspects are considered: boundary properties and boundary configuration.

As concerns boundary properties, the quality is determined on the basis of general properties such as "highest image gradient points" or on the basis of specific boundary information such as "specific image gradient values with their variations". In both cases boundary properties are expressed in terms of image and shape features.

Few variations are found in literature with regard to shape features. Shape is commonly based on bending information, such as (isophote) *curvature*, which is a popular descriptor due to its invariance to rotation and translation. An emerging class of deformable models, the statistical active shape [22] and appearance models [19], evaluates shape on the basis of prior probability distributions of boundary point coordinates. In this case boundary information in a training set is used to learn boundary properties (and their variation).

A large variety of features is found in literature to capture the image boundary properties. They are broadly divided into three types. In the first, boundaries in the image are defined on the basis of *edge information*, including distance information [17] and orientation information [130]. The most common type being used is differential image information, popular because of the invariance properties. The disadvantage is that image information is used only associated with the boundary of an object, whereas the interior may contain relevant information as well.

In the second type, objects in the image are defined on the basis of *region information*, generally specified on the basis of homogeneity or texture. The advantage is that the region of interest does not need to be detected directly from the initial one: as long as a good model of the data comprising the region as a whole is available, the region can grow over arbitrarily large distance to find the desired object region. The disadvantage is the lack of built-in constraints on connectivity of the extracted region and the smoothness of the boundary.

The third occurring type combines edge information with region information forming ***hybrid information.*** Edge information is either integrated with region information, as is the case in [113], or edge information and region information are used competitively, as is the case in [132]. The integration of various image cues of different origin leads to more robust results. Hence, they are preferred over single cue image information.

We consider learning shape and images features, their values and variations therein as a promising approach. Learning from a training set relieves the user from setting feature values and model parameters to specify an object. More importantly, feature values and model parameter settings are more accurate and reproducible and are based on natural observations rather than on user predictions.

Boundary configuration considers the arrangement of properties along the boundary. There are generally two types of configurations. ***Homogeneous*** boundaries in the image are defined using the same image and shape features. Although this is alluring from computational point of view, the homogeneity assumption is expected to fail for many segmentation applications, simply because most objects exhibit local inhomogeneities. Despite the broad range of features in literature, the majority of the deformable models do not take into account in their objective function the diversity of shape and gray-level appearance along boundaries.

The alternative is to consider object boundaries as ***heterogeneous.*** In this case, a repertoire of features along an object boundary is used to qualify it. Deformable models that scrutinize multiple features at each point along the boundary are expected to outperform those that omit to do so. They can be considered as a superset of homogeneous methods. One of the few reported methods which take into account heterogeneous boundaries are the piecewise deformable model [88] and the active contour with grammatical model [90]. The active shape and appearance model inherently take into account the inhomogeneities along object boundaries in the learning phase.

In conclusion, considering the increasing number of digital images, and the complexity of objects in medical images, it is natural to learn multiple image and shape features along object boundaries and to qualify an object boundary heterogeneously. We view the piecewise deformable model and the active shape and appearance model as most appropriate to this end.

Optimization

Optimization is concerned with actively finding the most optimal boundary in the image. Optimization departs from an initial estimate, usually provided by a higher-level mechanism or the user, assigning initialization an important role in optimization because it influences the final outcome of optimization greatly. Types of optimization strategies found in literature can be described according to their search area.

Global optimization tries to find the most optimal boundary in the entire image. An example of such an optimization is the dynamic programming in [1]. This type of optimization increases the accuracy of segmentation results because the deformable model is less likely to be trapped in a local minima. However, it is not efficient because it performs an exhaustive search.

Local optimization tries to find the most optimal boundary in the neighborhood of the initial model. The greedy algorithm [128] performs such a local search. This type of optimization is efficient but requires a good initialization to prevent being stuck into local minima. Initialization is an inherent difficulty in deformable models as the typically non-convex objective functions might have many local minima that mislead the model. In our interactive setting, we leave this problem to the user, making local a optimization strategy an appealing approach from computational efficiency point of view.

Oriented optimization tries to find the most optimal boundary on one side of the deformable model. The balloon [16] performs such an oriented optimization. An initialization of the initial model is required within the area bounded by the boundary of the object of interest or outside this area or else the balloon is not able to find the object boundary. Also, if part of the boundary has weak edge properties, this type of optimization can push the contour over that part towards a stronger edge in the vicinity. The disadvantage is that situation are difficult to handle where, parts of a model need to be inflated while other places need to be deflated.

Confined optimization tries to find the most optimal boundary in a confined area on both sides of the deformable model. This type of optimization is found in the dual snake [51]. Starting from two contours, one outside the boundary of interest and one inside the boundary of interest, the two contours are deformed towards each other to find the optimal boundary, capable of overcoming local minima in the energy landscape. The drawback here is that correspondence between the two contours has to be established and maintained during optimization.

Constrained optimization tries to find the most optimal boundary in the image by gradually optimizing specific points on the model. The ziplock [84] snake performs a constrained optimization. It first optimizes the outer points of an open contour and continues inward with the optimization of the other points under specific constraints, hence it is very efficient. This type of optimization generally requires a higher-level mechanism to decide which part of the model to optimize first. A disadvantage may be that wrongly optimized initial points negatively affect the remainder of the optimization.

We conclude that in order to achieve accurate results in an efficient manner, it is natural to use a local optimization strategy. The greedy algorithm is a positive example of such a strategy, requiring only an initialization close to the true boundary. In case prior information is present about the location, shape and appearance of the object boundary, it is obvious to select a local constrained optimization strategy which decreases even more the search space by taking into account this information. In this case, optimization like in the ziplock snake [84] is most appropriate.

2.3.4 Discussion

To summarize, we strive to qualify objects on the basis of boundary information rather than on the basis of area information. This is expected to facilitate human-computer interaction. Object boundaries are considered to be inhomogeneous and therefor require definition of multiple features. We choose to represent objects by local shape as well as image information. This combination provides a more detailed boundary description, and hence allows more accurate and controllable segmentation. Furthermore, in light of the interactive setting we endorse and for efficiency purpose we opt for a local optimization strategy. Of the reviewed method, the piecewise deformable model [88] and the active shape and appearance model, seem most suitable for interactive segmentation. While the piecewise deformable offers nice properties that we can take over in designing the interactive part of segmentation, the concept of learning features in the active shape and appearance models is very appealing for the computational part.

2.4 Research Questions

In this chapter we have reviewed a number of deformable model methods in literature, taking the point-and-click type of human computer interaction with visual object models as a basis for further investigation. We have examined deformable models in terms of their boundary model, their objective function and their optimization. This examination has given us a better insight in the benefits and drawbacks of deformable models for interactive image segmentation. It has made clear that while offering a promising platform for combining computation with interaction in an elegant way, deformable models lack the capacity to handle objects that are fractured, occluded, convoluted or inhomogeneous otherwise.

As medical images often contain anatomical structures with boundaries that are locally missing, vague, overlapping or abnormal and hence that violate the homogeneity assumption under which many deformable models operate, the above considerations rationalize the following two research questions. In the first place: *how can deformable models be exploited to capture objects that require the definition of multiple boundary features?* If deformable models are capable of capturing such objects: *how can they be used in combination with user interaction to exploit local boundary inhomogeneities in favor of optimizing user input?* These questions form the underlying motivation for the research described in this thesis. Inspired by the appealing properties of piecewise deformable models and active shape models we will conduct further research on the topic of inhomogeneous variational deformable models.

Chapter 3

Strings: Variational Deformable Models of Multivariate Ordered Features[*]

We propose a new image segmentation technique called strings. A string is a variational deformable model that is learned from a collection of example objects rather than built from a priori analytical or geometrical knowledge. An object boundary is represented by a one-dimensional curve in multivariate functional space rather than by a point as is the case in active shape models. In the learning phase, multiple shape and image feature functions along continuous object boundaries in a learning set are aligned, then subjected to functional principal components analysis and functional principal regression to model the feature space. A Mahalanobis distance model takes the natural variations in the learning set into account for evaluation of boundaries. In the segmentation phase, an object boundary in a new image is searched for with help of a deformable feature function, a string. The string is weighted by the regression model and evaluated by the Mahalanobis model. A curve is deformed in the image to produce feature functions with minimal distance. Strings have been compared with active shape models on 145 vertebra images, showing that strings produce better results when initialized close to the target boundary, and comparable results otherwise.

[*] Conditionally accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence

3.1 Introduction

In advanced image segmentation problems object boundaries frequently have inhomogeneous characteristics. The shape of a target boundary may be blunt at some parts and strongly convoluted at other parts. Or, the image gradient along a target boundary may be clearly visible and pointing outwards at some places, while it is hardly defined due to neighboring objects at other places. We note that in many advanced segmentation problems boundaries are fractured, occluded, convoluted or inhomogeneous otherwise, requiring the definition of multiple features for accurate description of the boundary. For this reason, it is imperative to construct inhomogeneous boundary models for application in image segmentation. We learn such boundary models by exploring the information contained in large collections of example images rather than constructing them from analytical or geometrical knowledge rules.

Learning in the context of boundary-based image segmentation has received considerable attention in literature, e.g. in [12], [43], [58], [71], [114], [108], in particular within the active shape model framework [22]. In active shape model methods, a boundary is learned by statistical analysis of feature values from a set of example boundaries. During image segmentation, the learned boundary model is used as a reference for deformation of an active shape and for evaluation of a boundary recorded by that shape. The generic approach of active shape models is very appealing. However, they often fail to fully exploit the multivariate continuous characteristics of a boundary. The question that is raised here is how multiple boundary features such as edge gradient and contour curvature, are exploited for learning a continuous variational image segmentation model.

We propose a unified approach to learning structurally different boundary features for multi-feature image segmentation. The problem of learning is transposed into one of analyzing the closed functional curves in feature space that best describe average feature values and the most important variations therein. Image segmentation is conceived of as an iterative procedure of recording multiple continuous feature values, weighting these feature values to amplify the statistically most descriptive features and evaluating the weighted features values with respect to the values seen in the learning set. The proposed segmentation approach combines theory from functional data analysis [95] with theory from chemometrics [69] to arrive at string segmentation models.

The chapter is organized as follows. Section 2 discusses previous work on active shape models. Section 3 introduces the string method. First the learning phase will be described in detail, then image segmentation with help of deformable strings is addressed. Implementation issues, experiments and results are described in section 4. Discussion and conclusion follow in section 5.

3.2 Related Work

A number of active shape models have been presented in literature. The method introduced by Cootes et al. in [22] makes use of established statistical techniques to construct a shape model from examples. In the reference, a shape is represented as an N-vector of vertices

$$\mathbf{x} = [(x_1, y_1), ..., (x_N, y_N)]^T. \tag{3.1}$$

Assuming two-dimensional vertices, each N vertex is a single point in a $2N$-dimensional vector space. The set of M shape examples subsequently forms a learning set of size M

$$\mathcal{L} = \{\mathbf{x}_1, ..., \mathbf{x}_M\}. \tag{3.2}$$

To remove variation from the learning set attributed to stretching, shearing and rotation of the shapes, the example shapes are aligned by Procrustes analysis [50], aiming at minimizing the sum of distances of each example shape to the average

$$\epsilon_{\mathcal{L}} = \sum_{m=1}^{M} ||\mathbf{x}_m - \bar{\mathbf{x}}||^2 \tag{3.3}$$

where $\bar{\mathbf{x}}$ is an initial estimate of the average shape, with $||\bar{\mathbf{x}}|| = 1$. In an iterative procedure, the M shape examples are aligned one-by-one, with each iteration refining the estimation of the average shape.

Assuming the cluster of aligned shapes forms an ellipsoid, in [22] principal components analysis is performed to reduce the dimensionality of the data using the covariance matrix

$$\mathbf{C}_{\mathcal{L}} = \frac{1}{M-1} \sum_{m=1}^{M} (\mathbf{x}_m - \bar{\mathbf{x}})(\mathbf{x}_m - \bar{\mathbf{x}})^T \tag{3.4}$$

where $\bar{\mathbf{x}}$ now denotes the average shape of the aligned example shapes. The eigenvalues λ_n of $\mathbf{C}_{\mathcal{L}}$, with $\lambda_n \geq \lambda_{n+1}$ for $n = 1, ..., 2N$, tell the amount of variance captured by each principal component. The largest fraction of the total variance is given by the first $Q \leq 2N$ eigenvalues

$$\lambda = \sum_{q=1}^{Q} \lambda_q. \tag{3.5}$$

The eigenvalues λ_q describe the most significant modes of variation and the corresponding eigenvectors describe the dimensions in which they occur. A shape instance $\mathbf{x}$ is explained as the mean shape plus some linear combination of these eigenvectors

$$\mathbf{x} \approx \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}^T \tag{3.6}$$

where the $Q \times 2N$ matrix $\mathbf{P}$ contains the first Q eigenvectors and $\mathbf{b}$ is a Q-vector of weighting coefficients. New plausible shapes are generated by varying $\mathbf{b}$ within suitable limits, derived by statistically examining the distribution of the weighting coefficients required to generate the learning data. On the basis of these feasible shapes high gradient boundaries are searched for in the image.

A number of shortcomings of the active shape model in [22] have been recognized and dealt with in literature.

One problem arises when the learning set $\mathcal{L}$ is contaminated with outliers, influencing the statistics adversely. For this reason, Duta et al. [35] remove outliers in an iterative approach using an inter-shape distance matrix that defines the mean alignment error between a polygonal approximation of a shape instance and an original shape from the learning set. The reduced learning set forms the basis for the construction of a less distorted statistical shape model. Application of the corrected shape model for image segmentation consequently leads to better performance. We adopt this idea and we remove outliers in one step using another distance measure.

Another problem is that minimization of equation 3.3 only takes into account pose and scale differences between instances and does not account for non-linear shape differences. This is solved by Duta et al. [35] using a flexible point matching technique that performs global similarity registration of two arbitrary sets of points and non-linear registration based on local similarity of two curves. Instead of minimizing equation 3.3, the trade-off between a compensated mean alignment error and the number of correspondences is minimized. This way the effect that unconstrained linear registration of two sets of points tends to shrink [42] one set with respect to the other is avoided. In addition, in [35] no correspondence between points is required when performing non-linear registration. Other work on the shape alignment problem includes the work by Kotcheff et al. [68], who optimize model compactness rather than variance. We also aim at linear and non-linear registration while avoiding problems with the cardinality of point sets.

A restriction of point distribution models is that deformation of the shape model by adjusting weighting vector $\mathbf{b}$ only allows limited deformation reflecting the variations in the learning set. For this reason, Wang et al. [125] use prior models based on principal component analysis of additional covariance matrices. By replacing equation 3.4 with a weighted and mixed covariance matrix that deals with independence and smoothness they are capable of building a wide range of shape models even when there are few examples shapes or if the learning set exhibits small variation. An earlier attempt to add artificial variation to the statistical model is found in [21]. We take a different road aiming at building models also capable of explaining objects dissimilar to the ones in the learning set.

An other limitation of active shape models concerns their inability to statistically capture image features around shape models. Cootes et al. [24] solve this problem by taking samples of the image intensity perpendicularly to a shape. By recording intensity profiles for all labeled points of the shape model, they arrive at an augmented learning set. As with the shape data, they compute the statistics of the intensity data using equations 3.1-3.6. The image features improve model specificity and hence segmentation accuracy. Other work on modeling image features includes Van Ginneken et al. [122], who propose a selection of optimal features by non-linear classifiers. We adopt the idea of modeling both shape and image features.

In [24], it is assumed that the image feature values sampled at different points along the boundary are independent of one another while commonly feature values are spatially highly correlated. Hence, independence of image feature values along a boundary cannot always be assumed. For this reason, Haslam et al. [56] propose a probabilistic fitness measure using concatenated intensity profiles, bringing in some continuity in feature values. This way, in addition two resolving the issue of dependence, they reduce the number of point distributions to two. Other work acknowledging and adhering to the often continuous characteristic of boundary features, particularly shape features, is found in e.g. [68], [27], [5], [56], [101], [71]. We embrace the thought of capturing spatial correlations among image features and we extend this idea to arbitrary features.

To overcome the problem of high correlation between structurally different features [56], Edwards et al. [37], and more recently Cootes et al. [19], propose an active appearance model that couples shape and image models more explicitly by using a single vector containing both shape and image feature values. By concatenating the feature values into a single vector and representing this vector as a point in a very high-dimensional space, a learning cluster is formed of combined shape and image features. This way a more specific boundary model is obtained and more accurate image segmentations are achieved. We adopt and generalize the idea of capturing correlations between structurally different boundary features.

In conclusion, many of the problems addressed by the aforementioned methods emanate from the discrete point representation of boundaries. Discrete points lead to discretization problems when point sets have different numbers of elements, requiring pseudo-continuous solutions. In addition, in discrete point representations, spatial interdependencies between features and correlations among structurally different features are not reckoned appropriately. Also, salient features in the learning set are insufficiently exploited when specified a priori rather than selected from the data by optimality. Finally, information valuable for model specificity is lost in the currently applied procedures for dimensionality reduction.

It is the purpose of this study to address these problems integrally. We do this with a technique we call *strings*.

3.3 Strings by Functional Data Analysis

Object boundaries are continuous. Hence, they should be represented by curves rather than by a collection of discrete points. Apart from this, boundaries are often described by multiple features. Hence they should be represented by multivariate models rather than by a combinations of univariate models. These boundary characteristics motivate the representation of boundary features by multi-variate curves in functional space, rather than by points in vector space. When boundaries are represented this way, the learning and segmentation problems can be solved by functional data analysis [95] (see figure 3.1).

Each of the components in figure 3.1 will be described in more detail in the following subsections. In section 3.4 we will again elaborate on these components one

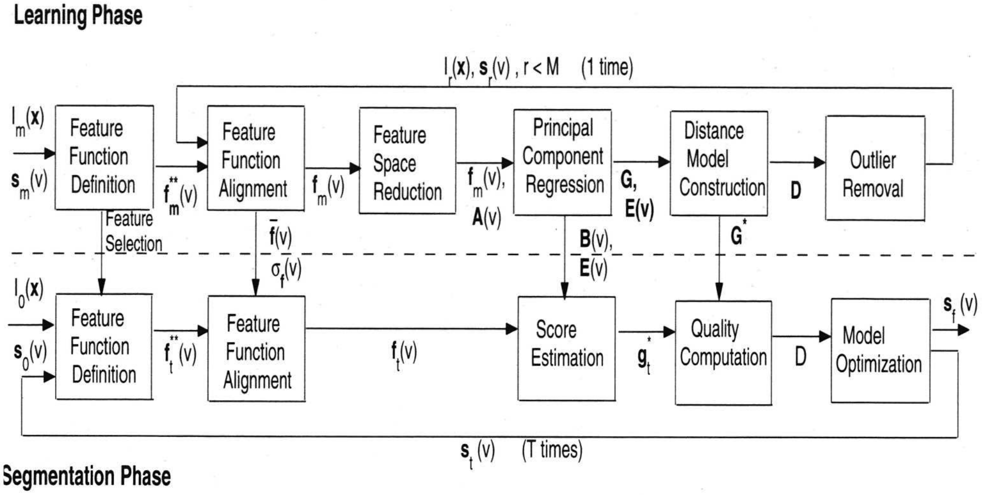


Figure 3.1: Overview of the string segmentation technique. Note that at the start of the learning phase, we need a set of images $I_m(\mathbf{x}), m = 1...M$ with corresponding known segmentations $\mathbf{s}_m(v)$. Invariably, in this chapter bold face upper case indicates a matrix of functions, e.g. $\mathbf{A}(v), \mathbf{B}(v), \mathbf{E}(v)$ or in the case of $\mathbf{G}$ of scalars, bold face lower case indicates a vector of functions, e.g. $\mathbf{f}_t$, or scalars, e.g. $\mathbf{g}_t^\star$, and regular lower case indicates a function or a scalar.

by one to show the (intermediate) results for a vertebra application.

3.3.1 Feature Function Definition

In order to construct a statistical boundary model, example features need to be specified and their values computed. Commonly, the feature values are a set of coordinate values of some boundary points and a set of image gradient values recorded orthogonally to those boundary points, see for example [20], [37], [19], [35].

We explore boundary features in example images using their known segmentation, represented by smooth curves $\mathbf{s} : \Re \to \Re^2$ parameterized by $v \in \Re$. Given the set of M input images $I_1(\mathbf{x}), ...I_M(\mathbf{x})$ with corresponding segmentations $\mathbf{s}_1(v), ...\mathbf{s}_M(v)$, the learning set consists of pairs of image and shape data (compare to equation 3.2)

$$\mathcal{L} = \{(I_1(\mathbf{x}), \mathbf{s}_1(v)), ..., (I_M(\mathbf{x}), \mathbf{s}_M(v))\}. \tag{3.7}$$

For the mth learning example, the shape $\mathbf{s}_m(v)$ relates to the image at points $I_m(\mathbf{s}_m(v))$. The relation is expressed in terms of N features derived from the shape (e.g curvature) as well as from the image (e.g. isophote curvature). The mapping $\mathbf{f} : \Re \to \mathcal{F}$ takes care of this, yielding feature functions $\mathbf{f}_m^{**}(v)$ in the N-dimensional functional space $\mathcal{F}$, where each dimension corresponds to one feature, i.e.

$$\mathbf{f}_m^{**}(v) = [f_{m1}^{**}(v), ..., f_{mN}^{**}(v)]. \tag{3.8}$$

To capture image and shape features on and off a boundary we exploit the local Taylor expansion [64] of the image up to the second order, sampled at discrete but dense points along the boundary. In this way, we have an approximately complete two-dimensional description of local boundary properties. Later, the principal components analysis, yet to be described, will cancel out linear dependencies in the Taylor set.

3.3.2 Feature Function Alignment

For statistical analysis of example boundaries, feature values at one point on an example boundary need to be compared with values at an equivalent point on an other example boundaries. Commonly, this is achieved by scaling, rotating and translating the examples so that they correspond as closely possible. This reduces to aligning a set of discrete points, when boundaries are represented by point distribution models as in [20], [37], [19].

In the context of our functional data, the alignment problem is a curve registration problem. Feature functions $\mathbf{f}_m^{**}(v)$ may differ due to the fact that they are not measured at the same path position v or due to small non-linear differences. A shift of feature values along v and a non-linear warping account for this. Alignment reduces to finding the warping function $\omega_m(v)$ that produces the warped feature function

$$\mathbf{f}_m^{*}(v) = \mathbf{f}_m^{**}(\omega_m(v)). \tag{3.9}$$

The warping function $\omega_m(v)$ is strictly monotonic and differentiable up to a certain order. It takes care of a shift and a non-linear transformation by the roughness penalty approach described in [95]. In this case, we penalize by the size of the third derivative of $\omega_m(v)$.

Alignment of $\mathbf{f}_m^{**}(v)$ is done by the Procrustes method [50] using a global alignment criteria that computes the least squares distance to $\hat{\mathbf{f}}(v)$, the overall average feature function. This reduces to finding $\omega_m(v)$ such that

$$\omega_m(v) = \underset{\omega_m^*(v)}{\operatorname{argmin}} \sum_{m=1}^{M} \int_v ||\mathbf{f}_m^{**}(\omega_m^*(v)) - \hat{\mathbf{f}}(v)||^2 dv. \tag{3.10}$$

The warping functions are estimated in an iterative process where argument values for a feature function are shifted and transformed so as to minimize the least squares error. The estimated average $\hat{\mathbf{f}}(v)$ is updated by re-estimating it from the partially aligned feature functions.

The final average feature function $\bar{\mathbf{f}}(v)$ is computed from the aligned set. It is subtracted from each feature function to normalize the range of feature values. This yields

$$\mathbf{f}_m(v) = \frac{\mathbf{f}_m^*(v) - \bar{\mathbf{f}}(v)}{\sigma_{\mathbf{f}}(v)} \tag{3.11}$$

with units of variance due to normalization by the variance vector of functions

$$\sigma_{\mathbf{f}}(v) = \left(\frac{1}{M}\sum_{m=1}^{M} ||\mathbf{f}_m^*(v) - \bar{\mathbf{f}}(v)||^2\right)^{1/2}. \tag{3.12}$$

Normalization is required to reduce the influence of differences in variation of differently measured features. The aligned normalized feature functions $\mathbf{f}_m(v)$ contain all information needed to statistically summarize features into a boundary model.

3.3.3 Feature Space Reduction

We perform principal components analysis to project the high dimensional functional data to a smaller feature space expecting that the essential structure in the original data is preserved. This is admissible as long as the features exhibit a small number of modes of variation, covering a large part of the variability in the data.

Functional principal component analysis [31] computes the main modes of variation in the collection of N-dimensional feature functions $\mathbf{f}_m(v)$. The number of modes to retain is derived from a given proportion of the variance as explained in the learning set. When the modes are numbered by $q = 1, ..., Q$, the central concept is that of taking the linear combination

$$g_{mq} = \sum_{n=1}^{N} \int_v f_{mn}(v)\alpha_{qn}(v)dv, \tag{3.13}$$

where $\alpha_{qn}(v)$ denotes a weighting function chosen so as to highlight variation in the data in dimension n. As before, $f_{mn}(v)$ is the nth dimension of the mth observed feature function. The values g_{mq} are the principal component scores. They will be used to produce more robust descriptions [32].

To obtain the value of g_{mq} for all q the corresponding vectors of weighting functions $\boldsymbol{\alpha}_q(v) = [\alpha_{q1}(v), ..., \alpha_{qN}(v)]$ need to be computed. They are sought for one-by-one in such a way that they explain most of the variation in the learning data

$$\begin{array}{rcl} \boldsymbol{\alpha}_q(v) & = & \underset{\boldsymbol{\alpha}_q^*(v)}{\operatorname{argmax}} \; \frac{1}{M}\sum_{m=1}^{M} g_{mq}^2 \\ & = & \underset{\boldsymbol{\alpha}_q^*(v)}{\operatorname{argmax}} \; \frac{1}{M}\sum_{m=1}^{M} \left(\sum_{n=1}^{N} \int_v f_{mn}(v)\alpha_{qn}(v)dv\right)^2 \end{array} \tag{3.14}$$

where $\boldsymbol{\alpha}_k(v)$, for each iteration k, is subject to the following orthonormal constraints

$$\sum_{n=1}^{N} \int_v \alpha_{qn}(v)^2 dv = 1 \tag{3.15}$$

$$\sum_{n=1}^{N} \int_v \alpha_{kn}(v)\alpha_{qn}(v)dv = 0, k \leq q. \tag{3.16}$$

Equations 3.15 and 3.16 ensure that the vector $\boldsymbol{\alpha}_1(v)$ contains most of the independent variation. After $\boldsymbol{\alpha}_1(v)$ has been established, the above process is continued until all Q significant modes of variation, each described by $\boldsymbol{\alpha}_q(v)$, are obtained.

The matrix of functions $\mathbf{A}(v) = [\boldsymbol{\alpha}_1(v), ..., \boldsymbol{\alpha}_Q(v)]^T$ indicates where along the boundary there is independent variation in the learning ensemble. $\boldsymbol{\alpha}_1(v)$ captures the location with largest correlated variation, $\alpha_2(v)$ the second largest of the remaining variance and so on until most of the variation is explained. Hence, the functional principal components capture the most important feature subspace by the matrix of weighting functions $\mathbf{A}(v)$.

3.3.4 Principal Components Regression

To construct an underlying model of feature values, the distribution of feature values seen in the learning set needs to be captured in statistical terms. In the active shape model literature [20], [37], [19], [35], feature reconstruction is done after projecting the features to the space spanned by the most important principal components. Hence, only part of the data in the learning set is subjected to modeling. We perform principal component regression [14] to obtain a predictive model from the feature functions in the learning set.

We define the matrix of functions $\mathbf{F}(v) = [\mathbf{f}_1(v), ..., \mathbf{f}_M(v)]^T$ and the matrix $\mathbf{G}$ by

$$\mathbf{G} = \mathbf{F}(v)\mathbf{A}(v) \tag{3.17}$$

with scalar elements according to the dot product defined in equation 3.13. We define a matrix of regression functions $\mathbf{B}(v) = [\boldsymbol{\beta}_1(v), ..., \boldsymbol{\beta}_Q(v)]^T$, with elements of the same N-dimensional functional form as the elements of $\mathbf{F}(v)$. To find the values of $\mathbf{B}(v)$, the matrix of feature functions $\mathbf{F}(v)$ is expressed as

$$\mathbf{F}(v) = \mathbf{G}\mathbf{B}(v) + \mathbf{E}(v) \tag{3.18}$$

with $\mathbf{E}(v) = [\boldsymbol{\epsilon}_1(v), ..., \boldsymbol{\epsilon}_M(v)]^T$ being the matrix of residual functions yet to be defined. Instead of regression on the original feature data, regression is performed on the principal component scores containing information on how the feature samples correlate with one another. The matrix of regression functions $\mathbf{B}(v)$ gives an estimate of how the principal scores relate to the feature functions and what the contribution of each is towards defining an unknown feature function. The regression functions are computed by least squares minimization such that

$$\mathbf{B}(v) = \underset{\mathbf{B}^*(v)}{\text{argmin}} \sum_{m=1}^{M} \int_v ||\mathbf{f}_m(v) - \mathbf{g}_m\mathbf{B}^*(v)||^2 dv. \tag{3.19}$$

Since there are no particular restrictions on the way in which the matrix of functions $\mathbf{B}(v)$ varies as a function of v, the solution can be obtained by minimizing the least squares difference for each v separately. After least squares minimization we have

$$\mathbf{E}(v) = ||\mathbf{F}(v) - \mathbf{G}\mathbf{B}(v)||^2. \tag{3.20}$$

With help of the estimated regression functions we predict scores for an unknown feature function in the segmentation phase, reconstruct it in reduced space according the regression model and evaluate it by examining the distance of its score to the cluster of scores corresponding to the feature functions in the learning set.

3.3.5 Mahalanobis Distance

In order to determine how a single boundary relates to the collection of boundaries in the learning set, a distance measure needs to be defined. In the active shape model literature, the evaluation of boundary feature values is performed in terms of the Mahalanobis distance, i.e. the distance to the average normalized by the variation in each dimension. Following Cootes et al. [20], we use a Mahalanobis distance model [32] to compute the distance of a feature function to the average of the learning set.

The Mahalanobis distance model is obtained by augmenting $\mathbf{G}$ with the vector of residuals $\boldsymbol{\epsilon} = [\epsilon_1, ..., \epsilon_M]$ with elements defined as

$$\epsilon_m = \frac{1}{N}\sum_{n=1}^{N}\left(\int_v \epsilon_{mn}(v)dv - \frac{1}{M}\sum_{m=1}^{M}\int_v \epsilon_{mn}(v)dv\right). \tag{3.21}$$

This yields an $(Q+1) \times M$ augmented principal components scores matrix $\mathbf{G}^{\star} = [\mathbf{G}, \boldsymbol{\epsilon}^T]$. This improves the robustness of Mahalanobis distance calculation [32]. The Mahalanobis distance matrix is then defined as

$$\mathbf{D} = \frac{\mathbf{G}^{\star T}\mathbf{G}^{\star}}{(M+1)}. \tag{3.22}$$

The Mahalanobis distance of $\mathbf{f}_m(v)$ to the average $\bar{\mathbf{f}}(v)$ is computed using $\mathbf{g}^{\star}_m = [\mathbf{g}_m, \epsilon_m]$ as follows

$$D^2(\mathbf{f}_m(v), \bar{\mathbf{f}}(v)) = \mathbf{g}^{\star}_m \mathbf{D}^{-1} {\mathbf{g}^{\star}_m}^T. \tag{3.23}$$

Note that D depends on $\mathbf{g}^{\star}_m$. Hence, the distance model not only takes into account variations encountered in the learning set but also the valuable additional discriminating factor of residual information [69]. In the segmentation phase, the Mahalanobis distance model will be used as an objective function to find a boundary in a new image from which a feature function emanates similar to the ones seen in the learning set.

3.3.6 Stochastic Outlier Removal

As outlier feature functions in the learning set can have a severe influence on the discrimination ability of the Mahalanobis distance model, they are removed following [35]. Those with a Mahalanobis distance exceeding a threshold τ_M are considered outliers. For a Gaussian distribution, $\tau_M = 3$ corresponds to removal of all instances that have a 1% probability of belonging to the class. These instances are removed, resulting in a new learning set in which each element has a Mahalanobis distance of τ_M or less. Learning is done once again to come up with more appropriate regression and distance models. For simplicity of notation, we assume in the following that the quantities are computed from the reduced set rather than from the full set of feature functions used this far.

3.3.7 Deformable Strings

Having constructed a statistical boundary model in the learning phase, in the segmentation phase we use this model as a reference for finding a boundary in a new image. To this end, we use the following information from the learning phase: the average feature function $\bar{\mathbf{f}}(v)$, the normalization function $\sigma_{\mathbf{f}}(v)$, the Mahalanobis distance model $\mathbf{G}^{\star}$ and the regression functions matrix $\mathbf{B}(v)$.

Now, consider the active feature function $\mathbf{f}_t^{**}(v)$, which will be deformed in time t. We call it a string. The string lives in the N-dimensional feature space $\mathcal{F}$, and hence we can compute its Mahalanobis distance to $\bar{\mathbf{f}}(v)$. As in the learning phase (see equation 3.9), we first obtain

$$\mathbf{f}_t^{*}(v) = \mathbf{f}_t^{**}(\omega_t(v)), \tag{3.24}$$

where, similar to equation to 3.11, the warping function $\omega_t(v)$ is computed such that

$$\omega_t(v) = \underset{\omega_t^*(v)}{\operatorname{argmin}} \int_v ||\mathbf{f}_t^{**}(\omega_t^*(v)) - \bar{\mathbf{f}}(v)||^2 dv. \tag{3.25}$$

We normalize the feature function in analogy to equations 3.11 and 3.12. Normalization yields

$$\mathbf{f}_t(v) = \frac{\mathbf{f}_t^{*}(v) - \bar{\mathbf{f}}(v)}{\sigma_{\mathbf{f}}(v)}. \tag{3.26}$$

The quality of $\mathbf{f}_t(v)$ with respect to the reduced feature functions matrix $\mathbf{F}(v)$ is determined from the relation of its corresponding score vector $\mathbf{g}_t$ to the cluster of example scores contained in $\mathbf{G}$. The vector $\mathbf{g}_t$ is estimated by solving

$$\mathbf{f}_t(v) = \mathbf{g}_t\mathbf{B}(v)^T + \epsilon_t(v). \tag{3.27}$$

That is, the equation estimates the score of a new feature function on the basis of the principal component regression model obtained in section 3.4. The principal component scores are estimated by least squares minimization such that

$$\mathbf{g}_t = \underset{\mathbf{g}_t^*}{\operatorname{argmin}} \int_v ||\mathbf{g}_t^*\mathbf{B}(v)^T - \mathbf{f}_t(v)||^2 dv. \tag{3.28}$$

In analogy to equation 3.20, we augment $\mathbf{g}_t$ with the residual of the least squares minimization as an additional discriminating factor for the string. The $(Q+1)$-vector $\mathbf{g}_t^{\star} = [\mathbf{g}_t, \epsilon_t]$ is obtained on the basis of

$$\epsilon_t(v) = ||\mathbf{g}_t\mathbf{B}(v)^T - \mathbf{f}_t(v)||^2 \tag{3.29}$$

by adding to $\mathbf{g}_t$ the residual sum

$$\epsilon_t = \frac{1}{N}\sum_{n=1}^{N}\left(\int_v \epsilon_{tn}(v)dv - \frac{1}{M}\sum_{m=1}^{M}\int_v \epsilon_{mn}(v)dv\right). \tag{3.30}$$

The Mahalanobis distance of $\mathbf{f}_t^{\star\star}(v)$ is defined as the distance of the newly augmented score $\mathbf{g}_t^\star$ to the average of the cluster of example scores

$$D^2(\mathbf{f}_t(v), \bar{\mathbf{f}}(v)) = \mathbf{g}_t^\star \mathbf{D}^{-1} \mathbf{g}_t^{\star T}. \tag{3.31}$$

We use this quality measure for image segmentation by strings. The string $\mathbf{f}_t^{\star\star}(v)$ is defined by features extracted from an active shape model $\mathbf{s}_t(v)$ and from the image in which that shape model lives. The string vibrates in feature space due to deformations of the shape model in image space. In this iterative procedure, the shape model is freely deformed rather than constructed in the reduced space as is the case in [20] and [37]. Forms of the shape model that are less plausible are punished by rather than prohibited.

3.3.8 Optimization

Finally, we formulate the segmentation problem as an optimization problem. The objective is to find a boundary that gives rise to a feature function with minimal Mahalanobis distance $D(.)$ to the ones seen in the learning set. To this end, the deformable shape model $\mathbf{s}_t(v)$ is deformed in the image $I_0(\mathbf{x})$ to suggest a statistically optimal boundary described by the deformable string $\mathbf{f}_t^{\star\star}(v)$. Starting from an initial shape configuration $\mathbf{s}_{t=0}(v)$ the shape model is deformed by tuning its shape parameters in such a way that the state of minimal energy provides the optimal feature function. This reduces to optimizing $\mathbf{s}_t(v)$ such that

$$\mathbf{f}(v) = \underset{\mathbf{f}_t(v)}{\operatorname{argmin}} \quad D(\mathbf{f}_t(v), \bar{\mathbf{f}}(v)). \tag{3.32}$$

We use simulated annealing [63] for optimization as it distinguishes between different local minima in the energy landscape. Starting off at the initial configuration, a sequence of iterations is generated, where each iteration consists of the random selection of a configuration from the neighborhood of the current configuration and the calculation of the corresponding change in the energy value. By a small perturbation in the neighborhood, a transition is achieved from one configuration into another one. If the change from time t to $t+1$ yields negative $\Delta D = D(\mathbf{f}_{t+1}(v), \bar{\mathbf{f}}(v)) - D(\mathbf{f}_t(v), \bar{\mathbf{f}}(v))$, the transition is accepted unconditionally; if the cost function increases the transition is accepted with a probability based upon the Boltzmann distribution $p = e^{|-\frac{\Delta D}{kT}|}$ where k is a constant and the temperature T is a control parameter. This temperature is gradually lowered throughout the segmentation from a sufficiently high starting value, i.e. a temperature where almost every proposed transition, both positive and negative, is accepted to a freezing temperature, where no further changes occur. The temperature is decreased in stages, and at each stage the temperature is kept constant until thermal quasi-equilibrium is reached.

3.4 Experiments and Results

We illustrate and discuss the use of functional data analysis for construction of the boundary model and the application of strings for image segmentation. We do this step by step according to figure 3.1. To this end, we use 145 annotated and digitized NHANES X-ray images of normal cervical vertebrae, acquired from the National Center for Health Statistics (NCHS) [86]. As can be seen from figure 3.8, the vertebra boundary in these images is ill-defined. It is characterized by the presence of interfering boundaries (e.g. the vertebra above and below), convoluted boundary parts (e.g. tips of the vertebral body), missing image evidence (e.g. at the pedicles) and in this case very poor image quality. These characteristics complicate model construction and image segmentation.

3.4.1 Feature Instantiation

For the description of the boundary we use a repertoire of features. Among the many features that have been proposed in literature (see [44] and [73] for a comprehensive survey of image and shape feature respectively), we confine ourselves to the use of invariant features. Invariant features generalize applicability, but more importantly, they minimize the need for feature alignment. For this reason, we use the features listed in table 3.1.

Dimension	Feature	Definition
1	contour curvature [44]	$\frac{\mathbf{s}_x(v)\mathbf{s}_{yy}(v)-\mathbf{s}_{xx}(v)\mathbf{s}_y(v)}{(\mathbf{s}_x^2(v)+\mathbf{s}_y^2(v))^{3/2}}$
2	isophote curvature [129]	$\frac{I_{xx}(\mathbf{x})I_y^2(\mathbf{x})-2I_x(\mathbf{x})I_y(\mathbf{x})I_{xy}(\mathbf{x})+I_{yy}(\mathbf{x})I_x^2(\mathbf{x})}{(I_x^2(\mathbf{x})+I_y^2(\mathbf{x}))^{3/2}}$
3	directional correspondence [79]	$\nabla I(\mathbf{s}(v)) \cdot \mathbf{n}(v)$

Table 3.1: Features in our implementation defining the dimensions of $\mathbf{f}^{**}(v)$. The first dimension is the contour curvature, the second the isophote curvature and the third the directional correspondence between the normal $\mathbf{n}(v)$ to the shape $\mathbf{s}$ at v and the image gradient at $\nabla I(\mathbf{x})$ with $\mathbf{x} = \mathbf{s}(v)$.

The features listed in table 3.1 and their derivatives up to second order are used. As a consequence $N = 9$ features are measured along each example vertebra boundary. As in this application the example vertebra boundary is represented by 7 discrete points, manually marked in the image by a medical expert, we compute a continuous approximation of the boundary by interpolation of a curve through the 7 points. The curve $\mathbf{s}(v)$ and its corresponding 9-dimensional feature function $\mathbf{f}^{**}(v)$ are represented by splines. They are formulated as tensor product B-Spline curves [93]

$$\mathbf{s}(v; \mathbf{p}) = \sum_{k=1}^{K} B_k(v)\mathbf{p}_k \tag{3.33}$$

$$\mathbf{f}^{**}(v; \mathbf{q}) = \sum_{j=1}^{J} B_j(v)\mathbf{q}_j \tag{3.34}$$

Basis functions $B_k(v)$ correspond to the $K = 7$ manually marked points $\mathbf{p}_k$. The $N = 9$ features, and also point coordinates values, are computed along $\mathbf{s}(v; \mathbf{p}_k)$ at

100 samples. The 100 coordinate values are used for redefinition of the 7-vector of control points $\mathbf{p}_k$ to a 100-vector of control points, to be used in active shape model segmentation (yet to be described). Feature functions $\mathbf{f}^{**}(v; \mathbf{q}_j)$ are defined by basis functions $B_j(v)$, corresponding to the $J = 100$ N-dimensional control points $\mathbf{q}_j$, i.e. the sampled feature values. As these feature values are not always smooth we impose regularity by using basis expansions with a relatively small number of basis functions [95].

3.4.2 Learning Phase

We discuss the results of the learning phase for the vertebra application. We only discuss the 0th-order derivative values of the features, i.e. only the first three dimensions of $\mathcal{F}$.

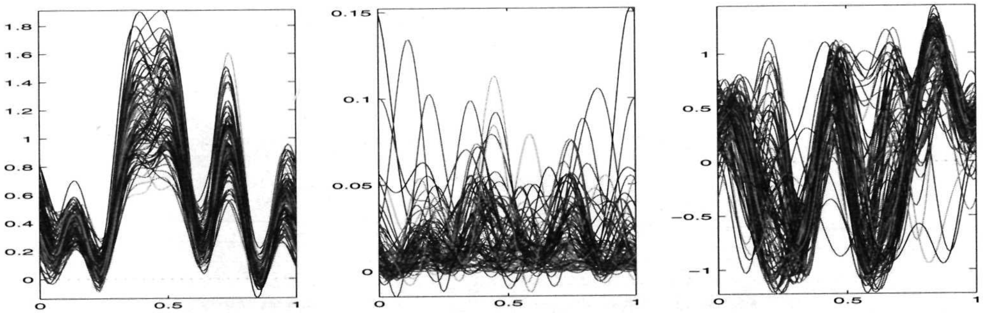


Figure 3.2: Feature values along path parameter v for all $M = 145$ examples. Left: contour curvature values $f^{**}_{m1}(v)$. Middle: isophote curvature values $f^{**}_{m2}(v)$. Right: directional correspondence values $f^{**}_{m3}(v)$.

Figure 3.2 illustrates contour curvature values $f^{**}_{m1}(v)$, isophote curvature values $f^{**}_{m2}(v)$ and directional correspondence values $f^{**}_{m1}(v)$ computed from $\mathbf{s}_m(v)$ and the image data arrays (see equation 3.8). The presence of correlated structure in the contour curvature values is apparent. The peaks in curvature values correspond with the tips of the vertebral body and correlate with the peaks in $f^{**}_{m2}(v)$. This is more clearly seen from the average functions in figure 3.5. Note that the isophote curvature is badly defined along most parts of the vertebra boundary. More correlated structure is seen in the feature values of directional correspondence.

The results of aligning feature functions $\mathbf{f}^{**}_m(v)$ by the iterative Procrustes procedure in equation 3.9 is illustrated in figure 3.3. In the vertebra application, the majority of the feature functions already has a good alignment thanks to the a priori manual registration of the discrete points by medical experts (from which the continuous shapes and feature functions are computed). As a consequence, alignment brings no significant changes to feature functions $\mathbf{f}^{**}_m(v)$. Alignment is more important during segmentation where the starting points for sampling features are unknown.

The effects of normalization of feature functions $\mathbf{f}^{*}_m(v)$ are shown in figure 3.4. Note that both the isophote curvature and directional correspondence show feature

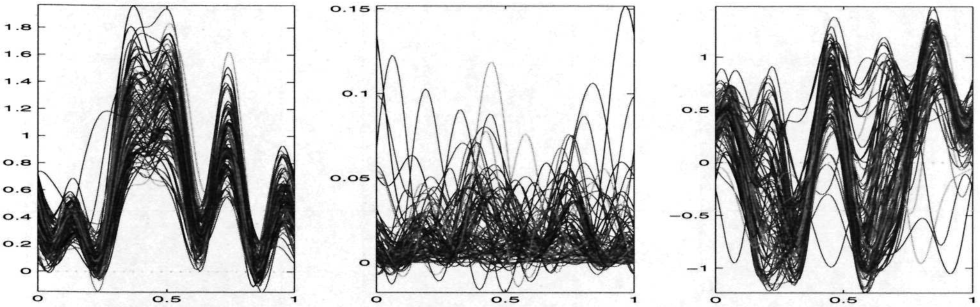


Figure 3.3: Results of registering feature functions vector $\mathbf{f}_m^{**}(v)$. Left: $f_{m1}^*(v)$. Middle: $f_{m2}^*(v)$. Right: $f_{m3}^*(v)$. Note that only three dimensions are shown out of the nine.

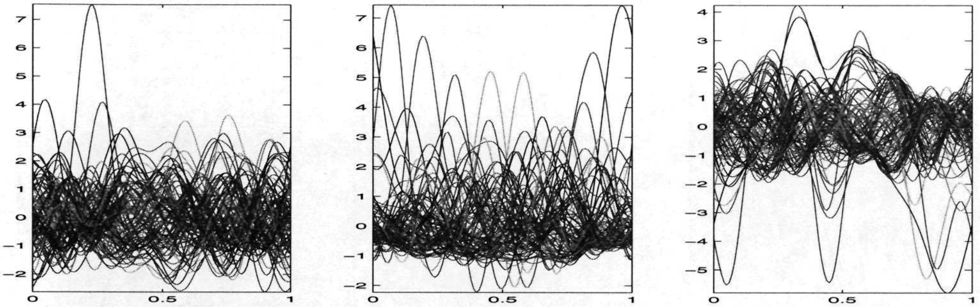


Figure 3.4: Results of normalizing feature functions $\mathbf{f}_m^*(v)$: Left: $f_{m1}(v)$ corresponding to contour curvature. Middle: $f_{m2}(v)$ corresponding to isophote curvature. Right: $f_{m3}(v)$ corresponding to directional correspondence.

functions with large variation, amplified due to centering to zero mean and unit variance according to equation 3.11. The feature functions with extreme variation are candidate outliers, to be removed later from the learning set.

The composite effect of adding and subtracting two standard deviations of the first principal component $\boldsymbol{\alpha}_1(v)$ to the average feature function $\overline{\mathbf{f}}_m(v)$ is shown in figure 3.5. We have chosen to reduce the feature space using $Q = 4$ principal components (see equation 3.14), together capturing 83.9 percent of the total variability in the data. Q has been set to 4 because we expect for our application that there are 4 corners, and hence places, in the model where the data in the learning phase show there is independent variation in the boundary feature values. Note that these displays remind of the diagrams in physics of modes of vibration in a string fixed at both ends, hence the name *strings*.

Finally, regression functions $\mathbf{B}(v)$, obtained using equation 3.19, are shown in figure 3.6. The function $\beta_{qn}(v)$ indicates how the nth feature along the curve contributes to the qth principal component. Hence, the regression functions indicate which features are locally most important to define the boundary characteristics, implying that

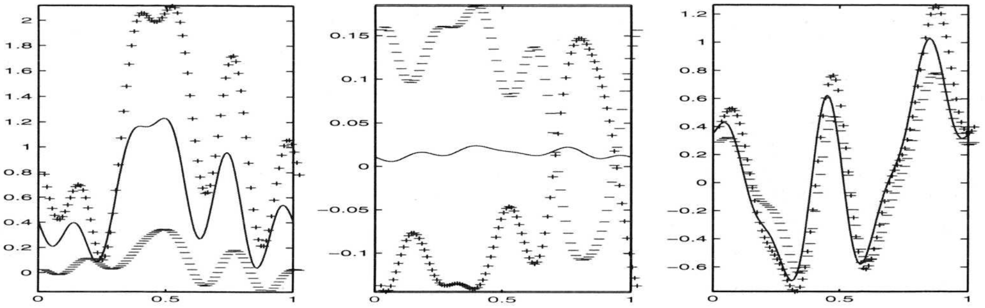


Figure 3.5: Average feature functions and the effects of adding (+) and subtracting (-) two standard deviations of the first principal component: $\bar{\mathbf{f}}(v)$ +/- 2 standard deviations of $\boldsymbol{\alpha}_1(v)$. From left to right the effects for, contour curvature, isophote curvature and directional correspondence.

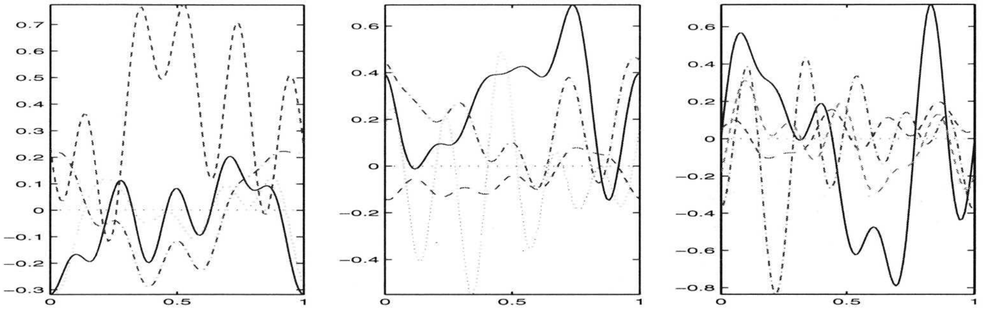


Figure 3.6: Regression functions corresponding to the $Q = 4$ principal components. Left: functions $\beta_{11}(v), ..., \beta_{41}(v)$ for contour curvature. Middle: $\beta_{12}(v), ..., \beta_{42}(v)$ for isophote curvature. Right: $\beta_{13}(v), ..., \beta_{43}(v)$ for directional correspondence.

weighting is done in a way that exploits the most correlated and descriptive features at each boundary point.

Not shown in illustrations is the effect of the optional outlier removal. A total of 11 example feature functions have been considered outliers and removed. After this, the whole learning procedure is repeated one more time. The average feature function vector $\bar{\mathbf{f}}(v)$, the normalization function vector $\boldsymbol{\sigma}_{\mathbf{f}}(v)$, the matrix of regression functions $\mathbf{B}(v)$ and of scalars $\mathbf{G}^{\star}$ are transferred to the segmentation phase.

3.4.3 Segmentation Phase

We perform string segmentation of the vertebra images using the above learning results as a reference. An active B-spline shape $\mathbf{s}_t(v; \mathbf{p}_k)$, defined by $k = 100$ control points, is deformed in the image by repositioning of these points in an iterative procedure. Each time a total of 100 samples are taken along $\mathbf{s}_t(v; \mathbf{p}_k)$ to construct a feature function from them. For simplification of comparison with active shape model segmentation (described below), the samples are taken at points $\mathbf{p}_k$. For computation of image features, derivatives are computed by convolution of the image with Gaussian derivatives at scale 4.0. Optimization of the active shape model is done by the simulated annealing optimization procedure, with the Boltzmann factor set to 0.99, thermal equilibrium defined as a 10 percent or smaller change in 10 random trials and a maximum of 50 iterations.

We also perform multi-resolution active shape model segmentation [23]. The active shape model learns the distribution of points $\mathbf{p}_k$. The normalized gradient is captured perpendicularly to the boundary in profiles of length 3. The number of levels of resolution is set to 6, with level 0 the original image, level the 2 the image with half the number of pixels etc.. After alignment of the shapes by rotation, translation and scaling, a shape model is constructed in which 84.2 percent of the variance is explained by the first 6 principal components. The neighborhood examined to find a better location for each point is 9. The initial shape is not taken to be the statistical average, rather it is an arbitrary shape which is projected onto the space spanned by the 6 principal components for evaluation [53]. Optimization stops after a maximum of 50 iterations.

We perform three experiments with strings and active shape models (software courtesy of [53]). In each experiment a shape is placed in the image on the correct position and perturbed a known amount to verify robustness against initialization. The perturbations include translation up to 15 pixels, rotation up to 30 degrees or scaling with respect to a center point $\mathbf{c}$ with a factor up to 0.1. The perturbed shape is then used to bootstrap segmentation. To measure the accuracy of the segmentation the distance from the resultant to the correct boundary is computed using the root squared metric error. The error measure is based on the $K = 100$ optimized control points and on control points $\mathbf{p}_k$ defining ground-truth, also by 100 samples. The experiments are performed systematically excluding each learning instance, from the learning set and using the excluded one to test the performance of the model built without it.

Figure 3.7 shows the average root squared metric of 145 segmentations for varying amounts of translation, rotation and scaling. As can be seen from the reduction in the root mean squared error, the initial shape almost always moves to the correct boundary for both strings and active shape models. The string segmentation method outperforms active shape model segmentation when the initial shape is close to the correct boundary, in spite of the fact that the ground-truth is poorly sampled. If not initialized close to the target, string segmentation produces results similar to those produced by active shape model segmentation, occasionally worse. This sensitivity to initialization is attributed to the fact that no pose parameters are optimized to explicitly account for pose corrections, in contrast to active shape model segmentation. The active shape model finds the correct boundary even from a large distance if that boundary is well defined. However, when the image evidence is vague, the active shape model tends to get trapped in a local minima far away from the correct boundary due to wrong pose optimization. For strings, we expect performance improvement by optimization of pose parameters in addition to shape parameters under the condition that pose parameters are restricted to admissible ranges. Equivalent improvements may be expected for active shape models by restricting pose parameters.

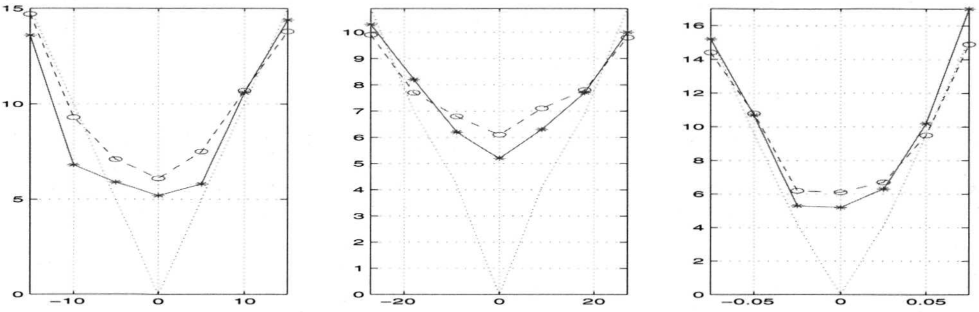


Figure 3.7: The average root mean square distance for initial and final curves as a function of translation (left), rotation (middle) and scaling (right).

From figure 3.7 we observe that even when the initial shape is the correct shape, optimization brings the initial shape to rest at an average of almost 5 pixel distance from correct shape. We note that the points marking the vertebra boundaries in our learning set have been placed by a single medical experts and that variation of 5 or more pixels in manual point placement can be expected. Hence, the structural error of approximately 5 pixels is largely ascribed to intra-observer variability. We expect an improvement of performance proportional to the accuracy of the ground-truth segmentation, either by more precise individual assignment or by using larger amounts of salient points per vertebra. As the construction of a general model for the cervical vertebra also contributes to the structural error, we expect that a dedicated boundary model, e.g. a model of the C1 vertebra, will bring in more accuracy and specifity to the boundary model.

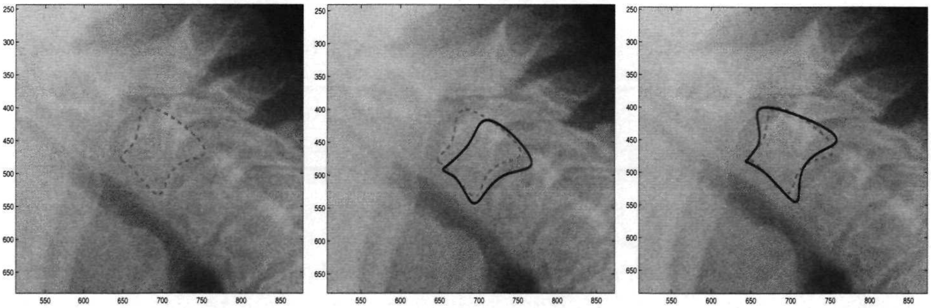


Figure 3.8: Segmentation of a NHANES cervical vertebra image. Left: typical NHANES image with ground-truth delineation. Middle: active shape model segmentation (solid line). Right: string segmentation (solid line). Note the vague image evidence along the vertebra boundary due to the very low image quality.

Figure 3.8 shows an example result for the active shape model segmentation and the string segmentation, which typically takes about 15 seconds and 95 seconds respectively on a standard machine. In spite of the elaborate matrix function manipulations, performance is still very good. Much of the processing time is attributed to feature function alignment. Also the exhaustive search of simulated annealing contributes to the high computational cost. We expect a significant reduction in computational cost with landmark-based alignment, a more efficient optimization technique and optimization of our code for speed. Note that much of the erroneous solutions of the active shape model [53] in figure 3.8 are global of nature. Apparently, the image evidence around the target boundary is too vague to be conclusive for such a model. Hence, the active shape model converges with a feasible shape in an unacceptable pose. For the string segmentation, the erroneous solutions are confined to boundary segments not on the correct boundary, but rather on other visually-detectable edges, such as tissue/background edges or even edges produced by gray-scale intensity variation within a vertebra.

3.5 Discussion and Conclusion

In conclusion, we have addressed a number of contemporary problems with statistical image segmentation models. We have represented boundaries by curves, solving some problems arising from discrete representations. By placing these curves in a multivariate functional feature space we have properly dealt with the problem of spatial and feature interdependencies. We have performed curve registration, releasing us from the problem of missing points when aligning. We have substituted feature data by principal component scores, thereby not only reducing the feature space but also endorsing more robust computations. We have constructed a regression model for predicting unknown boundaries, even ones dissimilar to the examples. We have built distance model that also accounts for residual information. Finally, we have defined image segmentation as string optimization in multi-dimensional feature space.

In this paper we have applied the string method for segmentation of rigidly shaped objects with ill-defined image evidence along their boundary, as often found in medical images. The method is also expected to work well for objects with more articulated shapes, provided there is correlated variation in the shape properties. When applied to objects of the real world, where objects often have an unknown and varying background, the definition of features should be such that values are computed from the inside of the object only. This provides a multi-variate statistical description of the object area, rather than a description of the object boundary with its surrounding. It is important to note that continuous ground-truth segmentations are required to fully exploit the capabilities of strings.

With [20], [37], [19], [35] we share the observation that features should be learned rather than constructed from a priori geometrical or analytical knowledge. Geometrical and analytical features such as smoothness act as constraints on the solution since the resulting shape then will often be smooth at most places, regardless whether that is the appropriate solution or not. As in [20], we conceive of shape and image evidence as features and learn where they are effective in describing the statistics of the model.

In fact, we have adopted the idea of mapping features to a space where the most important modes of variation are determined by principal components analysis and used for steering a shape for segmentation of an image.

The difference is that we conceive of a boundary as a multivariate continuous curve, requiring continuous functions to be learned. Cootes et al. [20] reduce the boundary to point sets. This introduces the following problems. First, points may be confused with other points if they are not labeled, leading to erroneous classification. Second, point location correspondence may be doted with error. Third, as recognized by Duta [35], points may be missing, requiring a pseudo continuous reconstruction of the boundary. Finally, measurements at discrete sample points are less reliable and lead to lose of spatial coherence.

We are less critically effected by these difficulties as we use closed continuous curves. We profit from functional data analysis in exploiting the spatial and feature correlations to explain the observed variations in the training data rather than removing or down weighting such correlations. Apart from this, in explaining these variations, we also consider residual information, rather than omitting it. Contrary to Cootes et al. [20] we built a regression model from which we try to explain unknown boundaries, including the errors made in doing so. Another difference is that, instead of producing new models in reduced space, and hence restricting the shapes the method can handle, we freely produce new segmentation shapes and punish implausible shapes.

In comparing the performance of our string implementation with an active shape model implementation using software by [53], we note the following. The active shape model performs better in the case the target object is visually well-defined and the initial shape is placed at a large distance from the target. This is thanks to the explicit optimization of pose parameters for translation, rotation and scaling. When the initial shape is initialized close to the target object, the active shape model performs better in the case the object has boundary properties like most objects in the learning set. If the target object has boundary properties that are dissimilar to the ones in the learning set, the string model produces more accurate results. We note that the prediction of feature values in reduced space, aids in explaining objects not seen in the learning set.

In a situation where the target boundary is visually ill-defined and the shape initialization is at a large distance from the target, both string segmentation and active shape model segmentation perform poorly. For that condition, we found active shape models to produce much better or much worse results than string segmentation. On average their performance is then similar. It has been acknowledged previously that proper initialization is required to guarantee satisfying result in view of the presence of disturbing attractors in the image [83]. Strings outperform active shape models if the target object is visually ill-defined and the initialization is close to the target boundary. This is due to the fact that strings locally exploit the discriminative power of a repertoire of features. Also the fact that strings cope better with spatial dependencies between feature values and the fact that they take into account correlations among features more explicitly improves segmentation under such highly demanding conditions.

Hence, we arrive at the conclusion that strings are particularly suited for learning variational models of objects that have inherently multivariate continuous boundary characteristics. Strings are also very well suited for segmentation of complex scenes, where the visual evidence is vague or where a multi-dimensional feature set is needed to capture an object boundary.

Chapter 4

Medical Image Retrieval by Browsing Population-Based Incrementally Learned String Segmentations*

This chapter proposes a browsing method for medical image retrieval by example objects. The characteristics of a population of normal objects are inductively learned by functional data analysis and summarized in a string segmentation model. The string model is used for segmentation of an example image of an abnormal object at the beginning of the browsing session. The segmentation result is used to adapt the string model towards one that can handle the recorded abnormality. The segmentation result, in addition, bootstraps retrieval of images containing similar abnormal objects. In an iterative process, good or specifically bad retrievals are segmented according to the new string model in order to refine the definition of the abnormality. The segmentations form a pilot for a population-based incremental learning technique that explores the repository and exploits previous retrieval results to arrive at a population of objects with the same common deviations from normal. Browsing ends with a set of images depicting same abnormalities as the initial example, ranked by the degree of content matching. The method has been successfully applied for retrieval of digitized X-ray images of vertebrae from a large collection of normal and abnormal instances.

*Submitted to IEEE Transactions on Medical Imaging

4.1 Introduction

Medical images come from an ever-increasing variety of sources and in ever-increasing amounts, bootstrapping the development of new techniques for content-based image retrieval. Traditionally, retrieval techniques utilize textual information in the patient record to recover images from a repository. This approach has proven to be successful for medical imaging [75], but it is apparent that the rich information contained in the images themselves is not fully exploited. Medical image data sets demand a higher degree of content understanding [118] to fully bestow visual information. In this context, content-based image retrieval by example provides an opportunity to tap the expertise present in an image repository based on visual features, once the textual information in the patient record is exhausted. We take retrieval one step further by explaining images based on objects contained in them. In its impact this is radically different from mainstream content-based image retrieval [111] where information is summarized from the entire image disregarding object properties.

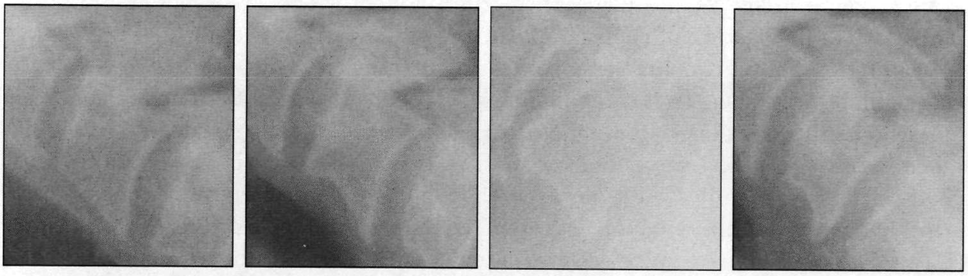
A number of methods in literature, e.g. [25],[30], [57], [72], [76], [91], [103], [104], [106], try to tackle the retrieval problem in an object-based manner using geometrical, analytical or grammatical segmentation models. The problem with these approaches is the premise that the segmentation model is capable of capturing an arbitrary object in an image, regardless of its natural variations. Apart from this, the approaches postulate that a single example object suffices to retrieve relevant images from the repository. This might be the case when objects have clear and consistent characteristics, easily distinguishing them from other ones. However, if this assumption is violated, as is often the case in medical imaging, segmentation models will fail due to their incapacity to precisely capture the user's retrieval intention, by learning from multiple query objects, user-interaction or from the history of a browsing session. Learning the search intention of the user is an important problem with direct implications on the ability of an algorithm to meet the user's demand.

The problem addressed in this work is: *how to learn the user's retrieval intention and recover images best matching his/her concept from large collections of images, commencing from a single example object*? The approach taken departs from the view that *a)* to accurately capture objects in images, segmentation models should be learned in terms of multiple features rather than constructed from a priori knowledge, *b)* to handle arbitrary objects, segmentation models should be adaptive rather than fixed *c)* to readily circumscribe the user's concept of the object, it should be described with help of multiple example objects exhibiting similar characteristics. In this context, medical image retrieval requires learning the characteristics of normal and abnormal object populations. The problem of retrieval is transposed into one of browsing the image repository in search for abnormal populations with the same deviating characteristics from normal as the example object.

The chapter is organized as follows. Section 2 describes the image material and the application to demonstrate the proposed retrieval method. Section 3 is devoted to the proposed retrieval scheme. The main issues are inductive learning, string-based segmentation, population-based incremental learning and content-based image retrieval. In section 4 automatic image classification is addressed. Experiments and results are described in section 5, then discussion and conclusion in section 6.

4.2 Clinical Example Application

The application to demonstrate the retrieval problem and proposed solution deals with cervical vertebra images exhibiting osteophytes. The image material is acquired from the approximately 17,000 X-ray films collected during the Second National Health and Nutrition Examination Survey (NHANES II) conducted by the NCHS [86]. In this cross-sectional population survey, X-rays were taken of persons aged between 25 and 74. Two X-rays of the spine, PA and lateral, were made except of pregnant women and women under 50 years of age, to provide evidence of osteophyte and degenerative disc diseases. The films were subsequently digitized at a horizontal and vertical sampling rate of 146 dpi using Lumisys laser scanning equipment [86].



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

Figure 4.1: Digital images of cervical vertebrae: a) normal, b) lower osteophyte, c) upper osteophyte, d) lower and upper osteophyte. Note the limited detail and the complexity of the boundary.

We concentrate on a repository consisting of 283 digitized NHANHES images of single cervical vertebrae. A medically certified diagnosis is attached to each image, including an expert delineation of the vertebra. The image repository contains both normal and abnormal vertebrae. The abnormal vertebrae are classified either as lower anterior osteophyte, upper anterior osteophyte or both lower and upper anterior osteophyte. To illustrate the differences between them consider figure 4.1.

An osteophyte is characterized by bony outgrowths on the anterior corners of the vertebral body. For example, the shape of the lower anterior of the C5 vertebra in figure 4.1b clearly extends from the body of the vertebra. The spurs are furthermore associated with the structural image boundary as opposed to histogram or texture characteristics [97]. We utilize shape and edge information related to the vertebra boundary for describing and retrieving images by visual features.

We simulate a situation in which a clinician wants to retrieve vertebra images from the repository of known and annotated cases to help determine the diagnosis of an unknown abnormal vertebra image, in this case one classified as lower anterior osteophyte. The clinician uses the abnormal vertebra as an initial example for retrieval similar images from the repository. The clinician then refines the initial query by indicating good and bad retrievals as new examples for the next retrieval step, this way browsing through the retrieved images. This brings the clinician to a population of images best matching the initial retrieval intention, hence best suited for supporting the diagnosis. This is a highly challenging task considering the subtle shape and image differences among vertebrae and in this case the marginal image quality.

4.3 Method for Browsing Image Populations

The proposed image retrieval scheme consists of four components as illustrated in figure 4.2. *Inductive learning* constructs a string model of a normal vertebra by learning from a given set of images with true segmentations. The model is used for *string-based segmentation* of the example image of the unknown abnormal vertebra at the beginning of the browsing session. The segmentation result is used to adapt the string model towards one that can handle the recorded abnormality. The segmentation result, in addition, is used for *content-based retrieval* of images depicting similar abnormalities. To refine the definition of the abnormality and hence retrieval, good or bad retrievals are indicated as new search examples. The examples form a pilot for the *population-based incremental learning* technique used to explore the repository and to exploit previous results. Browsing ends with a set of images depicting same abnormalities as the initial query image, ranked by the degree of content matching.

In the following sections we address each of the components in figure 4.2. We start with a brief description of the string segmentation model proposed in the previous chapter, which in contrast to other similar approaches (e.g. [11], [19], [20], [35], [43], [85]) has the capacity to use multiple continuous shape and image features in the definition of boundaries.

4.3.1 Defining a Population of Normals

We analyze a population of normal vertebrae in order to learn the important statistical characteristics of this population. We do this inductive learning on the basis of multiple continuous features recorded along normal vertebra boundaries.

Vertebra boundaries in the training images are represented by smooth curves $\mathbf{s} : \Re \rightarrow \Re^2$ parameterized by $v \in \Re$. Given the population of M input images $I_1(\mathbf{x}), ...I_M(\mathbf{x})$ for each of which the true shape $\mathbf{s}_1(v), ...\mathbf{s}_M(v)$ is known, the learning set consists of pairs of image and shape data

$$\mathcal{L} = \{(I_1(\mathbf{x}), \mathbf{s}_1(v)), ..., (I_M(\mathbf{x}), \mathbf{s}_M(v))\}. \tag{4.1}$$

For the mth learning example, the shape $\mathbf{s}_m(v)$ relates to the image at vertebra boundary points $I_m(\mathbf{s}_m(v))$. The relation is expressed in terms of N features measured

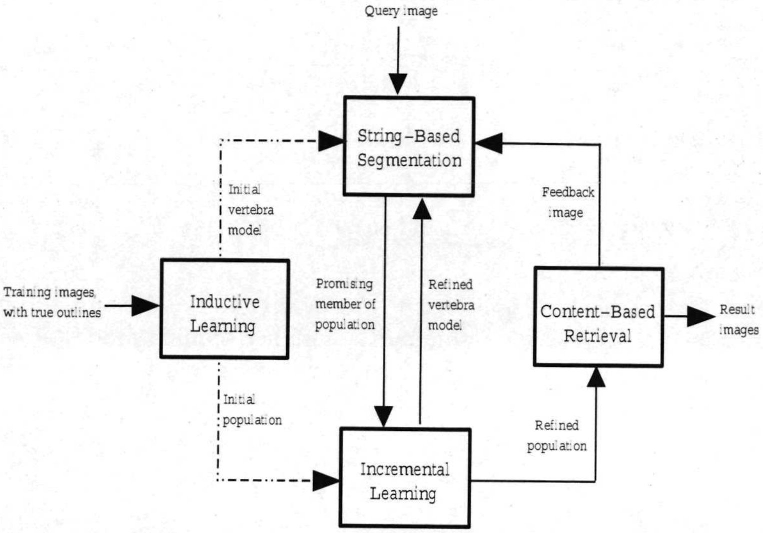


Figure 4.2: Overview of the retrieval scheme, consisting of components for inductive learning, string-based image segmentation, incremental learning and content-based image retrieval.

along $I_m(\mathbf{s}_m(v))$ derived from the shape as well as from the image. This yields M feature functions in the N-dimensional feature space $\mathcal{F}$

$$\mathbf{f}^{**}_m(v) = [f^{**}_{m1}(v), ..., f^{**}_{mN}(v)]. \tag{4.2}$$

The feature functions are aligned to remove variation from the learning data which is attributed to stretching, shearing and rotation of the feature data. This is accomplished using the Procrustes method [50] by finding for $\mathbf{f}^{**}_m(v)$ the strictly monotonic non-linear transformation $\omega_m(v)$ [95] that produces the warped feature function

$$\mathbf{f}^{*}_m(v) = \mathbf{f}^{**}_m(\omega_m(v)). \tag{4.3}$$

The warping functions and the aligned feature functions are computed iteratively with help of the estimated mean feature function $\hat{\mathbf{f}}(v)$. A global alignment criteria is used for computing $\omega_m(v)$ for $\mathbf{f}^{**}_m(v)$ such that

$$\omega_m(v) = \underset{\omega^*_m(v)}{\operatorname{argmin}} \quad \sum_{m=1}^{M} \int_v ||\mathbf{f}^{**}_m(\omega^*_m(v)) - \hat{\mathbf{f}}(v)||^2 dv. \tag{4.4}$$

The final mean feature function $\bar{\mathbf{f}}(v)$ is computed from this aligned set of feature functions and subsequently subtracted from each function $\mathbf{f}^*_m(v)$ to obtain feature functions $\mathbf{f}_m(v)$, containing all evidence of the learning phase properly aligned, centered and scaled to unit variance. The normalized feature data is defined by

$$\mathbf{f}_m(v) = \frac{\mathbf{f}^*_m(v) - \bar{\mathbf{f}}(v)}{\boldsymbol{\sigma}_{\mathbf{f}}(v)} \tag{4.5}$$

with units of variance due to normalization by

$$\boldsymbol{\sigma}_{\mathbf{f}}(v) = \left(\frac{1}{M} \sum_{m=1}^{M} ||\mathbf{f}^*_m(v) - \bar{\mathbf{f}}(v)||^2 \right)^{1/2}. \tag{4.6}$$

Principal components analysis is performed to project the N-dimensional functional data to a smaller feature space. Assuming the number of main modes of variation is Q, weighting functions $\boldsymbol{\alpha}_q(v) = [\alpha_{q1}(v), ..., \alpha_{qN}(v)]$ are sought for one-by-one, to find values

$$g_{mq} = \sum_{n=1}^{N} \int_v f_{mn}(v)\alpha_{qn}(v)dv \tag{4.7}$$

that explain most of the variation in the data. The function $\alpha_{qn}(v), q = 1, ..., Q$ is a weight chosen so as to highlight variation in the data in dimension n. The values g_{mq} form the principal component score $\mathbf{g}_m = [g_{m1}, ..., g_{mQ}]$, used in place of the actual feature function $\mathbf{f}_m(v)$ to summarize the bulk of feature data to a vector-valued quantity. The vector of weighting functions $\boldsymbol{\alpha}_q(v) = [\alpha_{q1}(v), ..., \alpha_{qN}(v)]$ is computed such that

$$\boldsymbol{\alpha}_q(v) = \underset{\boldsymbol{\alpha}^*_q(v)}{\operatorname{argmax}} \; \frac{1}{M} \sum_{m=1}^{M} g^2_{mq} \tag{4.8}$$

Principal component regression [14] is performed to build an underlying model of the data in the learning set. The matrix of regression functions $\mathbf{B}(v) = [\boldsymbol{\beta}_1(v), ..., \boldsymbol{\beta}_Q(v)]^T$ is computed by least squares minimization such that

$$\mathbf{B}(v) = \underset{\mathbf{B}^*(v)}{\operatorname{argmin}} \; \sum_{m=1}^{M} \int_v ||\mathbf{f}_m(v) - \mathbf{g}_m\mathbf{B}^*(v)||^2 dv. \tag{4.9}$$

The matrix $\mathbf{B}(v)$ indicates how each of the N measured features along the vertebra contour contributes to the principal component functions. Hence, the regression functions indicate which features are locally most important to define the vertebra characteristics. With help of the estimated regression functions, principal component scores are predicted for unknown feature function emanating from an unknown vertebrae.

A Mahalanobis distance model is computed from the $Q \times M$ matrix $\mathbf{G}$, composed of the q-dimensional principal component scores, one for each M feature functions. This differs from chapter 3 in that we omit residual information derived from minimizing equation 4.9. We omit this information in order to allow comparison of virtual feature functions with no residuals (yet to be described), with real feature functions that produce a residual when explained by the regression model. The Mahalanobis distance matrix is then defined as

$$\mathbf{D} = \frac{\mathbf{G}^T\mathbf{G}}{(M+1)}. \tag{4.10}$$

The Mahalanobis distance matrix allows to compute the distance of the individual $\mathbf{f}_m(v)$ or any other feature function to the population average, taking into account variation. In this sense, the Mahalanobis distance is a measure of quality of an individual with respect to an entire population. For $\mathbf{f}_m(v)$ the Mahalanobis distance is computed using $\mathbf{g}_m$

$$D^2(\mathbf{f}_m(v)|\mathbf{g}_m) = \mathbf{g}_m\mathbf{D}^{-1}\mathbf{g}_m{}^T. \tag{4.11}$$

To make the Mahalanobis distance computed in equation 4.11 only dependent on the shape of a cluster, not its size, we remove the scale difference using the root mean squared group size [32], contrary to what is done in chapter 3. To obtain the scale invariant Mahalanobis distance the quantity in 4.11 is divided by

$$R = \frac{1}{M-1}\sum_{m=1}^{M} D^2(\mathbf{f}_m(v), \bar{\mathbf{f}}(v)). \tag{4.12}$$

The following information is kept from the inductive learning phase: the population average $\bar{\mathbf{f}}(v)$, the normalization function $\sigma_{\mathbf{f}}(v)$, the matrix of regression functions $\mathbf{B}(v)$ and the principal components scores matrix $\mathbf{G}$. This information is transferred to the segmentation phase for finding a vertebra boundary in an unknown image on the basis of multiple weighted features.

4.3.2 Visualizing the Normal Population

We illustrate and discuss the appearance of a normal cervical vertebra according to the string model. In order to get better insight in the shape of the normal cervical vertebra and the natural variations therein, we first concentrate on $N = 2$ shape features, capturing the projectional alignment of vertebrae in addition to local shape.

We visualize the learning data in the image space. The images in the top row of figure 4.3 illustrate the average shape of a normal cervical vertebra plus (light grey values) and minus (dark gray values) up to three standard deviations away from the average shape. The four sub-figures 4.3a-4.3d correspond to variation in the first four principal component directions, together capturing 93 percent of the total variability in shape. The number of principal components has been set to Q =4 because we expect for our vertebra application there are 4 corners, and hence 4 places, in the string model where the data in the learning phase might exhibit independent shape variation.

Figure 4.3a shows that the main variation in the shape of the normal cervical vertebrae in our application occurs at the lower anterior corner. The variation extends

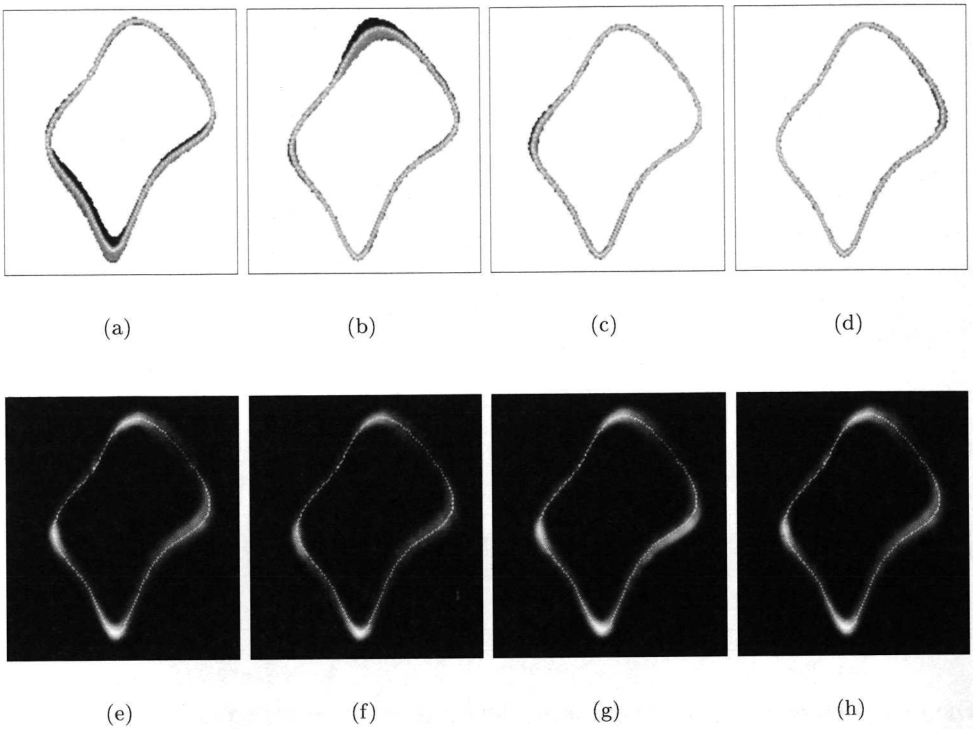


Figure 4.3: Visualization of shape and image characteristics of a population of normal cervical vertebrae. Top row visualizes the average shape plus (light) and minus (dark) up to three standard deviation in the a) first, b) second c) third and d) fourth principal components direction. Bottom row visualizes the average gradient along the average vertebra contour plus (positive normal) and minus (negative normal) up to three standard deviations away in the of e) first f) second g) third and h) fourth principal components direction.

to the entire anterior and lower part. Figure 4.3b shows that the second locus of variation is at the right upper corner. The third mode of variation concentrates at the upper anterior corner as illustrated in figure 4.3c, exhibiting minor shape variation. Figure 4.3d indicates little variation at the right lower corner in the fourth principal component direction. As expected in this application, shape variations concentrate at the four corners of the normal cervical vertebrae.

For the gray-level definition of the cervical vertebra boundary we consider $N = 1$ image feature. The image gradient magnitude along the vertebra boundary is recorded. The bottom row in figure 4.3 visualizes along the average vertebra contour the average gradient magnitude value and the variation therein. The variation is visualized in the normal direction of the average contour by plotting the average gradient value plus a multiple of a principal component in the positive normal direction and minus a multiple in the negative normal direction. The brighter the intensity the better defined the image gradient. The broader the scope of the intensity in perpendicular direction the more variation there is in gradient magnitude.

Figure 4.3e shows that the image gradient is well-defined at the upper and lower anterior corner and upper right corner of the vertebral body. As expected the gradient magnitude is ill-defined due to absence of intensity discontinuity at the part where the vertebral body joins the pedicles. The image gradient is also less steep at the middle upper and lower parts. It is not immediately visible where the main modes of variation occur. However, upon inspection of the image differences (data not shown here) it becomes clear that the first mode of variation focuses on the upper right corner, the second on the lower anterior corner, the third on the entire lower part of the vertebral body and the fourth along the entire right part of the vertebral body. From these observations we conclude that there is some correlation between shape and diagnostically significant image features.

4.3.3 String-Based Image Segmentation

At the beginning of a browsing session we need to segment the abnormal vertebrae in the unknown example image in order capture the characteristics of that vertebra. String-based image segmentation allows to do this in terms of multiple features, weighted according to the most important variations in the normal population.

An active string in the N-dimensional feature space $\mathcal{F}$, denoted by $\mathbf{f}_t^{**}(v)$, is used for image segmentation and feature extraction. This string is defined by features extracted from a shape model $\mathbf{s}_t(v)$ that deforms in time t and from the unknown vertebra image $I_0(\mathbf{x})$ in which that model lives. At time t the feature function is defined as

$$\mathbf{f}_t^{**}(v) = [f_{t1}^{**}(v|\mathbf{s}_t(v), I_0(\mathbf{x})), ..., f_{tN}^{**}(v|\mathbf{s}_t(v), I_0(\mathbf{x}))]. \tag{4.13}$$

After aligning $\mathbf{f}_t^{**}(v)$ to the population average $\bar{\mathbf{f}}(v)$ and weighting it by $\check{\mathbf{f}}(v)$, both derived from the learning phase, its quality with respect to the learning feature functions $\mathbf{f}_1(v), ..., \mathbf{f}_M(v)$ is determined from the relation of its corresponding score vector $\mathbf{g}_t$ to the cluster of principal component scores $\mathbf{g}_1, ..., \mathbf{g}_M$. First, the Q-vector $\mathbf{g}_t$ is estimated by solving

$$\mathbf{f}_t(v) = \mathbf{g}_t\mathbf{B}(v)^T + \epsilon_t(v) \tag{4.14}$$

where $\epsilon_t(v)$ is the residual function. This allows estimation of $\mathbf{g}_t$ such that the important features of $\mathbf{f}_t(v)$ according to the regression model in equation 4.9 are emphasized. The value of $\mathbf{g}_t$ is estimated by least squares minimization such that

$$\mathbf{g}_t = \underset{\mathbf{g}_t^*}{\operatorname{argmin}} \int_v (\mathbf{g}_t^*\mathbf{B}(v)^T - \mathbf{f}_t(v))^2 dv. \tag{4.15}$$

The Mahalanobis distance of $\mathbf{f}_t(v)$ to the average of the normal population is then defined in terms of the distance of the new score $\mathbf{g}_t$ to the average of the cluster constituting of $\mathbf{g}_1, ..., \mathbf{g}_M$. It is computed as follows

$$D_M^2(\mathbf{f}_t(v), \bar{\mathbf{f}}_t(v)) = \mathbf{g}_t \mathbf{D}^{-1} \mathbf{g}_t{}^T. \tag{4.16}$$

We use this quality measure for segmenting the unknown image $I_0(\mathbf{x})$. The string $\mathbf{f}_t(v)$ vibrates in feature space due to deformations of the shape model $\mathbf{s}_t(v)$ in the image, subsequently producing new scores evaluated by means of the Mahalanobis distance model. Starting from an initial configuration, $\mathbf{s}_{t=0}(v)$ is deformed by tuning its shape parameters in such a way that the state of minimal energy provides the optimal shape model $\mathbf{s}_f(v)$, assumed to be the outline of the vertebrae in the unknown image. This reduces to optimizing $\mathbf{f}_t(v)$, emanating from $I_0(\mathbf{s}_t(v))$, such that

$$\mathbf{f}(v) = \underset{\mathbf{f}_t(v)}{\operatorname{argmin}} \quad D_M^2(\mathbf{f}_t(v), \bar{\mathbf{f}}(v)). \tag{4.17}$$

At this point we have: a string model trained from a population of normal vertebra images and a string segmentation of an unknown abnormal vertebra for which we try to find similar images in the repository of known cases. The string segmentation of the abnormal vertebra is used to bootstrap the browsing image segmentations. The string model is refined during browsing in order to more precisely define the characteristics of the browsed population, i.e. of the abnormality.

4.3.4 Browsing a Population of Abnormals

As the string segmentation of the abnormal vertebra in the initial example image may not suffice to unequivocally describe the characteristics of that vertebra, the image repository of known cases is browsed for similar images to learn those characteristics more carefully. We do this by incrementally learning the vertebra features recorded in the retrieved images that are interactively specified as good or as bad retrievals.

For browsing we use a population-based probabilistic search algorithm based on modeling promising areas in the feature space by estimating their probabilistic distribution [3], [4]. The probabilistic model explores the feature space taking into account information from previous explorations. At the end of the exploration, the probabilistic model defines a narrow region of the feature space, frequently encountered during the exploration process, hence circumscribing the retrieval intention in terms of reoccuring characteristics.

We conceive of the feature set used for inductive learning as an initial population, bounding a feature subspace that defines the appearance of a normal population. That is, feature functions $\mathbf{f}_1^{**}(v), ..., \mathbf{f}_1^{**}(v)$ constitute the first generation of an evolving population. For simplicity of notation, from this point on we denote the evolving population by $\mathbf{f}_1^s(v), ..., \mathbf{f}_M^s(v)$. At generation $s = 0$ we have

$$\mathbf{f}_m^s(v) = \mathbf{f}_m^{**}(v) \tag{4.18}$$

with the corresponding principal component scores denoted by

$$\mathbf{g}_m^s = \mathbf{g}_m. \tag{4.19}$$

The probabilistic model $P^s(\mathbf{f}^s_m|\mathbf{g}^s_m)$ corresponding with this population is defined using Gaussian models with mean $\bar{\mathbf{f}}^s(v)$ and standard deviation $\sigma^s_{\mathbf{f}}(v)$. This is permissible as we expect a Gaussian distribution of feature functions. The probability of $\mathbf{f}^s_m(v)$ is computed on the basis of its principal component scores as follows

$$P^s(\mathbf{f}^s_m|\mathbf{g}^s_m) = |\mathbf{D}|^{-1/2}(2\pi)^{-Q/2}e^{\mathbf{g}^s_m \mathbf{D}^{-1} \mathbf{g}^{s\,T}_m}. \tag{4.20}$$

With help of $P^s(\mathbf{f}^s_m|\mathbf{g}^s_m)$ we maintain a probabilistic model of the promising regions of the search space. The aim is to adapt $P^s(\mathbf{f}^s_m|\mathbf{g}^s_m)$ by exploration and exploitation of the feature space such that at the end of the evolution, vertebra with abnormalities similar to the initial and subsequent examples have high probability according to this model. Representing the uth example vertebra by feature function $\mathbf{f}^s_u(v)$ for $u = 1, ..., U$ and corresponding score by $\mathbf{g}^s_u$, this means we want to find the probabilistic model $P^\star(\mathbf{f}^\star_u|\mathbf{g}^\star_u)$ with

$$P^\star(\mathbf{f}^\star_u|\mathbf{g}^\star_u) = \underset{P^s(\mathbf{f}^s_u|\mathbf{g}^s_u)}{\operatorname{argmax}} \sum_{u=1}^{U} P^s(\mathbf{f}^s_u|\mathbf{g}^s_u). \tag{4.21}$$

The U example vertebrae $\mathbf{f}^s_u(v)$ form the pilot for evolution, with the uth query example corresponding with the sth browsing step. In the initial browsing step the query example is the abnormal vertebrae in the unknown image. That is, for $s = 1$ and hence for $u = 1$, the query example is defined by the score $\mathbf{g}^s_u$ obtained by optimization of a string in the unknown image $I_u(\mathbf{x})$ such that

$$\mathbf{f}^s_u(v) = \underset{\mathbf{f}^s_t(v)}{\operatorname{argmin}} \quad D^2_M(\mathbf{f}^s_t(v), \bar{\mathbf{f}}(v)). \tag{4.22}$$

To determine whether a feature subspace corresponds to the concept of the sought abnormal vertebra we define an evaluation function based on the most promising individual of the population at hand. In the initial browsing step at $s = 1$ the promising individual is the one most resembling the abnormal vertebrae in the initial query image. The score $\mathbf{g}^s_u$ is used as a reference for selecting this individual. To this end a quantity is defined that measures the Euclidean distance between an individual $\mathbf{g}^s_m$ and $\mathbf{g}^s_u$

$$D_E(\mathbf{f}^s_m(v), \mathbf{f}^s_u(v)|\mathbf{g}^s_m, \mathbf{g}^s_u) = ||\mathbf{g}^s_m - \mathbf{g}^s_u||. \tag{4.23}$$

In an elitist approach, the individual is selected that has smallest Euclidean distance to $\mathbf{g}^s_u$. This individual forms the center of a new feature subspace that presumably better describes the features of the abnormal vertebra in the initial query than does $\bar{\mathbf{f}}^s(v)$. The new feature function is selected to define the new mean $\bar{\mathbf{f}}^*(v)$ for the new probabilistic model

$$\bar{\mathbf{f}}^*(v) = \underset{1 \leq m^* \leq M}{\operatorname{argmin}} \quad D_E(\mathbf{f}^s_{m^*}(v), \mathbf{f}^s_u(v)). \tag{4.24}$$

We also employ the least matching individual of the population so that we can rely on negative examples in addition to a search based on positive examples. This allows to explicitly formulate that the search is not targeted at normal vertebrae or at vertebrae with abnormalities not related to the initial example. Defining the new mean on the basis of the worst individual reduces to

$$\bar{\mathbf{f}}^*(v) = \underset{1 \leq m^* \leq M}{\operatorname{argmax}} \quad D_E(\mathbf{f}^s_m(v), \mathbf{f}^s_u(v)). \tag{4.25}$$

As the aim is definition of the new probabilistic model for the next generation we also need $\sigma^*_{\mathbf{f}}(v)$ in addition to $\bar{\mathbf{f}}^*(v)$. In the initial browsing step $\sigma^*_{\mathbf{f}}(v)$ is initialized to some large value in an attempt to ensure that the feature space is well-covered in terms of probability. In the subsequent browsing steps it is annealed towards a value that represents the variation in the browsed population. That is, given examples $\mathbf{f}^s_u(v)$ for $u = 1, ..., U$, we have

$$\sigma^*_{\mathbf{f}}(v) = \left(\frac{1}{U} \sum_{u=0}^{U} ||\mathbf{f}^s_u(v) - \bar{\mathbf{f}}^*(v)||^2 \right)^{1/2}. \tag{4.26}$$

The probabilistic model $P^*(\mathbf{f}^*_m|\mathbf{g}^*_m)$ corresponding to the newly suggested feature subspace is defined by $\bar{\mathbf{f}}^*(v)$ and and $\sigma^*_{\mathbf{f}}(v)$. The probability model $P^{s+1}(\mathbf{f}^{s+1}_m|\mathbf{g}^{s+1}_m)$ for the next generation is updated using information gained from $P^s(\mathbf{f}^s_m|\mathbf{g}^s_m)$ and $P^*(\mathbf{f}^*_m|\mathbf{g}^*_m)$. The probability update rule employed here is similar to weight update rule in competitive learning when an output is moved towards a particular sample feature function [3]. The next generation probability model is defined by mean

$$\bar{\mathbf{f}}^{s+1}(v) = (1-\gamma)\bar{\mathbf{f}}^s(v) + \gamma\bar{\mathbf{f}}^*(v) \tag{4.27}$$

and deviation

$$\sigma^{s+1}_{\mathbf{f}}(v) = (1-\gamma)\sigma^s_{\mathbf{f}}(v) + \gamma\sigma^*_{\mathbf{f}}(v) \tag{4.28}$$

The innovation rate γ offers explicit control of how fast the population converges. As the probability model is used to generate the next population, the learning rate also affects which portion of the feature space will be explored. It updates the probability model in the direction of the best individual of the current population, governed by γ. When the learning rate is 0, there is no exploitation of the information gained through search. As the learning rate is increased, the amount of exploitation increases, and the ability to search large portions of the feature space diminishes.

The next population of feature functions arises from the updated probability model $P^{s+1}(\mathbf{f}^{s+1}_m|\mathbf{g}^{s+1}_m)$. The feature functions are not real in the sense that they do not necessarily relate to real vertebrae. Rather they are virtual functions arising from exploring the feature space, possibly corresponding with unrealistic vertebrae. The virtual feature functions are obtained by sampling M times from $P^{s+1}(\mathbf{f}^{s+1}_m|\mathbf{g}^{s+1}_m)$

$$\mathbf{f}^{s+1}_m(v) \sim P^{s+1}(\bar{\mathbf{f}}^{s+1}, \sigma^{s+1}_{\mathbf{f}}(v)). \tag{4.29}$$

To continue exploring the feature space by evolving the probabilistic model, we need further query examples that are similar or specifically dissimilar to the abnormal vertebra in the initial query image. We fetch the example from the image repository.

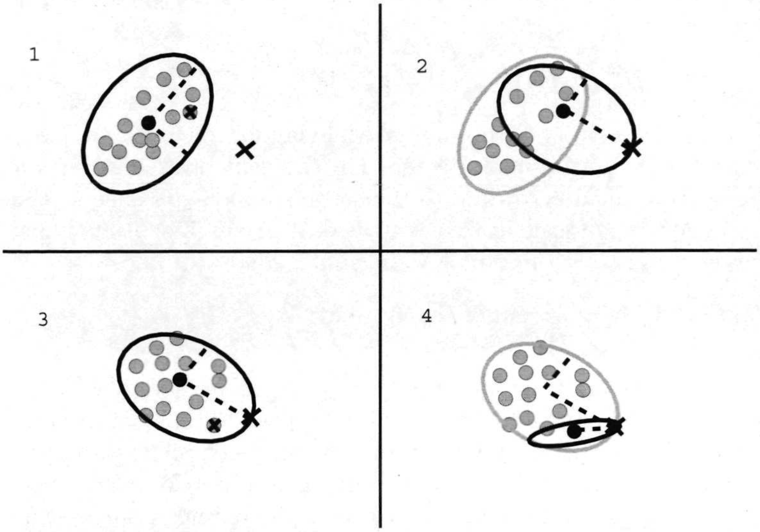


Figure 4.4: Schematic illustration of browsing the feature space using the same query example and $\gamma = 1$. In the first step the probabilistic model corresponding with the initial population (gray dots) is determined by mean (black dot) and variance (dashed lines). The example is the black cross. In the second step the one closest the the example (the dot with cross), is chosen as the new mean and the variance is computed from the difference between the example and the new mean. In the third step samples are drawn from the new probabilistic model for construction of the new population. In the fourth step the best individual is chosen from this population to form the new mean for the next generation probabilistic model. This procedure is performed a number of steps until the probabilistic model bounds a feature space representative for the query.

To this end, feature functions $\mathbf{f}_b(v)$ corresponding to the $b = 1...B$ vertebra images in the repository are evaluated using a string model that is trained from all M virtual feature functions $\mathbf{f}_m^{s+1}(v)$ by performing the same steps as described in equations 4.3-4.12. The principal component score $\mathbf{g}_b^{s+1}$ corresponding with the bth repository instance, computed according to the string model describing the population at generation $s+1$, is obtained by solving the following equation

$$\mathbf{f}_b(v) = \mathbf{g}_b^{s+1}\mathbf{B}^{s+1}(v)^T + \boldsymbol{\epsilon}^{s+1}(v) \tag{4.30}$$

using least squares minimization of the residual $\boldsymbol{\epsilon}^{s+1}(v)$. In here $\mathbf{B}^{s+1}(v)$ is the matrix of the re-computed regression functions. This way feature functions $\mathbf{f}_b(v)$ are reweighted after each browsing step so as to qualify them according to the common features recorded in the newly obtained population. To determine the promising images in the repository, we compute the Mahalanobis distance of each score $\mathbf{g}_b^{s+1}$

to the average of the evolved population on the basis of the updated distance model $\mathbf{D}^{s+1}$

$$D^2_M(\mathbf{f}_b(v), \bar{\mathbf{f}}^{s+1}(v)) = \mathbf{g}_b^{s+1}\mathbf{D}^{s+1^{-1}}\mathbf{g}_b^{s+1^T}. \tag{4.31}$$

The images in the repository with minimal Mahalanobis distance are the ones similar to the vertebra in the initial query image. For the next browsing step the images in the repository are ranked according to their Mahalanobis distance so that the most similar are placed first and the least one at the last place. The prime image suggested for browsing is the one with minimal Mahalanobis distance

$$\mathbf{f}_u(v) = \operatorname*{argmin}_{1 \le b^* \le B} \; D^2_M(\mathbf{f}_{b^*}(v), \bar{\mathbf{f}}^{s+1}(v))/R^{s+1}. \tag{4.32}$$

The above process of incrementally updating the probability model and the string model, and hence redefining and refining the concept of the sought abnormal vertebra, is repeated a number of times, until browsing has narrowed the feature space such that a sufficient number of relevant images of abnormal vertebrae from the image repository are fetched. A diagnosis can then be attached to the unknown abnormal vertebra in the initial example image based on the known classifications of the retrieved images.

4.3.5 Visualizing the Browsed Population

To demonstrate how the vertebra shape evolves when browsing the image repository with images of lower anterior osteophyte as the target, we visualize the vertebrae shapes emanating from the virtual feature functions $\mathbf{f}_m^{s+1}(v)$. We have selected an image of a vertebra possessing a lower anterior osteophyte which we know to be present in the image repository. Hence, the string model of the normal cervical vertebra is evolved to a model of a vertebra with a lower anterior osteophyte.

The shapes in the top row of figure 4.5 indicate the evolution trajectory from normal cervical vertebra to one which possesses a lower anterior osteophyte, requiring only four browsing steps. From the figures it can be seen which shapes are gradually explored to arrive at lower anterior osteophyte, with the modes of variation showing the parts of the vertebra contour that are adapted during browsing. It can be seen from figure 4.5d that there is almost no variation at the end of the browsing session as, in this case, the same abnormal vertebra has been used for exploration. When exploring the image repository on the basis of a population of abnormals, the average indicates the common characteristics of the browsed abnormals, whereas the modes of variation indicate parts of the abnormal vertebrae where the characteristics are equivocal. It might be postulated that the evolution from normal to abnormal vertebra perhaps corresponds to some pathological process.

The example demonstrates that the population-based incremental learning algorithm fetches relevant images from the repository within a small number of browsing steps. In this case the innovation rate γ for the trade-off between exploration and exploitation has been set to emphasis exploration rather than exploitation, allowing to browse quickly from the population of normal vertebrae to the population with lower anterior osteophyte. When the target of browsing is not one specific image

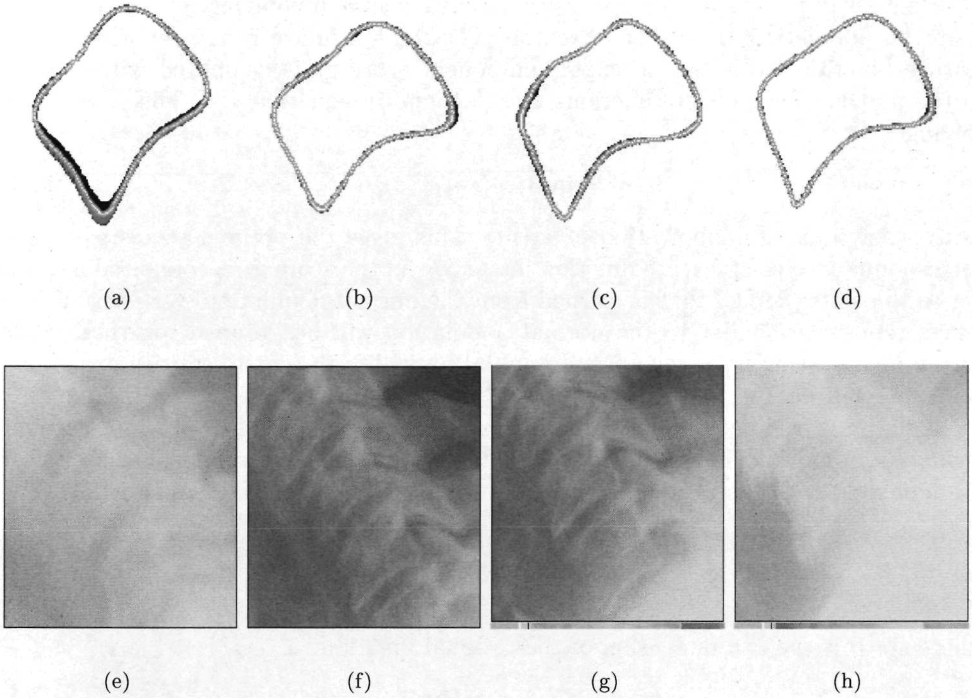


Figure 4.5: Top row shows the mean shape of a cervical vertebra plus (light) and minus (dark) up to three standard deviations away in the direction of the first principal component. The four pictures represent the condition at the $s = 4$ steps that were required to fetch the query image: a) $s = 0$, b) $s = 1$, c) $s = 2$ and d) $s = 4$. Note the difference with figure 4.3 in that the four figures correspond with the four browsing steps, not the variations in the four principal component directions. Bottom row displays the best intermediate retrieval results: e) after the first query, f) after the second query g) after the third query and h) after the fourth query step. Note that the initial query is the result in d.

but a population, emphasizing exploitation rather then exploitation enables to define more carefully what is sought for. Images of the abnormal vertebrae ranked highest after each of the 4 browsing steps are illustrated in the bottom row of figure 4.5.

4.4 Automatic Image Classification

This far we used the expert classification of the vertebrae in the image repository to draw conclusions about its appearance and about the appearance of unknown but similar vertebrae. As a side issue we dedicate this section to classifying the images automatically on the basis of their true segmentations.

We classify images in the repository on their commonly deviating characteristics from the population of normal vertebrae. For the bth image in the repository with feature function $\mathbf{f}_b(v)$, the principal component score $\mathbf{g}_b$ is computed with help of to the matrix of regression functions $\mathbf{B}(v)$ defined in equation 4.9. This is done by solving

$$\mathbf{f}_b(v) = \mathbf{g}_b\mathbf{B}(v)^T + \epsilon_b(v). \tag{4.33}$$

using least squares minimization of $\epsilon_b(v)$. This gives the optimal scores $\mathbf{g}_1, ..., \mathbf{g}_B$ corresponding to each feature function instance. As the score $\mathbf{g}_b$ is computed according to the string model that is learned from the normal population, vertebrae in the image repository similar to the normal population will be grouped together, while vertebrae deviating from the normal population will form remote entities depending on the deviation.

To improve the discrimination ability of the scores $\mathbf{g}_b$, we also take into account residual information. For each feature functions, its principal component score vector is augmented with the residual ϵ_b to obtain $\mathbf{g}_b^\star = [\mathbf{g}_b, \epsilon_b]$. The residual is defined by

$$\epsilon_b = \frac{1}{N}\sum_{n=1}^{N}\left(\int_v \epsilon_b^n(v)dv - \frac{1}{B}\sum_{b=1}^{B}\int_v \epsilon_b^n(v)dv\right) \tag{4.34}$$

where $\epsilon_b^n(v)$ is the nth dimension of the residual function

$$\epsilon_b(v) = ||\mathbf{f}_b(v) - \mathbf{g}_b\mathbf{B}(v)^T||^2. \tag{4.35}$$

The augmented scores $\mathbf{g}_1^\star, ..., \mathbf{g}_B^\star$ are clustered to obtain classes of scores with a similar deviations from the normal population. Unsupervised classification of the multidimensional score data is done by fitting a mixture of Gaussians with unconstrained covariance matrices and automatic choice of number of mixture components [55]. The result of clustering for $\mathcal{G} = \{\mathbf{g}_1, ..., \mathbf{g}_B\}$ is given by

$$\mathcal{C}(\mathcal{G}) = \{\mathcal{C}_1, ..., \mathcal{C}_C\}, \mathcal{C}_i \subset \mathcal{G}, \forall_{ij} : \mathcal{C}_i \cap \mathcal{C}_j = \emptyset. \tag{4.36}$$

Automatic image classification has two advantages. It releases the user from manually classifying images one by one. And, automatic clustering might reveal new classes of abnormal vertebrae only apparent when considering common deviations from the normal population in terms of multiple features.

4.5 Experiments and Results

The performance of the retrieval algorithm has been evaluated on the NHANES II cervical image data set [86]. As the data set includes a delineation of the vertebra boundaries in the form of point sets only, we obtain a continuous boundary representation by interpolating B-spline curves trough the points. B-spline curves are also used to represent feature functions, emanating from image and shape samples around the vertebra boundaries.

4.5.1 Experiments

For inductive learning we use a population of $M = 100$ normal cervical vertebra images. For commencing one browsing session we use one cervical vertebrae with lower osteophyte not present in the repository. This repository consists of $C = 5$ different images, a mixture of 45 normal vertebrae, 20 vertebrae with upper osteophyte, 68 with lower osteophyte and 49 with both upper and lower osteophyte, yielding a total of $B = 183$ images. The images for the different classes have been randomly selected out of the entire collection of 283 images. The composition of the images for learning and browsing is listed in table 4.1.

Class	Total	Learning	Testing
Normal cervical	145	100	45
Lower osteophyte	69	0	68
Upper osteophyte	20	0	20
Lower and Upper	49	0	49

Table 4.1: Categories of cervical vertebrae and in what data set they have been used

We record vertebra boundaries by $N = 4$ features measured at 50 sample points along their B-spline representation. The first dimension $f^1(v)$ of the feature space $\mathcal{F}$ is the distance between sampled x-coordinate values along the vertebra contour and the x-coordinate value of a single reference point on it. The second dimension $f^2(v)$ is the difference between sampled y-coordinate values and the y-coordinate value of the same reference point. The third dimension $f^3(v)$ is the local bending energy along the contour [79], measuring the curvature along the vertebra contour. The fourth dimension $f^4(v)$ is the gradient magnitude of the image along the vertebra contour, obtained by convolution of the image with the 2-dimensional first order fuzzy derivative at scale $\sigma = 4$ [44]. Hence, we have functions $\mathbf{f}(v) = [f^1(v), f^2(v), f^3(v), f^4(v)]^T$ in 4-dimensional feature space.

Automatic Image Classification

In the first experiment we automatically classify images on the basis of their visual content. The aim is to investigate how accurately the images in the repository, described by $\mathbf{f}_b(v)$, can be classified by considering their deviations from the normal cervical vertebra. We consider a case where $N = 2$ with only geometric features and a case where $N = 4$, with both shape and image features, i.e. $\mathbf{f}_b(v) = [f_b^1(v), f_b^2(v), f_b^3(v), f_b^4(v)]^T$. The classification is done unsupervised using the clustering algorithm in [55], without specifying in advance the number and distribution of the true classifications. For each cluster we count the number of vertebrae assigned to the same class as the class assigned by experts [86].

Image Browsing and Retrieval

The goal of the second experiment is to determine whether the algorithm retrieves images that are relevant to the initial query. After definition of the unknown abnormal vertebra by segmentation of its image at the start of the browsing session, each subsequent iteration consists of selecting a new positive segmented image out of the top $\lambda = 10$ retrieved images. This selected image is used as the query in the next iteration. When the result is a screen full of pictures of negative images, the top ranked image is used as a negative example. Normally, the result of a query will offer a better alternative than the user did the last querying with and the algorithm will converge to the desired population.

To asses how relevant the retrieved images are to the initial query, we measure precision and recall [112], defined as

$$precision = \frac{\text{No. relevant images retrieved}}{\text{Total No. images retrieved}}$$

$$recall = \frac{\text{No. relevant images retrieved}}{\text{Total No. relevant images in collection}}$$

where high precision indicates that from all the images returned by a query, a large proportion of the images are relevant to the search (purity of retrieval). A high recall indicates that from all the images in the repository that are relevant to the query, a large number of these images are indeed returned (completeness of retrieval). In our case, the number of relevant images is the number of images that are of the same class as the initial query image, i.e. images by expert consensus classified as vertebra with lower anterior osteophyte.

We also address the question of how well the retrieval algorithm searches the space of solutions. We express the trade-off between exploration of the feature space and exploitation of previous results by

$$exploration = \frac{\text{No. previously unretrieved relevant images}}{\text{Total No. of retrieved images}}$$

$$exploitation = \frac{\text{No. previously retrieved relevant images}}{\text{Total No. relevant images in collection}}$$

where previously unretrieved images are images that where no part of the top $\lambda = 10$ best results in the history of the browsing session. In this context, exploration is the ability of the retrieval algorithm to search the feature space thoroughly, while exploitation refers to the algorithms ability to use the information about the feature space it has gained to narrow its future search.

4.5.2 Results

Automatic Classification Correctness

Results of image classification on the basis of all $N = 4$ shape and image features are given in table 4.2. A total of $C = 5$ separate clusters are found by the clustering algorithm. All normal vertebrae are grouped into cluster 1. The vertebrae with lower osteophyte are predominantly found in cluster 3, whereas vertebrae with upper anterior osteophyte are also largely found in cluster 3. Cluster 2 primarily contains vertebrae with both lower and upper osteophyte. There is a clear distinction between classification correctness of normal and abnormal vertebrae. However, differences among abnormal vertebrae are less apparent. This is possibly due the fact that all vertebrae are explained with reference to the normal model, thereby possibly disregarding the characteristics of abnormal vertebrae. In comparison to classification on the basis of $N = 2$ shape features (data not shown here) application of both image and shape features improves the classification accuracy. Figure 4.6 shows individual images closest to centers of the clusters of table 4.2.

Class	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Normal cervical	45	0	0	0	0
Lower osteophyte	17	3	31	6	12
Upper osteophyte	8	2	8	0	2
Lower and Upper	3	25	9	4	8

Table 4.2: Automatic classification of the images in the repository on the basis of both shape and image features.

The use of residual information as defined in equation 4.35 contributes to a better classification than one without use of residuals (not shown here). However, as this information only signals when deviation is present not where, it is only suited to discriminate between, e.g. lower anterior osteophyte and lower-upper anterior osteophyte, for which the amount of deviation from the reference model is different. A better classification is expected on the basis of functional data, e.g. by functional discriminant analysis [95], rather than on the basis of principal component scores augmented with residual information.

The experiments with automatic image classification indicate there are no clear distinctions between features constituting the four classes of vertebrae. Rather the classes are characterized by common features, in addition to class-specific features, accentuating the importance of image browsing. For instance, if the user starts with an example image of lower osteophyte it is likely the algorithm retrieves normal vertebrae when that example happens to have many features in common with normal vertebrae. On the other hand, if the user can specifically indicate interest in lower parts of the vertebra but not in upper regions, by providing additional positive examples, the likelihood that only images from the lower osteophyte population will be returned increases. Forming a population of examples is expected to improve the discrimination ability of the algorithm and hence also that of retrieval accuracy.

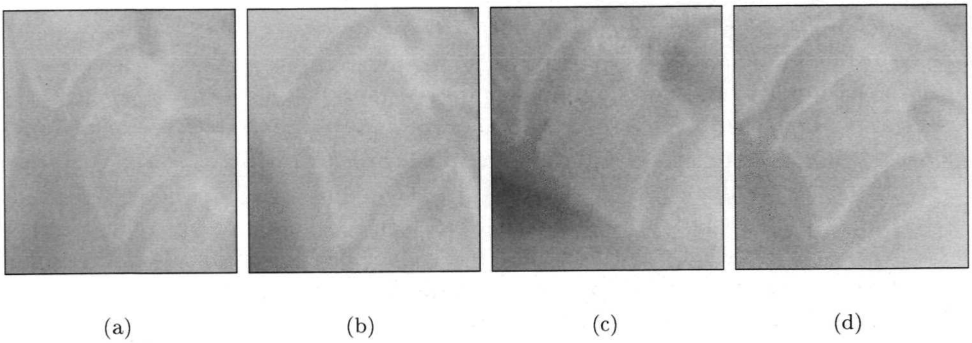


Figure 4.6: Individual images from the repository closest to centers of the found clusters: a) normal cervical vertebrae corresponding to cluster 4, b) vertebra with lower anterior osteophyte corresponding to cluster 5, c) normal vertebra which happens to be closest to the center of cluster 3 and d) a vertebra with both lower and upper anterior osteophyte, closest to the center of cluster 2.

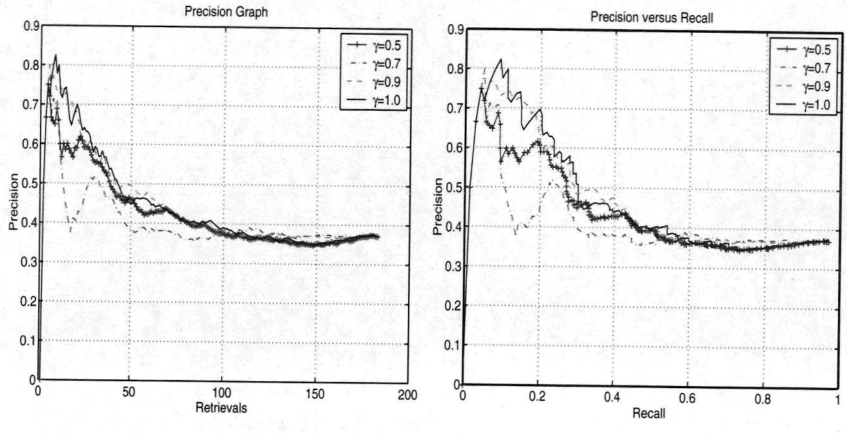
Precision and Recall for Retrieval

Figure 4.7 shows results for precision and recall. The results are the average of a total of 68 browsing sessions, each ending when the first $\lambda = 3$ of the ranked retrieved images are relevant. As the retrievals are ranked and the parameter λ acts as stopping criteria, it is primarily the first part of the precision and recall graphs that provides valuable information. For the innovation rate γ we have taken values between 0.5 and 1.0 as this range requires an acceptable number of iterations to convergence.

Figure 4.7a shows the precision graph, indicating the amount of relevant images shown on screen as a function of the number of retrievals. It can be seen that when a total of 50 images are retrieved about 45 percent of them are relevant for $\gamma = 0.5$ and $\gamma = 1.0$. The recall graph (not shown here) tells that at 50 retrievals about 30 percent of the relevant images in the repository are fetched. Concerning the innovation rate γ, the recall graph shows minor difference for the various values ofγ, whereas the precision graph indicates that, especially for small number of retrievals, high values for γ yield proportionally more precise results than lower values.

The precision versus recall graph shows for $\gamma = 0.5, \gamma = 0.7, \gamma = 0.9, \gamma = 1.0$ how precision decreases as increasingly large fractions of the collection are retrieved. It shows that to retrieve 50 percent of the relevant images from the repository, about 67 percent of the retrieved images will not be relevant, whereas for a recall of 10 percent, a precision of 75 percent is obtained. As concerns the influence of the innovation rate on the retrieval of relevant images, the graphs show that for $\gamma = 1.0$ in general the best performance is obtained.

For the vertebra application the precision and recall graphs tell that completeness of retrieval is difficult to control by population-based incremental learning due to



(a)

(b)

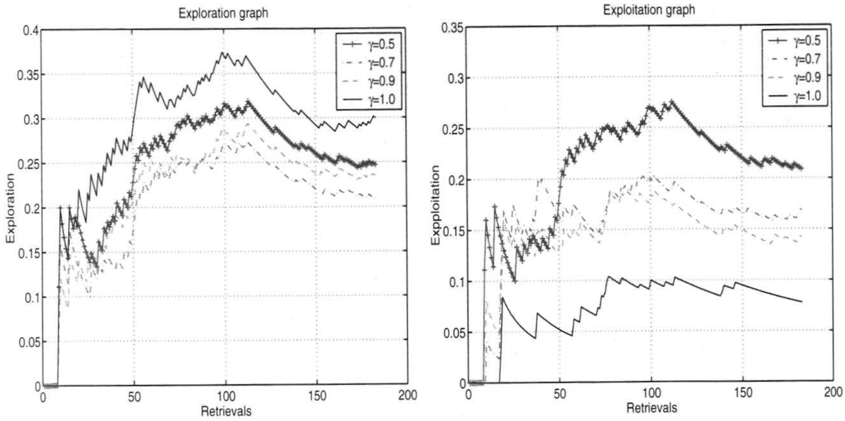
Figure 4.7: Average precision and recall values for 68 browsing sessions: a) recall graph b) precision versus recall graph .

diversity in the content of the images, while precision of retrieval is governed by the amount of focus on specific types of relevant images. From the above we conclude that image retrieval by exploration and exploitation on a population basis is promising when browsing an image repository with a specific target in mind.

Exploration and Exploitation for Browsing

Figure 4.8 shows the average exploration and exploitation graphs of 68 browsing sessions. Figure 4.8a illustrates the percentage of the retrieved relevant images that are not ranked in the top $\lambda = 10$ during the entire browsing session as a function of the number of retrieved images. It can be seen that up to approximately 10 retrievals all the retrieved images are suggested in the top $\lambda = 10$ at least once in the history of the browsing session. As expected most exploration is performed when $\gamma = 1.0$. For example, when a browsing session ends with 50 retrievals then for $\gamma = 1.0$, a percentage of 30 of the relevant retrieved images were not part of the top $\lambda = 10$ in the history of that session. For $\gamma = 0.5$ this is about 25 percent. We can state that high values for γ truly assist the exploration of the image repository by investigating parts of the feature space not entered before.

The graph in figure 4.8b tells the percentage of the retrieved images that are suggested in the top $\lambda = 10$ as a function of the number of retrieved images. It can be seen that when a browsing session ends with 100 retrievals, then for $\gamma = 1.0$ a percentage of 5 of the relevant retrieved images have been part of the top $\lambda = 10$ in the history of the browsing session. For $\gamma = 0.5$ this is about 17 percent. Low values



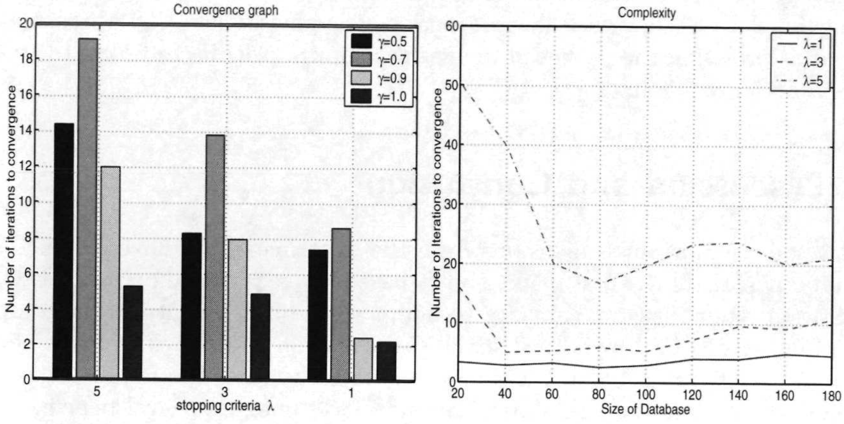
(a) (b)

Figure 4.8: Average exploration and exploitation values for 68 browsing sessions: a) exploration graph and b) exploitation graph.

for the innovation rate γ hence allow to exploit previous results when browsing the image repository. We conclude that exploration and exploitation by population-based incremental learning allows to explicitly control the way of navigation through the image repository to reach a target population efficiently and accurately.

We also investigate the convergence of retrieval for various values of stopping criteria λ. The stopping criteria simply indicates the number of relevant images in the first λ of the retrieved images. The graph in figure 4.9a shows the number of iterations required for convergence when $\gamma = 0.5, \gamma = 0.7, \gamma = 0.9$ and $\gamma = 1.0$. It can be seen that for $\lambda = 5$ a large number of iterations is required, especially for low values of the innovation rate γ. When the value of the innovation rate is large, on the other hand, even for $\lambda = 5$ a reasonable number of iterations is required for convergence. Hence, when aiming for retrieval of relevant images that highly resemble the query examples, a large number of browsing steps are required. The algorithm converges within an acceptable number of browsing steps when the emphasis is on retrieval of images that are less close to the example images, but are relevant to the query.

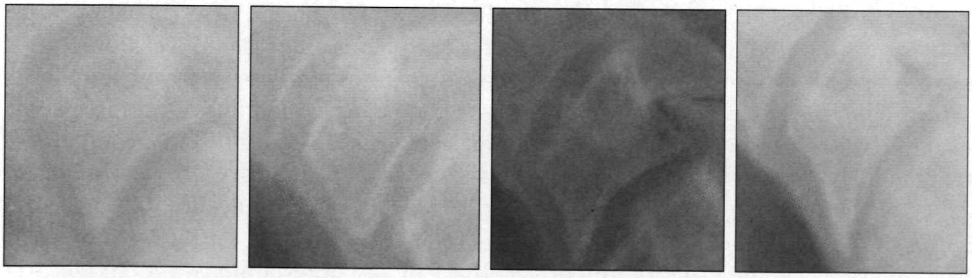
In determining the computational complexity of probabilistic image retrieval [123], we concentrate on scaling with size of the image repository. The graph in 4.9b gives the complexity when retrieving with $\gamma = 0.8$. The average number of iterations to convergence of ten browsing session is shown for increasing size of the image repository. As can be seen, for a small data set the average number of iterations is large. We partially attribute this to the fact that the query examples have features similar to a population not of interest for the query. In that case, the algorithm is not likely to



(a) (b)

Figure 4.9: Average convergence and complexity measurements of 10 browsing sessions: a) convergence graph, b) complexity graph.

fetch relevant images quickly. In fact, in the worst case it can oscillate. This problem diminishes with increasing size of the data set because then queries can be given far away from the overlapping areas. The complexity of the retrieval algorithm is difficult to determine due to the interplay of many of the components of the algorithm.



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

Figure 4.10: Illustration of a query image with retrieval results: a) initial query image is one of vertebra with lower anterior osteophyte, b) best matching image in repository after browsing, c) second best d) third best. Note that all the retrieved images are of vertebrae with lower anterior osteophyte.

Figure 4.10 shows example results of a browsing session. The initial query image is an image of a vertebra with lower anterior osteophyte. The best three matching images after browsing the retrieved images are shown. All three belong to the class of lower anterior osteophyte.

4.6 Discussion and Conclusion

We draw some important conclusions from the experiments we have conducted and the results thereof. In the first place, object-based image retrieval performs well if the segmentation model has the capacity to adapt to the user's concept. In comparing results of image retrieval by a fixed string segmentation model, with results obtained by an adaptive string segmentation model the latter reduces the minimal number of steps required for convergence. In the second place, image retrieval benefits greatly from exploring and exploiting the feature space by means of browsing retrieved images. The drawback of performing several browsing steps is paid back by an increase of the number of retrieved images that are relevant to the query. Thirdly, qualifying repository objects after each browsing step with respect to features that are recorded from the already browsed population with the same deviations from normal allows to quickly focus on the relevant images in the repository. In short, object-based image retrieval with help of an adaptive segmentation model as learned from a browsed population allows to learn the user's retrieval intention and recover relevant images in a limited number of steps.

A number of issues remain uninvestigated. It is expected that negative feedback improves the discrimination ability of the retrieval method in addition to dealing with local minima of the search space, e.g. when in response to user feedback the algorithm either returns exactly the same images as in a previous iteration or returns non-relevant images. The consequence of negative feedback to the exploration and exploitation of the feature space needs further study. Furthermore, in this application we have pre-selected a number of features for the definition of the object boundaries. A more precise definition of object-boundaries requires studying a larger number of features. When the different object classes are known, as in our clinical example application, features can be studied separately for each class in order to build a dedicated string model for each abnormal object. This is expected to improve the discrimination ability of the proposed method and hence also that of retrieval accuracy.

We note that, although in this paper we applied the method for retrieval of vertebrae images, it can be used for any other object in any other application. In fact, the components of the proposed method can be considered as individual contributions that are not restricted to image retrieval. Functional data analysis [95] of a given set of multiple continuous boundary features gives detailed in sight in the most important characteristics of objects and the natural variations therein. On top of this, such a statistical analysis is very much suited for multi-feature image segmentation and classification. It requires only a significant set of example images with delineations of objects. The population-based incremental learning technique originates from function optimization and has been used for a wide range application. This technique is appealing because of its simplicity and because it generalizes easily. The string representation of objects is not a prerequisite for this algorithm since it relies on evolving a probabilistic model of a population rather than on the evolution of individuals.

In conclusion, the difference with other image retrieval approaches is that we do not aim at finding an object closest to a query object in a single step. Rather we try to incrementally build a distribution of example objects that explain the appearance of the target object to the degree that ranking images in the repository by their visual features yields relevant images. Construction of such a collection of examples is compelling because: a) the query object may be in an area with overlapping distributions, prohibiting any conclusions about the class it belongs to b) the segmentation model may not capture the query object properly, requiring more input objects to learn what is to be found in an image c) the feature extracted from segmentation may not be representative for the user's retrieval intention, calling for a gradual optimization of features to consider. The combination of the string-based segmentation method with concepts borrowed from population-based incremental learning techniques [3] deals nicely with these problems. The proposed method works well for images of objects that are not easily defined by one feature, but require multiple features and multiple examples of these features to precisely capture the sought object.

Chapter 5

Necklaces: Inhomogeneous and Point-Enhanced Deformable Models*

In many advanced segmentation problems objects have inhomogeneous boundaries, hindering segmentation under uniform boundary assumption. We present a multi-feature image segmentation method, called *necklaces*, that exploits local inhomogeneities to reduce the complexity of the segmentation problem. Multiple continuous boundary features, deduced from a set of training objects, are statistical analyzed and encoded into a deformable model. On the deformable model salient features are identified on the basis of the local differential geometric characteristics of the features, yielding a classification into point landmarks, curve landmarks and sheet points. Salient features are exploited within a priority segmentation scheme that tries to find complete boundaries in unknown image, first by landmarks then by sheet points. The application of our method to segment vertebrae from CT data shows promising results despite their articulated morphology and despite the presence of interfering structures.

*Conditionally accepted for publication in Computer Vision and Image Understanding

5.1 Introduction

Image segmentation, the partitioning of an image into distinct areas whose points can be characterized according to some specific property, is commonly performed on the basis of object boundaries. Image information that is associated with these boundaries is extracted locally and is then linked in one or another way by a computational method. When images contain a complex ensemble of boundaries that are fractured, occluded, convoluted or inhomogeneous otherwise, computation may be seriously hampered under uniform boundary assumption. Segmentation under this premise only works well for a limited range of segmentation problems where objects are smooth and their edges are well-defined along the entire boundary. It is therefore imperative to strive for segmentation that exploits inhomogeneities rather then apprehending them as obstacles. This is viable as inhomogeneities often form salient information and hence are suitable boundary landmarks for image segmentation.

The importance of boundary landmarks has been recognized in several works. Methods that profit from point landmarks are proposed in e.g. [74], [70], [125], [54], [9]. In here, the user manually defines a set of point landmarks for computation to align it to the target object in the image. This is disadvantageous because manual definition is time-consuming and prone to failures. For this reason automatic landmark definition has been proposed in e.g. [121], [28], [36], [47], [109], [29]. Point landmarks as well as curve landmarks are defined on geometrical or analytical grounds. Interactive detection is advocated in e.g. [100], [99] and [46] since automatic definition and detection of point landmarks is complex and prone to errors [98]. Interaction in this case means that either a region of interest or an approximate position of a landmark is given by the user to aid computation. We commemorate that existing methods define and employ landmarks manually, automatically or interactively, the common denominator being the application of predefined features for landmark definition.

We conceive of landmark-based segmentation as a multi-feature selection procedure aiming at automatically defining and gradually exploiting landmarks. In a multi-feature approach object boundaries are defined using multiple image and shape features, often giving rise to a repertoire of landmarks. For example, when examining the bending along the heart's boundary in cardiac images, curvature feature values will give a peak at the lower tip of the outer wall. When, in addition observing the edge structure along the heart's boundary, this might give a highlight at the position where the aorta enters the heart. The local deviations in multi-feature observations can be seen as landmarks with differing degrees of freedom. We strive to learn such landmarks from a given set of training objects and to exploit them accordingly. The question addressed in this work is how to conveniently define landmarks and how to befittingly apply them for segmentation of volumetric image data.

The work presented in this chapter is organized as follows. In section 2 object boundary landmarks are discussed. Attention is devoted to object boundary representation, definition of landmarks and their application. Section 3 focuses on landmark-based segmentation by necklaces. Necklaces are introduced in terms of model construction, model qualification and model optimization. Experiments and results follow in section 4. The chapter concludes with a discussion in section 5.

5.2 Object Boundary Landmarks

We discuss requirements for boundary landmarks, their definition and application. Before we do that we first briefly describe object representation which plays an essential role in defining suitable landmarks.

5.2.1 Boundary Representation

In chapter 3, the appearance of a two-dimensional object is learned from a repertoire of features. Multiple continuous features are extracted from M training images $I_m(\mathbf{x})$ for $m = 1, ..., M$ and $I_m : \mathbf{x} \in \Re^2 \rightarrow \Re$, for each of which the known segmentation is represented by smooth curves $\mathbf{s}_m(u)$, $\mathbf{s} : u \in \Re \rightarrow \Re^2$. For the mth learning example, the shape $\mathbf{s}_m(u)$, relates to the image at points $I_m(\mathbf{s}_m(u))$. This is expressed in terms of N features derived from the shape as well as from the image. The mapping $\mathbf{f} : u \in \Re \rightarrow \mathcal{F}$ handles this, yielding *feature function*

$$\mathbf{f}_m(u) = [f_m^1(u), ..., f_m^N(u)]. \tag{5.1}$$

This way, features are integrally captured by space curves in the N-dimensional feature space $\mathcal{F}$, each dimension associated with one feature. This representation allows for conveniently localizing landmarks arising from multiple boundary features of the object. The landmarks are typically maxima, minima and zero-crossings of the space curves, identified at the level of derivatives by differential curve geometry.

Here, we take over the multi-feature object representation in order to exploit landmarks that appear in feature functions for image segmentation. We do this for three-dimensional objects, extending two-dimensional concepts from chapter 3 to three-dimensions. In this case, training images $I_m(\mathbf{x})$ are three-dimensional, i.e. $I_m : \mathbf{x} \in \Re^3 \rightarrow \Re$. Object outlines are smooth surfaces represented by $\mathbf{s}_m(\mathbf{u})$, $\mathbf{s} : \mathbf{u} \in \Re^2 \rightarrow \Re^3$. The mapping $\mathbf{f} : \mathbf{u} \in \Re^2 \rightarrow \mathcal{F}$ yields feature functions

$$\mathbf{f}_m(\mathbf{u}) = [f_m^1(\mathbf{u}), ..., f_m^N(\mathbf{u})]. \tag{5.2}$$

Features are now conveniently captured by manifolds [82] in the N-dimensional feature space $\mathcal{F}$. In search for landmarks, we have to consider high curvature points in these manifolds. Differential geometric surface analysis then localizes landmarks.

We briefly summarize requirements for landmarks, conforming to the ones in [92].

- Landmarks should be *selective*. Landmarks that do not posses characteristics that discriminate them from other boundary points have no additional value for segmentation.

- Landmarks should be *sparse*. Landmarks that are abundantly present are less suitable for concise representation of the segmentation problem.

- Landmarks should be *detectable*. Landmarks of which the presence can not be confirmed computationally may cause erroneous solutions.

- Landmarks should be *robust* in their appearance. Boundary landmarks that are inconsistent in their frequency of occurrence and that have highly variable appearance may hamper segmentation.

These requirements are general of nature; they hold for a large number of points on and off boundaries. We concentrate on boundary landmarks.

5.2.2 Definition of Landmarks

We take a differential geometric approach to find suitable landmarks from feature functions, reducing landmark definition to localizing surface landmarks. Surface landmarks are captured by the well-known first and second fundamental forms (see e.g. [65], [117]). In this section, we will discuss landmark definition for the case that the x, y and z-coordinates of a surface are the only features to consider, i.e. $\mathbf{f} : \mathbf{u} \in \Re^2 \rightarrow \Re^3$.

For the surface $\mathbf{f}(\mathbf{u})$ the *first fundamental* form $I(\mathbf{u}, d\mathbf{u})$ measures the length of a small movement $d\mathbf{f}$ from the point $\mathbf{f}(\mathbf{u})$ in the direction given by $d\mathbf{u}$. The first fundamental form $I(\mathbf{u}, d\mathbf{u})$ is determined on the basis of the surface derivatives $\mathbf{f_u}$ in the $\mathbf{u} = [u_1, u_2]^T$ direction. It is defined as

$$I(\mathbf{u}, d\mathbf{u}) = d\mathbf{u}^T \begin{bmatrix} \mathbf{f}_{u_1} \cdot \mathbf{f}_{u_1} & \mathbf{f}_{u_1} \cdot \mathbf{f}_{u_2} \\ \mathbf{f}_{u_2} \cdot \mathbf{f}_{u_1} & \mathbf{f}_{u_2} \cdot \mathbf{f}_{u_2} \end{bmatrix} d\mathbf{u}. \tag{5.3}$$

For the *second fundamental form* the unit normal vector is needed

$$\mathbf{n}(\mathbf{u}|\mathbf{f}) = \frac{\mathbf{f}_{u_1} \times \mathbf{f}_{u_2}}{|\mathbf{f}_{u_1} \times \mathbf{f}_{u_2}|}. \tag{5.4}$$

It measures the change $d\mathbf{n}$ of the normal vector and the change in the surface position $d\mathbf{f}$ given by

$$II(\mathbf{u}, d\mathbf{u}) = d\mathbf{u}^T \begin{bmatrix} \mathbf{f}_{u_1 u_1} \cdot \mathbf{n} & \mathbf{f}_{u_1 u_2} \cdot \mathbf{n} \\ \mathbf{f}_{u_1 u_2} \cdot \mathbf{n} & \mathbf{f}_{u_2 u_2} \cdot \mathbf{n} \end{bmatrix} d\mathbf{u}. \tag{5.5}$$

A surface is essentially expressed by mapping these two matrices into the normal curvature, defined as

$$\kappa_0(\mathbf{u}|\mathbf{f}) = -\frac{d\mathbf{f} \cdot d\mathbf{n}}{d\mathbf{f} \cdot d\mathbf{f}} \tag{5.6}$$

and the principal curvatures, which occur when $d\mathbf{n}$ and $d\mathbf{f}$ are aligned for a particular direction $d\mathbf{u}$. The maximum curvature κ_1 and the minimum curvature κ_2 are defined as

$$\kappa_1(\mathbf{u}|\mathbf{f}) = \frac{\mathbf{f}_{u_1 u_1} \cdot \mathbf{n}}{\mathbf{f}_{u_1} \cdot \mathbf{f}_{u_1}}, \tag{5.7}$$

$$\kappa_2(\mathbf{u}|\mathbf{f}) = \frac{\mathbf{f}_{u_2 u_2} \cdot \mathbf{n}}{\mathbf{f}_{u_2} \cdot \mathbf{f}_{u_2}}. \tag{5.8}$$

The directions **u** and **w** in which the principal curvatures occur, the *principal curvature directions*, are determined from $d\mathbf{u} = (du_1, du_2)$ by solving the quadratic equation

$$\begin{bmatrix} du_1 du_1 & -du_1 du_2 & du_2 du_2 \\ \mathbf{f}_{u_1} \cdot \mathbf{f}_{u_1} & \mathbf{f}_{u_1} \cdot \mathbf{f}_{u_2} & \mathbf{f}_{u_2} \cdot \mathbf{f}_{u_2} \\ \mathbf{f}_{u_1 u_1} \cdot \mathbf{n} & \mathbf{f}_{u_1 u_2} \cdot \mathbf{n} & \mathbf{f}_{u_2 u_2} \cdot \mathbf{n} \end{bmatrix} = 0. \tag{5.9}$$

An alternative to the principal curvatures are the direction independent *mean curvature* κ_3 and *Gaussian curvature*, κ_4:

$$\kappa_3(\mathbf{u}|\mathbf{f}) = \kappa_1 \kappa_2, \tag{5.10}$$

$$\kappa_4(\mathbf{u}|\mathbf{f}) = \frac{\kappa_1 + \kappa_2}{2}. \tag{5.11}$$

For specific applications it is desirable to make these quantities scale-independent by multiplication with the surface area

$$\delta A(\mathbf{u}|\mathbf{f}) = \sqrt{(\mathbf{f}_{u_1})^2 + (\mathbf{f}_{u_2})^2}. \tag{5.12}$$

This way two surfaces that are different in size but similar in shape can be considered the same with regard to curvature.

Two frequently used landmark localization approaches are discussed. They employ principal curvatures $\kappa_1(\mathbf{u}|\mathbf{f})$ and $\kappa_2(\mathbf{u}|\mathbf{f})$, the largest curvature $\kappa_1(\mathbf{u}|\mathbf{f})$ being the one with the highest value.

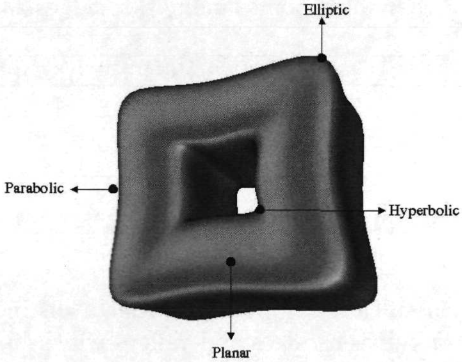


Figure 5.1: An object with four types of boundary points: flat/planar, ridge/parabolic, peak/elliptic, saddle/hyperbolic.

One way of classifying surface points is based on the principal curvatures $\kappa_1(\mathbf{u}|\mathbf{f})$ and $\kappa_2(\mathbf{u}|\mathbf{f})$ and associated principal directions $\mathbf{v}(\mathbf{u}|\mathbf{f})$ and $\mathbf{w}(\mathbf{u}|\mathbf{f})$ (e.g. [40], [121]). Two curvature functions $\kappa_1(s|\mathbf{c})$ and $\kappa_2(s|\mathbf{c})$ are determined which measure the curvature of the surface along a curve $\mathbf{c}(s)$, parameterized by s, that goes through the point of interest and has a tangent equal to the corresponding principal direction at that point. Zero-crossings of the directional derivatives of these curvature functions are used to classify points as elliptic, parabolic or planar (see table 5.1). Elliptic points can be viewed as point landmarks, parabolic points as curve landmarks.

	$\nabla\kappa_1(s\|\mathbf{c}) = 0$	$\nabla\kappa_1(s\|\mathbf{c}) \neq 0$
$\nabla\kappa_2(s\|\mathbf{c}) = 0$	elliptic	parabolic
$\nabla\kappa_2(s\|\mathbf{c}) \neq 0$	parabolic	planar

Table 5.1: Shape classification on the basis of zero-crossings of the curvature functions $\kappa_1(s|\mathrm{c})$ and $\kappa_2(s|\mathrm{c})$.

This approach provides a geometrically meaningful distinction between different types of surface points. The disadvantage is that on the basis of the zero-crossing only it is not possible to distinguish between "strong" and "weak" elliptic/parabolic points. Even almost flat structures and minor bumps in the surface are classified as elliptic or parabolic.

An other common way of classifying surface points is based on the signs of principal surface curvatures (e.g. [8], [117]). The possible configurations are listed in table 5.2. In this case, surface points are classified into one of six basic types: peak, pit, ridge, valley, saddle and flat. An equivalent classification is reached by using the signs of the Gaussian and mean curvature. Saddle points are then resolved into saddle ridge, saddle valley and minimal surface yielding eight basic surface point types. Peaks pits and saddle points can be viewed as point landmarks, ridges as curve landmarks.

	$\kappa_1(\mathbf{u}\|\mathbf{f}) < 0$	$\kappa_1(\mathbf{u}\|\mathbf{f}) = 0$	$\kappa_1(\mathbf{u}\|\mathbf{f}) > 0$
$\kappa_2(\mathbf{u}\|\mathbf{f}) < 0$	peak	ridge	saddle
$\kappa_2(\mathbf{u}\|\mathbf{f}) = 0$	ridge	flat	valley
$\kappa_2(\mathbf{u}\|\mathbf{f}) > 0$	saddle	valley	pit

Table 5.2: Shape classification on the basis of the signs of the principal curvatures.

The advantage of this classification approach is that a distinction is also made between pits and peaks and between ridges and valleys which have identical shape but are embedded differently in three-dimensional space. However, as with the previous classification scheme, minor bumps in the surface are classified as elliptic or parabolic. The number of landmark points derived this way can be very large. In addition the distinction between different types of landmarks is not always clear, making them less suited for aiding segmentation.

The above discussion motivates the definition of landmarks on the basis of a *threshold for the principal curvature values.* We modify the classification scheme based on the signs of the principal curvature and classify surface points depending on whether the absolute values of the principal curvatures $\kappa_1(\mathbf{u}|\mathbf{f})$ and $\kappa_2(\mathbf{u}|\mathbf{f})$ exceed a predefined threshold c_s, where the value of c_s is derived from a priori knowledge.

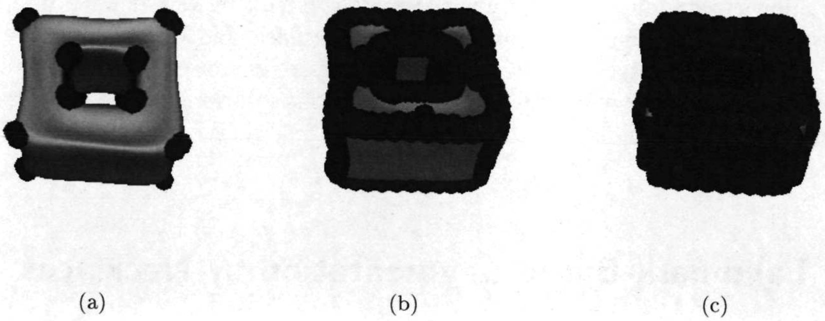


Figure 5.2: Classification of surface points on the basis of a threshold for the principal curvatures: a) point landmarks indicated by spheres, b) curve landmarks and c) sheet points.

	$\kappa_1(\mathbf{u}\|\mathbf{f}) < -c_s$	$-c_s \leq \kappa_1(\mathbf{u}\|\mathbf{f}) \leq c_s$	$c_s < \kappa_1(\mathbf{u}\|\mathbf{f})$
$\kappa_2(\mathbf{u}\|\mathbf{f}) < -c_s$	point landmark	curve landmark	sheet points
$-c_s \leq \kappa_2(\mathbf{u}\|\mathbf{f}) \leq c_s$	curve landmark	sheet point	curve landmark
$c_s < \kappa_2(\mathbf{u}\|\mathbf{f})$	point landmark	curve landmark	point landmark

Table 5.3: Shape classification on the basis of a threshold for the principal curvatures.

Three types of surface points are distinguished this way as illustrated in figure 5.2. We refer to them as point landmarks, curve landmarks and sheet points. Point landmarks are uniquely localized due the presence of a strong curvature in both principal directions. Curve landmarks are well defined in only one of the principal directions. Sheet points are only well defined in the normal direction. The points are listed in table 5.3.

5.2.3 Application of Landmarks

When surface points are classified into point landmarks, curve landmarks and sheet points as defined above, it is natural to adopt a priority scheme when objects are segmented. We employ the following order in finding objects:

1. Detect point landmarks. The result is a rough estimate of the position of the object by its point landmarks.

2. Given 1, localize curve landmarks. The result is an outline of the object.

3. Detect sheet points departing from the solution of 2. The result is the location of all boundary points.

This priority scheme enables to search for curve landmarks and sheet points once point landmarks are detected. The incremental approach has two advantages. In the first place it allows to handle complex image scenes by exploiting well-defined information first, reducing disturbance from unrelevant image data and diminishing the search space. Another advantage is that point landmarks reduce user interaction to point-to-point correspondence when dealing with erroneous solutions due to propagation of errors made in point landmark localization.

5.3 Landmark-Based Segmentation by Necklaces

Landmark-based segmentation can be applied adequately within the deformable model platform. Deformable model methods have been extensively used for image segmentation (e.g. [114], [119],[16],[15]). To accommodate deformable models for segmentation of inhomogeneous boundaries, we have developed one that uses multiple features to define and exploit boundary landmarks. As it reminds of a *necklace* because of its inhomogeneous and point-enhanced character we call it as such.

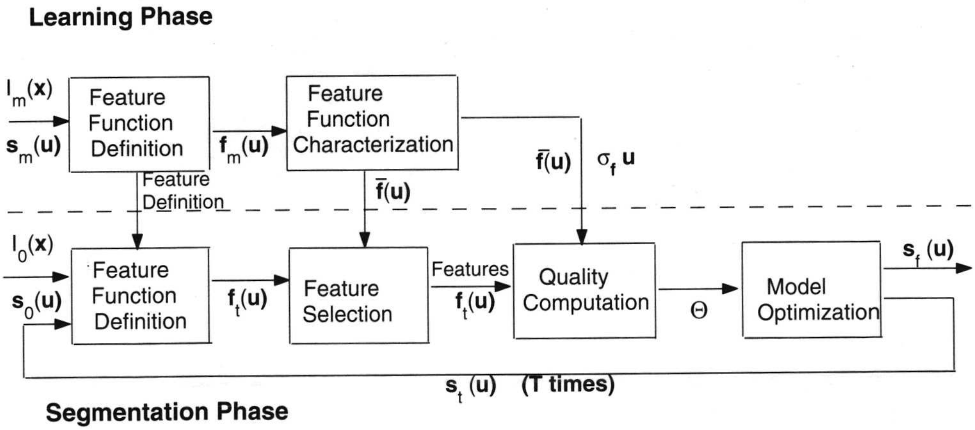


Figure 5.3: Overview of components of the necklace segmentation technique. Note that at the start of the learning phase, we need a set of images $I_m(\mathbf{x}), m = 1...M$ with corresponding correct ground-truth segmentation represented by $\mathbf{s}_m(\mathbf{u})$.

The essence of necklaces is formulated in terms of the following aspects: how the boundary model is constructed, how the objective function is formulated and how optimization is performed. Figure 5.3 shows the main components of the necklace segmentation technique. They will be addressed in the following sections.

5.3.1 Model Construction

As in chapter 3, we aim at learning object models rather than constructing them from a priori geometrical or analytical knowledge. However, in contrast to chapter 3, where image and shape features are weighted according to the variation seen in the training feature functions, here we select them according to saliency seen in these functions. We are aware of the fact that this may introduce shortcomings related to neglecting variational information. However, this has inconsequential affects to the concepts proposed here.

Shape Features

Object shape is learned from a training set of M surfaces $\mathbf{s}_m(\mathbf{u}), m = 1, ..., M$, embedded in example images $I_m(\mathbf{x})$. A number of shape features (e.g. curvature) are recorded along $\mathbf{s}_m(\mathbf{u})$ to obtain a population of features. The population average for the nth feature is

$$\bar{f}^n(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} f^n_m(\mathbf{u}|\mathbf{s}_m) \tag{5.13}$$

where $f^n_m(\mathbf{u}), n \in \{1, ..., N\}$ is the feature function emanating from fitting a surface through the feature samples taken along $\mathbf{s}_m(\mathbf{u})$. The feature functions are smoothed and aligned to properly compare features at same positions along the boundary. The variance in shape feature values is

$$\sigma_{f^n}(\mathbf{u}) = \left(\frac{1}{M} \sum_{m=1}^{M} (f^n_m(\mathbf{u}|\mathbf{s}_m) - \bar{f}^n(\mathbf{u}))^2 \right)^{1/2} . \tag{5.14}$$

As shape feature we select the translation and rotation invariant *mean curvature* made scale independent by multiplication with the local surface area $\delta A(\mathbf{u}|\mathbf{s}_m)$. Apart from this, we also learn the principal curvature values $\kappa_1(\mathbf{u}|\mathbf{s}_m)$ and $\kappa_2(\mathbf{u}|\mathbf{s}_m)$ to relieve landmark localization in subsequent steps. Scalar and vector features form dimensions of the feature space. Table 5.4 lists the features and their definition.

Feature	Dimension	Definition
1st principal curvature	$f^1_m(\mathbf{u}\|\mathbf{s}_m)$	$k_1(\mathbf{u}\|\mathbf{s}_m)$
2nd principal curvature	$f^2_m(\mathbf{u}\|\mathbf{s}_m)$	$k_2(\mathbf{u}\|\mathbf{s}_m)$
mean curvature	$f^3_m(\mathbf{u}\|\mathbf{s}_m)$	$(k_1(\mathbf{u}\|\mathbf{s}_m) * k_2(\mathbf{u}\|\mathbf{s}_m))/\delta A(\mathbf{u}\|\mathbf{s}_m)$

Table 5.4: Shape features in our implementation. First principal curvature values $f^1_m(\mathbf{u}|\mathbf{s}_m)$ form the first dimension of $\mathcal{F}$ and second principal curvature values $f^2_m(\mathbf{u}|\mathbf{s}_m)$ form the second dimension. The third dimension is formed by scale independent *mean curvature* values $f^3_m(\mathbf{u}|\mathbf{s}_m)$.

Image Features

As the gray-level patterns about a boundary point will often be similar in different images, we also statistically capture the image data along the object boundary. The statistics of image features along $I_m(\mathbf{s}_m(\mathbf{u}))$ are computed analogously to the shape statistics:

$$\bar{f}^n(\mathbf{u}) = \frac{1}{M}\sum_{m=1}^{M} f_m^n(\mathbf{u}|I_m, \mathbf{s}_m) \qquad (5.15)$$

and corresponding variance

$$\sigma_{f^n}(\mathbf{u}) = \left(\frac{1}{M}\sum_{m=1}^{M}\left(f_m^n(\mathbf{u}|I_m, \mathbf{s}_m) - \bar{f}^n(\mathbf{u})\right)^2\right)^{1/2}. \qquad (5.16)$$

To highlight specific structures in the image while suppressing unrelevant ones, we use filter information obtained from the structure tensor. The structure tensor is commonly used to capture the local three-dimensional structure of the image (e.g. [52], [87], [28], [127]) as it contains more information about the image than gradient information only. For an image position $I(\mathbf{x})$ it is defined by

$$\mathbf{M}(\mathbf{x};\sigma_d) = \begin{pmatrix} \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_1}(\mathbf{x};\sigma_d)} & \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \\ \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_2}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_2}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \\ \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_1}(\mathbf{x};\sigma_d)} & \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \end{pmatrix} (5.17)$$

where σ_d is the differentiation scale and $\overline{(.)}$ denotes smoothing with scale σ_s to obtain the dyadic product. The matrix elements I_{x_i} denote partial image derivatives defined as convolution of the image with derivatives of a Gaussian function

$$I_{x_i}(\mathbf{x};\sigma_d) = I(\mathbf{x}) * G_{x_i}(\mathbf{x};\sigma_d).$$

The eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of $\mathbf{M}(\mathbf{x};\sigma_d)$ are used to define image feature functions. When all eigenvalues are sufficiently large this indicates a point-like structure. Feature function $f_m^4(\mathbf{u}|I_m, \mathbf{s}_m)$ records a boundary in an image filtered to highlight such point-like structures. Two eigenvalue, λ_1, λ_2, a multitude larger than the smallest eigenvalue λ_3 indicates a point on a curve-like structure. Feature function $f_m^5(\mathbf{u}|I_m, \mathbf{s}_m)$ records a boundary in an image filtered to highlight such tubular-like structures. Similarly, $f_m^6(\mathbf{u}|I_m, \mathbf{s}_m)$ measures intensity values in an image which has been processed to highlight sheet-like structures. The image features are listed in table 5.5.

The normalization constant c_t in table 5.5 reflects the minimum required image variation for an image point to be highlighted as one of the three boundary point types. This way we can focus on strong point landmarks and curve landmarks while disregarding image variations caused by minor bumps in the object boundary or by noise.

At this point we have a statistical description of the object boundary of interest. The relevant information is contained in the $N = 6$ dimensional population average feature function $\bar{\mathbf{f}}(\mathbf{u}) = [\bar{f}^1(\mathbf{u}), ..., \bar{f}^6(\mathbf{u})]$ and variation $\boldsymbol{\sigma}_{\mathbf{f}}(\mathbf{u}) = [\sigma_{f^1}(\mathbf{u}), ..., \sigma_{f^6}(\mathbf{u})]$. We transfer this information to the segmentation phase.

Feature	Dimension	Definition
image point points	$f^4_m(\mathbf{u}\|I_m, \mathbf{s}_m)$	$1 - e^{-\frac{\lambda_3}{c_t}}$
image sheet landmarks	$f^5_m(\mathbf{u}\|I_m, \mathbf{s}_m)$	$1 - e^{-\frac{\lambda_2}{c_t}}$
image curve landmarks	$f^6_m(\mathbf{u}\|I_m, \mathbf{s}_m)$	$1 - e^{-\frac{\lambda_1}{c_t}}$

Table 5.5: Image features in our implementation: $f^4_m(\mathbf{u}|I_m, \mathbf{s}_m)$ highlights point landmarks, $f^5_m(\mathbf{u}|I_m, \mathbf{s}_m)$ highlights curve landmarks, $f^6_m(\mathbf{u}|I_m, \mathbf{s}_m)$ highlights sheet points.

5.3.2 Model Qualification

For segmentation of an unknown image we exploit landmarks found in $\bar{\mathbf{f}}(\mathbf{u})$.

Deformable Surface

The *deformable surface* $\mathbf{s}_t(\mathbf{u}), \mathbf{s}_t : \mathbf{u} \in U \subset \Re^2 \rightarrow \Re^3$ is used to actively find an object in the unknown image $I_0(\mathbf{x})$. It deforms in time t to suggest new features extracted from it and the image it lives in, to be compared to $\bar{\mathbf{f}}(\mathbf{u})$ and $\sigma_{\mathbf{f}}(\mathbf{u})$ as the statistics from learning. The initial deformable surface $\mathbf{s}_0(\mathbf{u})$ is the average of the aligned training shapes

$$\mathbf{s}_0(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} \mathbf{s}_m(\mathbf{u}). \tag{5.18}$$

On the deformable surface, point landmarks, curve landmarks and sheet points are localized. This is done by investigating at path positions $\mathbf{u}$ whether the principal curvatures of the population average $\bar{\mathbf{f}}(\mathbf{u})$ exceed the predefined threshold c_s. For instance, if at $\bar{f}^n(\mathbf{u})$ the statistical average of both principal curvatures exceeds the predefined threshold, than $\mathbf{s}_t(\mathbf{u})$ is considered a point landmark. Hence,

$$\mathcal{U}_A = \{\mathbf{u} \mid -c_s > \kappa_1(\mathbf{u}|\bar{\mathbf{f}}) > c_s, -c_s > \kappa_2(\mathbf{u}|\bar{f}) > c_s \}\ , \tag{5.19}$$

$$\mathcal{U}_B = \{\mathbf{u} \mid -c_s > \kappa_1(\mathbf{u}|\bar{\mathbf{f}}) > c_s, -c_s < \kappa_2(\mathbf{u}|\bar{\mathbf{f}}) < c_s \}\ , \tag{5.20}$$

$$\mathcal{U}_C = \{\mathbf{u} \mid -c_s < \kappa_1(\mathbf{u}|\bar{\mathbf{f}}) < c_s, -c_s < \kappa_2(\mathbf{u}|\bar{\mathbf{f}}) < c_s \}\ . \tag{5.21}$$

The disjoint sets $\mathcal{U}_A$,$\mathcal{U}_B$,$\mathcal{U}_C$ together contain all path positions, with $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_A$ being point landmarks, $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_B$ curve landmarks and $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_C$ sheets points.

Objective Function

To qualify the deformable surface when it is in an unknown image $I_0(\mathbf{x})$, an objective function is defined that measures how much a recorded boundary deviates from the population average. The deviation is calculated in terms of the above defined local shape and image features. An integration is carried to compute a global measure of deviation for all N features measured along $I_0(\mathbf{s}_t(\mathbf{u}))$. Denoting the N feature functions emanating from $I_0(\mathbf{s}_t(\mathbf{u}))$ by $\mathbf{f}_t(\mathbf{u}) = [f^1(\mathbf{u}), ..., f^N(\mathbf{u})]$, the global objective function summarizes the fit as follows

$$\Theta(\mathbf{s}_t|\bar{\mathbf{f}}, \sigma_{\mathbf{f}}, \mathbf{f}_t) = \int_{\mathbf{u}} \frac{1}{N} \sum_{n=1}^{N} \omega^n(\mathbf{u}) \theta^n(\mathbf{u}|\bar{f}^n, \sigma_{f^n}, f^n) d\mathbf{u}. \tag{5.22}$$

The weight function $\omega^n(\mathbf{u})$ is the actual feature selector. For each feature n it indicates the relative importance of that feature based on the *mean curvature* of the population average for feature n, denoted by $\kappa_3(\mathbf{u}|\bar{f}^n)$. Currently, $\omega^n(\mathbf{u})$ selects one feature and turns off others:

$$\omega^n(\mathbf{u}) = \begin{cases} 1 & \text{if } \kappa_3(\mathbf{u}|\bar{f}^n) > \kappa_3(\mathbf{u}|\bar{f}^i), i \in \{1, ..., N\}, i \neq n \\ 0 & \text{otherwise} \end{cases} \tag{5.23}$$

The local objective function $\theta^n(\mathbf{u}|\bar{f}^n, \sigma_{f^n}, f^n)$ measures the distance between expected and recorded values for feature n. To ensure a controllable distance measure, the Mahalanobis distance [34] is computed using population average and variation information obtained from learning. For the nth feature this means

$$\theta^n(\mathbf{u}|\bar{f}^n, \sigma_{f^n}, f^n) = \upsilon(\mathbf{u}) \left(\frac{\bar{f}^n(\mathbf{u}) - f^n(\mathbf{u})}{\sigma_{f^n}(\mathbf{u})} \right)^2. \tag{5.24}$$

The local objective function is controlled by means of weighting function $\upsilon(\mathbf{u})$. It regulates the contribution of point landmarks, curve landmarks and sheet points to the objective function value using prespecified weights υ_A, υ_B and υ_C

$$\upsilon(\mathbf{u}) = \begin{cases} \upsilon_A & \text{if } \mathbf{u} \in \mathcal{U}_A \\ \upsilon_B & \text{if } \mathbf{u} \in \mathcal{U}_B \\ \upsilon_C & \text{if } \mathbf{u} \in \mathcal{U}_C \end{cases} \tag{5.25}$$

The weights are positive and subjected to the following constraint

$$\upsilon_A + \upsilon_B + \upsilon_C = 1. \tag{5.26}$$

In the priority scheme discussed in section 5.2.3, the first step is performed using the following weight settings $\upsilon_A = 1, \upsilon_B = 0, \upsilon_C = 0$, step two using $\upsilon_A = 0, \upsilon_B = 1, \upsilon_C = 0$, and three using $\upsilon_A = 0, \upsilon_B = 0, \upsilon_C = 1$. The weights may also be set such that features along the entire surface contribute to the definition of the object, constrained according to equation 5.26.

5.3.3 Model Optimization

The optimization of the deformable surface corresponds to the minimization of the objective function. Optimization only affects the geometry of the initial surface $\mathbf{s}_{t=0}$

by changing the position of surface points while keeping the same uniform parameterization. The aim is to find the optimal deformable surface $\mathbf{s}_f(\mathbf{u})$, such that

$$\mathbf{s}_f = \underset{\mathbf{s}_t}{\operatorname{argmin}} \quad \Theta(\mathbf{s}_t|\bar{\mathbf{f}}, \boldsymbol{\sigma}_\mathbf{f}, \mathbf{f}_t) \tag{5.27}$$

Optimization involves two main steps for each surface point. In the first step a new suggested position is calculated based on the fit quality of the deformable surface. This is followed by movement of the deformable surface in order to reposition each point as close as possible to the newly preferred position.

Search Scheme

We adopt a priority scheme when optimizing the deformable surface; different surface points are fitted to the image data at different times and in different number of dimensions. The aim is to first search for well-defined points in the image and to exploit solutions thereof for obtaining an optimal global fit.

We consider optimization a function of scalars α, β and γ. These parameters are associated at each surface point with the normal vector $\mathbf{n}(\mathbf{u}|\mathbf{s}_t)$, the maximum principal direction $\mathbf{v}(\mathbf{u}|\mathbf{s}_t)$ and the minimum principal direction $\mathbf{w}(\mathbf{u}|\mathbf{s}_t)$ respectively. These local surface properties allow to steer segmentation in ways that take into account physical object properties.

The suggested movement, or drive $\mathbf{d}(\mathbf{u}|\mathbf{s}_t)$ that works on a surface point $\mathbf{s}_t(\mathbf{u})$ to move it to the newly preferred position, is formulated as a linear combination:

$$\mathbf{d}(\mathbf{u}|\mathbf{s}_t) = \alpha\mathbf{n}(\mathbf{u}|\mathbf{s}_t) + \beta\mathbf{v}(\mathbf{u}|\mathbf{s}_t) + \gamma\mathbf{w}(\mathbf{u}|\mathbf{s}_t). \tag{5.28}$$

This way, movement is restricted to well-defined directions, permitting search spaces of different dimensions for different types of points as illustrated in figure 5.4. Moreover, the magnitude of the movement can be easily controlled by predefining the range of acceptable values for the optimization parameters.

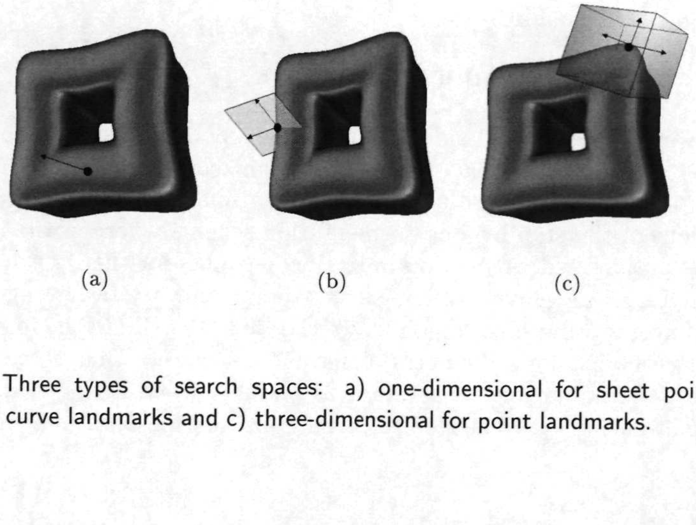

(a) (b) (c)

Figure 5.4: Three types of search spaces: a) one-dimensional for sheet points, b) two-dimensional for curve landmarks and c) three-dimensional for point landmarks.

Fitting the deformable surface to the image data is performed in four steps, following the priority scheme discussed in section 5.2.3. In the first step deformable surface point landmarks are fitted to corresponding image point landmarks. The fit is performed in three dimensions where the search space is naturally spanned by the normal vector $\mathbf{n}(\mathbf{u}|\mathbf{s}_t)$ and the principal curvature directions $\mathbf{v}(\mathbf{u}|\mathbf{s}_t)$ and $\mathbf{w}(\mathbf{u}|\mathbf{s}_t)$. Accordingly, optimization reduces to finding the optimal values for α, β and γ, defining the the quantity in equation 5.28.

The user may fit a restricted number of selected point landmarks by pointing and clicking in the image to steer the segmentation. User interaction yields initial values for parameters α, β and γ. That is, interaction with image point $\mathbf{x}_p$ yields parameter values for which holds

$$\mathbf{x}_p = \mathbf{s}_t(\mathbf{u}) + \mathbf{d}(\mathbf{u}|\mathbf{s}_t), \mathbf{u} \in \mathcal{U}_A. \tag{5.29}$$

The solution obtained by solving this linear system of equations forms the departing point for the computational method. The optimal values for α, β and γ are automatically searched for in the vicinity of the point of interaction.

In the next step, curve landmarks are fitted to the image data departing from the previous result. Optimization is done away from point landmarks, i.e. first curve landmarks close to point landmarks are optimized, then curve landmarks at greater distances. This way we further exploit earlier found solutions. The location of curve landmarks in the image is searched for in a two-dimensional space spanned by the normal vector $\mathbf{n}(\mathbf{u}|\mathbf{s}_t)$ and the maximum principal curvature $\mathbf{v}(\mathbf{u}|\mathbf{s}_t)$. Optimization corresponds to finding the optimal values for α and β, while γ is set to zero. Consequently, the driving force working on curve landmark $\mathbf{s}_t(\mathbf{u}), \mathbf{u} \in \mathcal{U}_B$ is fixed by

$$\mathbf{d}(\mathbf{u}|\mathbf{s}_t) = \alpha\mathbf{n}(\mathbf{u}|\mathbf{s}_t) + \beta\mathbf{v}(\mathbf{u}|\mathbf{s}_t). \tag{5.30}$$

Sheet points are then fitted to the image data to find all remaining boundary points. First those sheet points are optimized that are close to curve landmarks than sheet points at greater distance for the same reason as mentioned above. The location of sheet points in the image is obtained by fitting surface points $\mathbf{s}_t(\mathbf{u}), \mathbf{u} \in \mathcal{U}_C$ to the image data in the normal direction only:

$$\mathbf{d}(\mathbf{u}|\mathbf{s}_t) = \alpha\mathbf{n}(\mathbf{u}|\mathbf{s}_t). \tag{5.31}$$

So far point landmarks, curve landmarks and sheets points have been optimized separately. In the final step all surface points are optimized once again in their respective dimensions to obtain a global solution and to fine tune results. This is especially important in light of the step-by-step segmentation scheme where errors made in early steps may be propagated. A final optimization accounts for this.

Furthermore, as movement of one surface point almost always suggest a similar movement of neighboring surface points, we choose to distribute the force working on a single surface point along the entire deformable surface. That is, we preserve the shape of the deformable surface as much as possible when fitting a specific surface point by simultaneously estimating the correct position for deformable surface points

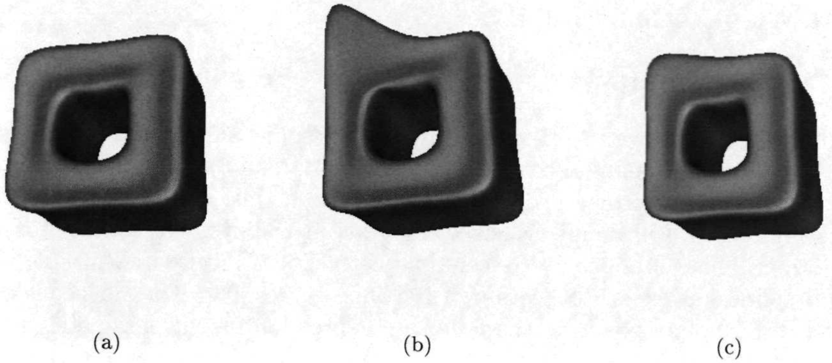


Figure 5.5: The left upper point-landmark is repositioned causing deformation of the entire surface: a) initial surface, b) deformation with a low distribution constant, c) deformation with a large distribution constant.

that have not yet been optimized. This means that, given drive $\mathbf{d}(\mathbf{u}_i|\mathbf{s}_t)$ working on surface point $\mathbf{s}_t(\mathbf{u}_i)$, the following movement of $\mathbf{s}_t(\mathbf{u}_j)$, $\forall \mathbf{u}_j \in U$ is performed to obtain the next preferred position $\mathbf{s}_{t+1}(\mathbf{u}_j)$

$$\mathbf{s}_{t+1}(\mathbf{u}_j) = \mathbf{s}_t(\mathbf{u}_j) + \mathbf{d}(\mathbf{u}_i|\mathbf{s}_t)e^{-(\delta/c_d)} \tag{5.32}$$

where $\delta = D(\mathbf{s}_t(\mathbf{u}_i), \mathbf{s}_t(\mathbf{u}_j))$ denotes the Euclidean distance between surface points $\mathbf{s}_t(\mathbf{u}_i)$ and $\mathbf{s}_t(\mathbf{u}_j)$. The constant $c_d > 0$ is a predefined value controlling the magnitude of the distribution. A small value for the distribution constant influences the shape of the surface in the immediate neighborhood of the point under optimization, while a large value also effects the shape of the deformable surface at large distances. This is illustrated in figure 5.5.

Initialization

The inherent difficulty in deformable model methods is that searching for a minimum over a non-convex function is possible only under predefined conditions that lead to the desired solution [18]. As the objective function is typically non-convex, it might have many local minima. As a consequence the use of a local optimization method will usually find only local minima, and therefore it becomes necessary to place the initial surface close to the true boundary to obtain acceptable results.

We allow the user to specify an initial guess that is close to a local minimum. The initial surface for the object at hand is placed in the image and aligned with the sought object boundary by transformation. The user translates, rotates and scales the a priori deformable surface until an acceptable first guess is-obtained. Starting from the user initialization the optimization method refines the initial guess to fit to the image data. In this context, the purpose of initialization here is to bootstrap optimization.

5.3.4 Implementation

In our implementation we use B-spline surfaces. B-spline surfaces have been chosen because they allow for analytic computation ([93], [41]), and hence an easy derivation of local second order surface properties. Apart from this, the influence of control points is limited to well defined surface intervals providing local control to surface manipulation. This simplifies user-steered adjustments of locally undesirable results.

The B-spline surface is a collection of B-Spline curves [93]. The surface is defined as the set of all points given by the following expression for all parameter values of $\mathbf{u} = [u_1, u_2]^T$

$$\mathbf{s}(\mathbf{u}) = \sum_{j=1}^{J} \sum_{l=1}^{L} B_p^j(u_1) B_q^l(u_2) \mathbf{b}^{j,l} \tag{5.33}$$

where $\mathbf{b}^{j,l}$ is the array of J x L control points. The $B_p^j(u_1)$ are B-spline basis functions of degree $p-1$ in u_1 direction, which are $p-2$ times continuously differentiable. The $B_q^l(u_2)$ are the basis functions of degree $q-1$ in u_2, which are $q-2$ times continuously differentiable. A set of knots in a path parameter interval relating to the control points is used to define the basis functions. For an analytic expression of B-splines basis functions see [93].

For the calculation of new shapes during the optimization process we need the points $\mathbf{p}^{\jmath,\ell}, \jmath = 0, ..., J, \ell = 0, .., L$, which are the interpolation points of $\mathbf{s}(\mathbf{u})$ at $\mathbf{u} = [u_1^\jmath, u_2^\ell]^T$. They are defined as

$$\mathbf{p}^{\jmath,\ell} = \mathbf{s}(u_1^\jmath, u_2^\ell) = \sum_{j=1}^{J} \sum_{l=1}^{L} B_p^j(u_1^\jmath) B_q^l(u_2^\ell) \mathbf{b}^{j,l}. \tag{5.34}$$

Powell's method is used in combination with Brent's line minimization method [13] for calculating new suggested positions for the control points. This variant of coordinate descent optimization minimizes each parameter in turn using Brents line search minimization. The method cycles repeatedly through all parameters. It discards one coordinate dimension in favor of another one if this is a better candidate. The process is repeated after each cycle of minimization, until a stable solution is found for the objective function. Although being slower than gradient-descent approaches, Powell's method has the distinct benefit that no derivatives need to be known for the objective function.

5.4 Experiments and Results

The application to evaluate the proposed necklace method deals with 20 CT images of vertebrae. The CT images are acquired on a Philips SR 700 CT at 140 KV (Philips

Medical Systems, Best, The Netherlands). They were scanned at a resolution of approximately 225 × 225 $\mathcal{W}m$ and slice thickness of 0.5 mm. The images concern the lumbar part of the human spine, exhibiting considerable variation in shape and gray-level appearance even among normal patients.

Four steps are involved in the evaluation procedure: (1) manually outlining and aligning vertebral boundaries for composing the training set, (2) statistically analyzing features in the training set for construction of the vertebra model, (3) segmenting vertebral boundaries in new unknown images, and (4) comparing segmentation results with the known solutions. These steps are performed in two distinct phases: learning and segmentation.

In the learning phase the medical expert provides ground-truth segmentations by indicating three-dimensional points in three two-dimensional orthogonal slices of the volume data. This is done only one time and within one day. A total of 144 points are positioned along a single vertebra boundary, forming the interpolation points for B-spline surfaces. The resulting B-spline surfaces and corresponding images are added to the training set. Subsequently, image and shape features are taken at 400 samples along each surface in the training set, then statistically analyzed.

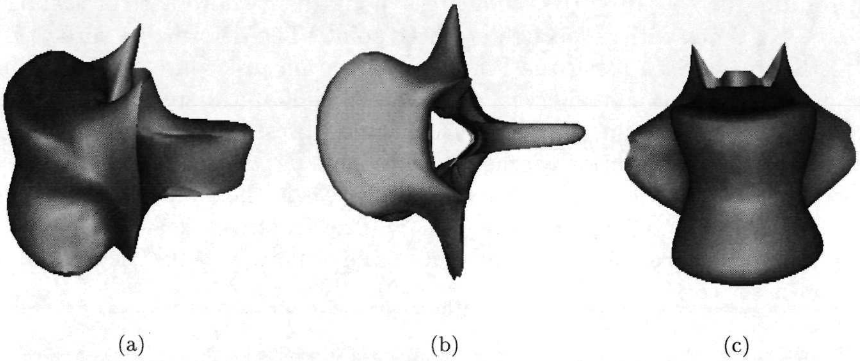


Figure 5.6: The population average vertebra surface $\mathbf{s}_{t=0}(\mathbf{u})$: a) side view b) upper view and c) frontal view. Its shape is controlled by 144 interpolation points.

The testing phase involves segmentation of the vertebra images with help of the necklace model. The segmentations are then compared with the ground truth delineations. This is done using a cross-validation technique [66] as splitting the limited amount of 20 vertebra images into separate fixed learning and testing categories would result in an insufficient number of test and learning cases. Figure 5.6 shows the population average shape of the vertebrae in our training set.

5.4.1 Experiments

The experiments investigate the benefit of landmark-based segmentation with help of necklaces over segmentation under uniform and smooth boundary assumption. We concentrate on landmarks that emanate from the curvature of the vertebra surface, i.e. landmarks are defined only on the basis of high value positions in $\bar{f}^1(\mathbf{u})$ and $\bar{f}^2(\mathbf{u})$ under the assumption that in this application saliency is predominantly due to vertebra shape. Apart from this, feature selection is done according to the type of surface point: at point landmarks features $f^3(\mathbf{u})$ and $f^6(\mathbf{u})$ are evaluated, at curve landmarks features $f^3(\mathbf{u})$ and $f^5(\mathbf{u})$ and at sheet points $f^3(\mathbf{u})$ and $f^4(\mathbf{u})$. This way we always take into consideration shape features while exploiting saliency in the image data.

To compare landmark-based segmentation using necklaces with conventional segmentation we also study segmentation under uniform boundary assumption. To this end we reduce the necklace method to one without landmarks *a)* by learning and applying $f^3(\mathbf{u})$ and $f^4(\mathbf{u})$ along the entire boundary and *b)* by optimizing each point of the the deformable surface in three dimensions, regardless of the dimensionality of the features at the surface points.

We perform three segmentation experiments both for landmark-based segmentation and the conventional way of segmentation. The ground-truth shape model is transformed to the initial deformable surface by translating it with $-15 \leq \chi \leq 15$ voxels, rotating it by $-10 \leq \psi \leq 10$ degrees along the z coordinate or scaling it by $0.9 \leq \sigma_t \leq 1.1$ times with respect to a center point. The deformable surface is then fitted to the image data using the Powell optimization procedure, with equilibrium defined as a 10 percent or smaller change in maximal 50 iterations for line search and a maximum of 50 calls to line search. The curvature threshold is fixed at $c_s = 0.03$. In the first three steps of the priority scheme the weights v_A, v_B and v_C are either turned on or off depending on the segmentation aim, whereas in the global optimization step they are set to: $v_A = 0.5, v_B = 0.3$ and $v_C = 0.2$. The image features are obtained using differentiation scale $\sigma_d = 3.0$ and smoothing scale $\sigma_s = 4.0$. The normalization constant c_t is set to 100.

5.4.2 Results

Evaluating the spatial accuracy of the necklace method amounts to comparing the extracted vertebral segments with ground truth, which is not predetermined, but rather is a function of the test data and the accuracy of the human expert. The distance between the segmented surface and the target surface is measured using the root squared metric error, which computes the average Euclidean distance between points in the proposed solution and points in the ground truth. The Euclidean distance is computed using a discrete error formula:

$$\epsilon_d = \min_{0 \leq \delta < L} \frac{1}{L} \sum_{l=0}^{L-1} ||\mathbf{s}_m(\mathbf{u}_l) - \mathbf{s}_f(\mathbf{u}_l + \delta)||. \tag{5.35}$$

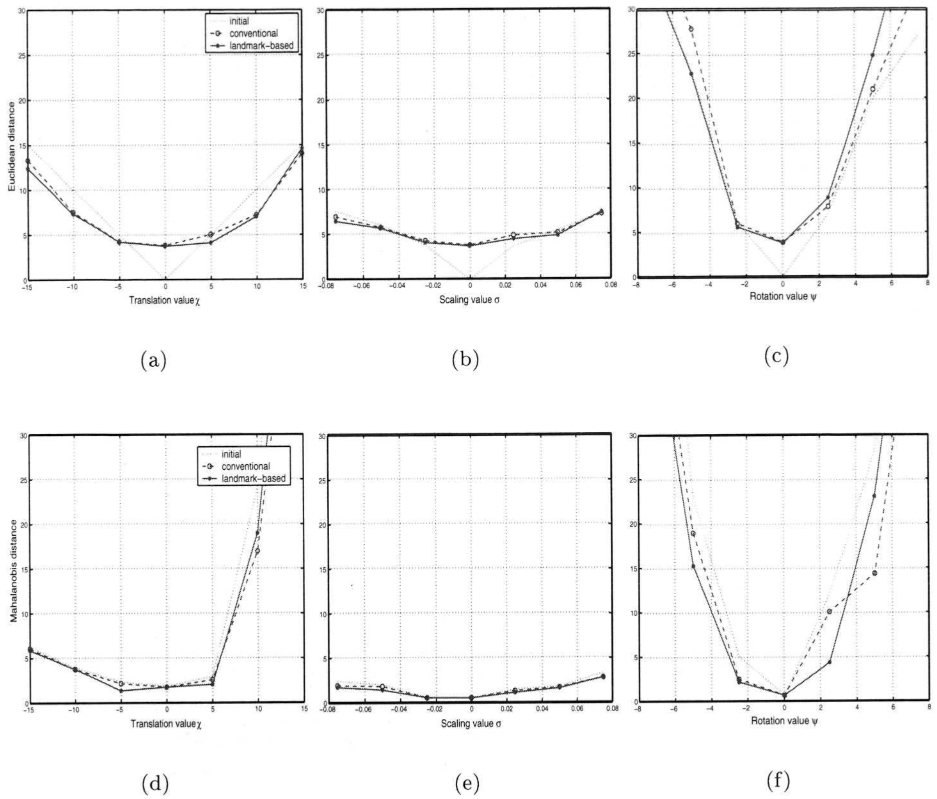
where $\mathbf{s}_m(\mathbf{u})$ is the ground-truth solution and $\mathbf{s}_f(\mathbf{u})$ is a point on the optimized deformable surface. This error metric is used commonly for measuring the accuracy of a point sample distribution.

The first row in figure 5.7 shows the average Euclidean distance between the segmentation result (with and without landmarks) and the ground-truth solution for different transformations. The flatness of the graphs is an indication of the capture range of the deformable surface. It can be seen that even small rotations negatively influence segmentation, while translations and scaling up to +/-5 pixels and scaling with +/- 0.04 produce acceptable results with Euclidean distance less than 5. In most cases landmark-based segmentation achieves a higher reduction in Euclidean distance, i.e. the result is closer to the true solution. Sometimes the result is worse. We attribute this to the fact that when point landmarks are localized and fixed wrongly, e.g. because the sought image landmark point is out of the capture range of the deformable surface, it has a considerable negative affect on the subsequent localization of other points. We conclude that landmark-based segmentation using necklaces outperforms conventional segmentation when point landmarks are localized well.

The second row in figure 5.7 indicates how the average Euclidean distance corresponds to the average Mahalanobis distance of the segmentation result to the population average. It can be seen that the Mahalanobis distance after optimization is lower then at the beginning, meaning that the final deformed surface has feature values closer to population average than the initial deformable surface. Landmark-based segmentation obtains a larger reduction in Mahalanobis distance when starting from a scaled version of the ground-truth shape. For translation and rotation landmark-based segmentation yields higher reduction in some cases, and lower in other ones. Note that for the vertebra application the objective function is much more sensitive for translation and rotation than for scaling, presumably due to the dense context of the segmentation scene with ribs and other organs easily disturbing the fit of the vertebra model

From figure 5.7 we observe that even when the initial deformable surface is the ground-truth, the optimization brings the deformable surface to rest at an average of a little under 4 voxel distance. The minimal voxel distance of 4 is largely due to variation in assignment of ground-truth by the expert. For our stochastic model such a variation is the lower bound in accuracy of what the algorithm is capable of finding back. In a four time repetition of the ground-truth assignment under similar conditions we found a variation of 4.4, 4.9 and 6.7 voxels for point-landmarks, curve landmarks and sheet points respectively. We expect an improvement of performance proportional to the accuracy of the ground-truth segmentation, either by more precise individual assignment or by using larger amounts of salient points per vertebra.

Furthermore, we partially ascribe the minimal voxel distance of 4 to fact that the model was built from different types of lumbar vertebrae (L1, L2, L3, L4, and L5), which may have led to a population average that does not sufficiently resemble any of the segmented vertebra. This causes the deformable surface sometimes to shift away from its correct position in order to comply to the population average. We note that it is essential to use as much invariant features as possible to reduce the effect



(a) (b) (c)

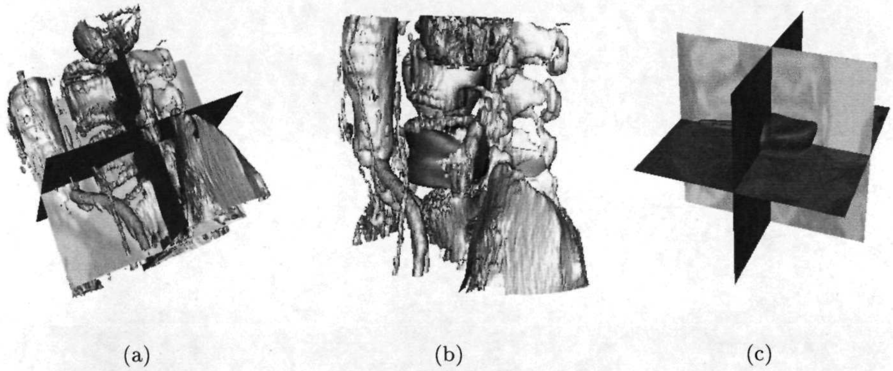
(d) (e) (f)

Figure 5.7: Comparison of segmentation results with and without application of landmarks. Top row shows the average Euclidean distance of results to ground-truth shape for a) translation, b) scaling, c) rotation. Bottom row shows the average Mahalanobis distance of result to population average for : d) translation, e) scaling, f) rotation

of the population average features to the accuracy of the segmentation result, e.g. considering image features after histogram equalization.

To obtain an impression of the computational gains of our algorithm, we also observe how fast the initial model converges to a stable solution. We perform 10 segmentations where we observe the number of calls to the objective function, the reduction in Mahalanobis distance and the reduction in Euclidean distance. For landmark-based segmentation a total of 4104 calls are required on average, taking on average a little over 1 minute to reach equilibrium. The average reduction in Mahalanobis distance is 0.7, corresponding with an average reduction in Euclidean distance of approximately 1.4. Conventional segmentation requires on average 9055 calls, with an average reduction in Euclidean distance of 0.43 and in Euclidean distance of 1.1.

We attribute the computational efficiency to the reduction of search space of solutions for curve landmarks (2D) and sheet points (1D). The fact that landmark solutions are propagated to estimate adjacent boundary points on the same and other vertebrae, also contributes to an increase in computational efficiency.



(a) (b) (c)

Figure 5.8: Three views of the segmentation scene: a) volume rendering of the 3D data block, b) same with initial model superimposed, c) same with different orthogonal intensity planes.

We give a segmentation illustration of the fourth lumbar vertebra. Segmentation departs from the population average vertebra shape in contrast to the above experiments which have been conducted starting from the ground-truth shape. To get an impression of the tree-dimensional context of the image data and the deformable surface in it, figure 5.8 gives three different views of the scene. Figure 5.8a shows a volume rendering of the image data on the basis of a threshold for gray-value intensity, providing a visual means of comparing results from the deformable surface approach with those based on thresholding. The three-orthogonal slices through the image data show intensity values of the original image. Figure 5.8b shows how the deformable surface and the volume rendering relate by superimposing the surface on the rendering. Figure 5.8c shows another view of the deformable model. As the planes cut the deformable surface, only parts of it in front of the plane are visible.

The top row in figure 5.9 illustrates the initial deformable surface from three different perspectives (see figure 5.8 for context). The bottom row illustrates the segmentation result. The images show how the deformable surface correctly moves to the upper and lower surface of the vertebral body. At some parts the model moves only slightly towards the target boundary or even moves away from it to comply to the population statistics.

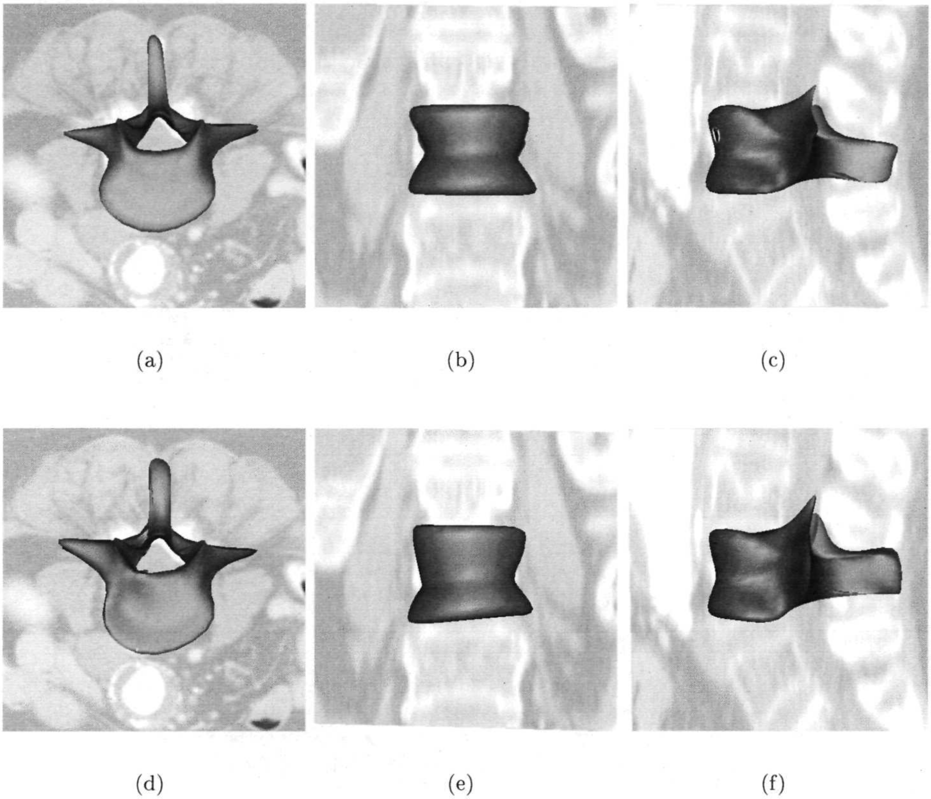


Figure 5.9: An example segmentation with initial and final deformable surface superimposed on the intensity slices through the volume data. Top row shows the initial situation: a) upper view, b) frontal view, c) side view.. Bottom row shows result after deformation: d) upper view, e) frontal view, f) side view.

5.5 Discussion and Conclusion

In conclusion, in this work we present an inhomogeneous and point-enhanced deformable model called a necklace. The necklace aims at exploiting salient information in images containing sophisticated objects and constellations. Saliency is determined on the basis of multiple features observed along object boundaries in a given training set. Multiple features from multiple objects, forming a collection of manifolds in multi-dimensional feature space, are averaged and inspected for landmarks on the basis of the second order structure of the population average. A distinction is made between point landmarks, curve landmarks and sheets points depending on the number dimensions with extreme values for the curvature of the manifolds. Along the entire manifold features are selected that give saliency. They are subsequently employed in a priority scheme to search for boundaries in new unknown images. Point landmarks are detected in three-dimensional image areas, plausibly bootstrapped by interaction. Starting from detected point-landmarks, curve and sheet landmark are detected in respectively two-dimensional and one-dimensional areas. A final detection of all points provides a global segmentation solution.

The method has in common with other landmark-based methods such as [74] and [29] that it exploits salient image and shape information for segmentation. However, in contrast to mainstream landmark-based segmentation, we focus not on a single basis for a landmark but rather try to learn landmarks emanating from multiple features. What makes our approach different from methods with the same aim such as [99] is that we cast more than just one features in a multivariate functional feature space to identify a repertoire of landmarks automatically. Apart from this, feature selection is carried out to exploit the most discriminative feature at each point along the object boundary. This way the uniqueness of features is exploited as much as possible along the entire boundary. Furthermore, to make segmentation less distracted and more efficient the search for boundaries is performed in relevant dimensions only, specified by the directions in which points are uniquely defined.

The necklace method is suited for interactive segmentation. Point landmarks are sparse and have a unique zero-dimensional character forming excellent conditions for steering purposes in three dimensions. They permit the user to conveniently perform segmentation by easily making a one-to-one correspondence between point landmarks on the deformable surface and point landmarks in the image, enhancing indigenous integration of interaction and computation to avoid graphically editing solutions in a post mortem sense. Where the experiment results suggest that a better localization of point-landmarks improves the performance of the necklace method it may give the operator a natural place in accrediting the whole segmentation process.

In the experiments described in the previous section we determine landmarks on the basis of a single feature, the curvature, and apply two features for fitting the deformable surface to the image data. We did not investigate how the necklaces behaves with landmark selection from a large number of features. It is straightforward to extend equation 5.23 to the case where more features are selected at each point. We expect that more accurate results are obtained by *a)* tuning v_A, v_B, v_C locally such that features along the entire surface contribute to the definition of the object at all steps of the priority scheme and *b)* re-estimating the position of deformable surface points at all steps of the priority scheme rather than fixing them. In the definition of features a number of other issues remain unexplored when the sign of the principal curvatures is included in the definition of landmarks. Also the influence of the path parameterization, the number and distribution of the sample points over the surface requires more study. We leave these aside for future work.

We arrive at the conclusion that the necklace method, in particularly when used in combination with user interaction, may help segmentation of complex scenes where a multi-dimensional feature set is needed to properly capture a boundary locally. Necklaces work well when object segmentation using one or two straightforward features fails due to lack of a simple definition of the object boundary.

Chapter 6

Three-dimensional Segmentation of Spinal Images Using an Integral Deformable Spine Model*

Segmentation of the spinal column directly from three-dimensional image data is desirable to accurately capture its morphological properties. We describe a method that allows true three-dimensional spinal image segmentation using a deformable integral spine model. The appearance of vertebrae is learned from multiple continuous features recorded along vertebra boundaries in a given training set of images. Important summary statistics are encoded into a necklace model on which landmarks are differentiated on their free dimensions. The landmarks are used within a priority segmentation scheme to reduce the complextity of the segmentation problem. Necklace models are coupled by string models. The string models describe in detail the natural variability in appearance of spinal curvatures from multiple continuous features recorded in the training set. Strings are used for restricting segmentation within feasible solutions. In the segmentation phase, the necklace and string models are used to find the spinal column in unknown image data via elastic deformations. The driving application in this work is analysis of CT scans of the human lumbar spine. A segmentation illustration shows the method is promising for assessing morphological properties of the spinal column.

*Submitted to IEEE Transactions on Engineering in Biomedicine

6.1 Introduction

Segmentation of three-dimensional spinal images is an essential task for analyzing the morphology of the spinal column. This is difficult to achieve fully automatically with current computer vision methods due to the articulated structure of vertebrae and their dense context with ribs and other organs. Vertebrae exhibit many predicaments, violating the smoothness assumption under which many segmentation methods operate. Apart from this their image appearance is mostly far from evident. Insufficient image contrast, interfering anatomical structures, and other local irregularities lead to an inhomogeneous gray-level appearance that is hard to capture when assuming the image structure is homogeneous. Spinal image segmentation may encounter serious problems when it does not appraise the sophisticated image and shape appearance of the spinal column. It is therefore desirable to construct a segmentation model that exploits vertebral inhomogeneities rather than being hampered by them.

Commonly, segmentation of spinal images is done on the basis of geometrical models (e.g. [62], [124], [116], [49], [60]). The models capture the shape of vertebrae. They often also utilize their spatial inter-relationships, reducing image segmentation to fitting geometrical and spatial models to highlighted edges in the image data. There are two important shortcomings to this approach. Geometrical models often lack expressiveness to capture the full range of feasible vertebral shapes and constellations that can be expected in an image. And, geometrical models are targeted at exploiting a priori shape and spatial information, while in spinal images the spinal column is not only defined by these properties but also by its image structure. There is a need to also model image structure [96]. In order to construct apt segmentation models for the spinal column it is natural to observe multiple spinal properties and to address their natural variability.

The lack of segmentation models that exploit in full the many features defining the spinal column has incited us to tackle the segmentation problem by learning from example images. When subjected to statistical analysis a set of example images might reveal new and relevant features, conceivably enriching segmentation models. That is, if a good description of the spinal column is not possible from a priori geometrical knowledge, it may be learned from multiple features observed in a significant set of example images. Learning in this context helps focusing on the most relevant information in a potentially overwhelming quantity of image data full of boundaries of various scales and spatial configurations. Hence, we concentrate on learning multiple features of the spinal column rather than attempting to improve segmentation models by fixing on more a priori geometrical knowledge.

This chapter presents a segmentation method that exploits salient and variational information deduced from multiple continuous features as observed in a given training set of spinal images, under the assumption that the appearance of vertebral structures can be well captured in a statistical sense [2], [78], [7]. It is organized as follows. In section 2 related work on segmentation of spinal images is discussed. Section 3 briefly describes the image material used in this work and introduces the proposed method. The following issues are addressed: the necklace model for capturing vertebral structures, the string model for expressing spinal curvatures and the spine model for capturing the entire spinal column by elastic deformation in the image reminiscent to a marionette with interrelated structures moved by strings. In section 4 an illustration is given. Discussion and conclusion follow in section 5.

6.2 Related Work

In literature a number of two- and tree-dimensional model-based approaches are proposed for segmentation of spinal images. The majority exploits vertebral landmarks to reduce the complexity of the segmentation problem.

Kauffmann et al. [62] first detect the axis of the spinal column by manually placing points along it and fitting a curve through them. The fitted curve is used to initialize and rigidly match templates of the vertebral body with the image data to obtain vertebral outlines. Landmarks are extracted from the best fitting contours and subsequently used for three-dimensional reconstruction. Verdonck et al. [124] manually indicate specific landmarks in the image and find others using an interpolation technique. The landmarks, together with a manually indicated axis of the spinal column, are used to automatically compute endplates on vertebrae and the global outline of the spine. A rigid spine model is then fitted to two sets of landmarks, one in frontal and the other in lateral view, to obtain a three-dimensional description. In [2], Aykroyd and Mardia use wavelets to model one-dimensional deformation functions of spinal curvatures from front and side view X-ray images, treating each vertebra as a well defined landmark. They statistically analyze spinal shape and shape differences for the two views.

These methods not always catch the three-dimensional properties of the spine as they rely on reconstruction from two-dimensional images using biplanar or stereographic techniques [116]. They yield three-dimensional descriptions from manually marked landmark points on multiple image views, resulting in inaccurate and irreproducible measurements. Apart from this, the sparse set of landmark points is often connected by straight lines in order to give a wire frame representation [62], yielding non-realistic geometrical shapes and often lacking expression to capture local morphology, such as for vertebral deformities. This and the steady increase of patient-friendly three-dimensional imaging methods have motivated the development of true three-dimensional landmark-based methods.

The approach developed by Dansereau et al. [26] uses a three-dimensional personalized parametric model of the spine. Simple geometric primitives like elliptic cylinders and prism are adapted to fit corresponding landmarks in the image, giving a rough impression of the three-dimensional character of the spinal column. In [92] a model of the cervical spine is used consisting of a finite-element model augmented by additional structures used to locate landmarks, contours, surfaces, and regions. The surfaces and volumes are used by statistical estimation modules to automatically extract landmarks, using the finite-element model as a road map. Once the landmarks are found they are employed to refine the vertebra models. In [74], a point-based statistical model is proposed for segmentation of single vertebrae. Manually selected points associated with anatomical landmarks on the vertebra boundary are used for point-to-point correspondence between model and image data, resulting in a close but not exact fit of the model. A mesh relaxation method moves ordinary model points to the target boundary for an exact match. The point distribution model is easily extended to capture multiple vertebrae but without the capacity to explicitly quantize properties of the entire spinal column.

Three-dimensional methods are a necessity to capture the truly intricate characteristics of the vertebral column. The reported methods are capable of doing that. However, they do this on the basis of a priori geometric models only, while adscititious exploitation of image models would enhance segmentation. Apart from this, most of the three-dimensional methods rely on manual localization of anatomical landmarks, which is subjective and labor intensive due to the huge amount of image data.

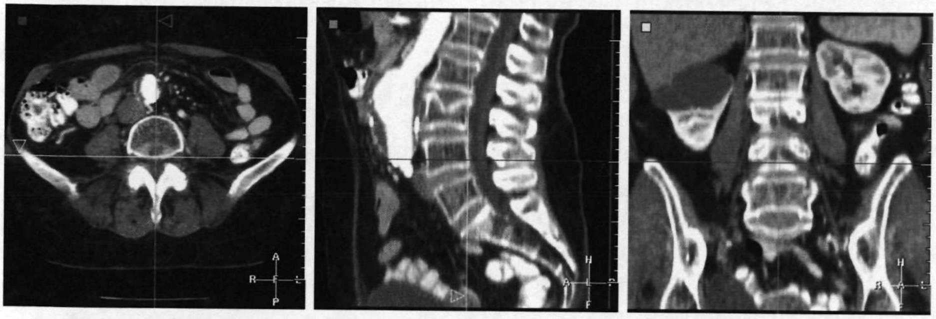
We take a different approach and aim at segmenting the spine by a learned model of continuous image and shape features on which landmarks are automatically localized on their free dimensions and applied accordingly.

6.3 Materials and Method

In this work we make use of an image data set consisting of CT scans of the abdomen part of a group of 18 elderly people. All subjects have been scanned with a Philips SR 700 CT at 140 KV (Philips Medical Systems, Best, The Netherlands) at a resolution of approximately 225 × 225 Wm and slice thickness of 0.5 mm. The images are originally taken for the purpose of investigating the aorta but most of them also display the lumbar part of the spinal column. Concentrating on the lower four lumbar vertebrae (L2, L3, L4, L5), we demonstrate our method on 6 CT images of subjects with minimal spinal and vertebral deformities. In figure 1 transversal, sagittal and coronal slices of an image from the data set are shown, exhibiting normal and abnormal vertebrae.

For segmentation of these images we use deformable model methods (e.g. [15], [115], [79]). First introduced by Kass et al. [61] and Staib et al. [114], the idea behind them is to treat segmentation as an optimization problem, typically by minimizing a model fitting function that rewards locally smooth boundaries that pass through high-gradient image regions. A model is deformed in the image trying to find a compromise between features derived from the image and features obtained from a shape model. The deformation stops when an equilibrium is reached. The deformable model is then assumed to lie on the target boundary in the image.

We adopt the deformable model approach with the difference that we aim at learning vertebral features rather than defining them on the basis of a priori knowledge. In addition, to exploit salient and variational information as observed in a given training set of spinal images we construct *a)* models of the lumbar vertebrae using necklaces, and *b)* models of the spinal curvatures using strings and *c)* an integral spine model using coupled necklace and string models. In the following section we describe the models one by one.



(a) (b) (c)

Figure 6.1: Axial, sagittal and coronal cross sections of a spinal CT image. The images reveal an infrarenal aortic aneurism, renal and oesteoporotic fracture of the plate only obvious on the sagittal reconstruction.

6.3.1 Necklace Model for Vertebral Structures

A visual inspection of the vertebra surface suggests that the surface is bended in ways significantly different from a flat surface, i.e. it has many concave and convex surfaces differing from weakly to strongly curved. Analogous to diversity in shape, the gray-level appearance of the vertebra boundary exhibits many different structures. At some parts the vertebra boundary has well-defined intensity discontinuities, while at other parts there is vague pictorial evidence or none at all due to bad image quality or interfering structures in the neighborhood.

To appropriately capture and exploit the locally sophisticated geometrical and image appearance of the vertebra surface we need to observe multiple features along its boundary. To that end we employ necklaces. As discussed in the previous chapter, the necklace model allows for the analysis of inhomogeneous boundaries by recording a repertoire of image and shape features. Specifically it allows for exploitation of salient features as landmarks for segmentation.

Feature Definition

The appearance of the spinal column is learned from a training set of M examples consisting of three-dimensional spinal images $I_m : \mathbf{x} \in \Re^3 \to \Re, m = 1, ..., M$ and true vertebral outlines as edited in the volume data by a medical expert in three two-dimensional orthogonal slices of the volume data. We represent each vertebral outline $\vartheta = 1, ..., V$ in the M training images by B-spline surfaces $\mathbf{s}_m^\vartheta : \mathbf{u} \in \Re^2 \to \Re^3$ [93]. After manual alignment of the surfaces, image and shape features are extracted at the surface and conveniently captured as a manifold in an N-dimensional feature space, adopting the multi-feature object representation of necklaces. Considering the

ϑth vertebrae the set of multivariate continuous features is deduced from the training data by the mapping $\mathbf{f}^{\vartheta} : \mathbf{u} \in \Re^2 \rightarrow \mathcal{F}^{\vartheta}$

$$\mathbf{F}^{\vartheta}(\mathbf{u}) = [\mathbf{f}_1^{\vartheta}(\mathbf{u}), ..., \mathbf{f}_M^{\vartheta}(\mathbf{u})]. \tag{6.1}$$

Here, we have chosen to compute the location, rotation and scale invariant *mean curvature* as the local shape feature to be recorded along the surface outlines. We also compute the principal curvatures to simplify landmark definition in subsequent steps. The feature values are analytically computed from the B-spline surfaces [93] $\mathbf{s}_m^{\vartheta}(\mathbf{u})$. The feature values are used to construct feature functions $\mathbf{f}_m^{\vartheta}(\mathbf{u})$ by interpolating a smooth surface through them.

We also compute three image features described in the previous chapter. To recapitulate, the first feature has a large value when it concerns a point on a flat structure, e.g. the upper surface of the vertebral body. The second feature has a large value when it concerns a point on a tubular structure, e.g. where the upper and lower surface merge with the anterior surface of the vertebral body. The third feature has a large value when it concerns a point on a tip-like structure, e.g. the tip of the spinal process. The features are extracted from the training images along the training surfaces, i.e. at image positions $I_m(\mathbf{s}_m^{\vartheta}(\mathbf{u}))$

The extraction of $N = 6$ features for each vertebra yields V sets of M surfaces in a 6-dimensional feature space, to be analyzed statistically for model construction.

Landmark Selection

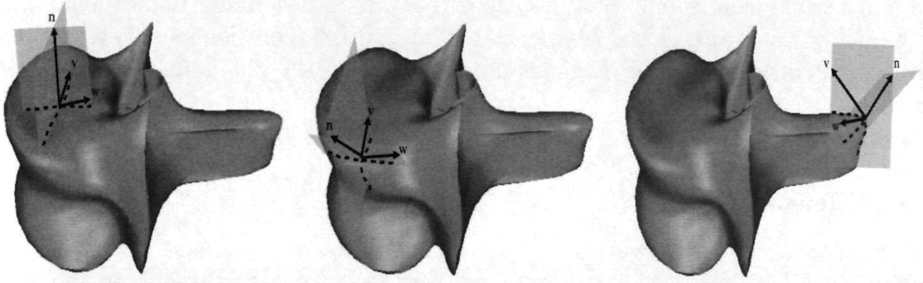
We aim at exploiting landmarks that are defined by the multiple features recorded along the continuous vertebra boundary. Vertebral landmark definition reduces to localizing peaks in feature function values $\mathbf{f}_m^{\vartheta}(\mathbf{u})$. However, rather then separately investigating each training instance m for landmarks, we first compute the statistics of the training sets. Then we try to obtain robust landmarks from the population average function values. After alignment of the feature functions, the population average for vertebra ϑ is computed as

$$\bar{\mathbf{f}}^{\vartheta}(\mathbf{u}) = \frac{1}{M}\sum_{m=1}^{M} \mathbf{f}_m^{\vartheta}(\mathbf{u}|I_m, \mathbf{s}_m). \tag{6.2}$$

This two-dimensional surface in the N-dimensional feature space is obtained by averaging each training surface $\mathbf{f}_m^{\vartheta}(\mathbf{u}|I_m, \mathbf{s}_m)$ in each dimension. The variance in shape feature values is

$$\sigma_{\mathbf{f}^{\vartheta}}(\mathbf{u}) = \left(\frac{1}{M}\sum_{m=1}^{M} ||\mathbf{f}_m^{\vartheta}(\mathbf{u}|I_m, \mathbf{s}_m) - \bar{\mathbf{f}}^{\vartheta}(\mathbf{u})||^2\right)^{1/2}. \tag{6.3}$$

The average feature function $\bar{\mathbf{f}}^{\vartheta}(\mathbf{u})$ is searched for high curvature points on the basis of its local second order properties [65], [117]. These properties are obtained from the infinite set of planes passing through and containing the normal at a specific point on the population average $\bar{\mathbf{f}}^{\vartheta}(\mathbf{u})$. For example, when the only features considered are the x,y and z-coordinates of the vertebral surfaces $\mathbf{s}^{\vartheta}_{m}(\mathbf{u})$, each of the normal planes intersects the surface by a planar curve. The curvature at the point of interest is an identifying curvature measure for the surface. The pair of directions $\mathbf{v}$ and $\mathbf{w}$ are defined such that these curvatures reach their maximum and minimum curvatures κ_1 and κ_2 as illustrated in figure 6.2. We use these principle curvatures and locally associated directions to define landmarks.



(a) (b) (c)

Figure 6.2: Surface point properties are derived from the curves defined by the intersection of the surface with the two orthogonal planes that go through that point and contain the normal vector $\mathbf{n}$. The curves with minimal and maximal principal curvatures in corresponding directions $\mathbf{v}$ and $\mathbf{w}$ define the type point: a) sheet points have low bending and freedom to move in two directions, b) curve landmark have a fixed position in all but one dimension c) point landmark have no free dimensions and are precisely localized.

We make distinction between point landmarks, curve landmarks and sheet points by evaluating the principal curvature values at each point of $\bar{\mathbf{f}}^{\vartheta}(\mathbf{u})$. Point landmarks are surface points $\mathcal{U}_A$ where both principle curvatures have an extreme absolute value. They are precisely localized in three dimensions. Surface points where the absolute value of one of the principal curvatures is extreme, are curve landmarks, denoted by $\mathcal{U}_B$. They are precisely localized in two dimensions. At sheet points $\mathcal{U}_C$ both values of the absolute principle curvature is low. Typically they are well-defined in only one dimension. The sets $\mathcal{U}_A$,$\mathcal{U}_B$,$\mathcal{U}_C$ together contain all relevant path positions. A threshold for the principal curvature values may be chosen such that the definition of these sets, i.e. the distribution of geometrical landmarks, largely coincides with anatomical landmarks.

At this point we have $V = 4$ necklace models: one for each vertebra captured. In the segmentation step, the information contained in $\bar{\mathbf{f}}^{\vartheta}(\mathbf{u})$ and $\sigma_{\mathbf{f}^{\vartheta}}(\mathbf{u})$ is used as a reference model for feature selection and qualification. The sets $\mathcal{U}_A$,$\mathcal{U}_B$,$\mathcal{U}_C$ are used for landmark-based segmentation.

6.3.2 String Model for Spinal Curvatures

At the level of the spinal column we combine each of the vertebra models to use knowledge about the spinal curvatures. The cervical and lumbar curvatures are characterized by a convex shape, while the thoracic and sacral curvatures are characterized a convex forward shape. Knowing that these spinal curvatures are almost always present with some variation across and among subjects, this information can assist image segmentation.

We aim at capturing the spinal curvatures using string models. As discussed in chapter 3, the string model has the capacity to build a detailed underlying statistical model of open and closed boundaries from multiple continuous shape and image features, in contrast to other similar approaches (e.g. [19], [20], [35], [43], [85]). Here, we use strings to capture the common appearance of spinal curvatures in our training data and the main modes of variation therein, in ways similar to [2].

Feature Definition

The shape of the spinal curvatures is learned from the same training set of M examples images and V outlines associated with each vertebra. Assuming the landmarks on the lumbar vertebrae occur at approximately the same position, we select point landmarks $\mathbf{u}_\ell \in \mathcal{U}_A$ for $\ell = 1, ..., L$ and learn the appearance of the L curves that pass through the V surface points $\mathbf{s}_m^1(\mathbf{u}_\ell), ..., \mathbf{s}_m^V(\mathbf{u}_\ell)$. Each curve captures the spatial relation between corresponding point landmarks on adjacent vertebrae, derived automatically from the vertebral outlines. We represent the $\ell$the curve in the mth training image, by the B-spline curve $\mathbf{c}_m^\ell : u \in \Re \to \Re^3$. Multiple continuous image and shape features are extracted along the curves in the training set and captured by space curves in feature space $\mathcal{F}^\ell$, yielding feature function

$$\mathbf{F}^\ell(u) = [\mathbf{f}_1^\ell(u), ..., \mathbf{f}_M^\ell(u)]. \tag{6.4}$$

We observe two features along the curve $\mathbf{c}_m^\ell(u)$. The first feature is the bending of the curve, capturing shape statistics. We have chosen this feature because of its invariance properties and because it might reveal new and interesting anatomical knowledge. For example, the value and location of the maximum curvature along the spinal column is a relevant clinical measure for spinal deformities [2]. The second feature measures the image gradient magnitude, supporting the definition of spinal curvature by means of image evidence, which is mainly confined at tips of the vertebral structures.

The extraction of 2 features for each spinal curvature yields L sets of M curves in a two-dimensional feature space, to be analyzed statistically for model construction.

Variational Information

We aim at statistically modeling the natural variability of the spinal curvatures in terms of shape and gray-level features. This differs from the mainstream methods in that we do not confine the definition of spinal curvatures based on a priori geometrical knowledge such as smoothness on intra- and inter-curve properties of the spinal column [126]. Rather by learning what the common appearance of the spinal curvatures is, the search criteria can be based on natural variations. Our training data can be summarized as

$$\bar{\mathbf{f}}^{\ell}(u) = \frac{1}{M}\sum_{m=1}^{M} \mathbf{f}_m^{\ell}(u|I_m, \mathbf{c}_m^{\ell}). \tag{6.5}$$

The one-dimensional curve $\bar{\mathbf{f}}^{\ell}(u)$ in the multi-dimensional feature space is obtained by averaging each training curve $\mathbf{f}_m^{\ell}(u|I_m, \mathbf{c}_m)$ in each dimension separately. The variance is

$$\boldsymbol{\sigma}_{\mathbf{f}^{\ell}}(u) = \left(\frac{1}{M}\sum_{m=1}^{M} ||\mathbf{f}_m^{\ell}(u|I_m, \mathbf{c}_m^{\ell}) - \bar{\mathbf{f}}^{\ell}(u)||^2\right)^{1/2} \tag{6.6}$$

The population average feature function $\bar{\mathbf{f}}^{\ell}(u)$ and variation $\boldsymbol{\sigma}_{\mathbf{f}^{\ell}}(u)$ contain important evidence in the training data. Variational information is captured in more detail by functional data analysis [95], producing string models of the spinal curvatures. The string models allow to observe multiple features along one-dimensional curves, to weight the features according to the most important natural modes of variation and to explain unknown instances by a statistically determined feature weighting procedure. Strings resemble necklaces in that multiple continuous features are recorded and statistically analyzed, however, the emphasis lies on weighting features according to their natural variations, whereas necklaces mainly focus on selecting features for defining landmarks.

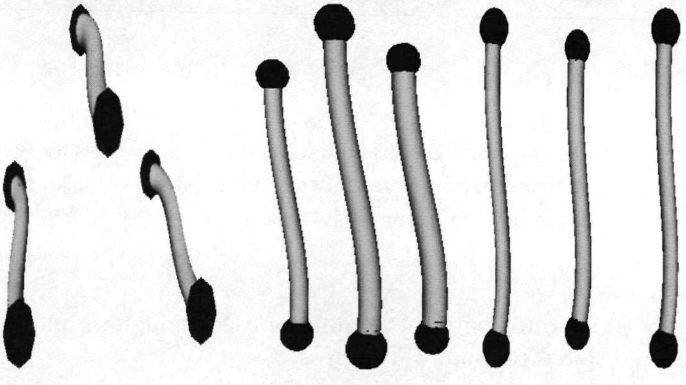


Figure 6.3: Three string models found as an average over 6 normal patients viewed from three different perspectives: a) axial b) sagittal and b) coronal.

At this point we have a detailed statistical description of the spinal curvatures. For simplicity, we assume all the relevant information is contained in $\bar{\mathbf{f}}^{\ell}(u)$, and $\boldsymbol{\sigma}_{\mathbf{f}^{\ell}}(u)$.

We proceed with only these two quantities. Figure 6.3 illustrates the population average shape of the spinal curvatures of the lumbar part of the spinal column.

6.3.3 Integral Model for Spinal Column

To form an integral model of the spinal column we couple the necklace models of vertebrae and the string models of the spinal curvatures. This is accomplished by stacking the V necklace models at positions that are statistically determined from the strings models. This is illustrated in figure 6.4

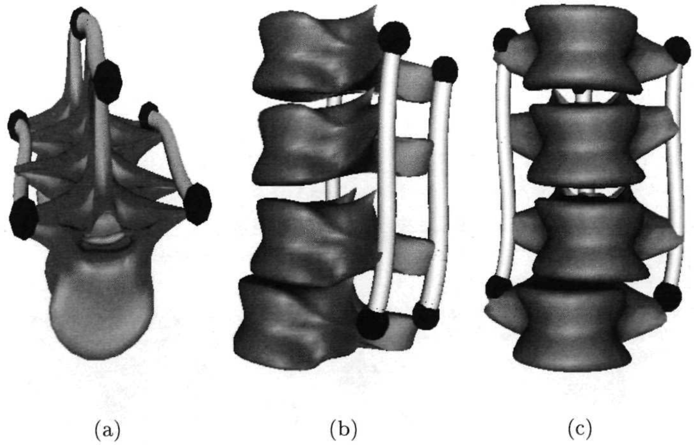


Figure 6.4: The integral spine model consists of multiple necklace models for vertebrae coupled by string models as in a marionette from three different perspectives: a) axial b) sagittal and b) coronal.

The spine model is deformed onto to the data of a new unknown image such that it fits best a recorded spinal column.

Qualification

A *deformable integral spine model* consists of deformable surfaces $\mathbf{s}_t^{\vartheta}(\mathbf{u})$ and deformable curves $\mathbf{c}_t^{\ell}(u)$, which account for variability among vertebral structures and their interrelationships. For all V vertebra models, the initial surface $\mathbf{s}_{t=0}^{\vartheta}(\mathbf{u})$ and curve $\mathbf{c}_{t=0}^{\ell}(u)$ are the population average. That is, assuming models $\mathbf{s}_m^{\vartheta}(\mathbf{u}), \vartheta = 1, ..., V$ and $\mathbf{c}_m^{\ell}(u), \ell = 1, ..., L$ are properly aligned and uniformly parameterized to establish point correspondence they are defined as

$$\mathbf{s}_{t=0}(\mathbf{u}) = \frac{1}{M}\sum_{m=1}^{M} \mathbf{s}_m^{\vartheta}(\mathbf{u}) \tag{6.7}$$

$$\mathbf{c}_{t=0}(u) = \frac{1}{M}\sum_{m=1}^{M} \mathbf{c}_m^{\ell}(u). \tag{6.8}$$

The fit quality is determined on the basis of features $\mathbf{f}_t^{\vartheta}(\mathbf{u})$ and $\mathbf{f}_t^{\ell}(u)$ emanating from $\mathbf{s}_{t=0}^{\vartheta}(\mathbf{u})$ and $\mathbf{c}_{t=0}^{\ell}(u)$ respectively. The model fitting function makes a compromise between the fit quality of the necklace models and the string models:

$$\Theta_{spine} = \frac{1}{V}\sum_{v=1}^{V} \omega^{\vartheta} \cdot \theta^{\vartheta}(\bar{\mathbf{f}}^{\vartheta}, \sigma_{\mathbf{f}^{\vartheta}}, \mathbf{f}_t^{\vartheta}) + \frac{1}{L}\sum_{l=1}^{L} \omega^{\ell} \cdot \theta^{\ell}(\bar{\mathbf{f}}^{\ell}, \sigma_{\mathbf{f}^{\ell}}, \mathbf{f}_t^{\ell}). \tag{6.9}$$

The first term measures the distance between the expected and the sampled values for each vertebra ϑ. To ensure a controllable distance measure, the Mahalanobis distance [34] is computed using mean and variation information obtained from learning. For the ϑth vertebra this means

$$\theta^{\vartheta}(\bar{\mathbf{f}}^{\vartheta}, \sigma_{\mathbf{f}^{\vartheta}}, \mathbf{f}^{\vartheta}) = \int_{\mathbf{u}} \upsilon(\mathbf{u})||\frac{\bar{\mathbf{f}}^{\vartheta}(\mathbf{u}) - \mathbf{f}^{\vartheta}(\mathbf{u})}{\sigma_{\mathbf{f}^{\vartheta}}(\mathbf{u})}||^2 d\mathbf{u}. \tag{6.10}$$

The fit is controlled by means of the function $\upsilon(\mathbf{u})$ which weights the fit at each point of the deformable surface $\mathbf{s}_t^{\vartheta}(\mathbf{u})$. Weighting is done according to the type of surface point under consideration: for point landmarks there is a predefined weight υ_A, for curve landmarks υ_B and for sheet points υ_C. For example, the search for point landmarks is performed using settings $\upsilon_A = 1, \upsilon_B = 0, \upsilon_C = 0$. The weights may also be set such that features along the entire surface contribute by setting all values larger than 0, but constrained to add up to 1.

The second component in equation 6.9 measures the distance between the expected and the sampled values for each string model. This way it seeks at all times resemblance between the reference spinal curvature $\mathbf{f}^{\ell}$ and the deformed curve $\mathbf{f}_t^{\ell}$ normalized by the common modes of variation. For the ℓth string model the fit quality is formulated as

$$\theta^{\ell}(\bar{\mathbf{f}}^{\ell}, \sigma_{\mathbf{f}^{\ell}}, \mathbf{f}_t^{\ell}) = \int_{u} ||\frac{\bar{\mathbf{f}}^{\ell}(u) - \mathbf{f}^{\ell}(u)}{\sigma_{\mathbf{f}^{\ell}}(u)}||^2 du \tag{6.11}$$

The model fitting function is regulated by means of weights ω^{ϑ} and ω^{ℓ}, which are positive and add up to 1. Their value is defined by the user and generally tuned such that they emphasis either the fit of the necklace models or the fit of the strings models. The fit quality forms the basis for deformation.

Optimization

Having specified the model fitting function, we must choose how to optimize the degrees of freedom of the spine model. Optimization involves two main steps for each point on the deformable necklace model. In the first step a newly suggested position is calculated based on the fit, followed by deformation of the necklace model to move the point into the newly preferred position. Optimization only affects the geometry of the deformable surfaces: in our implementation the curves constituting the string models are automatically derived from them. The aim is to find the optimal deformable surfaces for all $\mathbf{s}_t^{\vartheta}(\mathbf{u}), v = 1, ..., V$, such that

$$\Theta_{spine}^{optimal} = \underset{\mathbf{s}_t^{\vartheta}}{\operatorname{argmin}} \quad \Theta_{spine}. \tag{6.12}$$

As the deformable surfaces know the type of point they search for, movement of surface points can be explicitly made dependent on that type, contradicting the usual way of repositioning all points in a three-dimensional space regardless of their dimensionality. Sheet points, curve landmarks and point landmarks are searched for in one, two and three dimensions respectively (see figure 6.5). This allows a better control of the segmentation as movement of surface points is restricted to well-defined directions.

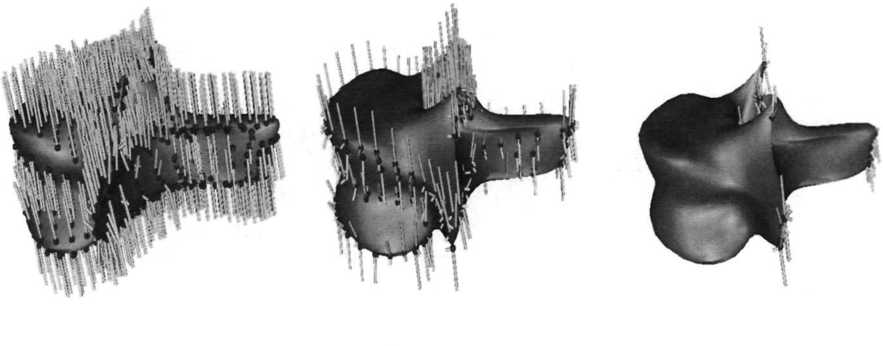


Figure 6.5: For each type of surface point the search area is specified in terms of local surface properties: a) sheet points are optimized in the normal direction (small stripes indicate the direction at each sheet point) perpendicular to the surface, b) curve landmarks in a two-dimensional area spanned by the normal and first principal direction and c) point-landmarks in a three-dimensional area specified by the normal and principal directions.

A priority scheme is used when optimizing the deformable surface $\mathbf{s}_t^{\vartheta}(\mathbf{u})$. For each deformation of a surface onto the image data the following scheme is employed:

1. Optimize point landmarks of the vertebra in a three-dimensional area. The result is a rough estimate of the position of the vertebra boundary by its point landmarks.

2. Optimize curve landmarks in a two-dimensional area departing from 1). The result is the location of surface curve points determining the outline of the vertebrae.

3. Optimize sheet points in a one dimensional area departing from 2). The result is the location of all boundary points.

4. Optimize all points one more time in their respective dimensions departing from 3), to fine tune the result and to obtain a global optimum.

In searching for a specific vertebra we also estimate the position of other vertebrae. We do the estimation only when searching for point landmarks so that movement of one point landmark affects the entire spine model. We accomplish this by distributing the force that works on a single point to all other points, in this case, weighted according to distance. For example, if there is a drive $\mathbf{d}(\mathbf{u}_i)$ working on surface point $\mathbf{s}_t^{\vartheta}(\mathbf{u}_i), \mathbf{u}_i \in \mathcal{U}_A$, which is also the connection point $\mathbf{c}_t^{\ell}(u_i)$ for a string model, this yields the following estimation for the discrete points $\mathbf{s}_t^{\vartheta}(\mathbf{u}_j), j = 1, ..., J$ on all vertebra models $\vartheta = 1...V$

$$\mathbf{s}_{t+1}^{\vartheta}(\mathbf{u}_j) = \mathbf{s}_t^{\vartheta}(\mathbf{u}_j) + \mathbf{d}(\mathbf{u}_i)e^{-(\delta_C/c_d)}. \tag{6.13}$$

The distance $\delta_C = D(\mathbf{c}_t^{\ell}(u_i), \mathbf{c}_t^{\ell}(u_j))$ between points $\mathbf{c}_t^{\ell}(u_i)$ and $\mathbf{c}_t^{\ell}(u_j)$ is used to determine the extent of the distribution. A small value for the distribution constant c_d influences the shape of the deformable necklace model above or beneath the one being optimized, while a large value also effects the shape of the surfaces at large distances. This way segmentation of a single vertebral structure influences the entire spine model.

6.4 Illustration

We give an illustrations of segmentation of vertebral structures from CT images with help of the deformable integral spine model. First the 4th lumbar vertebra is segmented using a necklace model, then part of the lumbar spine is segmented with help of the spine model.

For construction of the vertebra surfaces, a total of 144 points are used, with articulated structures such as the spinal process and transverse processes requiring more points than flat parts such as the vertebral body. These points are interpolated by a B-spline surface to obtain a continuous representation. For each vertebra surface in each training instance, image and shape features are recorded at 400 sample points. Feature functions are constructed from them and statistically analyzed to build the necklace models according to equations 6.1-6.3. Landmarks on the learned necklace models are defined on the basis of of curvature threshold $c_t = 0.05$. The following point landmarks are selected to form string models according to equations 6.4-6.6: two points corresponding to the two tips on the spinal process, and the two tips at the two transverse processes. These six points are also used for human computer interaction.

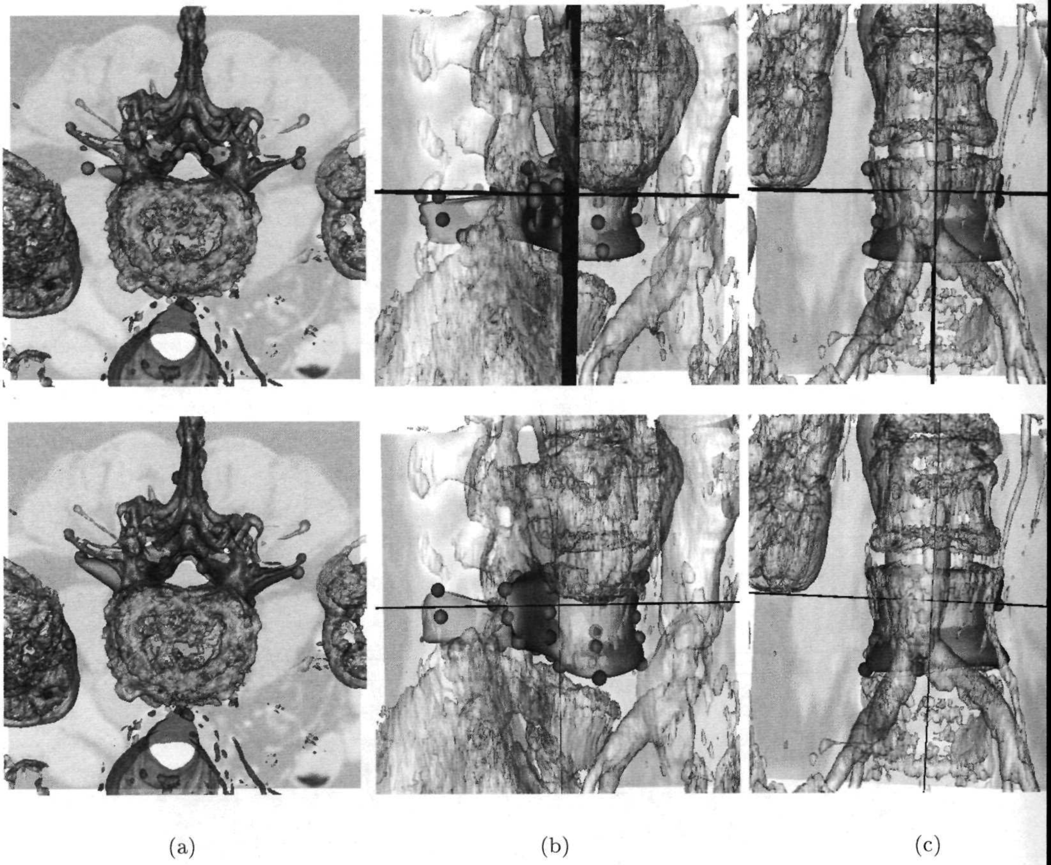
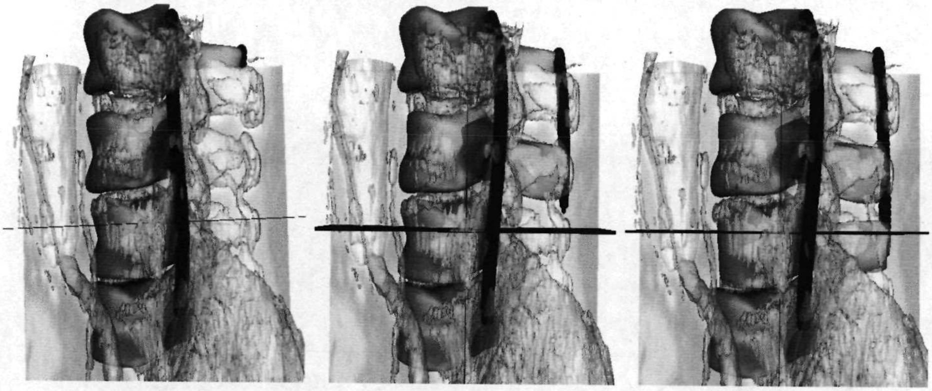


Figure 6.6: The segmentation scene for three different perspectives. The image data is rendered transparently and the deformable model in it opaquely. First row: the initial necklace model. Second row: the deformation result.

For initialization of the spine model in the image data, a fixed point is selected to enable quick correspondence between the model and the target boundary. The deformable spine model is interactively bootstrapped by pointing and clicking at the corresponding point in the image. In this segmentation session no translation, rotation or scaling is required as the first guess is acceptable. The first row in figure 6.6 shows the condition after initialization from three different perspectives, with the image data rendered with opacity 0.5 and the model in it with opacity 1. The local fit quality at a number of control points is indicated with colored spheres. The color varies from green, indicating a good fit to red, expressing a bad fit.

Following the priority scheme described in the previous section, we optimize the initial deformable model in four steps. In the first step point landmarks are automatically fit to the image data after marking of the landmarks. This produces a preliminary solution which is closer to the target boundary then in the starting position. In the second step, curve landmarks, in particular at the lower part of the vertebral body, move towards the target boundary. In the third step, the lower and the upper planes of the vertebral body are reasonably found by deformation of the surface in one dimension. The result after optimizing all surface points once again in their respective dimension is illustrated in the second row of figure 6.6. The deformable surface has found an optimal solution, weighting both image and shape features along the entire surface. The majority of the surface points was fitted well to the target boundary. At some parts the necklace model moves away due to attraction by neighboring structures or due to locally too much deviation of the target boundary from the population average.



(a) (b) (c)

Figure 6.7: Landmark-based segmentation of the 3rd and 4th lumbar vertebrae. The spine model is visualized together with gray-level planes through the original image and transparent renderings of the high intensity objects: a) initial condition, b) condition after fitting the 3rd lumbar vertebra model and c) condition after fitting the 4th lumbar vertebra model.

To illustrate how the spinal column is extracted using the spine model, segmentation of part of the lumbar spine is performed. First the $L3$ is segmented by deformation of the corresponding deformable surface as before. Simultaneously the position of the $L4$ is estimated by changing the geometry of the corresponding deformable surface according to the solution for the $L3$. On the basis of the preliminary solution for the $L4$, point landmarks on the $L4$ are sought under the constraints that the expected spatial relation is maintained as much as possible, i.e. spinal curvatures comply to the statistics. No interaction is required as the spine model is accurate enough. Then curve landmarks and sheet points are sought in the image data. The position of the initial deformable surfaces and curves and their position after optimizing are illustrated in figure 6.7. The step -by-step automatic segmentation of the $L3$ and $L4$ succeeds despite the articulated vertebral structures and their constellation.

6.5 Discussion and Conclusion

The segmentation of vertebral structures in the previous section is of exemplary nature. The primary goal is to illustrate how the spinal column can be captured using multiple continuous features that are averaged and weighted to exploit salient information and patient-to-patient variation learned from a set of images of normal patients. We have used image data from 6 patients to train vertebra models for the $L2, L3, L4$ and $L5$, six models of the spinal curvatures and one integral spine model. Clearly a larger image data is required to asses the accuracy of the method and its usefulness for clinical use. However, we have demonstrated that the application of multiple continuous features along vertebral surfaces and spinal curves is promising for segmentation of unknown spinal images. For further validation a particular interest is in the long term goal of analyzing average and variation characteristics of normal and pathological spinal columns from longitudinal studies [96], in ways similar to [2] and [7], for the purpose of calculation of local and global deformity quantifying parameters.

We accentuate some important items of our method in comparison with [2], [7], [96]. In the first place, our method works completely in three dimensions, allowing to measure truly three-dimensional properties of vertebral structures and spinal curvatures, rather than relying on two-dimensional features. In the second place, we capture not only shape properties but also image intensity properties as they also define the appearance of vertebral structures. Multiple continuous features are extracted, then statistically analyzed by multivariate functional techniques [95] to obtain important population statistics. This is done a) to exploit natural variation in appearance for constraining the deformation of the spine model, rather restricting it on a priori geometrical constraints and b) to exploit salient information, defined as differential geometrical landmarks by multiple features, for reducing the complexity of the segmentation problem. Furthermore we aim at a step-by-step automatic segmentation departing from geometrically well-defined landmarks on a particular vertebra rather than at aiming for finding a one shot integral solution for the spinal column using manually marked anatomical landmarks.

As concerns the implementation, in our method we used continuous curves and surfaces in the form of B-splines to represent the spinal column and features derived thereof. This in contrast to most other methods that usually use point distribution models, finite element models or geometric primitives. The advantage of a continuous representation is that quantitative information can be analytically computed, allowing for more complete and accurate measurements, especially in the clinically relevant differences in curvature and torsion of the spinal column. No manual or other additional heuristic techniques are required to compute the positions of landmarks, contrary to e.g. [62]. We found that the continuous curves and surfaces provide a compact representation in the form of control points. This is beneficiary because a restricted number of points are required to define the spinal column and to control the spine model when segmenting images.

A drawback of the reported method is an inherent difficulty of deformable models often getting trapped in local minima of optimization. It has been acknowledged previously that proper initialization is required to guarantee satisfying result in view of the presence of disturbing attractors in the image [83]. This is particularly true for complex images such as of the spine. As automatic initialization of deformable models is still an open problem [45], we deal with these problem by minimal human-computer interaction. The user simply points and clicks in the image to make one-to-one correspondence between model and image by means of six point landmarks. The use of point landmarks alleviates the problem of interaction in three-dimensional space [45] due to their zero-dimensional property. During segmentation the user controls the entire spine model as a marionette by interaction with a few point landmarks and propagation of landmark solutions to other parts of the spinal column.

We have only considered CT images of normally appearing spinal columns. When dealing with spinal deformities such as scoliotic spines that exhibit lateral curvatures and vertebral rotations, segmentation using the integral spine model may face problems due to considerable deviations from the normal spinal column. In this context it is essential to select invariant features in order to capture a broad range of natural variations and simultaneously minimizing the effects of non-essential variations in ground-truth delineations. Also, it important to carefully handle the alignment problem has this will be difficult, if not possible, when dealing with much variation. A possible approach is construct statistical spine models for different classes of conditions. This already has been proposed in [2] where, similar to our approach, a large collection of X-ray images from a longitudinal study of idiopathic scoliosis is examined from side and front view to obtain important summary statistics such as curvature. However, as stated by Aykroyd and Mardia, even when different spine models are constructed for different spinal conditions, there would still be substantial variability within these conditions to obtain good statistical summary. This remains an open problem.

Bibliography

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

[2] R.G. Aykroyd and K.V. Mardia. Shape analysis of spinal curves by wavelet warping using an mcmc approach. Technical Report No. STAT-96/10, The University of Leeds, Leeds, UK, 1996.

[3] S. Baluja. Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning. Technical Report CMU-CS-94-163, Carnegie Mellon University, Pittsburgh, PA, 1994.

[4] S. Baluja and R. Caruana. Removing the genetics from the standard genetic algorithm. In *Proceedings of International Conference on Machine Learning*, 1995.

[5] A. Baumberg and D. Hogg. Learning flexible models from image sequences. *Proceedings of European Conference on Computer Vision*, A:299–308, 1994.

[6] D. Beard, D. Eberly, B. Hemminger, and S. Pizer. Interacting with image hierrachies for fast and accurate object segmentation. In *Proceedings of SPIE Image Processing*, volume 2167, pages 10–17, 1994.

[7] S. Benameur, M. Mignotte, S. Parent, H. Labelle, W. Skalli, and J. De Guise. Three-dimensional biplanar reconstruction of scoliotic vertebrae using statistical models. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 577–582, 2001.

[8] P.J Besl and R.C. Jain. Invariant surface characteristics for 3D object recognition in range images. *Computer Vision Graphics and Image Processing*, 33(1):33–80, jan 1986.

[9] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, June 1989.

[10] F.L. Bookstein. Shape and the information in medical images: A decade of the morphometric synthesis. *Computer Vision and Image Understanding*, 66:97–118, 1997.

[11] M Brejl. *Automated Initialization and Automated Design of Border Detection Criteria in Edge Based Image Segmentation.* PhD thesis, 1999.

[12] M. Brejl and M. Sonka. Object localization and border detection criteria design in edge-based image segmentation: Automated learning from examples. *IEEE Transactions on Medical Imaging*, 19:973-985, 2000.

[13] R.P. Brent. *Algorithms for Minimizationwithout Derivatives.* Prentice-Hall, Englewood Cliffs, NJ, 1973.

[14] S. Chatterjee, A.S. Hadi, and B. Price. *Regression Analysis by Example.* John Wiley and Sons, New York, 2000.

[15] I. Cohen, L.D. Cohen, and N.J. Ayache. Using deformable surfaces to segment 3-d images and infer differential structures. In *Proceedings of European Conference on Computer Vision*, pages 648–652, 1992.

[16] L.D. Cohen. On active contour models and balloons. *Computer Vision Graphics and Image Processing*, 53(2):211–218, March 1991.

[17] L.D. Cohen. Auxiliary variables and 2-step iterative algorithms in computer vision problems. *Mathematical Imaging and Vision*, 6(1):59–83, January 1996.

[18] L.D. Cohen and R. Kimmel. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision*, 24(1):57–78, August 1997.

[19] T.F. Cootes, G.J. Edwards, and C.J Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, July 2001.

[20] T.F. Cootes, A. Hill, C.J. Taylor, and J. Haslam. Use of active shape models for locating structure in medical images. *Image and Vision Computing*, 12(6):355–365, July 1994.

[21] T.F. Cootes and C.J. Taylor. Combining point distribution models with shape models based on finite element analysis. *Image Vision and Computing*, 13:403–409, 1995.

[22] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models: Their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995.

[23] T.F. Cootes, C.J. Taylor, and A. Lanitis. Multi-resolution search with active shape models. *Proceedings of International Conference on Pattern Recognition-A*, 94:610–612.

[24] T.F. Cootes, C.J. Taylor, A. Lanitis, D.H. Cooper, and J. Graham. Building and using flexible models incorporating grey-level information. In *Proceedings of IEEE International Conference on Computer Vision*, pages 242–246, 1993.

[25] D. Daniels, D. Campenhout, W. Niblack, W. Equitz, R. Barber, E. Bellon, and F. Fierens. Interactive outlining: an improved approach using active contours. In *Proceedings of SPIE Storage and Retrieval for image and video databases*, pages 226–233, 1993.

[26] J. Dansereau, A. Chabot, N.T. Huyn, H. Labelle, and J. deGuise. Three-dimensional reconstruction of vertebral endplate wedging. In M. D'Amico and G.C. Santambrogio, editors, *Three dimensional analysis of spinal deformities.*, pages 69–73, 1995.

[27] R.H. Davies, T.F. Cootes, C. Twining, and C.J. Taylor. An information theoretic approach to statistical shape modelling. In Tim Cootes and Chris Taylor, editors, *Proceedings of the British Machine Vision Conference*, pages 3–11, 2001.

[28] G. de Vries and P.W. Verbeek. Scale-adaptive landmark detection, classification and size estimation in 3d object-background images. In *Proceedings of International Conference on Pattern Recognition*, pages Vol III: 1026–1029, 2000.

[29] J. Declerck, G. Subsol, J.P. Thirion, and N. Ayache. Automatic retrieval of anatomical structures in three-dimensional medical images. In *Proceedings of Computer Vision, Virtual Reality and Robotics in Medicine*, 1995.

[30] A. del Bimbo and P. Pala. Visual image retrieval by elastic matching of user sketches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):121–132, February 1997.

[31] I.L. Dryden and K.V. Mardia. *Statistical shape analysis.* John Wiley and Sons, New York, 1998.

[32] J. Duckworth. What is chemometrics. In *Proceedindgs of International Conference on Near Infrared Spectroscopy*, 1995.

[33] J.S. Duncan and N. Ayache. Medical image analysis: Progress over two decades and the challenges ahead. *PAMI*, 22(1):85–106, January 2000.

[34] B.S. Duran and P.K. Odell. *Cluster Analysis-A Survey.* Springer-Verlag, 1974.

[35] N. Duta, A.K. Jain, and M.P. Dubuisson-Jolly. Autòmatic construction of 2d shape models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5):433–446, May 2001.

[36] D. Eberly, R. Gardner, B. Morse, S. Pizer, and C. Scharlach. Ridges for image analysis. *Mathematical Imaging and Vision*, 4:353–373, 1994.

[37] G.J. Edwards, A. Lanitis, C.J. Taylor, and T.F. Cootes. Statistical models of face images: Improving specificity. *Image and Vision Computing*, 16(3):203–211, March 1998.

[38] D.V. Beard et.al. The magic crayon: An object definition and volume calculation testbed. *Spie Image Processing*, 1898:789–796, 1993.

[39] A.X. Falcao, J.K. Udupa, S. Samarasekera, S. Sharma, B.E. Hirsch, and R.d.A. Lotufo. User-steered image segmentation paradigms: Live wire and live lane. *Graphical Models and Image Processing*, 60(4):233–260, July 1998.

[40] T.-J. Fan, G. Medioni, and R. Nevatia. Segmented description of 3-d surfaces. *IEEE transactions on Robotics and Automation*, 32:527–538, 1987.

[41] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press, Boston, 1993.

[42] J. Feldmar and N.J. Ayache. Rigid, affine and locally affine registration of free-form surfaces. *International Journal of Computer Vision*, 18(2):99–119, May 1996.

[43] S.D. Fenster and J.R. Kender. Sectored snakes: Evaluating learned-energy segmentations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):1028–1034, 2001.

[44] L. Florack. *Image Structure*, chapter Local Image Structure, pages 133–171. Kluwer, Dordrecht, 1997.

[45] A. F. Frangi, W.J. Niessen, R.M. Hoogeveen, van W. Walsum, and M.A. M.A. Viergever. Model-based quantitation of three-dimensional magnetic resonance angiographic images. *IEEE Transactions on Medical Imaging*, 18(10):946–956, 1999.

[46] S. Frantz, K. Rohr, and H.S. Stiehl. Multi-step procedures for the localization of 2-d and 3-d point landmarks and automatic roi size selection. In *Proceedings of European Conference on Computer Vision*, 1998.

[47] P. Giblin and B.B. Kimia. A formal classification of three-dimensional medial axis points and their local geometry. In *Proceedings of IEEE Computer Vision Patern Recognition Conference*, pages I:566–573, 2000.

[48] S.F.F. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical report, Mitsubishi Electronic Technology Center, Cambridge Massachusetts, 1997.

[49] A.P. Godillon, F.X. Lepoutre, and D. Chopin. Evaluation of a three-dimensional geometrical spinal model. In M. D'Amico and G.C. Santambrogio, editors, *Three dimensional analysis of spinal deformities.*, pages 191–195, 1995.

[50] C. Goodall. Procrustes methods in the statistical analysis of shape. *Royal Statistical Society. Series B. Methodological*, 53(2):285–33, 1991.

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

[52] L. Haglund. Adaptive multi-dimensional filtering. *Ph.D thesis*, 1992.

[53] G. Hamarneh. Active shape model:modeling shape variations and gray level information and an application to image search and classification. Technical Report R005/1998, Department of Signals and Systems, Chalmers University of Technology, Sweden, 1998.

[54] Kenneth M. Hanson, editor. *Shape-based Interactive Three-Dimensional Medical Image Segmentation*, volume 3034, 1997.

[55] K.D. Harris, D.A. Henze, J. Csicsvari, H. Hirase, and G. Buzsaki. The accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. *Journal of Neurophysiology*, 4:401–414, 2000.

[56] J. Haslam, C.J. Taylor, and T.F. Cootes. A probabalistic fitness measure for deformable template models. In *Proceedings of British Maschine Vision Conference*, pages 33–42, 1994.

[57] A.K. Jain and A. Vailaya. Image retrieval using color and shape. *Pattern Recognition*, 29(8):1233–1244, August 1996.

[58] A.K. Jain, Y. Zheng, and S. Lakshmanan. Object matching using deformable templates. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 18:267–278, 1996.

[59] A.K. Jain, Y. Zhong, and M.P. Dubuisson-Jolly. Deformable template models: A review. *SP*, 71(2):109–129, December 1998.

[60] S.B. Jang, K.S. Booth, C.W. Reily, B.J. Sawatzky, and S.J. Tredwell. Three-dimensional analysis of scoliosis surgery using stereo photogrammetry. In *Proceedings SPIE Electronic Imaging Science and Technology*, 1994.

[61] M. Kass, A.P. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *Proceedings of IEEE International Conference on Computer Vision*, pages 259–268, 1987.

[62] C. Kauffmann and J.A. de Guise. Digital radiography segmentation of scoliotic vertebral body using deformable models. In *Proceedings of SPIE Medical Imaging*, volume 3034, pages 243–251, 1997.

[63] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simmulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(3):267–286, May 1983.

[64] J. J. Koenderink. The structure of images. *Biol. Cybern.*, 50:363–370, 1984.

[65] J. J. Koenderink. *Solid Shape.* MIT Press, 1989.

[66] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of International Joint Conference on Artificial Intelligence*, 1995.

[67] K. Konstantinides and J.R. Rasure. The khoros software development environment for image and signal processing. *IEEE Transactions Image Processing*, 3(3):243–252, 1994.

[68] A.C.W. Kotcheff and C.J. Taylor. Automatic construction of eigenshape models. *Medical Image Analysis*, 2:303–314, 1998.

[69] R. Kramer. *Chemometric Techniques for Quantitative Analysis.* Marcel-Dekker, 1998.

[70] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):743–756, July 1997.

[71] M. Leventon, E. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proceedings of Computer Vision and Pattern Recognition*, 2000.

[72] S. Liu and S. Sclaroff. Medical image segmentation and retrieval via deformable models. In *Proceedings of IEEE International Conference on Image Processing*, pages 1071–1074, 2001.

[73] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[74] C. Lorenz and N. Krahnstver. Generation of point-based three-dimensional statistical shape models for anatomical objects. *Computer Vision and Image Understanding*, 77(2):175–191, February 2000.

[75] H.J. Lowe, I. Antipov, W.R. Hersh, and C.A. Smith. Towards knowledge-based retrieval of medical images.the role of semantic indexing, image content representation, and knowledge-based retrieval. In *Proceedings of the 1998 AMIA Annual Fall Symposium*, 1998.

[76] G. Lu. An approach to image retrieval based on shape. *Journal of Information Science*, 23(2):119–128, July 1997.

[77] F. Maes, D. Vandermeulen, P. Suetens, and G. Marchal. Computer-aided interactive object delineation using an intelligent paintbrush technique. In *Proceedings of Computer Vision, Virtual Reality and Robotics in Medicine*, 1995.

[78] K.V. Mardia, A.N. Walder, P.A. Millner, and R.A. Berry, E.and Dickson. Analysis of three-dimensional spinal shape from two x-ray views. In *Image Fusion and Shape Variability Techniques*, pages 123–128, 1996.

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

[80] F. Meyer and S. Beucher. Morphological segmentation. *Visual Communication and Image Representation*, 1(1):21–46, September 1990.

[81] E.N. Mortensen and W.A. Barrett. Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing*, 60(5):349–384, September 1998.

[82] M. Nakahara. *Geometry, Topology and Physics.* Adam Hilger, Bristol and New York, 1998.

[83] W. Neuenschwander, P. Fua, G. Szekely, and O. Kubler. Initializing snakes. In *Proceedings of IEEE Computer Vision Patern Recognition Conference*, pages 658–663, 1994.

[84] W.M. Neuenschwander, P. Fua, L. Iverson, G. Szekely, and O. Kubler. Ziplock snakes. *IJCV*, 25(3):191–201, December 1997.

[85] A. Neumann and C. Lorenze. Statistical shape model based segmentation of medical images. *Computerized Medical Imaging and Graphics*, 22:133–143, 1998.

[86] National health and nutrition examination survey (1976-1980), public use data tape documentation, physicians examination. national center for health statistics. available at national library of medicine, communications engineering branch, http://archive.nlm.nih.gov/.

[87] W. Niessen. *Multiscale Medical Image Analysis.* PhD thesis, Utrecht University, 1997.

[88] S. Olabarriaga, A.W.M. Smeulders, A.C.A Marijnissen, and K.L. Vincken. An intelligent interactive method for the joint space in osteoarthritic ankles. *Information Processing in Medical Imaging*, pages 394–399, 1999.

[89] S.D Olabarriaga and A.W.M. Smeulders. Interactive segmentation in medical image segmentation: A survey. *submitted to Medical Image Analysise*, 2000.

[90] B. Olstad and A.H. Torp. Encoding of a-priori information in active contour models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):863–872, September 1996.

[91] E.G.M. Petrakis. Content-based retrieval of medical images. *International Journal of Computer Research*, Forthcoming.

[92] R. Pichumani. *Construction of a Three-dimensional Geometric Model for Segmentation and Visualization of Cervical Spine Images.* PhD thesis, Stanford University, 1997.

[93] L. Piegl and W. Tiller. *The NURBS Book.* Springer-Verlag, Berlin, 1997.

[94] Elliot P.J., J.M. Knapman J.M, and Schlegel W. Interactive image segmentation for radiation treatment planning. *IBM Systems Journal*, 31(4):441–449, 1992.

[95] J.O. Ramsay and B.W. Silverman. *Functional data analysis.* Springer-Verlag, Berlin, 1997.

[96] L. Rodney Long and G. R. Thoma. Landmarking and feature localization in spine x-rays. *Journal of Electronic Imaging*, 4(10):939–956, 2001.

[97] R. Rodney Long and G.R. Thoma. *Image Query and Indexing for Digital X-rays.* Communications Engineering Branch, Lister Hill National Center for Biomedical Communications, National Library of Medicine, 1998.

[98] K. Rohr. Landmark-based elastic matching of three-dimensional medical images. In *Proceedings of Second Germany-Korea Joint Conf. on Advanced Medical Image Processing*, pages 60–63, 1997.

[99] K. Rohr. *Landmark-Based Image Analysis:Using Geometric and Intensity Models.* Kluwer, 2001.

[100] K. Rohr, H.S. Stiehl, R. Sprengel, W. Beil, T.M. Buzug, J. Weese, and M.H. Kuhn. *Point-Based Elastic Registration of Medical Image Data Using Approximating Thin-Plate Splines*, pages 297–306. Springer Berlin Heidelberg, 1996.

[101] S. Rowe and A. Blake. Statistical feature modelling for active contours. *Proceedings of European Conference on Computer Vision*, B:560–569.

[102] R.Sedgewick. *Algorithms.* Addison-Wesley Press, 1983.

[103] Y. Rui. A modified fourier descriptor for shape matching in mars. In *Image Databases and Multimedia Search*, pages 165–180, 1998.

[104] Y. Rui, A.C. She, and T.S. Huang. Automated shape segmentation using attraction-based grouping in spatial-color-texture space. In *Proceedings of IEEE International Conference on Image Processing*, pages 816–821, 1996.

[105] Tieck S, Gerloff S, and Stiehl H.S. Interactive graph-based editing of watershed-segmented 2d images. In *Proceedings of Interactive Segmentation of Medical Images Workshop*, 1998.

[106] R. Samadani and C. Han. Computer-assisted extraction of boundaries from images. In *Proceedings of SPIE Storage and Retrieval*, 1993.

[107] J.A. Schnabel and S.R. Arridge. Active contour models for shape description using multiscale differential invariants. In *Proceedings of British Machine Vision Conference*, 1995.

[108] S. Sclaroff and A.P. Pentland. Modal matching for correspondence and recognition. *T-PAMI*, 17:545–561, 1995.

[109] O. Shinagawa, T.T. Kunii, A.G. Belyaev, and T. Tsukioka. Shape modeling and shape analysis based on singularities. *The International Journal of Shape Modeling*, 2:85–102, 1996.

[110] G.J. Sivewright and P.J. Elliot. Interactive region and volume growing for segmenting volumes in mr and ct images. *Medical Informatics*, 19(1):71–80, 1994.

[111] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R.C. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.

[112] D.M. Squire, W. Muller, H. Muller, and J. Raki. Content-based query of image databases, inspirations from text retrieval. *Pattern Recognition Letters*, 21(11):1193–1198, 2000.

[113] L.H. Staib, A. Chakraborty, and J.S. Duncan. An integrated approach for locating neoroanatomical structure from mri. *International Journal of Pattern Recognition and Artificial Intelligence*, 11(8):1247–1269, December 1997.

[114] 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.

[115] L.H. Staib and J.S. Duncan. Model-based deformable surface finding for medical images. *IEEE Transactions on Medical Imaging*, 15(5):720–731, October 1996.

[116] I.A.F. Stokes. Biplanar radiography for measurement of spinal shape and motion. *Automedica*, 5:37–49, 1985.

[117] D.J. Struik. *Lectures on classical differential geometry*. Addison-Wesley Press, 1950.

[118] H.D. Tagare, C.C. Jaffe, and J. Duncan. Medical image databases: A content-based retrieval approach. *American Medical Informatics Association*, 4:184–198, 1997.

[119] D. Terzopoulos and D. Metaxas. Constraining deformable superquadrics and nonrigid motion tracking. In *Proceedings of IEEE Computer Vision Patern Recognition Conference*, pages 337–343, 1991.

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

[121] J.P. Thirion. Extremal points: Definition and application to three-dimensional image registration. In *Proceedings of IEEE Computer Vision Patern Recognition Conference*, pages 587–592, 1994.

[122] B. van Ginneken, A.F. Frangi, J.J. Staal, B. ter Haar Romeny, and M. Viergever. A non-linear gray-level appearance model improves active shape model segmentation. In *Proceedings of IEEE Workshop on Mathematical Models in Biomedical Image Analysis*, pages 205–212, 2001.

[123] N. Vasconcelos. On the complexity of probabilistic image retrieval. In *Proceedings of IEEE International Conference on Computer Vision*, 2001.

[124] B. Verdonck, R. Nijlunsing, F.A. Gerritsen, J. Cheung, D.J. Wever, A. Veldhuizen, S. Devillers, and S. Makram-Ebeid. Computer assisted quantitative analysis of deformities of the human spine. In W.H. Wells, A. Colchester, and S. Delp, editors, *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, LNCS, pages 822–831. Springer, 1998.

[125] Y.M. Wang and L.H. Staib. Boundary finding with prior shape and smoothness models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):738–743, July 2000.

[126] G.Q. Wei, J.Z. Qian, and H. Schramm. Generalized dynamic programming approaches for object detection: Detecting spine boundaries and vertebra endplates. In *Proceedings of IEEE Computer Vision Patern Recognition*, pages 954–959, 2001.

[127] C.F. Westin, A. Bhalerao, H. Knutsson, and R. Kikinis. Using local three-dimensional structure for segmentation of bone from computer tomography images. In *Proceedings of IEEE Computer Vision Patern Recognition*, pages 794–800, 1997.

[128] D.J. Williams and M. Shah. A fast algorithm for active contours and curvature estimation. *Computer Vision Graphics and Image Processing*, 55(1 January):14–26, January 1992.

[129] M. Worring, A.W.M. Smeulders, L.H. Staib, and J.S. Duncan. Parameterized feasible boundaries in gradient vector-fields. *Computer Vision and Image Understanding*, 63(1):135–144, January 1996.

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

[131] A.L. Yuille, D.S. Cohen, and P.W. Hallinan. Feature extraction from faces using deformable templates. *Computer Vision*, 8(2):99–111, August 1992.

[132] S.C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900, September 1996.

Summary

In today's medical practice, digital images have evolved to such a degree that they have become a key and pervasive source of anatomical and functional information. Literally, millions of images are generated everyday and the number keeps escalating with advances in imaging technology. It will be impossible to cope with this explosion of image data without the aid of computers. Computers will be indispensable for such tasks as storage, processing, analysis and retrieval of large digital image collections. Thus, it is not surprising to see in contemporary medical imaging science a vivid interest in the development of qualified technology. One of the topics that has received considerable attention is the problem of image segmentation. Numerous methods have been proposed to attack the segmentation problem for a specific task, anatomical structure or imaging sensor. The efforts described in this thesis have concentrated on the development of more generic and robust image segmentation methods. Taking medical imaging problems as example applications, the basic questions we have addressed are *a)* how to automatically learn what constitutes the boundary of an anatomical structure in an image and *b)* how its constituents can be exploited for image segmentation and image browsing in large datasets.

Departing from the view that most contemporary segmentation methods still need significant user input to generate acceptable results, in chapter 2 we have focussed on interactive segmentation. We have set our goal to combine computation and interaction in a more structured and integral manner. To this end, we have first characterized human-computer interaction in terms of user input, interpretation of user input and purpose thereof. On the basis of this characterization, we have examined interactive methods in the image segmentation literature. We have found deformable model methods to form a suitable platform for interactive segmentation. For a better understanding of the benefits and drawbacks of deformable models we have characterized, reviewed and examine some well-known deformable model methods in literature. This examination and characterization has allowed us to get more insight in the main components of deformable models and to identify places for interaction within and between these components. We have come to the conclusion that inhomogeneous deformable models and statistical deformable models have many properties required for successful segmentation of complex medical images. This has motivated us to conduct further research on *a)* the construction of inhomogeneous deformable models by statistical analysis of boundary features derived from a set of annotated example images and *b)* the application of such deformable models for image segmentation and

image browsing.

Chapter 3 tackles the problem of how boundary features such as edge gradient and contour curvature, are properly captured, summarized and applied for model construction and image segmentation. An important contribution has been the representation of a boundary as a multivariate curve, emanating from recording multiple continuous features along that boundary. This representation permits the combination of techniques from functional data analysis with techniques from chemometrics to arrive at a method which we have called *strings*. In this method two phases are distinguished: learning and segmenting. The problem of learning is transposed into one of statistically analyzing the closed functional curves in feature space that best describe average feature values and the most important variations therein. Segmentation is conceived of as an iterative procedure in which a potential boundary in an unknown image *a)* is recorded in terms of multiple continuous features *b)* its feature values are weighted to amplify the statistically most descriptive features and *c)* its weighted feature values are evaluated with respect to the values seen in the learning set for acceptance or rejection. Strings have been compared with active shape models on 145 digitized X-ray images of cervical vertebrae. The results indicate that strings produce better results when initialized close to the target boundary, especially for visually ill-defined boundaries, and comparable results otherwise.

Chapter 4 focuses on the problem of how to learn the search intention of a user performing content-based image retrieval by example, e.g. for assistance in the diagnosis of the contents of an unknown image. In an attempt to solve this problem, we have developed a method to browse images on the basis of string segmentations of anatomical structures therein. In this method, first feature values of a population of normal anatomical structures in a learning set are captured in a string model. The string model is used to bootstrap retrieval of images that depict abnormal anatomical structures similar to the one contained in, and segmented from, the example image using the string model. In an iterative process, good or specifically bad retrievals are indicated interactively to refine the string model to one that describes the sought anatomical structure more precisely. The features of the structures in the indicated images also form a pilot for a population-based incremental learning technique. This technique explores the feature space using a probabilistic model. The probabilistic model aims at defining relevant parts of the feature space by moving to those area's were the feature values correspond to those seen in the, as positive indicated, retrieved images. At the end of the exploration, the probabilistic model defines a narrow region of the feature space frequently encountered during the browsing process, hence circumscribing the retrieval intention in terms of reoccuring features. Browsing ends with a set of images depicting same abnormalities. The method has been successfully applied for retrieval by example of digitized X-ray images of abnormal cervical vertebrae.

Chapter 5 addresses the question of how to exploit salient boundary features, landmarks, to reduce the complexity of a multi-dimensional image segmentation problem. We have proposed a method in which we conceive of segmentation as a procedure in which *a)* multiple continuous features are recorded, *b)* optimal features are selected locally and *c)* the selected features are step-wise exploited for image segmentation.

As in chapter 3, multiple continuous boundary features in a set of annotated images are extracted and statistically analyzed to obtain a model of the boundary features. The statistical model is used to select optimal features rather than to weight them as done in chapter 3. Feature selection is done for each boundary point on the basis of the curvature properties of the feature values around that point, i.e landmarks are determined on geometrical grounds. Each boundary point is, in addition, classified as sheet point, curve landmark or point landmark based on the number of dimensions in which the selected feature is well defined. Results of feature selection and classification are input to a priority segmentation scheme which tries to automatically find boundaries in an unknown image, first by point-landmarks, then by curve landmarks and finally by sheet points. To this end a deformable model is used that has been accommodated to identify and exploit boundary landmarks. A user has the liberty to indicate a number of landmark points in the image to guide model deformation. As this deformable model uses different features and different search dimensions at different boundary points, we have called the segmentation method *necklaces*. The application of necklaces for segmentation of three-dimensional CT images of vertebral structures has shown acceptable results when used in an interactive setting.

Unlike the other chapters, chapter 6 takes on a specific segmentation problem, namely that of finding the spinal column in three-dimensional images. The method we have proposed to solve this inherently three-dimensional problem combines strings and necklaces to arrive at a true three-dimensional solution. Necklace models of vertebrae are constructed by statistical analysis of multiple features along boundaries in a given learning set. The necklace models define for each vertebra optimal features and their dimensionality. They are used for deformable model segmentation of single vertebral structures. Necklace models are coupled by string models, which express the spatial relationship between adjacent vertebrae in terms of multiple global spinal features such curvature and length. The string models define natural variations in spinal curvature values as observed in a learning set. The necklace and string models are coupled to form an integral spine model. The spine model is used as a reference during the segmentation of an unknown image with help of a deformable model. The deformable model fits to the image data by elastic deformation in ways reminiscent to a marionette with interrelated structures moved by strings. The user has the opportunity to guide the deformation process of the entire spine model by pointing and clicking at landmarks in the image. To illustrate, we have segmented a CT image of part of the human lumbar spine. In combination with user interaction the step-by-step segmentation has produced promising results.

Samenvatting

In de dagelijkse klinische praktijk spelen digitale beelden een steeds grotere rol in de voorziening van anatomische en functionele informatie. Elke dag worden letterlijk miljoenen beelden gegenereerd en dit aantal zal alleen maar toenemen met de snelle ontwikkelingen in de digitale beeldtechnologie. Het is onmogelijk om zulke grote bestanden van digitale beelden te onderhouden zonder ondersteuning van computers. Computers zullen onmisbaar zijn voor taken als opslag, verwerking, analyse en ontsluiting van grote collecties digitale beelden. Het is dan ook niet verwonderlijk dat in de medische wetenschap grote belangstelling is ontstaan voor ontwikkeling van adequate beeldverwerkingstechnologieën. Als een van de meest urgente onderwerpen heeft het probleem van beeldsegmentatie hierbij immens veel aandacht ontvangen. Een groot aantal beeldsegmentatiemethoden is ontwikkeld dat zich richt op een specifiek anatomisch orgaan, een specifieke taak of specifieke beeldmodaliteit. Het onderzoek waarvan dit proefschrift verslag doet, heeft zich ten doel gesteld generieke en robuuste beeldsegmentatiemethoden te ontwikkelen. In de context van medische applicaties, wordt ingegaan op de fundamentele vragen *a)* hoe de bouwstenen die randen van anatomische structuren in beelden definiëren geleerd kunnen worden, en *b)* hoe deze bouwstenen gebruikt kunnen worden voor beeldsegmentatie en het zoeken in grote collecties beelden.

In hoofdstuk 2 staat interactieve segmentatie centraal. Aangenomen wordt dat beeldsegmentatie nog altijd niet volledig automatisch kan geschieden en dat een bepaalde mate van mens-computer interactie nog nodig is voor het behalen van acceptabele resultaten. De vraag waarop wij ons gericht hebben is hoe berekening en interactie gecombineerd kunnen worden op een gestructureerde en integrale manier. Voor het beantwoorden van deze vraag hebben wij de mens-machine interactie beschreven en gekarakteriseerd in termen van het type, interpretatie en doel van gebruikersinput. Op basis van de karakterisering hebben we interactieve segmentatiemethoden in de literatuur bestudeerd. We zijn tot de conclusie gekomen dat deformeerbare modellen een geschikt platform vormen voor het integraal combineren van berekening en interactie. Om beter inzicht te krijgen in de voor- en nadelen van toepassing van deformeerbare modellen voor beeldsegmentatie, hebben we een aantal deformeerbare modellen uit de literatuur in meer detail beschreven en gekarakteriseerd. De beschrijving en karakterisering van deformeerbare modellen heeft zich vooral gericht op het begrijpen van de diverse componenten van deze modellen en het identificeren van mogelijkheden voor interactie binnen en tussen deze componenten. Gebleken is dat inhomogene deformeerbare modellen en statistische deformeerbare modellen goede eigenschappen bezitten voor interactieve segmentatie van complexe medische beelden. Dit heeft ons er toe gebracht onderzoek te doen naar de constructie van inhomogene deformeerbare modellen door statistische analyse van randkenmerken van anatomische structuren die geëxtraheerd zijn uit geannoteerde voorbeeldplaatjes. Het heeft ons tevens gemotiveerd onderzoek te doen naar de toepassing van geleerde inhomogene modellen voor segmentatie van beelden en het zoeken ervan in grote beelddatasets.

Hoofdstuk 3 houdt zich bezig met de vraag hoe kenmerken van de randen van anatomische structuren in digitale beelden, zoals kromming en de sterkte van intensiteitovergangen, naar behoren kunnen worden vastgelegd, samengevat en toegepast voor beeldsegmentatie. Een belangrijke bijdrage is de representatie van een rand als een multi-variate kromme, welke ontstaat na metingen van meerdere kenmerken langs een continue rand. Deze representatie heeft het mogelijk gemaakt een methode (*snaren* genaamd) te ontwikkelen die gebruik maakt van technieken uit de functionele data analyse en chemometrie. De methode kent twee stappen: leren en segmenteren. In de leerfase, worden meerdere kenmerken langs continue randen in een set van geannoteerde voorbeeldplaatjes geëxtraheerd en statistisch geanalyseerd. Dit komt neer op het bepalen van de multi-variate krommen die het best de gemiddelde kenmerkwaarden en de meest belangrijke variaties in de leerset beschrijven. Beeldsegmentatie kan dan gezien worden als een iteratief proces waarin een potentiële rand in een onbekend beeld *a)* wordt vastgelegd in termen van meerdere kenmerken *b)* wordt gewogen om de meest belangrijke kenmerken te benadrukken, en *c)* wordt geëvalueerd ten opzichte van de kenmerkwaarden in de geannoteerde voorbeeldplaatjes om het al dan wel niet te accepteren als de gezochte rand. Het gedrag van snaren is vergeleken met dat van 'active shape' modellen op basis van 145 gedigitaliseerd röntgenbeelden van wervels. De resultaten tonen aan dat snaren, vooral voor complexe beelden, betere resultaten opleveren als initialisatie ervan in de buurt van de gezochte rand geschiedt.

Centraal in hoofdstuk 4 staat de vraag hoe, in de context van beelddatabases, de zoekintentie van een gebruiker bepaald kan worden. Het scenario dat we als voorbeeld hebben genomen is een gebruiker die op basis van een voorbeeldplaatje soortgelijke beelden in een dataset zoekt voor assistentie bij de diagnose van de inhoud van dat plaatje. In een poging een antwoord te geven op voorgenoemde vraag, hebben wij een methode ontwikkeld die het mogelijk maakt, als het ware te bladeren door beelden in een dataset op basis van segmentaties van anatomische structuren in die beelden. In deze methode vindt eerst een statistische analyse plaats van kenmerkwaarden van een populatie van normale anatomische structuren in een gegeven leerset. De analyse resulteert in een snaarmodel dat gebruikt wordt om het bladeren op gang te brengen door segmentatie van een voorbeeldplaatje. Op basis van de initiële segmentatie worden databasebeelden opgehaald met een soortgelijke segmentatie. In een iteratief proces worden vervolgens de goede en/of slechte beelden in de opgehaalde set interactief aangegeven om het snaarmodel te verfijnen. De kenmerken van de rand van de anatomische structuren in de als relevant aangewezen beelden vormen een gids voor een zoekalgoritme dat is gebaseerd op een probabilistische model. Het probabilistische model probeert belangrijke kenmerkwaarden te bepalen door naar die delen van de kenmerkruimte te bewegen, waar de kenmerkwaarden corresponderen met degene die zijn vastgelegd in de als relevant aangewezen beelden uit de dataset. Aan het einde van het iteratieve proces definieert het probabilistisch model die kenmerkwaarden die herhaaldelijk zijn vastgelegd tijdens het bladeren en daarmee representatief zijn voor datgene waar de gebruiker naar zoekt. Het bladeren eindigt met een set beelden van anatomische structuren met een afwijking die overeenkomt met de afwijking die is vastgelegd in het initiële voorbeeldplaatje. De methode is succesvol toegepast voor het vinden van relevante röntgenbeelden van wervels op basis van een voorbeeldplaatje van een afwijkende wervel.

In hoofdstuk vijf richten we ons op de vraag hoe ijkpunten in randkenmerken gebruikt kunnen worden voor beeldsegmentatie. We beschrijven een methode, *kralensnoer* genaamd, waarin segmentatie wordt beschouwd als een procedure waarin *a)* meerdere randkenmerkwaarden worden vastgelegd, *b)* optimale randkenmerken lokaal worden geselecteerd en *c)* de geselecteerde kenmerken vervolgens stapsgewijs worden toegepast voor segmentatie. Net als in hoofdstuk 3 worden meerdere continue randkenmerkwaarden geëxtraheerd uit een set van geannoteerde voorbeeldplaatjes en vervolgens geanalyseerd om een statistisch model te construeren. Het statistisch model wordt gebruikt om optimale kenmerken te selecteren, dit in tegenstelling tot hoofdstuk 3 waar kenmerken worden gewogen. Voor elk punt op een rand wordt het optimale kenmerk bepaald op basis van de mate van kromming van de kenmerkwaarden rondom dat punt: ijkpunten worden geometrisch bepaald. Elk punt van de rand wordt bovendien geclassificeerd als vlak-achtig, lijn-achtig of punt-achtig op basis van het aantal dimensies waarin dat kenmerk goed is gedefinieerd. Kenmerkselectie en -classificatie wordt vervolgens gebruikt voor het automatisch zoeken naar de rand van een specifieke anatomische structuur in een onbekend beeld. Dit gebeurt stapsgewijs: eerst worden de punt-achtige, dan lijn-achtige en ten slotte vlak-achtige randpunten gezocht. Segmentatie gebeurt met behulp van een deformeerbaar model dat gebruik maakt van verschillende kenmerken en zoekruimtes voor het vinden van verschillende randpunten. Hierbij stellen punt-achtige randpunten een gebruiker in staat door middel van een beperkt aantal kliks in het beeld de deformatie te leiden. De toepassing van kralensnoeren voor interactieve segmentatie van drie-dimensionale CT beelden van wervels heeft aangetoond dat kralensnoeren acceptabele resultaten produceren als deze interactief worden gebruikt.

Het laatste hoofdstuk concentreert zich op een specifiek segmentatieprobleem, namelijk het segmenteren van de wervelkolom in een drie-dimensionaal beeld. De methode die we hebben ontwikkeld combineert snaren en kralensnoeren voor het genereren van drie-dimensionale segmentatieoplossingen. Kralensnoeren worden gebruikt voor het construeren van wervelmodellen door statistische analyse van meerdere, continue randkenmerkwaarden in een leerset van geannoteerde volume beelden van wervels. De wervelmodellen definiëren voor ieder punt langs de wervelrand welk kenmerk optimaal is en wat de dimensionaliteit van dat kenmerk is. De wervelmodellen worden gebruikt als referentie tijdens segmentatie van een wervelbeeld met behulp van een deformeerbaar model. De wervelmodellen, één voor iedere wervel, zijn gekoppeld aan elkaar met behulp van meerdere snaren. De snaren definiëren de spatiele relatie tussen de wervels in termen van meerdere globale wervelkolomkenmerken, zoals de kromming en lengte van de wervelkolom. De kralensnoeren en de snaren vormen samen een integraal model van de wervelkolom dat tijdens segmentatie gebruikt wordt als referentie door een deformeerbaar model. Het deformeerbare model past zich aan aan de wervelkolom in het te segmenteren beeld op een manier die doet denken aan een bewegende marionet. Hierbij speelt de gebruiker een centrale rol: de gebruiker wijst een aantal ijkpunten aan in het beeld voor ondersteuning van het automatische deformatieproces. Ter illustratie hebben we de derde en vierde lumbale wervels gesegmenteerd met behulp van het integrale model.

Acknowledgements

Pursuing a Ph.D. has been a tremendous experience to me in many different ways. For this, I am principally grateful to my supervisor and promotor, Arnold Smeulders. He has given me the space to explore my thoughts in my own way. He also has given me the time to exploit my thoughts in my own paste. Most importantly, however, I feel he has equipped me with some means to systematically expose, explain and solve problems. The innumerable conversations and discussions with him, full of scientific feedback and personal support on his part, have been an important mine of information and motivation to me. Looking back, I can now say that in the last couple of years I have learned much more than I hoped for at the beginning of this journey. Not only for use in science but also for use in everyday life. Arnold, thanks for your tireless support and patience in guiding me in the fields of the scientific world!

In the course of completing this thesis a number of other people have made contributions. The feedback on chapter 4 by Carl Jaffe (Yale University) has been very welcome, especially as it is his research on medical image databases that has motivated me to conduct some research in this field. Pia Pfluger (University of Amsterdam) has been kind enough to improve many of my mathematical formulations in chapter 5, making my somewhat unclear notation more readable to a larger audience. Even though my request for help in the painstaking task of delineating set of three-dimensional images came on a short notice, Diana Bosch (Academic Hospital of the Free University of Amsterdam) has been kind enough to immediately assist me in doing so. I would also like to acknowledge the feedback by Jörg Neuerburg (Academic Hospital of the Heinrich-Heine University) and Dagnew Menghesha (Johanniter-Hospital) on chapter 6. Finally, many thanks to the members of my Ph.D defense committee for their valuable comments, suggestions and naturally for the approval of the thesis.

I have spent in total almost a year and a half with the Image Processing and Analysis group (IPAG) at Yale University, first as a M.Sc. student thanks to Marcel Worring, then two years later, as Ph.D. student. It was here where I came in touch with practical medical imaging problems and where I got the opportunity to work with others on related topics. I would like to thank Jim Duncan for making this possible and for providing me with scientific ideas that are definitely present in this thesis in one way or another. A special word of thanks is also due to Carl Jaffe for his very kind hospitality. I also would like to thank Hemant Tagare, Larry Staib, Anand Rangarajan and Francois Meyer for spending time on answering many of my questions. Working in the lab with, among others, Ravi Bansal, Yongmei Wang, Xiaolan Zeng, Xenios Papademetris, and later Rik Stokking and Collin Studholm has been a pleasure and a basis for continued cooperation.

Many people have contributed to the diverse and pleasant working environment at the Intelligent Sensory Information Systems group (ISIS). Among them are: Andrew Bagdanov, Kees Bergsma, Carlo, de Boer, Rein van den Boomgaard, Aristeidis Diplaros, Jan-Mark Geusenbroek, Dennis Koelma, Hoang Anh Minh, Hieu Nguyen, Giang Nguyen, Thang Pham, Ioannis Patras, Frank Seinstra, Cees Snoek Leon Todoran, Joost van de Weijer, Marcel Worring, Jeroen Vendrig and Frans Verster. A special thanks goes to Theo Gevers, whose dissipative way of communicating has been very welcome on many occusions. I would also like to extend my thanks to Virginie Mes: it is always nice to have someone around to easily rely on for taking care of practical issues. Finally thanks to Erik Engbers, my room mate for about five years. His probing questions and challenging criticism have often been very helpful to me. I reckon his favorite role of playing the advocate's devil has been more of a help to me than my less incentive and intuitive approach to him.

I have recently started working as a post-doctoral fellow at the Biomedical Imaging Group (BIGR) at the Erasmus Medical Center. I would like to thank Jan Grashuis, Rik Stokking, Jifke Veenland and Henri Vrooman for creating the possibility to continue research on medical imaging problems and for giving me the opportunity to finish my thesis in a lenient atmosphere. A special thanks also to Rik Stokking for bringing me in touch with the group. I look forward to conducting further research on medical imaging problems in a medical environment.

I consider myself blessed to have the steady and life-long support of my of sisters, Winta and Mical, my parents, Aster Isahac and Tesfay Ghebreab, and my grandmother Harègu Neamin. They have always formed a solid and secure environment from which I have tapped inspiration and motivation to patiently and cautiously proceed towards my objectives, especially in trying circumstance as in the last couple of years. I hope that the completion of one of these objectives in tangible form, this thesis, is a source of an inspiration and motivation to them. Finally, special gratitude and appreciation are due to my wife, Bet-El Teklemariam. She has been a source of love and of intellectual and spiritual support in so many ways, both before and during the writing of this thesis, that I cannot even begin to enumerate them. If ever a cliche were true, it would be that without her, this thesis would never have been finished (....this way). I really look forward to enjoying "real" life with you, more consciously, more intensively and more vividly.