Segmentation Of Regions In The Medial Temporal Lobe

Bjørn Skovlund Dissing

Kongens Lyngby 2007

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

Preface

This thesis was prepared at the section for Image Analysis, Informatics and Mathematical Modelling (IMM), Technical University of Denmark (DTU) as a partial fulfillment of the requirements for acquiring the degree Master of Science in Engineering (M.Sc.Eng). The work was carried out over a period of 7 months for the amount of 35 ECTS points. Lyngby, December 2007

Bjørn Skovlund Dissing

Acknowledgements

I would like to thank the following people for their help and support during my thesis period: My academic supervisor, Associate Professor Dr. Rasmus Larsen for taking the time to give me feedback and comments at any given time. Ph.d students Mads Fogtmann Hansen and Arnold Skimminge for their help and guidance. Thomas Zöega Ramsøy for delivering the manual annotations of the data. Furthermore I would also like to thank my family and friends for support through this thesis-period, and especially those helping me proof-reading the report.

Abstract

This thesis describes a method used to construct a segmentation algorithm for the Medial Temporal Lobe(MTL) in the human brain. The regions of interest in the MTL consists of four closely connected cortical areas, hippocampus and amygdala. A set consisting of 13 MRI scans of different individuals with matching manual expert annotations of the MTL were delivered, and the intensity images was standardized across all 13 images to get corresponding intensity values for corresponding tissue types.
The regions of the MTL are located fairly deep in the brain where the contrast is quite low, and region-boundaries can be difficult to find. Many classical segmentation approaches will fail to segment these regions of interest, and the key to a more successful segmentation is incorporation of prior knowledge. The manual annotations are represented as Level Set Functions, and a coupled statistical shape model is trained to capture the spatial variation across the dataset. This model is employed in a region-based energy formulation which maximizes the mutual information between the region labels and the image voxel intensity values of the image. A rather significant improvement is seen when the coupled model is compared to individual segmentations of each region.

Resumé

Denne afhandling beskriver en metode der er brugt til at lave en segmenterings algoritme i den Medial Temporale Lob(MTL). Interesseområderne i MTL består af fire tæt knyttede kortikal-områder samt hippocampus og amygdala i hver af de to hjerne hemisfærer. Et sæt bestående af 13 MRI scanninger af forskellige individer med tilhørende manuelle annoteringer af MTL blev udleveret. Intensitets-billederne er alle blevet standardiseret for at opnå en korresponderende intensitet for tilsvarende vævstyper i de enkelte scanninger.
Regionerne i MTL er lokaliseret dybt inde i hjernen hvor kontrasten på MRI scanninger er meget lav, og de enkelte region-skel kan være svære at finde. Derfor vil mange af de klassiske segmenterings metoder fejle og nøglen til en mere successfuld segmentering er inkorporering af forhåndsviden. De manuelle segmenteringer er blevet repræsenteret som *Level Sets*, og en koblet statistisk form model er blevet trænet til at indfange den spatielle variation der er i data-sættet. Denne model er anvendt i en regions baseret energi formulering som maksimerer den gensidige information mellem de enkelte regionsmærkater og billede- voxel intensiteten. Når den koblede model sammenlignes med enkeltvis segmentering af de enkelte regioner ses en tydelig forbedring.

Contents

Chapter 1

Introduction

1.1 Motivation and Main Objectives

Are different forms of memory in the human brain the result of distinct anatomical regions and processing mechanisms?. The answer to this question is not an easy one, and is an active area of ongoing research among neurologists and experts in brain anatomy.
A large number of people with different professions are currently collaborating to answer this and similar questions in a large project on the investigation of The Medial Temporal Lobe(MTL), Section 3.3, at e.g. the Danish Research Center for Magnetic Resonance(DRCMR), Section 3.1.1.

The MTL is the name of a brain region which spans a set of different anatomical brain-regions. It is known that the MTL establishes functional connections with widespread areas of the neocortex, Section 3.3, when the brain is processing declarative memory[1]. However, the roles of the individual regions in the MTL are manyfold, and their precise relative role in the processing of different kinds of memory content and their individual processing stages causes discussion among neurologists. A study of the precise functional relations can be done by observing the activity in the brain.
So how can these areas be investigated when the brain is active? The answer to this is by using functional Magnetic Resonance Imaging(fMRI), Section 3.1, which is a well developed method used to study neural activity in e.g. the brain, in vivo.
Having well defined spatial areas, marking the locations for the study of the full-brain fMRI recording, will make the final results of the study more evident and robust. However, localizing these regions is not an easy task, even for brain-anatomy experts. The process of a manual localization is a very time-demanding and exhaustive job, which ideally should be performed automatically by an intelligent system. This is of course where this thesis fits into the big puzzle, having the objectives:

- Investigate the possibilities of making an intelligent system which is able to localize the regions of interest.
- Elaborate and analyze a sensible approach to an automatic or semi-automatic labeling of these regions
- Develop a prototype which demonstrates the approach and can carry out a labeling of the regions of interest

Besides the neurological motivations described above, the task of building intelligent systems for brain image processing is a motivation in itself and has

[1] Part of the human memory which is responsible for storing facts

many interesting aspects from an engineers point of view. In the research area of intelligent imaging systems, segmentation, Chapter 2, is a vast topic that concerns many people. Especially segmentation of brain regions covers a broad range of literature because of the many challenges this type of imaging system faces. To work with a daunting segmentation task as the one in this dissertation involves many practical problems which will be presented throughout the thesis, and thereby offers good opportunities to exploit many of the skills, achieved in field of machine learning, pattern recognition and image processing.

1.2 List Of Abbreviations

CSF	Cerebro Spinal Fluid
FOV	Field Of View
GM	Gray Matter
LSF	Level Set Function
MI	Mutual Information
MPRAGE	Magnetization Prepared Rapid Acquisition Gradient Echo
MTL	Medial Temporal Lobe
PCA	Principal Component Analysis
PDE	Partial Differential Equation
SDM	Signed Distance Map
SNR	Signal to Noise Ratio
TR	Repetition Time
TE	Echo Time
TI	Inversion Time
VOI	Volume Of Interest
WM	White Matter

Chapter 2

Segmentation Approaches

In the last decades, segmentation of images has been of great importance to many different industries. The process of segmenting an image is the process of distinguishing objects in the image from the background. Of course, since this is of such importance, a vast amount of different techniques useful for different scenarios have been developed, ranging from simple thresholding approaches to more sophisticated statistical approaches. A complete overview of different segmentation methods is obviously out of the scope of this thesis, but a brief listing of different methods will be appropriate, putting more emphasis on techniques relevant for solving the problem faced in this project.

2.1 An overview

In Figure 2.1 a brief overview of different segmentation techniques are grouped after type. The region- and edge-based segmentation groups, contain some low-level segmentation methods. The classification methods use more advanced techniques, although by itself only suited for certain problems. Finally the deformable models and atlas methods are very specialized techniques, interesting for the segmentation of regions in the MTL. Issues such as spatial resolution, poor contrast, ill-defined boundaries, noise or acquisition artifacts, make segmentation a difficult task and it is illusory to believe that it can be achieved by using gray-level information alone ie. edge or region-based. A priori knowledge has to be used in the process, and low-level processing algorithms have to cooperate with higher level techniques such as deformable and active models or atlas-based methods. These low- and high-level techniques will be listed and described in the following in accordance with Figure 2.1.

2.2 Region-based and Morphology Approaches

Region based methods aim to distinguish and label different objects with respect to their regional intensity behaviors. Different methods for doing this exist, where a small subsection is listed in Figure 2.1.The region-based segmentation approach has the following advantages:

- It is guaranteed (by definition) to produce coherent regions.
- Linking edges, gaps produced by missing edge pixels, etc. is not an issue.

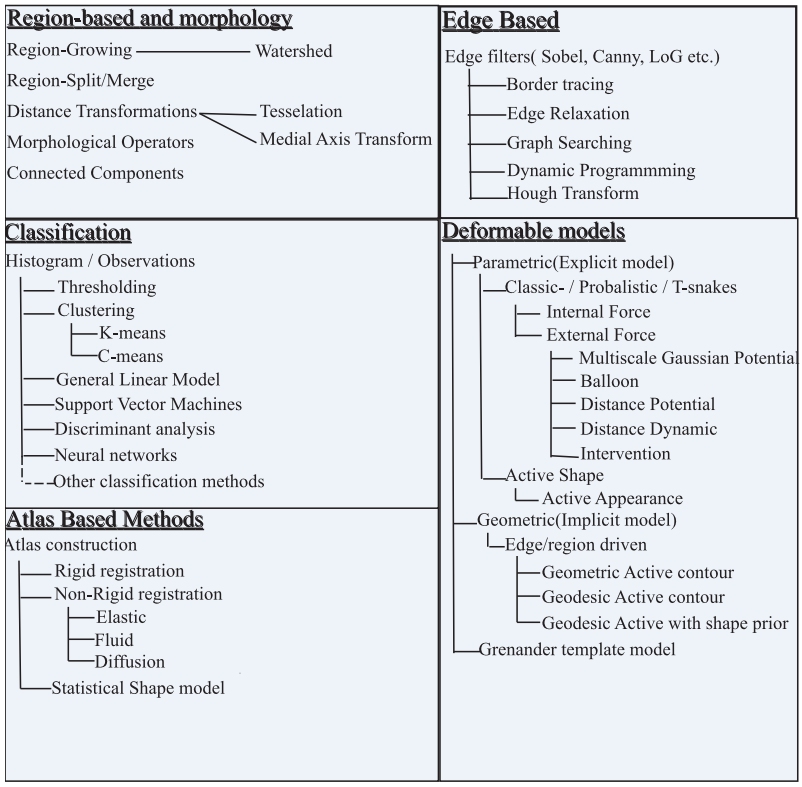

Figure 2.1: *A brief overview of segmentation methods*

- It works from the inside out, instead of the outside in. The question of which object a pixel belongs to is therefore immediate.

However, the region-based segmentation approach also has drawbacks:

- Decisions about region membership are often more difficult than applying edge detectors.
- It cannot find objects that span multiple disconnected regions. (Whereas edge-based methods can be designed to handle "gaps" produced by occlusion.)

The Basic Idea of **Region Growing** is to start with a single pixel p and expand from that seed pixel to fill a coherent region, based on some similarity measure until a certain threshold have been reached.
Many different Similarity Measures has been proposed in the literature, from the most simple neighboring pixel comparison, to more sophisticated measures using texture, gradient, or geometric information. Naturally the algorithm can be improved by growing multiple seeds simultaneously.
The **Split and Merge Algorithms** is an improvement to the region growing algorithm due to the fact that pure merging methods are computationally expensive because they start from such small initial regions. By recursively splitting the image into smaller and smaller regions until all individual regions are coherent, then recursively merging these to produce larger coherent regions the same result is achieved with less computations.
A vast amount of algorithms improving the basic approach just outlined exists, using different merging and splitting approaches. One very famous is the **Watershed** algorithm which sees the input-image as a topographic surface, high pixel intensities being peaks and low ones being grooves. The algorithm fills the holes in this topographic surface leaving only the barriers between the holes, also called the watersheds. Different approaches have been suggested to implement this algorithm. In one approach a euclidean distance map is used to determine the boundaries of the *ponds*. In another approach mathematical morphology is applied to the image, first eroding the image, remembering positions of highest peaks, whereafter a special dilation is performed to connect these peaks. In a third approach called tobogganing neighboring top-peaks are connected by finding common local minima in an iterative fashion. This algorithm is very good for separating particles.

2.3 Edge-based Approaches

Edge-based segmentation represents a large group of methods based on information about edges in the image. The goal of these segmentation methods is to reach at least a partial segmentation, meaning to group local edges into an image where only edge chains with a correspondence to existing objects or image parts are present.
The edges are found using edge detecting operators, often realized as edge-kernels which are convoluted with the image. Some famous edge filters are the *Laplacian*, the *Canny*, the *Sobel* and the *Prewitt* filters. A simple gradient calculation of the image also reveals edges. These edges mark image locations of discontinuities in gray level, color, texture, etc.
However, the resulting edges can for obvious reasons not be used directly as segmentation results. Therefore supplementary processing steps must follow to

combine edges into edge chains that correspond better with borders in the image. This can be quite difficult due to image noise or the presence of unsuitable information such as spurious borders and cusps.
As listed in Figure 2.1 the reconstruction of the edges after the use of an edge operator can be done in various ways. A simple **brute-force approach** tracks the edges in an iterative way starting from the first border pixel encountered, in either a four or eight connection setting. Naturally, this can be further refined by incorporating a graph for edge searching. In this graph, the notes are encoded with boundary candidate pixels, and the transition-cost between nodes could be the euclidean distance in image space. Node-costs could be the edge-probability returned by the edge operator. Now it is just a matter of finding the optimal minimum cost path in an efficient manner, thus extracting the most probable and continuous boundaries. Dynamic programming can be used in a similar manner on a cost matrix expressing the same costs.
Finally, the well known **Hough transformation**, also listed in Figure 2.1 is excellent for finding correct boundaries. The reader is referred to (Ballard; 1981) for further information about this algorithm.

2.4 Classification(Pure intensity based)

The task of classification is the task of calculating a probability for each of a number of predefined classes, for a number of observations and selecting the most probable candidate. This can be done using super-, or unsupervised techniques. An entire branch of science, Machine learning and pattern recognition, deals with the classification of n-dimensional signals which of course means that only a very small subsection of existing techniques has been listed in Figure 2.1. A common feature for most classification techniques is that the probability calculation for each observation is based on a n-dimensional signal. Since MRI-scans can be multi-modal, i.e. represented in T1-,T2 and Proton Density weighted images, these classification techniques are well suited to label certain regions in these scans. As classification is based purely on probability calculated from pixel/voxel intensities, certain segmentation tasks such as white-matter, grey-matter or CSF-fluids labeling in brain-scans are very well suited for classification in contrast to eg. finding specific areas such as the hippocampus or corpus callosum. This would require spatial prior knowledge due to the fact that pixel/voxel intensities within these regions are similar to pixels outside. However, in combination with other methods, classification methods can yield powerful segmentation techniques. This topic is dealt with in the next subsection. Due to the fact that classification is not used in this project, a further description of methods will not be necessary, and the reader is referred to the literature such as (Hastie et al.; 2001) for further information.

2.5 Deformable models

Deformable models are used in different industries, and are very popular in the medical imaging communities, which is relevant for this project.
The human brain can be classified into a number of different distinct anatomical regions where a few is seen in Section 3.3. Each of these anatomical regions has a shape, a spatial position and consists of certain tissues. Often it is the case that these are difficult to locate in medical image scans, which is why local search methods have been developed, which are briefly presented in this section.

2.5.1 The Classical Deformable Models

A deformable model is a contour moving in image space, investigating the properties of local regions. The first of its kind, which has drawn much attention, is the active contour, popularly known as the **Snake** (Kass et al.; 1988). Snakes are planar deformable contours that are useful in several image analysis tasks. They are often used to approximate the locations and shapes of object boundaries in images, based on the reasonable assumption that boundaries are piecewise continuous or smooth. In its basic form, the mathematical formulation of snakes draws from the theory of optimal approximation involving functionals. Deformable models are curves or surfaces defined within an image domain that can move under the influence of *internal forces*, which are defined within the curve or surface itself, and *external forces*, which are computed from the image data. So basically, classical deformable models build on edge-based segmentation, but have restricted degrees of freedom due to the regularizing curve. This is not a complete review of snakes, so for more information the reader is referred to (Kass et al.; 1988), (Sonka and Fitzpatrick; 2000),(McInerney and Terzopoulos; 1996). The basic form of the snake is seen in Equation 2.1.

$$E(C) = S(C) + P(C) \tag{2.1}$$

This can be rewritten into Equation 2.2

$$E(C) = \alpha \int_0^1 |C'(q)|^2 dq + \beta \int_0^1 |C''(q)|^2 dq - \lambda \int_0^1 |\nabla I(C(q))| dq \tag{2.2}$$

The two first terms denote the inner force, $S(C)$, where α damps the smoothness of the deforming curve and β damps the rigidity of the curve. The third term in Equation 2.2, $P(C)$ denotes the external forces, which couples the deforming contour to the image. Here λ controls the attraction strength of the boundary attraction. As seen in Figure 2.1 there exists a series of external forces, which can be concatenated in an additive way ie. $P(C) = P_1(C) + P_2(C) + \cdots + P_N(C)$.

The balloon force is a rather vital force, and was suggested by (Cohen; 1991) as an additional force given by $P(C) = wN(C)$, ie. an addition of the inward normal to the curve C to the external force, which makes the snake prevent getting trapped by isolated spurious edges. A short description of these forces is listed below. Again, for a more thorough overview of these the reader is referred to (Sonka and Fitzpatrick; 2000) and (McInerney and Terzopoulos; 1996):

Multiscale gaussian a scalespace version of the normal gaussian attraction force

Distance Calculating a distance map to edges, obtaining force field with large attraction range

Intervention Makes the user able to interact with the contour while evolving. Consists of e.g. a *Spring* and *volcano* which pulls and pushes the evolving contour respectively towards the points specified by the user.

In spite of the elegant design, snakes has some drawbacks.

- Topology - Cannot model bifurcations, splitting and merging of objects
- Problems with numerical implementations
- Non-intrinsic - Energy depends on parametrization of curve, not directly related to the objects geometry

To overcome some of these problems, (Caselles et al.; 1993) and (Malladi et al.; 1995) have proposed the implicit active contour model, also known as the geometric active contour, which constitutes very interesting applications of level set ideas within the active contour framework.

$$\frac{\partial \phi}{\partial t} = g(I)|\nabla \phi| \left(\operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + k \right) \tag{2.3}$$

In this setting, the model is motivated by a curve evolution approach and not an energy minimizing one. The active contour is embedded as a level set in a suitable image evolution that is determined by a partial differential equation. On convergence the final contour is then extracted.
ϕ denotes the evolving level set, initially calculated as a signed distance map or using the *fast marching algorithm*(Sethian; 1999). t is the time step. g is the speed function, and is normally calculated as

$$g(I) = \frac{1}{1 + |\nabla I|^2} \quad , \quad g : \mathbb{R}^p \rightarrow [\, 0, 1 \,] \tag{2.4}$$

I being the input-image. So g will make sure the evolution stops when a boundary is reached, meaning that $|\nabla I|$ will yield high values, dragging g to zero. The term $\text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)$ is responsible for the regularizing effect of the model and plays the role of the internal energy term in the snake model Equation 2.2. The constant k is a correction ensuring that the "internal force" remains positive, so it can be interpreted as a force pushing the curve towards 0 when the curvature becomes zero or negative.
The main advantages of an implicit active contour over the classical explicit snake is the automatic handling of topological changes, high numerical stability and independence of parametrization. However, a main drawback is the additional computational complexity.

2.5.2 More Recent Deformable Models

(Caselles et al.; 1997) proposed the **Geodesic Active Contours** for improved performance over the geometric contour model. The basic idea is to see the boundary detection in a geometric model from another point of view. In Riemannian space with a metric derived from the image content, the boundary detection is equivalent to finding the geodesic curve (ie. curve of minimal weighted length). In Riemannian space, the contour model can thereby accurately track boundaries with high variation in their gradient,including small gaps which was difficult in the previous model. (Caselles et al.; 1997) and (Leventon; 2000) give a good derivation of the Geodesic Active Contour formulation given by Equation 2.5.

$$\frac{\partial\phi}{\partial t} = |\nabla\phi|\text{div}\left(g(I)\frac{\nabla\phi}{|\nabla\phi|}\right) + cg(I)|\nabla\phi| \tag{2.5}$$

The term $cg(I)|\nabla\phi|$ is a constant motion term and may help to avoid certain local minima, i.e. it works as the balloon force proposed in (Cohen; 1991).
To restrict the evolving curve (Leventon; 2000) introduces a shape regularization term in the force calculation, thus creating what is also seen in Figure 2.1 as the **geodesic active contour with shape prior**. In short terms, initial signed distance maps, Section 4.1, of training shapes created by an expert are aligned, Section 5.1 thus capturing the pose. A Principal Component Analysis, Section 3.4, is then performed on the aligned distance maps, extracting principal modes of variation and corresponding principal components. The pose and modes of variation are incorporated into Equation 2.5, thus yielding

$$\frac{\partial\phi}{\partial t} = \lambda_1\left(|\nabla\phi|\text{div}\left(g(I)\frac{\nabla\phi}{|\nabla\phi|}\right) + cg(I)|\nabla\phi|\right) + \lambda_2\left(u^* - u\right) \tag{2.6}$$

where u^* is the maximum a posteriori final curve controlled by λ_2, meaning this term incorporates information about the shape of the object being segmented.

So finally, the entire formulation in Equation 2.6 can be seen as a concatenation of the internal force regularizing the curve-smoothness, a balloon force which protects the curve against collapsing and finally the force that regularizes the shape form and position.
The final implicit deformable shape model listed in Figure 2.1 is the **region-based shape model** (Chan and Vese; 2001) (Tsai et al.; 2001) and (Tsai et al.; 2004). Inspired by (Mumford and Shah; 1989) and (Caselles et al.; 1997), a region-based attraction force has been developed in (Chan and Vese; 2001) to avoid some of the drawbacks of the edge-based attraction force. This is in particular when edges contain singularities and the SNR is very low, causing the speed function $g(|\nabla I|)$ to have a value greater than zero when the contour reaches an edge. A natural approach in this case would be to try and smooth the image first with a gaussian kernel, maybe in a scale-space fashion to avoid noise. This, however, might also fail, so a region-based approach has been developed to create another speed function for the evolving contour. This means that the evolving curve in (Chan and Vese; 2001) takes the form Equation 2.7.

$$\frac{\partial \phi}{\partial t} = \delta_\epsilon(\phi) \left[\mu \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \nu - \lambda_1 (u_0 - c_1)^2 + \lambda_2 (u_0 - c_2)^2 \right] \tag{2.7}$$

Again the evolving curve given by ϕ is defined by a set of forces, internal and external. As in the geometric snakes, the first term controls the smoothness of the curve. ν can be seen as a force controlling the area, i.e. it prevents the curve shrinking to a point, and finally the two last terms are measures of the variance of the inner and outer region of the contour. The last two terms will make the contour dwell in an area where both the variance of the inside and outside region is at a minimum, meaning one or two homogeneous areas.
(Tsai et al.; 2001) adopted this approach together with the shape prior approach created by (Leventon; 2000), and formulated an active contour driven by a region-based force and regularized by a shape term describing the shape of the desired object. Here the evolution of the curve embedded in a level set function is however redefined in a way which drastically reduces its degrees of freedom. The basic idea is to only allow the surface to deform in an affine and very constrained non-rigid way, controlled by the modes of principal components found in the shape model analysis. This means that the segmentation task actually can be seen as a registration task where the goal is to minimize the cost-function that describes the difference of the inner and outer region. In 2D the amount of parameters to optimize will then be four plus the amount of significant modes of variations found. This same approach was used in (Tsai et al.; 2004) in a setting of multiple shapes and another differentiation between regions. Based on the performance of this model, and the performance of the same model implemented in (Hansen; 2005), this might be a suitable model for segmentation of the MTL.
Tim Cootes has developed the **Active Shape Model(ASM)**(Cootes et al.; 1995) and **Active Appearance Model(AAM)**(T.F.Cootes et al.; 1998) which

essentially is a moving contour as described previously, only propagated using a different technique. Active shape and appearance models are very high-level techniques which use prior information based on the knowledge of expert landmarks. Like Leventon and Tsai's approaches, Active Shape Models builds a statistical model of a set of training data using PCA. A Procrustes algorithm is used to remove linearities from the training set before the statistical analysis is performed. However, the difference is that active shape models are based on landmarks instead of level set functions. This is an immediate drawback since it is often infeasible to manually annotate 3-dimensional shapes as described in Section 4.1. Having a mean shape, estimated from the aligned training-set, the pose parameters from the procrustes alignment and a set of modes of variation from the PCA, the shape model is set free in the image space to search for a similar object. This is done by sampling in the image along the normal of each point in the model, checking to see how well each sample fits to the model. When best samples have been found for each point, the parameters are updated, and thereby the curve is moved closer to a minimum. The AAM is an improvement of the active shape model, where the texture across the target object is included as information for the model fitting.
ASM and AAM have proven to perform very well on a wide range of segmentation tasks, not the least for MRIs of the human brain. An AAM algorithm would also be an interesting model for the segmentation of the MTL. These models are also referred to as deformable atlases.

2.6 Atlas Based Segmentation

Digital brain atlases are used very often for a multitude of tasks and have many applications in medical image analysis. Atlases take on many forms, ranging from an intensity image of the *average* subject to more detailed shape, intensity and functional models of specific structures. Atlases are used in basic research on population analysis, as guides in segmentation and seed point selection, as context in navigation tasks, and as models to overcome signal limitations and indistinct boundaries. Atlases may be based on a single individual or on a sample of a population. Atlases can be deterministic, where each region of space is associated with a single structure, or atlases can be probabilistic, where each region of space is assigned a likelihood of belonging to a variety of structures.
When atlases are constructed from a sample population, the imagery for the subjects in the sample are transformed into a common coordinate frame prior to consolidating their information. This step of *rooting* the atlas is common to both deterministic and probabilistic atlas construction. Establishing this common coordinate frame is a critical step that impacts the quality of the resulting atlas. A common practise is to select one subject from the sample on which to

base the atlas. But if the selected subject is far from the population mean, the resulting atlas will be biased towards this individual. This, in turn, can bias any inferences made with the atlas. Certainly a number of publications dealing with this problem have been published.
If a good atlas has been created, and it can be successfully matched to a novel scan, then all information present in the atlas (such as the tissue labels) is then known with respect to the image, achieving segmentation of the new scan. So actually, this is more of a image registration task than a segmentation task due to the fact that the difficult part lies in registering the novel scan to the atlas. A multitude of registration techniques exists for among other things warping novel brains to atlases which is an individual scientific area. This process is also known as **Deformation-based morphometry**
Boiled down to unfairly few words, the task of registering two brains starts by removing linearities using an affine warp to account for patient movements, brain size, scanner artifacts etc. leaving only biological differences. These differences are usually of main interest and are described by a non-rigid warp. Non-rigid registration is done using higher dimensional transformations, such as polynomial mappings or displacement fields. Other frequently used non-rigid registration techniques based on PDEs are:

- Elastic Registration
- Fluid Registration
- Diffusion Registration

Often a scale-space approach is used to avoid local minima, where a very coarse sub-sample of the image is first registered. When converged, a finer edition of the image is registered. This repeats until the top image converges to a minimum, hopefully being a global one.

2.7 Discussion

Having presented an overview of different segmentation methods this would also be a natural place to clarify the reason for selecting the method used for segmentation in this project. In order to do this, it should be noted that Section 3.3, Section 3.2 and Appendix B should be consulted for insight in the problem at hand. Furthermore the method will be justified in Section 4.3.
Due to the fact that the MTL resides deep within the brain the SNR is very low. Furthermore the contrast of different tissue-types is low and the boundaries of

the regions are not very well defined. The most distinct region is hippocampus, which is notorious for its difficulty of segmentation. Furthermore, these 12 regions are, in each of the hemispheres, very spatial closely located, in some places even adjacent. Naturally this immediately leads to think of a local search algorithm with a shape prior to regularize the evolution. (Tsai et al.; 2004) solved a similar problem with medical images of the pelvic area. Here, three indistinct regions where the goal of the segmentation, which is an inspiration to the problem faced here. Due to the close spatial relationship an indistinct properties, it is a natural thought to try and use the this spatial relationship as prior knowledge. Therefore it has been chosen to test a coupled shape model which will be described further in the next chapters. Furthermore it has been chosen to represent the shapes as level sets, given their nice properties, Section 4.1. A utilization of the AAM would likely also do a good task of segmentation, and the decision of the best of these methods should depend on a practical test rather than theory. The AAM has not been implemented in this project though. Normally a strong reason to use level sets is their ability to make topologic splits and merges. This property is not a motivation in the case of the MTL, as the regions neither split nor merge, but are closed surfaces.

Chapter 3

Background Theory

This chapter contains separate section describing different theory used in this thesis. Furthermore an introduction to the dataset is given here.

3.1 Basic Concepts of Magnetic Resonance Imaging and DRCMR

Here follows a presentation of the data acquisition site, and a short introduction to the concepts of Magnetic Resonance Imaging.

3.1.1 Danish Research Centre for Magnetic Resonance

The Danish Research centre for Magnetic Resonance(DRCMR)[1] was founded in 1985 following a donation of Denmark's first MR scanner to the centre by the Simon Spies Foundation.
The centre is responsible for providing a clinical diagnostic MR radiological service and is based at the MR unit in Hvidovre Hospital. In 2005, 3411 patients underwent MR investigations at the MR-centre. The centre is also involved in education both via direct teaching, undergraduate and PhD programs.
The centre's research activities are concentrated mainly on studies of brain diseases, such as stroke, dementia, epilepsy and multiple-sclerosis and has special expertise in the field of neuroradiology. The centre currently controls four different scanners:

- Siemens Trio 3.0 Tesla
- Siemens Vision 1.5 Tesla
- Siemens Impact 1.0 Tesla
- Varian Inova 4.7 Tesla

in which the Siemens Trio 3.0 Tesla was used to obtain the dataset for this thesis.

[1] http://www.drcmr.dk/

3.1.2 MRI in short

For deeper discussions of MRI and further references the reader is referred to (Hornak; 1996-2006).
Magnetic Resonance imaging(MRI) was first introduced in medical imaging in 1971 and was developed from knowledge gained in the study of nuclear magnetic resonance. This technology was first described independently by Felix Bloch and Edward Mills Purcell, in 1946, both of whom shared the Nobel Prize in physics in 1952 for their discovery.
Since its introduction to medical imaging in the early 1980's, the use of MRI has increased rapidly over the years. In 2003, there were approximately 10,000 MRI units worldwide, and approximately 75 million scans performed per year(Hornak; 1996-2006).
This excellent non-invasive imaging technique allows the anatomical study of the human brain, and together with techniques such as Single photon emission computed tomography (SPECT) and Positron emission tomography(PET), a non-invasive study of the activations within the brain, also known as functional-MRI (fMRI), can be performed. These techniques are being used in a vast amount of different studies, and as a diagnostic tool for various diseases. Whereas structural MRI scanning have a large place in medicine, fMRI and its brethren techniques are still largely devoted to neuroscience research, even though these techniques are becoming more common in medicine. Even though fMRI is used for the most part in the project surrounding this thesis, see Section 1.1, it is not used in this thesis since only the anatomy of the brain is of interest here. Therefore fMRI will no longer be mentioned.

3.1.3 The physics of MRI in short

Every proton, neutron and electron possess what is called a spin. This spin can be thought of as a magnetic moment vector, causing the particle to behave like a tiny magnet with a north and south pole. When the particle is placed in an external magnetic field, the spin vector of the particle aligns itself with the external field, just like a magnet would. When removed again from the electric field, the spin vector will fall back to a steady state, while emitting a small Radio Frequency(RF) signal. This natural behavior is exactly what the MR-scanner utilizes to create intensity images.
In order for the scanner to be able to register these small emitted pulses, an abundance of a certain type of atom must be present. Luckily such an atom is abundant in all biological systems, as these consists mainly of water, where the most frequent atom is hydrogen, which is also what the MR-scanner utilizes.

So, when a biological system, in which the spin of the atoms are naturally unaligned, enters a scanner, the atoms are affected by a strong static magnetic field(most current scanners operate at 1-3 Tesla [2]). The atoms are now aligned in parallel or anti-parallel direction, to the direction of the magnetic field. The atoms are not strictly aligned parallel to the magnetic field but at a small flip angle in which they precess around the magnetic field at a frequency called the Larmor frequency. As the atoms are kept in this state, another external frequency is pulsed at the Larmor frequency, perpendicular to the original field. This causes the atoms to precess away from the original field, and towards the volatile pulsed field momentarily. As this pulse stops, the atoms precess back to alignment of the original field, and pulses a small RF which is collected and used to produce an image.
When receiving the transmitted RF pulses, measurements are taken at important relaxation times called Time1 and Time2(T1 & T2). T1 is the time required for a certain percentage of the tissue's nuclei to realign. T2 is the decay of the RF signal after it has been created. These are also known as Longitudinal relaxation time and Transversal relaxation time respectively. Both these measures are tissue dependent which gives the MRI its ability to distinguish between different tissues in the body.
The image contrast is created by using a selection of image acquisition parameters that weights signal by T1 or T2, or no relaxation time ("proton-density images"). In the brain, T1-weighting causes the nerve connections of white matter(WM) to appear white, and the congregations of neurons of gray matter(GM) to appear gray, while cerebrospinal fluid(CSF) appears dark. The contrast of "white matter," "gray matter'" and "cerebrospinal fluid" is reversed using T2 imaging, whereas proton-weighted imaging provides little contrast in normal subjects.

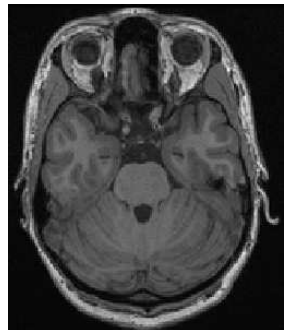

Figure 3.1: *T1 weighed image in sagittal aspect*

[2]SI derived unit of magnetic flux density (or magnetic inductivity), named after Nikola Tesla. Earth's magnetic field on the equator at a latitude of 0° is $3.1 \cdot 10^{-5} T$

3.2 An overview of the provided data

A set of 13 MRI-image volumes(3D), Figure 3.2, were supplied together with expert annotated masks, Figure 3.3, indicating 12 Volume Of Interests(VOI) per image, see Appendix B for more detailed plots. Each image is identified by a unique key[3] , and their Probability Density Functions(PDF) are found in Appendix D. Furthermore, binary masks with value 1 indicating GM voxels and value 0 indicating other tissue type was supplied.

The masks of the VOIs were drawn after a protocol developed by Thomas

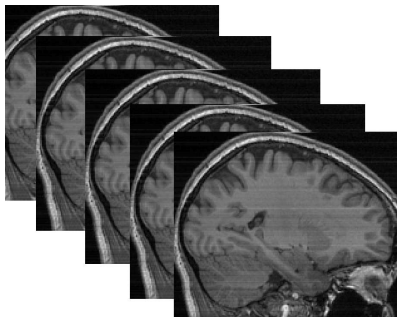

Figure 3.2: *Intensity images - Sagittal view*

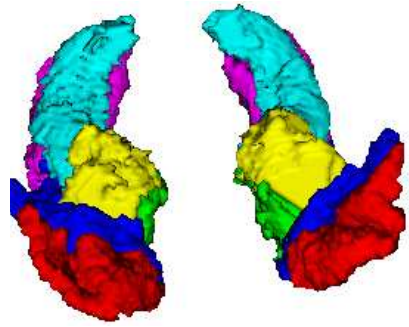

Figure 3.3: *Manual segmentation masked by GM masks - Coronal frontal view*

Zoëga Ramsøy. The reader is also referred to (Pruessner et al.; 2000), (Pruessner et al.; 2002) and (Insausti et al.; 1998) for some specifications of biological landmarks used to draw the masks.

Due to the fact that the precise localization of each region is very troublesome, the masks specifies the area of the neighborhood of each region. These areas should then be masked with the GM mask to get a precise indication of the areas.

The intensity volumes were as previously mentioned acquired with a Siemens Trio 3.0 Tesla, with the following parameters(See abbreviations in Section 1.2) Each volume has been corrected for non-uniformity artifacts using the Non-parametric Non-uniform intensity Normalization(N3) (Sled et al.; 17).

By investigating the intensity properties of the 13 MRI images it has become clear that the Probability Density Functions(PDF) varies some between the 13 intensity images. In Figure 3.4 the normalized mean and variance are seen for each of the 13 images. The first four images seem to differ significantly in both mean and variance from the rest. The intensity distributions in Appendix D

[3] *f1371,f1374,f1387,f1388,f1512,f1577,f1593,f1736,f1737,f1740,f1777,f1778,f1830*

Scanner	Siemens Magnetom Trio 3T MR scanner with an eight-channel head coil(Invivo, FL, USA)
Weighing	MPRAGE, T1 weighed images
Voxel dimension	$1 \times 1 \times 1$ mm.
FOV	256 mm.
Dimensions	$182 \times 218 \times 182$
TR/TE/TI	$1540/3,93/800$ ms
Flip-angle	$9°$

Table 3.1: *Scan specifications*

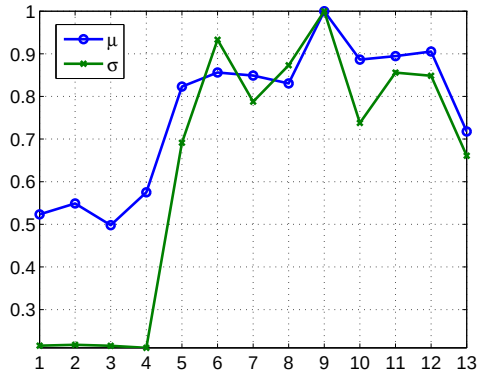

Figure 3.4: *Normalized gray-scale statistics, mean(blue), standard deviation(green), of the 13 images intensity images*

reveals that the first four images with lower mean are less heavy tailed than the rest, which is the reason for the low variance and mean. This difference in intensity distribution could cause problems if a method such as Mutual Information(MI), used in Section 5.3.4, is trained on probability distributions estimated across all images.
Surely there can be different causes for this difference, such as the size of the head, and other anatomic differences. It is however quite suspicious that the distributions seems to be grouped into two groups.
A way to make the distributions similar is to standardize the data i.e. $\mathbf{X}_s = \frac{\mathbf{X}-\mu}{\sigma}$. As the data will then be centered around the origin with unit variance, this would make the more heavy tailed distributions such as image 9 (f1737) more similar to image 1-4 as seen in Figure 3.5.
This, however, might be a bit dangerous because the precise cause of density difference is not known, thus it could degrade information in the data. Nevertheless the data have been standardized.

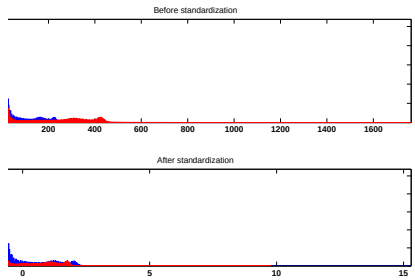

Figure 3.5: *Probability Distribution for image one(red) and nine(blue) before and after standardization*

3.3 The Medial Temporal Lobe

This section will mainly present the anatomic properties of the regions in the Medial Temporal Lobe(MTL) which is of special importance to this project. Furthermore it will give a short introduction to the anatomy of the human brain in general, the functional responsibilities of the MTL and how it is connected to the rest of the brain.
For a thorough description of the anatomy of the brain, the reader is referred to (Moos and Møller; 2006).
Seen from a top-down point of view, the brain consists of the *Prosencephalon*[4], the *Cerebellum*[5] and *Truncus encephalicus*[6]. The *Prosencephalon* is again split up into *Diencephalon*[7] and the *Cerebrum*[8]. The *Cerebrum* is subdivided into two hemispheres, where each of these are separated into four different lobes, *frontal*, *parietal*, *temporal* and *occipital* as seen in Figure 3.6 coded as yellow, green, purple and blue respectively. Each of these have areas of responsibilities and can of course be further subdivided into smaller regions, which is out of the scope of this thesis. However, the temporal lobe is of special interest here, which is why this region will be stressed a bit further.
Within the Temporal lobes are the MTLs (near the Sagittal plane that divides left and right cerebral hemispheres) which are thought to be involved in episodic/declarative memory. Deep inside the MTL, the hippocampus is of particular importance for the memory function - particularly transference from short to long term memory and control of spatial memory and behavior. Another central region in the MTL is the Amygdala which has been shown in research to perform a primary role in the processing and memory of emotional reactions such as fear and pleasure. Both of these structures interact with the cortical areas of the MTL, Temporopolar, Entorhinal, Perirhinal and Parahippocampal Cortices while performing memory tasks, and are all part of the limbic system[9]. The outermost surface of the Cerebrum is called the *Neocortex* and is known to establish connections with the MTL during memory processing to bind together the distributed storage sites in the neocortex that represent a whole memory. The role of these connections are only temporary and as time passes after learning, memory stored in neocortex gradually becomes independent of MTL structures, consequently forgotten.
The temporal lobe is shown in Brodmann's map(Brodmann; 1909) of the human brain in Figure 3.6 as purple. An exterior outline (the convex hull) of the regions of interest in the MTL can also be seen in Figure B.1 in Appendix B.

[4] Forhjernen
[5] Lillehjernen
[6] Hjernestammen
[7] Mellemhjernen
[8] Storhjernen
[9] Structures in the human brain involved in emotion, motivation, and emotional association with memory

The six before mentioned regions in the MTL are listed in table 3.2 together

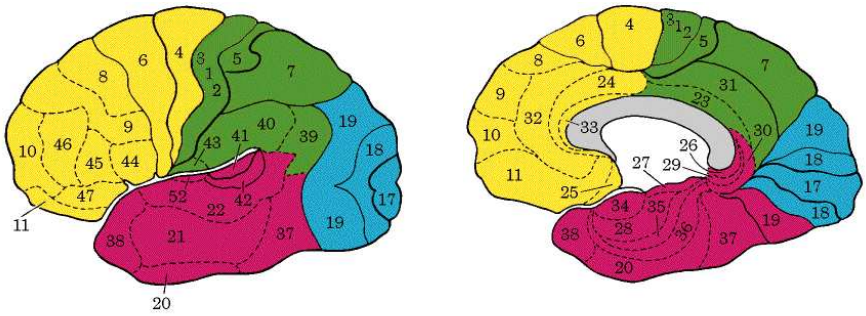

Figure 3.6: *Lateral(left) and Mid(right)-sagittal view of the human brain encoded in Brodmann's map of cytoarchitectonics*

with their abbreviation, corresponding location in the Brodmann model, color-code and a reference to a figure, located in Appendix B which shows a close-up of the region in three orthogonal views. The color-codes and abbreviations applied here will be used in the remainder of this thesis.

Region	Abb.	Figure 3.6	Color	Sample figures
Temporopolar Cortex	TPC	38	Red	B.2,B.3
Entorhinal Cortex	EC	28,34	Green	B.4,B.5
Perirhinal Cortex	PRC	35,36	Blue	B.6,B.7
Parahippocampal Cortex	PHC	27,36	Magenta	B.8,B.9
Hippocampus	HC	-	Cyan	B.10,B.11
Amygdala	AD	-	Yellow	B.12,B.13

Table 3.2: *Medial Temporal Lobe regions*

Furthermore the regions are shown in Figure 3.7 in volumetric views. These are screen-shots taken from an application developed for this project to explore the regions interactively.

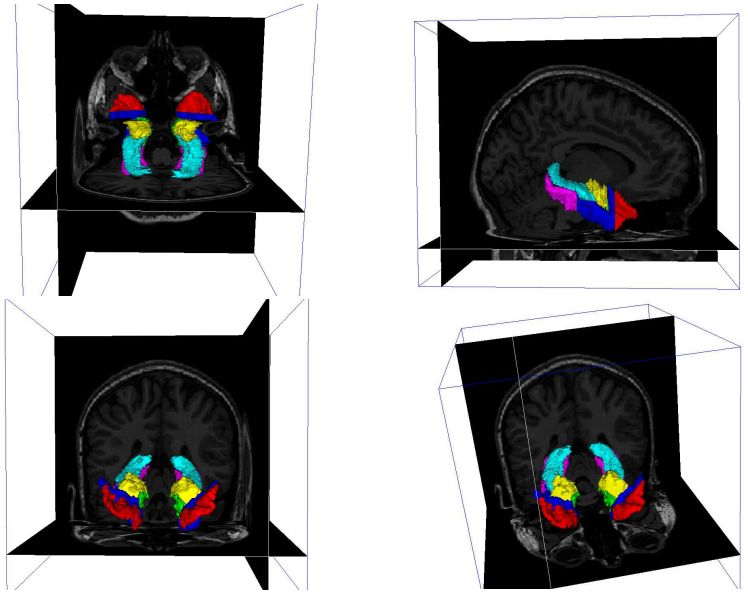

Figure 3.7: *A volumetric view of MTL.* ***Top left:*** *Axial view.* ***Top right:*** *Sagittal view* ***Bottom left:*** *Frontal Coronal view.* ***Bottom right:*** *Slanted view*

3.4 Principal Component Analysis

In high-dimensional data problems it can be difficult to immediately understand and visualize how the data behaves. To uncover the latent data structure, the eigenvalue problem of the dispersion matrix can be solved and analyzed to find the eigenvalues with the largest magnitude. By projecting the data linearly onto the calculated eigenvectors, a linear orthogonal transformation of the data is performed, which rotates the coordinate system in a way such that the axes of the system will correspond to the eigenvectors. The variation along the axes now represents the directions of most variation in the data in a descending order, the 1st. axis being the one representing most variance.
This technique is called Principal Component Analysis(PCA) and is one of the oldest and a very commonly used multivariate statistical technique, created initially by Karl Pearson in the beginning of the 19th. century and refined later by Harold Hotelling in 1933.
In the following the method is stated in a more formal setting.
When dealing with large amounts of data, the *central limit theorem* (Pitman; 1993) often justifies the use of the normal distribution to describe the density of data. This assumption will be used in the remainder of this thesis in order to use PCA.
Given a stochastic vector $\mathbf{X}$ Equation 3.1 consisting of n stochastic d-dimensional observations,

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \tag{3.1}$$

henceforth called the *covariate matrix*, and the data can be assumed gaussian, a model describing the data can be set up using the multivariate normal distribution. As the gaussian model is given by $\mathcal{N}(\mu, \sigma^2)$, it is necessary to estimate the mean and variation from the data. Thus, the gaussian model of $\mathbf{X}$ is

$$\mathbf{X} \sim \mathcal{N}\left(\widehat{\mu}, \widehat{\mathbf{\Sigma}}\right) = \mathcal{N}\left(\begin{bmatrix} \widehat{\mu_1} \\ \widehat{\mu_2} \\ \vdots \\ \widehat{\mu_n} \end{bmatrix}, \begin{bmatrix} \widehat{\sigma_{11}^2} & \widehat{\sigma_{12}^2} & \dots & \widehat{\sigma_{1n}^2} \\ \widehat{\sigma_{21}^2} & \widehat{\sigma_2^2} & \dots & \widehat{\sigma_{2n}^2} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{\sigma_{n1}^2} & \dots & \dots & \widehat{\sigma_n^2} \end{bmatrix}\right) \tag{3.2}$$

$\widehat{\mu}$ and $\widehat{\mathbf{\Sigma}}$ are estimated using a maximum likelihood estimates seen in Equation 3.3 and Equation 3.4.

$$\widehat{\mu}_{ML} = \mathbb{E}(\mathbf{X}) = \begin{bmatrix} \mathbb{E}(\mathbf{x})_1 \\ \mathbb{E}(\mathbf{x})_2 \\ \vdots \\ \mathbb{E}(\mathbf{x})_n \end{bmatrix} \tag{3.3}$$

$$\begin{aligned} \widehat{\mathbf{\Sigma}}_{ML} &= \mathbb{E}\left((\mathbf{X} - \mathbb{E}(\mathbf{X}))(\mathbf{X} - \mathbb{E}(\mathbf{X}))^T\right) \\ &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x_i} - \widehat{\mu})(\mathbf{x_i} - \widehat{\mu})^T \\ &= \frac{1}{n}\mathbf{X_c}\mathbf{X_c}^T \end{aligned} \tag{3.4}$$

$\mathbf{X_c}$ are the covariates $\mathbf{X}$ centered around the origin.
Both the covariance, also known as the dispersion, and the correlation matrix can be used for the PCA and for the remainder of this thesis, the PCA will be based on the covariance matrix.
There are different ways of obtaining the principal components of the covariates. Different ways of obtaining the principal components and modes can be used such as:

1. By solving the eigenvalue problem on the big covariance matrix Equation 3.4
2. By solving the eigenvalue of the smaller covariance matrix, Equation 3.5.
3. By doing a Singular Value Decomposition(SVD) on the centered data, Equation 3.7

When the dimensionality greatly exceeds the number of observations, the problem is ill posed and a number of eigenvalues will be zero. Furthermore the computational task of solving the eigensystem for huge matrices is infeasible. Therefore as done by (Leventon; 2000), solving for the smaller covariance matrix Equation 3.5 is faster and easier.

$$\begin{aligned} \widehat{\mathbf{\Sigma}_s}_{ML} &= \mathbb{E}\left((\mathbf{X} - \mathbb{E}(\mathbf{X}))^T(\mathbf{X} - \mathbb{E}(\mathbf{X}))\right) \\ &= \frac{1}{n}\mathbf{X_c}^T\mathbf{X_c} \end{aligned} \tag{3.5}$$

If $\mathbf{d}_i$ is an eigenvector of $\widehat{\mathbf{\Sigma}}_s$ with corresponding eigenvalue λ_i, it can be shown that $\mathbf{w}_i = \mathbf{X}_c\mathbf{d}_i$ is an eigenvector of Equation 3.4 with eigenvalue λ_i. This is

shown in (Leventon; 2000). Normalizing the length of the resulting components is necessary

$$\mathbf{w}_i = \frac{\mathbf{w}_i}{\|\mathbf{w}_i\|} \tag{3.6}$$

Another way of finding the principal modes of variation is by using a *Singular Value Decomposition*(SVD) (Hastie et al.; 2001).
The SVD is very general in the sense that it can be applied to any $m \times n$ matrix as opposed to the eigenvalue decomposition stated above. The SVD factorizes the covariate-matrix $\mathbf{X}$, as shown in Equation 3.7,

$$\mathbf{UDV}^T = \mathbf{X}_c \tag{3.7}$$

into $\mathbf{U}$ and $\mathbf{V}$, two orthogonal matrices of size $n \times n$ and $p \times p$ containing the principal components of $\mathbf{X}_c$, and $\mathbf{D}$ which is a diagonal matrix, with singular values on the diagonal $d_1 \geq d_2 \geq \ldots \geq d_p \geq 0$. The principal components contained in $\mathbf{V}$ are equal to those found in the eigenvalue-decomposition of $\widehat{\mathbf{\Sigma}}$ as seen in Equation 3.8.

$$\begin{aligned} \frac{1}{n}\mathbf{X_c}^T\mathbf{X_c} &= \frac{1}{n}\mathbf{V}\mathbf{D}^T\mathbf{U}^T\mathbf{U}\mathbf{D}\mathbf{V}^T = \frac{1}{n}\mathbf{V}\left(\mathbf{D}^T\mathbf{D}\right)\mathbf{V}^T \\ \frac{1}{n}\mathbf{X_c}\mathbf{X_c}^T &= \frac{1}{n}\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}^T\mathbf{U}^T = \frac{1}{n}\mathbf{U}\left(\mathbf{D}\mathbf{D}^T\right)\mathbf{U}^T \end{aligned} \tag{3.8}$$

where the relationship between the empirical, centered covariance matrix and the factorized matrices are shown. Furthermore the singular values of $\frac{1}{n}\underline{\underline{D}}$ corresponds to the square-roots of the first $\min(n,p)$ eigenvalues of $\widehat{\underline{\underline{\Sigma}}}$.
Finally when the principal modes has been calculated, a suitable amount of components has to be selected to get a good dimensionality reduction. This is done empirically in this thesis by setting a threshold for the amount of variance-explanation needed. Each component gives a contribution, and the amount of components, m, is chosen such that Equation 3.9 is fulfilled.

$$\frac{\sum_{i=1}^{m}\lambda_i}{\sum_{i=1}^{p}\lambda_i} \geq t \tag{3.9}$$

t is an empirical threshold value.
As a short example, data sampled from a bivariate normal distribution with an anisotropic dispersion structure is seen in Figure 3.8. It is obvious from this simple example, in which direction the data has highest variation. The above described PCA is performed on the data, and the two principal modes found are seen in Figure 3.8 centered around the origin, the principal axes being the principal modes of variation of the data. Figure 3.9 shows the data projected down on the principal components, which yields an isotropic data-cloud as all the variance was described by these principal components.

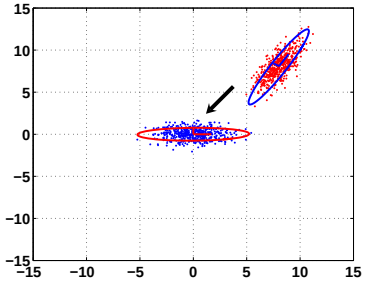

Figure 3.8: *Rotation of coordinate system*

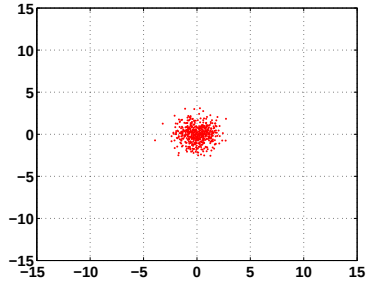

Figure 3.9: *Projection of data on principal axes*

3.5 Image Warping

Image Warping and image Registration are two closely connected terms. An image registration uses a *warp* together with a *similarity measure* to find a suitable spatial transformation such that one transformed image becomes as similar as possible to another, which can limited by a *regularization*.
A deep discussion of registration and warping techniques is not within the scope of this project, and the reader is referred to (Pluim et al.; 2003), (Gramkow; 1996) (Modersitzki; 2004) or (Glasbey and Mardia; 1998) for further discussion.
Usually, one of the images is viewed as the *Image* $\mathbf{I}$ and the other as a deformable *Template* $\mathbf{T}$. The warp operation

$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \mathcal{T}[p]x \tag{3.10}$$

maps the pixel $\mathbf{x}$ residing in the coordinate frame of $\mathbf{T}$, to a sub-pixel location $\tilde{x}$ in the coordinate frame of $\mathbf{I}$. This is done w.r.t. $\mathbf{p}$, using a given transformation $\mathcal{T}[\mathbf{p}]$ which also denotes the transformation type. In this thesis $\mathcal{T}[\mathbf{p}]$ is a composition of a set of linear transformations listed in Appendix E.
In practise, the warp is performed backwards, $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$, so that $\mathbf{I}$ is warped back into the coordinate frame of the template. This is due to the discrete nature of the images which means that when performing a forward transformation, it is not certain that an input pixel maps to an output pixel, thus causing holes in the new image. Especially when the template is smaller than the image, this is a very good way of moving the template around in the image. This is very well explained in (Baker and Matthews; 2002).
The *similarity measure* can be done in different ways, and is a part of a cost-function which should be minimized using appropriate warps, which like the warp-type should be chosen to suit the problem at hand.

3.6 Estimation of Probability Distribution Functions

This section briefly presents a non-parametric method of estimating a PDF based on a set of observations.
This is a well studied field, and many methods exists for estimating PDFs, both parametric and non-parametric. A parametric approach requires an assumption of the type of PDF the data is sampled from e.g. a gaussian distribution. The PDF is then estimated by calculating maximum likelihood estimates of the parameters in the assumed distribution using e.g. the Expectation Maximization algorithm.
Parametric density estimates are very precise and good if it is possible to make assumptions of the underlying PDF. Although very good, it is sometimes not possible to make this assumption and a non-parametric method is required instead. In this area techniques such as histograms, splines, clustering, kernel methods and other exists.
Kernel functions are widely used for PDF estimation, and these are essentially "just" an interpolation of data with a specific kernel κ weighing each observation.

$$\widehat{p}(x) = \frac{1}{n\overline{\omega}} \sum_{i=1}^{n} \kappa \frac{x - x_i}{\overline{\omega}} \tag{3.11}$$

in this project a gaussian kernel is used and the bandwidth $\overline{\omega}$ is chosen according to (Silverman; 1986) as were done in (Tsai et al.; 2004).

3.7 Validation and generalization methods

There are different ways of estimating the generalization/prediction error of a mathematical model. The generalization error is a quantity for how well a model generalizes to novel input data. The Akaike's and Bayesian information criteria are two methods. Another more simple, but widely used approach are the K-fold- or Leave-One-Out Cross Validation(LOOCV) methods in which the training set is split up into minor parts, and the model is trained without a certain part. This part is then presented to the model as novel input, and the error is calculated. (Hastie et al.; 2001) gives a good explanation of the LOOCV, which is suitable for smaller training sets such as the one in this project. Each entry in the data set is successive kept out of the training-phase, and used in the test-phase, and thereby obtaining a generalization error for each sample. The generalization error is then estimated normally as the squared sum of the individual generalization errors

$$\widehat{e}_{LOOCV} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2 \tag{3.12}$$

For calculating the correctness of a segmentation between the ground truth and a given segmentation result, the DICE measure, (Dice; 1945), which is a common measure is used. The DICE measure is merely a quantity describing the overlap of two binary sets.

$$\mathbf{DICE} = \frac{2\,(A \bigcap B)}{|A| + |B|} \tag{3.13}$$

where $|.|$ denotes the area of a set.

Chapter 4

Methods

The concrete methods used for the segmentation of the regions in the MTL are here described. Theory from previous chapter is used.

4.1 Medical Shapes

4.1.1 Representation of Medical Shapes

The representation of shapes in medical images has shown to be a important and challenging problem. One may broadly categorize shape representations into two categories; *explicit* and *implicit* which will be presented here with focus on a implicit representation.

4.1.1.1 Explicit representation

In an explicit framework, the shape is represented by a set of, usually connected, primitives(e.g. points, triangles etc.). To mention a couple of examples, (Cootes et al.; 1995) uses landmarks which are points in cartesian space, often created by an expert, and represents samples from a certain surface or contour. (Shenten et al.; 2002) used a dense surface mesh for representing certain brain regions. A drawback of the explicit representations is that there has to be correspondence of the descriptors across a population of shapes. So while locating these points in simple 2D structures can be done, but if the object of interest is a complex 3-dimensional structure it is infeasible for a human interactor to place the landmarks in a consistent manner. Methods have been developed to solve this problem in an automatic way by creating corresponding landmarks for the shape model using numerical optimization. Minimum Description Length (MDL) is a such method, and has been developed for some years now, beginning in 1998 with (Kotcheff and Taylor; 1998).

4.1.1.2 Implicit representation

In the implicit category, binary masks and the elegant level set functions(LSF) (Sethian; 1999) are used to describe anatomical structures. Often the signed distance maps(SDM) (Danielsson; 1980) realizes the level set function, which will be explained more thoroughly.
Discrete binary masks are normally created by an expert as were the case with

the landmarks. The binary shape should be understood as a division of the image domain Ω into two sets, the foreground $\mathcal{S}$ and the background $\overline{\mathcal{S}}$ as showed in Equation 4.1.

$$\begin{aligned} &\forall \mathbf{x} \in \Omega : \{0,1\}, \\ &\forall \mathbf{x} \in \mathcal{S} : \{1\}, \quad \Omega \text{ is the image domain} \\ &\forall \mathbf{x} \in \overline{\mathcal{S}} : \{0\} \end{aligned} \tag{4.1}$$

In contrast to the landmarks, this representation does not require correspondence across the set of shapes in the same manner, which makes it more feasible to annotate for an expert, even on more complex objects.
As mentioned, the SDM is often used as a realization of the LSF which have widespread use in different areas of research and production. LSFs are especially popular in the field of fluid simulation. The boundary of a binary shape can be embedded as a closed curve $\overrightarrow{C}$ in a SDM, $\mathbf{\Psi}$, defined through the transformation

$$\begin{aligned} L(\mathbf{x}) &= \min\left(d\left[\mathbf{x}, \overline{\mathcal{S}}\right]\right), \mathbf{x} \in \mathcal{S} \\ L(\mathbf{x}) &= \min\left(d\left[\mathbf{x}, \mathcal{S}\right]\right), \mathbf{x} \in \overline{\mathcal{S}} \end{aligned} \tag{4.2}$$

so that there are negative distances inside the curve and positive distances outside the curve. This implicit shape representation is the realization of a PDE, evolving from the shape boundary in all directions. Figure 4.1 shows a shape boundary represented in both binary and LSF form on a small regular lattice.

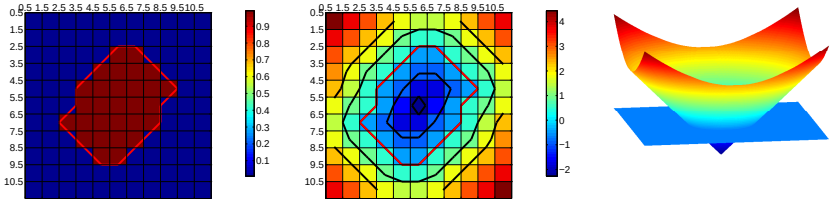

Figure 4.1: *Two different types of shape boundary representation, Binary(left) and Level Set Function(middle). The shape boundary is marked as red. Both images are* 11×11 *pixels. 3D view of LSF, zero Level Set is marked by the slab(right)*

Of course there are both advantages and disadvantages to this representation:

Advantages and Disadvantages of the LSF shape representation

+ Handles singularities in the boundary very well
+ Is able to track sharp corners and change in topology such as breakage and merging
+ Point correspondence is not required as in a point model
+ Provides simple linearization of binary shapes
÷ Is computationally more heavy than parametric curves
÷ Is difficult to adapt to new problems

When SDMs are used for shape representation this usually involves different computations on these, such as difference of SDMs, addition of SDMs etc. These operations has some problems which will be outlined in the next section.

4.1.2 Shape Modeling

Combining a set of the above mentioned shapes into a statistical shape model is a well studied field of research. Although statistical shape models of various representations can be built , this section will only concern shape modeling of shapes represented by LSF.
The usual approach when building a statistical shape-model is by performing PCA on a set of aligned LSFs, $\{\mathbf{\Psi}^1, \mathbf{\Psi}^2, \ldots, \mathbf{\Psi}^n\}$. This can be done for multiple shapes as well, which will result in a coupled shape model as proposed by (Tsai et al.; 2003). Considering a single shape first, the shape model or deformable atlas is after PCA defined by the eigenvector modes of variation $\{\mathbf{\Phi}^1, \mathbf{\Phi}^2, \ldots, \mathbf{\Phi}^n\}$, the eigenmodes $\mathbf{\Lambda}$ and the mean shape $\overline{\Phi}$. The q first modes of variation are selected, thus reducing the description space while maintaining the most significant variance directions. This deformable atlas is now able to encode novel shapes within that atlas space by varying the parameter $\mathbf{w}$

$$\mathbf{\Phi}_{new}[\mathbf{w}] = \overline{\mathbf{\Phi}} + \sum_{i=1}^{q} w_i \lambda_i \mathbf{\Phi}_i \tag{4.3}$$

4.1.2.1 Issues with SDM-shapemodels

Although very elegant, the encoding of new shapes in Equation 4.7 has some issues as noted by (Tsai et al.; 2003) and (Golland et al.; 2005), which one should at least be aware of before using a deformable atlas. Linear operations are not closed under the set set of SDMs. This means that for example that the addition of two SDMs generally does not lead to an SDM. Therefore the shape model produces incorrect LSFs, which means that the produced shapes might not be real samples of the deformable atlas space. This issue leads to some concern which can be dealt with by further approximation, necessitating an additional computational step. (Golland et al.; 2005) deals with this by projecting novel shapes back onto the manifold of valid SDMs. (Hansen; 2005) made a process called *Shape reinitialization* to approximate the true SDM. At *MICCAI 06* an interesting approach to solve this problem called *Logarithm Odds Maps* (Pohl et al.; 2006) was presented. This approach deals with the so-called Log-Odds space $\mathbb{L}$, which embeds embeds SDMs and relates them to *probalistic atlases*[1] using a *logistic*[2] function. This is in fact a new shape representation called a Space Conditioned Probability map(SCP) which assigns a probability to each voxel, describing its relationship to the shape. The foreground of the shape is represented by probabilities greater than 0.5, while the background is represented by probabilities less than 0.5. Unfortunately due to the fact that it is a probalistic atlas it has to be used together with a probalistic segmentation method e.g. as proposed in (Pohl et al.; 2005).
The segmentation method chosen in this project is not based on probabilities, but rather on the nature of LSFs, which means that the problem of inaccurate SDMs has been dealt with by simply recalculating the SDM based on the extracted zero level set curve $\overrightarrow{C}$ encoded in a new shape. There is no doubt that this is a very simple "solution" and it causes additional computational overhead which should be avoided.

[1] shape defined by probabilities. The value of each node defines how probable it is that it belongs to the foreground.

[2] $\mathcal{P}(t) = \frac{1}{1+e^{-t}}$

4.1.2.2 A strongly coupled model

Considering a set of multiple aligned shapes represented as SDMs as seen in Equation 4.4

$$\begin{bmatrix} \boldsymbol{\Psi}_1^1 & \boldsymbol{\Psi}_1^2 & \cdots & \boldsymbol{\Psi}_1^m \\ \boldsymbol{\Psi}_2^1 & \boldsymbol{\Psi}_2^2 & \cdots & \boldsymbol{\Psi}_2^m \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\Psi}_n^1 & \boldsymbol{\Psi}_n^2 & \cdots & \boldsymbol{\Psi}_n^m \end{bmatrix} \tag{4.4}$$

together with the mean SDMs, $\overline{\boldsymbol{\Psi}}^k$ for each shape class, a covariate matrix can be created. The mean SDMs can be considered as shape classes (Turk and Pentland; 1991), $\overline{\boldsymbol{\Phi}}^k$ which represents the common features across all subjects. What is desired in this model is to capture the differences in and between shapes, meaning each subjects difference from the mean class. Therefore a set of mean-offset functions are calculated as

$$\tilde{\boldsymbol{\Psi}}_i^k = \boldsymbol{\Psi}_i^k - \boldsymbol{\Phi}^k \tag{4.5}$$

which are used for the data analysis. Due to the fact that each shape is represented as a level set this gives some problems when creating the shape model. As the boundary of the shape propagates as a function of time, variations on surface of the shape may drown in the marching front as shown in Figure 4.2, thus biasing the true variation in the shape model.

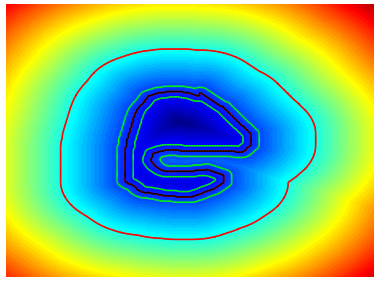

Figure 4.2: *Propagation of a level set function for a given boundary. The black contour is the zero iso-line, the green iso-line represents the ±5 level set curve and the red represents the 30th level set curve.*

It is seen that the kink in the lower part of the shape is present at t=5, but at t=30 the kink is no longer present. Therefore it would be desirable to base the shape model on a narrowband and thereby avoid "drowning" of important information. Furthermore this gives less data which obviously will lead to less computations.

Narrowbands are extracted from the mean-offset functions and arranged as column vectors $\tilde{\psi}_i^k$, stacked so that a single column in the covariate matrix represents the narrowbands of all shapes in a single subject, and a single row represents a single observation across all subjects as illustrated in Equation 4.6. Of course there has to be correspondence of the narrow bands across the subjects.

$$X = \begin{bmatrix} \tilde{\psi}_1^1 & \tilde{\psi}_2^1 & \cdots & \tilde{\psi}_n^1 \\ \tilde{\psi}_1^2 & \tilde{\psi}_2^2 & \cdots & \tilde{\psi}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\psi}_1^m & \tilde{\psi}_2^m & \cdots & \tilde{\psi}_n^m \end{bmatrix} \tag{4.6}$$

A PCA on the small $n \times n$ covariance matrix is performed as described in Section 3.4. The most significant principal components q are chosen so that $q < n$ according to Equation 3.9. As done in (Turk and Pentland; 1991), these are reshaped back into volumes thus creating a set of eigenshapes $\mathbf{\Phi_i^k}$.
Combining these eigenshapes with the shape classes, $\overline{\mathbf{\Phi}}_i$, renders us able to generate m novel shapes simultaneously

$$\begin{aligned} \Phi^1[\mathbf{w}] &= \overline{\mathbf{\Phi}}^1 + \sum_{i=1}^{q} w_i \lambda_i^1 \mathbf{\Phi}_i^1 \\ \Phi^2[\mathbf{w}] &= \overline{\mathbf{\Phi}}^2 + \sum_{i=1}^{q} w_i \lambda_i^2 \mathbf{\Phi}_i^2 \\ &\vdots \\ \Phi^m[\mathbf{w}] &= \overline{\mathbf{\Phi}}^m + \sum_{i=1}^{q} w_i \lambda_i^m \mathbf{\Phi}_i^m \end{aligned} \tag{4.7}$$

where $\mathbf{w}$ is a set of weights used to control the degree of influence each principal component has on the final eigenshape.
Now being able to create a set of new shapes by varying the parameter $\mathbf{w}$ it is also necessary to incorporate pose in the shape model to be able to handle a larger dispersion of shapes. The incorporation of pose in a single shape class is seen in Equation 4.8

$$\mathbf{\Phi}^1[\mathbf{w}, \mathbf{p}](\mathbf{x}) = \overline{\mathbf{\Phi}}(\mathbf{W}(\mathbf{x}; \mathbf{p}))^1 + \sum_{i=1}^{q} w_i \lambda_i^1 \mathbf{\Phi}_i^1 \tag{4.8}$$

where $\mathbf{W}(\mathbf{x}; \mathbf{p})$ denotes a usual warp as described in Section 3.5. This means that for each shape a set of parameters $\mathbf{w}$ and $\mathbf{p}$ are used to describe the form of a specific novel shape.

4.1.3 Extraction of the shape boundary

Working with LSFs, requires a method to extract $\overrightarrow{C}$ from the LSF. This task is very common in e.g. medical imaging, where iso-surfaces often are extracted from various patient scans to make visualizations or the like. Algorithms has been developed to find such iso-curve/surfaces, and a very popular method for finding these is *the marching cubes algorithm*, (Wyvill et al.; 1986), which is used in this project to find curves and surfaces.

4.2 Shape Alignment

The initial step of building a shape prior for a segmentation algorithm is the spatial alignment of the training set to create a common coordinate frame for all the shapes. This filtering of linearities is performed to obtain the shapes as given by *David George Kendall* (Dryden and Mardia; 1998):

> *Shape is the geometrical information that remains when location, scale and rotational effects are filtered out from an object.*

Aligning shapes is essentially a registration task as described in Section 3.5, which means that the warping theory will be used in this chapter. This also means that the alignment can be performed in a number of ways using different transformation types and similarity measures depending on what the result should be. For the purpose of shape alignment here, a rigid body transformation is chosen. Since this means only filtering out translational and rotational effects it is not entirely coherent with the definition of shapes given by Kendall above, which also states that the scale of the objects should be filtered out. However, it is the belief of the author that scaling could be a biological factor which the shape model should be able to incorporate, and therefore a rigid body transformation has been chosen. Furthermore, leaving scale out of the warping process, avoids the process of recomputing the distance map after each transformation as mentioned in Section 4.1.

Aligning the shapes is here considered a two-fold objective. Because shapes in theory could be quite dispersed, an initial crude alignment is necessary. The crude alignment chosen here, registration by moments, is a nice and very computationally efficient approach and is performed on binary masks. After the initial alignment a more accurate alignment is necessary. This alignment is performed using LSF representation of the shapes, Section 4.1, for improved matching and speed. This step is a common registration task which will be presented in Section 4.2.2.

4.2.1 Registration By Moments

A registration by moments of the shapes will be used to recover a rigid transformation and create a qualified first guess for all shapes, even if they have large spatial dispersion.

The moments of an image describes certain characteristics. Especially the first two order moments, the center of mass and the spatial dispersion matrix respectively describes the translational and rotational relationship, which is valuable information when making spatial inference on a object. The first order moment in 3D is defined in Equation 4.9

$$\mathbf{c} = \frac{\sum_{\mathbf{x}}^{N} \mathbf{I}(\mathbf{x})\mathbf{x}}{\sum_{\mathbf{x}}^{N} \mathbf{I}(\mathbf{x})}, \mathbf{x} = \begin{bmatrix} i \\ j \\ k \end{bmatrix} \tag{4.9}$$

and the second order moment is defined in Equation 4.10

$$\mathbf{S} = \frac{\sum_{\mathbf{x}}^{N} I(\mathbf{x})(\mathbf{x}-\mathbf{c})(\mathbf{x}-\mathbf{c})^T}{\sum_{\mathbf{x}}^{N} \mathbf{I}(\mathbf{x})} \tag{4.10}$$

As shown in (Gramkow; 1996), if two images $\mathbf{T}$ and $\mathbf{I}$ are characterized by their first order moments $\mathbf{c_t}$ and $\mathbf{c_i}$ and second order moments $\mathbf{S_T}$ and $\mathbf{S_I}$, the rigid transformation

$$\mathbf{y} = \mathbf{R}\mathbf{x} + \mathbf{t} \tag{4.11}$$

which relates $\mathbf{T}$ to $\mathbf{I}$ by transforming the moments of $\mathbf{T}$ is given by

$$\begin{aligned} \mathbf{R} &= \mathbf{U_I}{\mathbf{U_T}}^{-1} \\ \mathbf{t} &= \mathbf{c_i} - \mathbf{c_t} \end{aligned} \tag{4.12}$$

$\mathbf{U_T}$ and $\mathbf{U_I}$ denotes the eigenvectors of the spatial dispersion matrices $\mathbf{S_T}$ and $\mathbf{S_I}$, which can be factorized by

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}^{1/2}\left(\mathbf{U}\mathbf{\Lambda}^{1/2}\right)^T \tag{4.13}$$

where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues sorted in a descending order. The proof has been left out, and the reader is referred to (Gramkow; 1996) for further information. The eigenvalues, $\mathbf{\Lambda}$ are on purpose not used in the transformation due to the fact that scaling is a variation that the shape model should have possibility of incorporating, and therefore it is not desirable to remove this linearity now. If scaling was a factor to be removed as well, the calculation of the rigid transformation would be Equation 4.14

$$\begin{aligned} \mathbf{R} &= \mathbf{U_I}{\mathbf{\Lambda_i}}^{1/2}{\mathbf{\Lambda_T}}^{-1/2}{\mathbf{U_T}}^{-1} \\ \mathbf{t} &= \mathbf{c_i} - \mathbf{c_t} \end{aligned} \tag{4.14}$$

In Figure 4.3 two artificial shapes are shown together with their moments. The gross alignment approximations seems to give a qualified start guess for a more

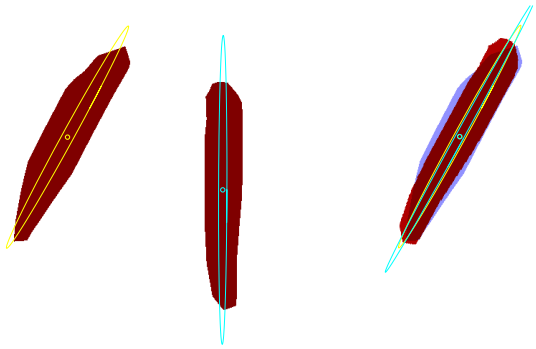

Figure 4.3: *Synthetic shapes rigidly aligned by moments up to order two*

sensitive algorithm.

Since this method is a rather simple approach, and "only" aligns the principal axes of the shapes, it can give "wrong" answers. Certainly the task of aligning the principal axes of two shapes is an ill-posed problem, so if the shapes are spatially different enough as seen in Figure 4.4 there is a high possibility of a "wrong" answer. Here the principal axes and center of mass have been aligned

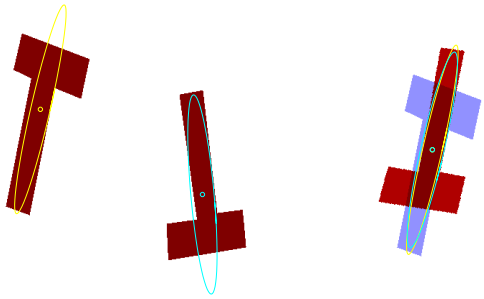

Figure 4.4: *A rather unlikely shape configuration causing a wrong alignment answer.*

correct, but the initial guess found makes a correct alignment impossible for a more finely tuned algorithm since the shape is stuck in a local minimum, and therefor the answer is wrong. Fortunately this is an extreme case which is rather unlikely to happen.

4.2.2 Fine tuning the shape alignment

The fine-tuning of the shape alignment can be done in more or less complex ways. Before setting up a registration framework it is necessary to define what sort of images must be registered. Figure 4.5, 4.6 and 4.7 shows a profile taken from a square shaped 2D object. In the figures, the shape is represented as a binary shape, a binary shape smoothed by a gaussian kernel and a SDM. The blue signal illustrates the deformable template $\mathbf{T}$ and the green illustrates the image $\mathbf{I}$. Since most fast optimization schemes uses gradient information, the binary

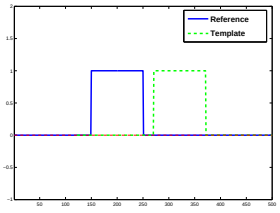

Figure 4.5: *Profile of binary image*

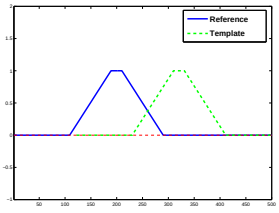

Figure 4.6: *Profile of smoothed binary image*

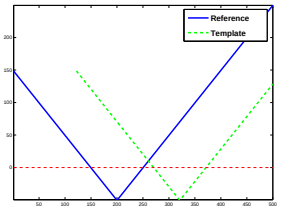

Figure 4.7: *Profile of LSF*

shapes seems to pose a problem in this regard, as they lack gradient information. (Tsai et al.; 2001) overcame this problem by calculating a maximum overlap criterium of all the shapes jointly. Because he did this he also overcame the problem of the small area of gradient information. Another way to help this problem is by creating artificial gradient information. This could be done by blurring the edges with a gaussian filter, Figure 4.6, although this still has limited gradient information. Finally it seems that LSFs have a lot of gradient information everywhere. This could mean that a registration of two LSFs should be able to converge rather fast if a good cost function was to be chosen.
A good and simple similarity measure is the Sum of Square Difference(SSD) which has proved to be a robust measure, Equation 4.15.

$$E_{SSD}(\mathbf{x},\mathbf{p}) = \frac{1}{2}\sum_{\mathbf{x}} \left(I(\mathbf{W}(\mathbf{x};\mathbf{p})) - T(\mathbf{x})\right)^2 = \frac{1}{2}(I(\mathbf{W}(\mathbf{x};\mathbf{p}))-T)^T(I(\mathbf{W}(\mathbf{x};\mathbf{p}))-T) \tag{4.15}$$

This is commonly used for measuring difference in e.g. grayscale images, but could also be used to measure difference between LSF's, enabling us to align these.
As it was chosen in the beginning of this chapter to leave out scaling from the alignment process, this poses a problem for binary shapes using a maximum overlap criterion. When two objects are of different size, and the smaller is nested within the bigger, there will ostensibly be numerous optima, making it an extremely ill-posed problem. An example of this is seen in Figure 4.8

to 4.10. Here the template shown in red slides from right to left in intervals of 1, calculating the energy at each step. When using binary representation, Figure 4.9 it is seen that there is optimum everywhere the template is completely encapsulated in the reference shape. When using the level set representation and SSD similarity measure, it is obvious that it is a convex function with minimum in the center of the shape as wanted. The optimization should optimize the

Figure 4.8: **I** *is the green shape,* **T** *is the red.*

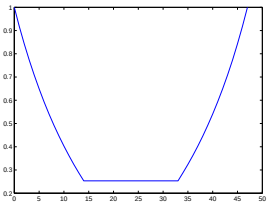

Figure 4.9: *Maximum Overlap energy, Binary representation*

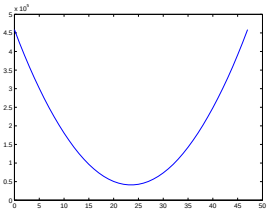

Figure 4.10: *SSD energy, Level set representation*

pose-parameters $\mathbf{p}$ which constitutes the rigid body transformation

$$\mathbf{p} = \begin{bmatrix} R_x & R_y & R_z & T_x & T_y & T_z \end{bmatrix} \tag{4.16}$$

The derivative of the SSD is calculated as a composition of the difference image, the numerical gradient of the warped template and the euclidian jacobian for the pose parameter,see also (Modersitzki; 2004), (Tsai et al.; 2003) or (Hansen; 2005)

$$\begin{aligned} \nabla_p^i E_{SSD} &= (\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p})) - \mathbf{T}) \\ &\begin{bmatrix} \frac{\partial \mathbf{T}(\tilde{x},\tilde{y},\tilde{z})}{\partial \tilde{x}} & \frac{\partial \mathbf{T}(\tilde{x},\tilde{y},\tilde{z})}{\partial \tilde{y}} & \frac{\partial \mathbf{T}(\tilde{x},\tilde{y},\tilde{z})}{\partial \tilde{z}} & 0 \end{bmatrix} \frac{\partial \mathcal{T}[\mathbf{p}]}{\partial \mathbf{p}^i} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \\ &\tilde{\mathbf{x}} = \mathbf{W}(\mathbf{x};\mathbf{p}) \end{aligned} \tag{4.17}$$

where i is the i'th component of the rigid warp transformation matrices listed in Appendix E. Together, these form the Jacobian which should be used in the optimization scheme. An efficient optimization schemes such as the Gauss-Newton is often utilized in image registration when using SSD similarity. The Gauss-Newton scheme is a nonlinear least squares numerical optimization method based on implemented first order derivatives of the components of a vector function $\mathbf{r}(\mathbf{x})$ and can achieve quadratic convergence in special cases. The Gauss-Newton method is looking for a local minimizer $\mathbf{p}^*$ for the cost-function defined in Equation 4.15,

$$\mathbf{r}(\mathbf{x};\mathbf{p}) = (\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p})) - \mathbf{T}) \tag{4.18}$$

where the images has been arranged as vectors. So the approximation for $\mathbf{r}(\mathbf{p}+\Delta\mathbf{p})$ is given as

$$\mathbf{r}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p}) = \mathbf{r}(\mathbf{p}) + \mathbf{J}\ \Delta\mathbf{p} \tag{4.19}$$

where $\mathbf{J}$ is the Jacobian of $\mathbf{r}(\mathbf{p})$. Inserting Equation 4.19 into Equation 4.15 yields

$$E_{SSD}(x;\mathbf{p}+\Delta\mathbf{p}) \simeq L(\Delta\mathbf{p}) = E_{SSD}(\mathbf{x};\mathbf{p}) + \Delta\mathbf{p}\ \mathbf{J}^T\mathbf{r}(\mathbf{x};\mathbf{p}) + \frac{1}{2}\Delta\mathbf{p}\ \mathbf{J}^T\mathbf{J}\mathbf{p} \tag{4.20}$$

Finding the increment $\Delta\mathbf{p}$ is done by minimizing $L(\Delta\mathbf{p})$, where

$$\nabla L(\Delta\mathbf{p}) = \mathbf{J}^T\ \mathbf{r}(\mathbf{x};\mathbf{p}) + \mathbf{J}^T\mathbf{J}\Delta\mathbf{p} \tag{4.21}$$

where $\nabla^2 L(\Delta\mathbf{p}) = \mathbf{J}^T\mathbf{J}$ ie. the Hessian, a positive definite matrix. The gradient is zero at a minimizer for L, so the Gauss-Newton step can be found as

$$\Delta_{gn} = -\nabla^2 L(\Delta\mathbf{p})^{-1}\mathbf{J}^T\ \mathbf{r}(\mathbf{x};\mathbf{p}) \tag{4.22}$$

and the parameters can be updated accordingly with a damping coefficient α determining the step-length

$$\mathbf{p} = \mathbf{p} + \alpha\Delta\mathbf{p} \tag{4.23}$$

α can be determined with a linesearch algorithm.
As a working example in $\mathbb{R}^2$, two circle formed shapes are registered in Figure 4.11.

4.2.3 Aligning all shapes

Having the basic steps of how to align two shapes , a setup for the entire alignment process can be set up. To align a set of shapes to create a common coordinate-frame, a good way is by using a procrustes approach. The steps in the procrustes approach is outlined below, (Cootes and Taylor; 2004):

1. Choose one example as an initial estimate of the mean shape.
2. Record the first estimate as $\overline{x}_0$ to define the default reference frame.
3. Align all the shapes with the current estimate of the mean shape
4. Re-estimate mean from aligned shapes.
5. Apply constraints on the current estimate of the mean by aligning it with $\overline{x}_0$

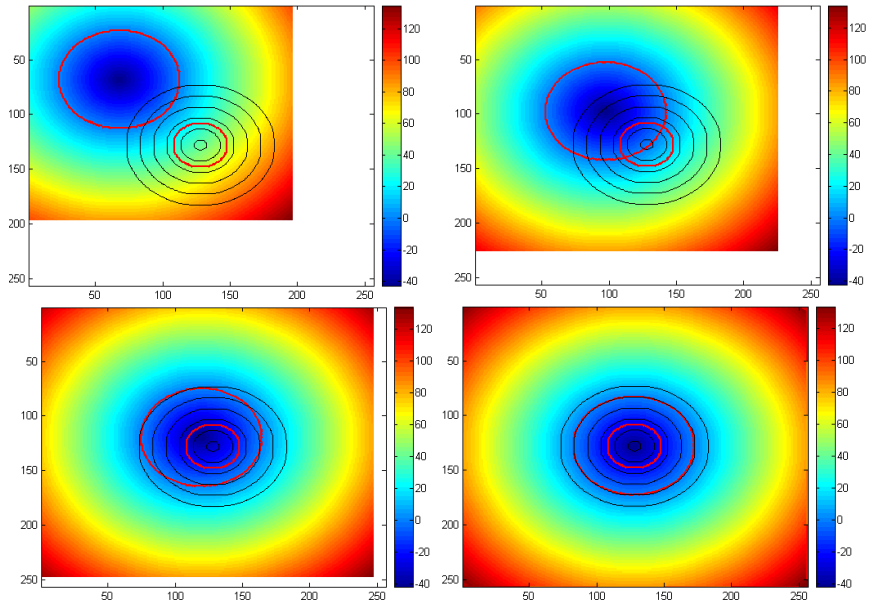

Figure 4.11: *The process of aligning two LSFs. The contour-lines is the template image and the underlying LSF is the image which the template is registered to. From left up to bottom right the process is seen in its state after 0, 12, 28 and 65 iterations respectively.*

6. If not converged, return to 3. Convergence is declared when the the mean does not change significantly, and is set empirically

The Procrustes algorithm has been developed for aligning instances of a single shape. A joint alignment of LSF could also be performed instead, like performed on binary shapes in (Tsai et al.; 2004). This would undoubtedly yield a different result since the difference of all shapes would here be taken into consideration.

Finally this chapter concerns the concrete segmentation of the MTL.

4.3 An Active Implicit Parametric Shape Model

This section describes the theory behind a region-based segmentation technique which will be used in a later section. The shapes spawned by the deformable atlas described in Section 4.1 defines a segmenting surface which spans multiple regions. This segmenting surface should be fitted in image space by minimizing some energy functional with respect to the pose and deformation parameters presented in Section 4.1.2.2.

4.3.1 Why not an edge-driven model?

As presented in Chapter 2 there are different ways of driving a contour in the image space towards certain interesting areas such as specific boundaries or regions. Here a region based approach is explored and utilized due to the nature of the regions having near to no boundaries, and intensities only slightly differing from its surroundings, i.e. very low contrast.
This is however an easy allegation to make, and to justify this, the driving term in Equation 2.4 should be analyzed.

$$g(I) = \frac{1}{1 + |\nabla I|^2} \quad , \quad g : \mathbb{R}^p \rightarrow [\ 0, 1\ ] \tag{4.24}$$

∇I denotes the gradient of the image, and is a force pulling the contour towards boundaries. This means that the gradient of the image space is extremely decisive for the performance of the segmentation. Figure 4.12 shows a sagittal slice of the scan of test person f1512, together with the corresponding gradient image - (∇I). The contour of the regions of interests are color-coded according to Table 3.2.
It should be noticed that even though this is merely a sub-sample of a single slice in the entire volume, it generally represents the boundaries of the regions quite well in all three dimensions.
The direction of the arrows in the gradient image shows in which direction the contour would be pulled, and the length of the arrows shows the strength of the pull. There is no significant attraction capabilities seen near either of the region boundaries, witnessing that an edge-based approach would yield lesser results. However, in the adjacent corresponding intensity image, a difference in the texture is relatively clear when comparing the inside of the regions with the outside. It is seen that the inside of the regions consists of GM and the outside

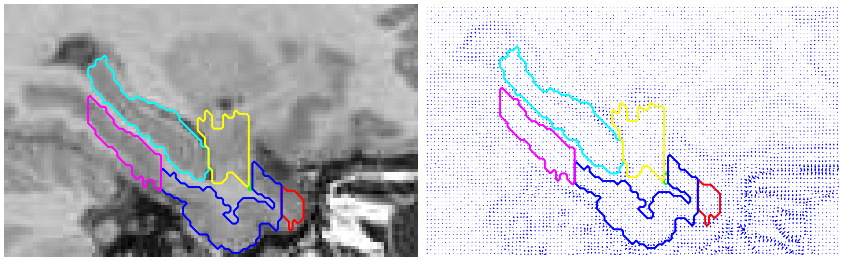

Figure 4.12: *Sagittal zoom of MTL. Left: Intensity image. Right:* ∇I*, the gradient image*

of among other things WM. The boundary between each region is however a bit unclear, which might cause some trouble. We will return to the properties of these regions in a short while. A region-based driving force might be able to detect this difference in the texture by comparing the intensity distribution of the inside and outside of the shape as done in (Kim et al.; 2002).

4.3.2 Exploring a Region-Based model

Having decided upon using a region-based model, a couple of definitions needs to be made before proceeding.
As described in Section 4.1, the closed segmenting or active surface, denoted $\overrightarrow{C}$ is embedded in a LSF as the zero-level set, Equation 4.25

$$\overrightarrow{C} = \left\{(\mathbf{x}) \in \mathbb{R}^N : \mathbf{\Phi}[\mathbf{w}, \mathbf{p}](\mathbf{x}) = 0\right\} \tag{4.25}$$

The region encapsulated by this closed surface is denoted as the inner region R^i, while the surrounding region is denoted as the outer region R^o, Equation 4.26

$$\begin{aligned} R^i &= \left\{(\mathbf{x}) \in \mathbb{R}^N : \mathbf{\Phi}[\mathbf{w}, \mathbf{p}](\mathbf{x}) \leq 0\right\} \\ R^o &= \{(\mathbf{x}) \in \mathbb{R}^N : \mathbf{\Phi}[\mathbf{w}, \mathbf{p}](\mathbf{x}) > 0 \ \wedge \ \mathbf{\Phi}[\mathbf{w}, \mathbf{p}](\mathbf{x}) < r\} \end{aligned} \tag{4.26}$$

The width of the outer region r is chosen empirically to optimize the performance of the segmenter. To get a more intuitive idea of the regions, they have been plotted in Figure 4.13. To recapitulate the properties of the above-mentioned regions in image-space Figure 4.14, 4.15 and 4.16 shows the right-side regions of subject f1512 with an outer region of $r = 10$.
The regions have been plotted in three different images for the sake of clarity. The reader should notice how indistinguishable the regions are, and their proximity to each other. As described in Figure 4.14, and seen in all three figures,

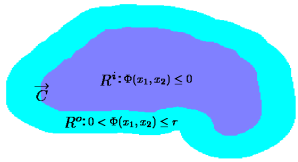

Figure 4.13: *Simple illustration of the inner and outer region of a shape and the shape boundary as defined in Equation 4.25 and Equation 4.26*

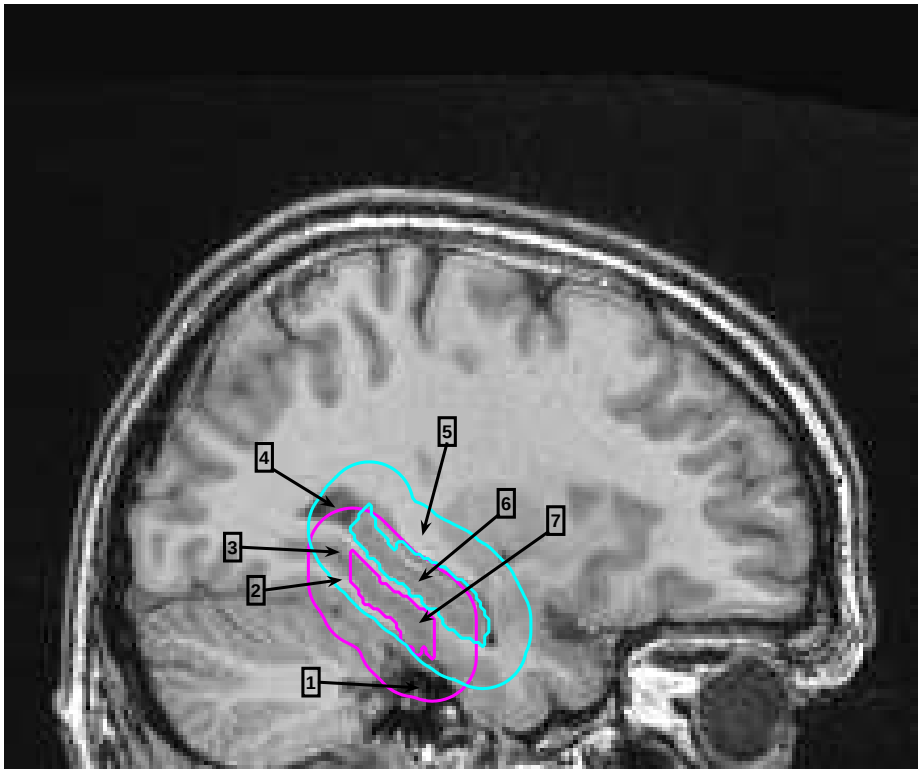

Figure 4.14: *Hippocampus and parahippocampal cortex shown with inner and outer regions, $r = 10$. The outer regions are seen to contain several brain tissue types, while the inner regions only contains GM.* ***1.*** *Parahippocampal outer - CSF* ***2.*** *Parahippocampal outer - WM* ***3.*** *Parahippocampal outer - GM and Hippocampus outer -GM* ***4.*** *Hippocampus outer -GM/CSF* ***5.*** *Hippocampus outer -WM* ***6.*** *Hippocampus inner -GM* ***7.*** *Parahippocampal inner -GM*

all the inner regions consists mainly of GM tissue. The outer regions on the other hand consists of a variety of different tissues as indicated in Figure 4.14,

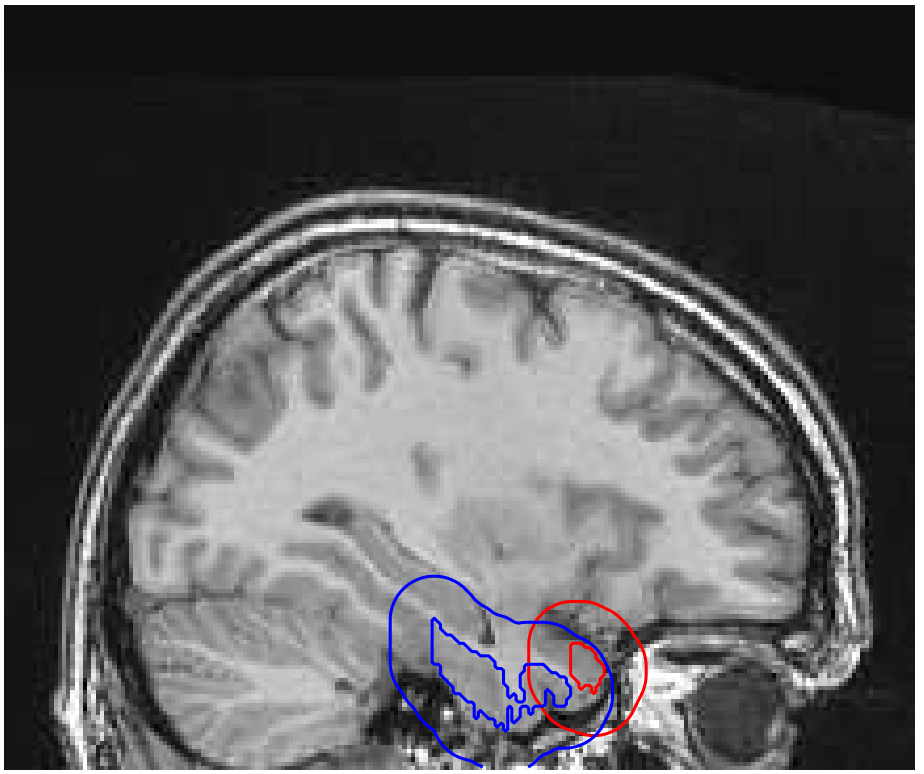

Figure 4.15: *Temporolar(red) and Perirhinal(blue) cortices shown with inner and outer, $r = 10$, regions.*

WM, GM and CSF. Of course the radius of the outer region could be altered to cover a smaller area, and thereby reducing the probability of it containing all tissue types. The width of the outer region should be adjusted empirically to improve the performance of the segmentation.

4.3.3 Utilizing an information theoretic approach

Having looked at the regions in practise rises a new question - How should an initial LSF be fitted to find these regions in novel images. (Tsai et al.; 2003) proposes as previously mentioned to optimize the LSF in terms of the parameters $\mathbf{p}$ and $\mathbf{w}$, - i.e. by warping and deforming each LSF with respect to a well defined cost function. A such cost function should be based on a

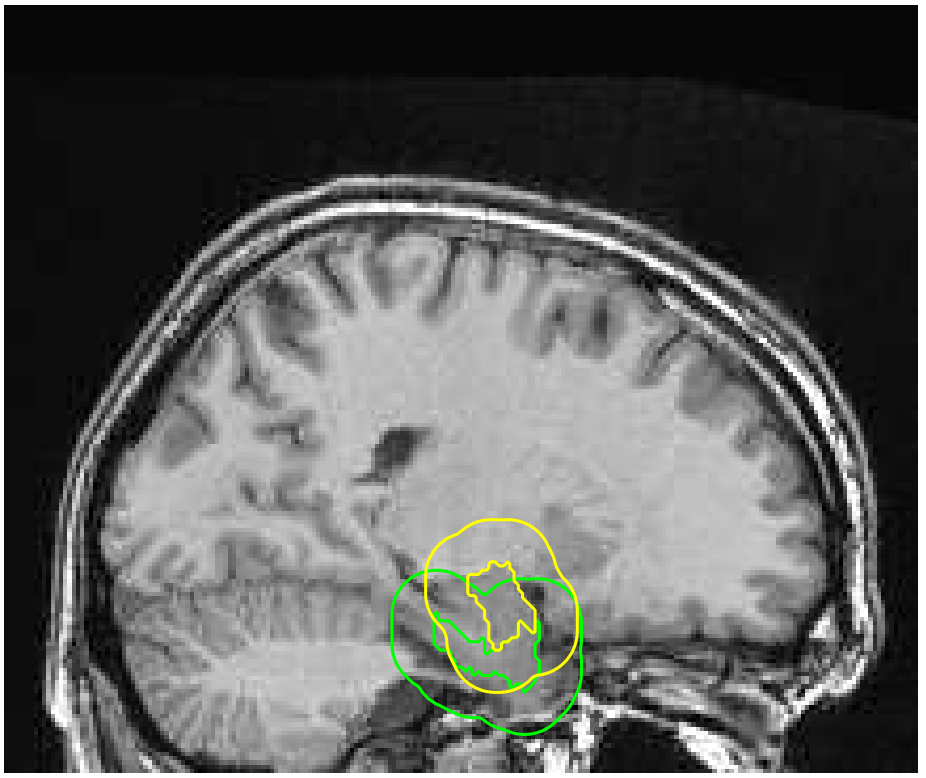

Figure 4.16: *Enthorinal(green) cortex and Amygdalar(yellow) shown with inner and outer, $r = 10$, regions.*

relationship between the inner and the outer region just outlined. This can of course be done in many ways, and different approaches has been proposed in the literature. (Chan and Vese; 2001) proposed a region-based segmentation algorithm based on the difference of variance in the inner and outer region.

$$E_{CV} = R^i_{\sigma^2} + R^o_{\sigma^2} \tag{4.27}$$

(Fisker; 2000) used the difference in mean

$$E_{CV} = R^i_{\mu} - R^o_{\mu} \tag{4.28}$$

to detect nanoparticles with a deformable template model. However, for either of these cost functions to work, both regions needs to be rather homogeneous which does not really seem to be the case for the outer regions. We saw in Figure 4.14, 4.15 and 4.16 that all outer regions consisted of various intensities, making it highly inhomogeneous.

An extremely powerful similarity measure often used in connection with image registration is the Mutual Information(MI) similarity measure. For a thorough examination of MI and its properties, the user is referred to (Pluim et al.; 2003). (Kim et al.; 2002) proposed a region-based active contour which is evolved using a Mutual Information criterion between a binary region label and the intensity values of an image, written in Equation 4.29.

$$\begin{aligned}\mathbf{I}(\mathbf{I}(\mathbf{x}); L(\mathbf{x})) &= \widehat{h}(\mathbf{I}(\mathbf{x})) - \widehat{h}(\mathbf{I}(\mathbf{x})|L(\mathbf{x})) \\ &= \widehat{h}(I(\mathbf{x})) - \pi^i \widehat{h}(\mathbf{I}(\mathbf{x})|L(\mathbf{x}) = R^i) \\ &\quad - \pi^o \widehat{h}(\mathbf{I}(\mathbf{x})|L(\mathbf{x}) = R^o)\end{aligned} \tag{4.29}$$

Here $\mathbf{x}$ is a stochastic variable uniformly distributed over the VOI(i.e. inner and outer region). $L(\mathbf{x})$ assigns a label to $\mathbf{x}$ being either in the inner or outer region according to Equation 4.26. π^i denotes the prior probability of a voxel belonging to the inner region e.g.

$$\pi^i = \frac{|R^i|}{|R^i| + |R^o|} \ , \tag{4.30}$$

$|.|$ is the cardinality of a set, and is calculated as the area.
$\widehat{h}$ is the estimated differential or Shannon entropy which is a good way of measuring similarity between probability density functions, when dealing with Mutual Information methods. $\widehat{h}$ is calculated for the inner and outer regions as:

$$\begin{aligned}\widehat{h}(\mathbf{I}(\mathbf{x})|L(\mathbf{x}) = R^i) &= -\frac{1}{|R^i|} \sum_{R^i} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^i\right)) \\ \widehat{h}(\mathbf{I}(\mathbf{x})|L(\mathbf{x}) = R^o) &= -\frac{1}{|R^o|} \sum_{R^o} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^o\right))\end{aligned} \tag{4.31}$$

$\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^i\right)$ is the probability of $\mathbf{I}(\mathbf{x})$, belongs to the inner region.
If $L(\cdot)$ is not the correct segmentation, then knowing $L(X)$ is not enough to determine which distribution $I(X)$ came from, inner or outer, and thus I(X) is not independent of X. Therefore the mutual information between the label and the image as a function of X is maximized if $L(\cdot)$ gives the correct segmentation. To sum up, the goal is to maximize the mutual information between the region labels and the image pixel intensity values of the image, based on two probability distributions. Since $\widehat{h}(\mathbf{I}(\mathbf{x}))$ in Equation 4.29 is independent of $L(\mathbf{x})$ it is also independent from $\overrightarrow{C}$ and will have no influence on the evolution of the curve and can therefore be removed. Furthermore it would be desirable to change the problem from a maximization to a minimization problem. So we end up with the following cost-function

$$\begin{aligned}E_{MI} &= -\mathrm{MI}(\mathbf{I}(\mathbf{x}); L(\mathbf{x})) \\ &= \pi^i \widehat{h}(\mathbf{I}|L = R^i) + \pi^o \widehat{h}(\mathbf{I}|L = R^o),\end{aligned} \tag{4.32}$$

Before the cost function can be used efficiently a few more definitions needs to be made. The goal is to be able to drive the optimization of the parameters $\mathbf{w}$ and $\mathbf{p}$ using a gradient method. Thus the derivative of the energy function needs to be defined.

$$\begin{aligned}\nabla_{\mathbf{w}} E_{MI} =& \pi^i \nabla_{\mathbf{w}} \widehat{h}(\mathbf{I}(\mathbf{x})|L = R^i) + \pi^o \nabla_{\mathbf{w}} \widehat{h}(\mathbf{I}(\mathbf{x})|L = R^o) \\ \nabla_{\mathbf{p}} E_{MI} =& \pi^i \nabla_{\mathbf{p}} \widehat{h}(\mathbf{I}(\mathbf{x})|L = R^i) + \pi^o \nabla_{\mathbf{p}} \widehat{h}(\mathbf{I}(\mathbf{x})|L = R^o)\end{aligned} \tag{4.33}$$

$$\begin{aligned}\nabla_{\mathbf{w}} \widehat{h}(\mathbf{I}|L = R^i) &= \frac{1}{|R^i|} \left(\oint_{\overrightarrow{C}} \nabla_{\mathbf{w}} \mathbf{\Phi} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^i\right)) ds \right) \\ \nabla_{\mathbf{p}} \widehat{h}(\mathbf{I}|L = R^i) &= \frac{1}{|R^i|} \left(\oint_{\overrightarrow{C}} \nabla_{\mathbf{p}} \mathbf{\Phi} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^i\right)) ds \right) \\ \nabla_{\mathbf{w}} \widehat{h}(\mathbf{I}|L = R^o) &= \frac{1}{|R^o|} \left(\oint_{\overrightarrow{C}} \nabla_{\mathbf{w}} \mathbf{\Phi} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^o\right)) ds \right) \\ \nabla_{\mathbf{p}} \widehat{h}(\mathbf{I}|L = R^o) &= \frac{1}{|R^o|} \left(\oint_{\overrightarrow{C}} \nabla_{\mathbf{p}} \mathbf{\Phi} \log(\mathbb{P}\left(\mathbf{I}(\mathbf{x}) = R^o\right)) ds \right)\end{aligned} \tag{4.34}$$

$\nabla_p \Phi$ is the gradient of the LSF and is given in Equation 4.17 with parameters to be optimized shown in Equation 4.16. $\nabla_w \Phi$ is the deformation of the LSF, and is given as Equation 4.35

$$\nabla_w \Phi = \Phi \tag{4.35}$$

The PDFs for the calculation of E_{MI} should be based on already known segmentations i.e. the manually annotated images which has been supplied.
A problem arises when the evolving LSFs starts to overlap. This means that a voxel is assigned multiple labels, which of course not should be possible. To deal with this a heuristic method of checking prior to the update of $\mathbf{w}$ if this causes an overlap. If so, the update of $\mathbf{w}$ is not performed, leaving only the update of $\mathbf{p}$. Having a new spatial position, a new update of $\mathbf{w}$ might not cause an overlap.

Chapter 5

Segmentation in The Medial Temporal Lobe using a Deformable Atlas

5.1 Shape Alignment

The alignment procedure described in Section 4.2 will here be carried out in practice and demonstrated on the structures of the MTL. Initially the shapes are roughly aligned by their first and second order moments. In practice the displacements are not of a very big magnitude, but the registration does however yield a better offset for the gradient method. To avoid bias the center of mass is initially calculated for all shapes, and the one which is centered the most serves as the shape to which the rest of the shapes are registered. Figure 5.1

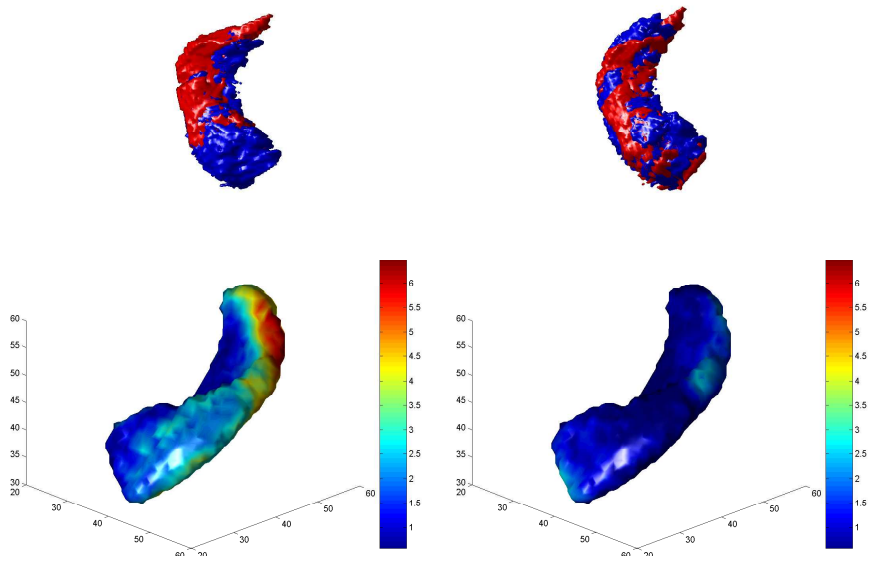

Figure 5.1: *Registration by moments(top row) and gradient method(bottom row): Hippocampus is shown prior to after alignment*

shows hippocampus before and after the initial crude alignment. Only a smaller misalignment is seen, which is corrected efficiently. Furthermore it shows the result of the gradient method described in Section 4.2. The plot of the gradient method depicts the estimated mean shape with the variance of the distances from the mean to all other shapes as the surface color, red being high variance. The distances have been calculated in a set of surface samples. The rest of the aligned regions are found in Appendix F. In Table 5.1 the sum the distances is seen before and after the gradient alignment. Shapes such as region 2 have had a very successful alignment, whereas region 5 and 6 were less successful, - it has however improved in alignment. The worse alignment results of region 5 and 6

has an obvious reason. If these figures are compared to the sizes of the volumes in Appendix C, and especially the standard deviation of the sizes, it is seen that region 5 and 6 varies greatly in scale. As the alignment procedure does not deal with shape, this variance will obviously still be high. So now, the variation that remains in the dataset is the scaling and the biological differences, which was the endpoint of the alignment procedure.

Region	Sum of distances before alignment	Sum of distances after alignment	Improvement
1	2416	794.60	67.11%
2	5150	371.30	92.79%
3	4180	1365.00	67.34%
4	3273	992.10	69.69%
5	4886	4038.00	17.36%
6	5235	3993.00	23.70%
7	7628	175.80	97.70%
8	6404	577.30	90.99%
9	6331	342.70	94.59%
10	8307	751.00	90.96%
11	3813	279.50	92.67%
12	3864	174.30	95.49%

Table 5.1: *Alignment results of the twelve shapes. The total distance between mean-shape and the actual shapes have decreased.*

5.2 Implicit Parametric Shape Model of the MTL

Previous section considered the definition and handling of external shape parameters. This section will consider the intra-shape variation and the co-variation between shapes in a strongly coupled shape model. It was seen in last chapter that there still were differences in the shapes after alignment, meaning that some latent data variation is present. These are the biological variations which should be incorporated in a statistical shape model as described in Section 4.1. Certainly larger dataset gives better statistical properties of the shape model, so a dataset of only 13 patients makes it rather difficult to build a good statistical model.

5.2.1 Capturing variations in the coupled model

Due to the fact that the shapes already exists as SDMs, the building of a coupled shape model is straight forward. The aligned SDMs of the 12 different shapes and 13 patients from Section 5.1 are combined and built into the statistical shape model outlined in Section 4.1.2.2. As seen in Figure 5.2, $\sim 95\%$

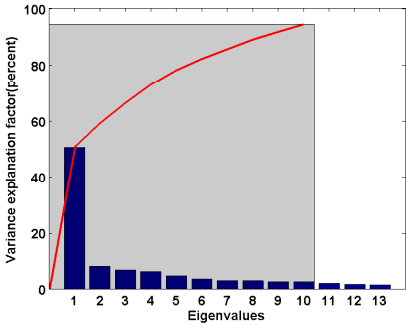

Figure 5.2: *Decay of eigenmodes in the coupled shape model*

of the shape variations can be modeled using 10 parameters. Considering the fact that we wish to model the elements of a SDM, the dimensionality exceeds the number of training elements by far $n >> p$. This means the system is underconstrained, and a great number of eigenvalues will be zero, indicating the Gaussian has collapsed into the number of dimensions spanned by the training set. This is why the small covariance matrix was used for the eigen analysis in the first place. Seen in the light of this many parameters, a model of only 10 parameters describing most of the variance is rather good. The matter of

the fact is however that it is 10 out of 13 possible parameters, which is not a very substantial reduction. Figure 5.3 displays all thirteen principal components plotted as functions of each other. In the diagonal each principal component is seen individually. No clear structure is observed in either of principal com-

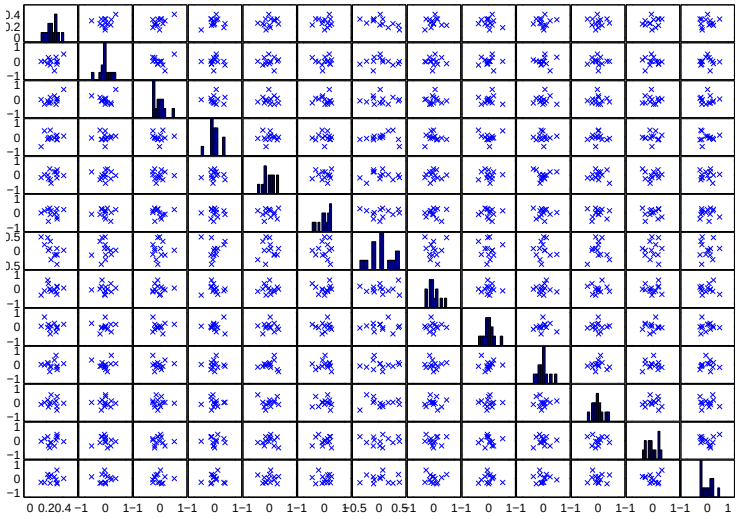

Figure 5.3: *Scatter plot of all principal components*

ponents, thus concluding that the variation of the MTL can be meaningfully described by the linear PCA transformation. Would there have been some sort of non-linear tendency, a PCA would have problems describing the variance.
Looking back at Figure 5.2, the first mode is very dominating. To find the reason for this, the attention should be drawn to the first row of Figure 5.4. This row displays the meanshapes and their variation by the first principal component with $\pm 1\sigma$. This mode clearly describes the overall scaling of the model, and is apparently a very dominating influence. This is a direct consequence of leaving out scaling in the alignment process. Going back to the discussion of leaving out scaling from the alignment model, one might of course argue that a simple anisotropic or even isotropic scaling might be able to compensate for this sort of variation when the shape model is put to use in the segmentation algorithm. It is however the belief of the author that the scaling of the MTL is more specific than such i.e. scales in more than three directions, and therefore have to be described by a principal component. In the second principal mode an asymmetrical variation of the left, right amygdala(yellow) and left, right temporopolar(red) cortex seems to be dominant. The Enthorinal cortices are also

seen to vary some. In total it is rather difficult to see distinct deformations in the remaining modes, and in some modes similar deformations are seen, such as the two hippocampus(cyan) regions in mode three and four, which shows a sort of contraction.

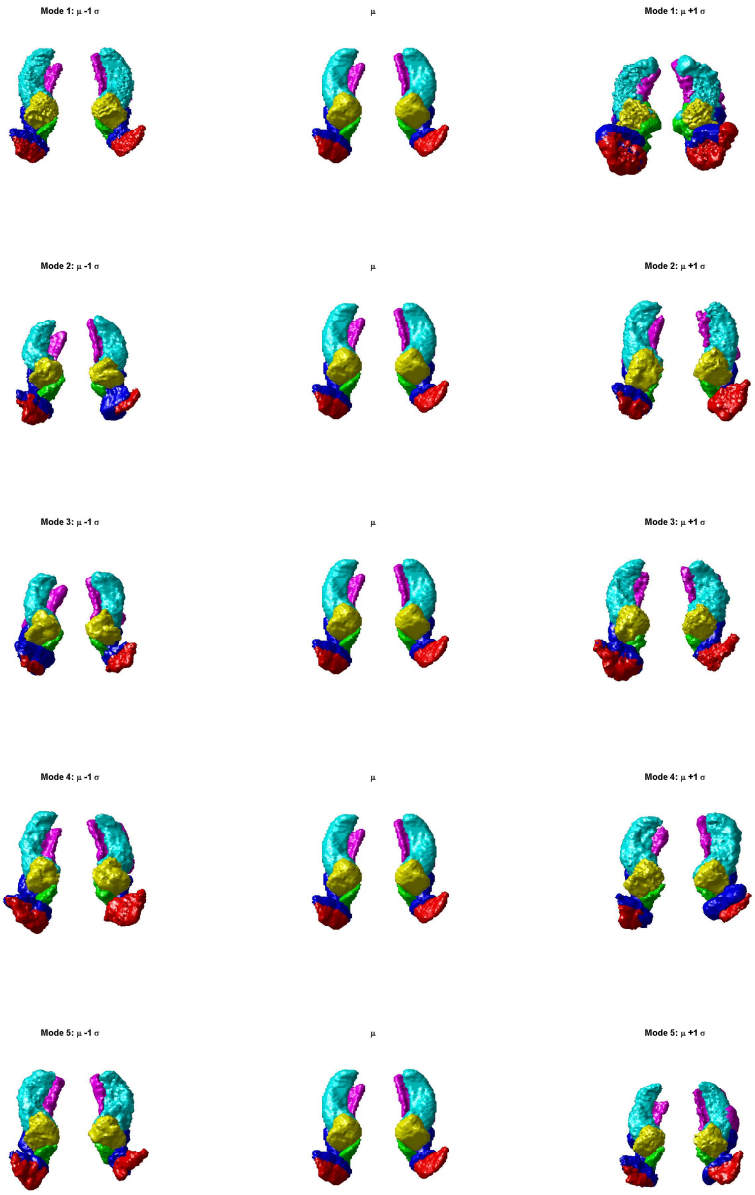

Figure 5.4: *Mean shape deformation. influence From top to bottom: Mode 1 to 6. From left to right:* $\mu - \sqrt{\sigma}$,μ,$\mu + \sqrt{\sigma}$

5.3 Segmentation with the Implicit Parametric Shape

Recalling Section 4.3, a method for using the implicit parametric shape model was defined. A region based segmentation approach was described, which will be utilized in the MTL in this section. In order to make sure that the model will work as expected, some conditions for the segmentation will be set up. Furthermore the validity of the model will be tested on a set of synthetic examples. As the model generalizes in both 2D and 3D, these exercises will be performed in 2D for clarity. After this, the model is tested in 3 dimensions on each region separately, and the generalization error calculated according to the methods in Section 3.7. This will give some insight to which regions might be harder to locate. Finally the coupled shape model is tested on all 12 shapes simultaneously and the results are validated.

5.3.1 The region-based model in a practical setting

In order to get acquainted with the segmentation problem, the formulated energy description in Section 4.3 will be reviewed in a practical setting. This means that the necessary conditions this method here will be summed up, illustrated and described in practise.

5.3.1.1 Initial Grids

For most numerical optimization methods, the initial guess of the optimal parameter is crucial for the entire optimization sequence. This case is of course not any different, which means that a well suited initial spatial location and deformation is required. For this purpose the mean shapes $\mathbf{\Phi}_k^i$ specified in Section 4.2 is utilized. For the exact spatial initial guess the center of mass of each meanshape is used, as shown in Figure 5.5 for a single shape. The crucial importance of this initial guess will be demonstrated during the next sections, especially in the final model validation.

5.3.1.2 Estimation of region-PDFs in the MTL

As the model is driven on a probability machine, i.e. the differential entropy of the inner R^i and outer R^o region, the PDFs of the ground truth regions needs

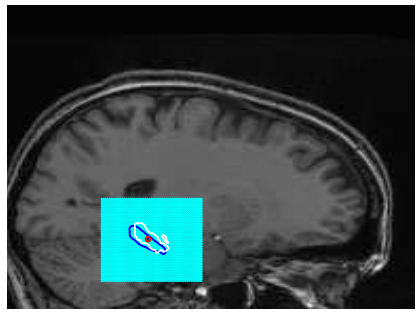

Figure 5.5: *Initial grid conditions for a shape. The white region is ground truth of a slice of PHC. The blue region is the meanshape of PHC. The red circle is the center of mass for the blue shape. The cyan grid is the template space which is warped into image space.*

to be evaluated. This is done using the parzen window, Section 3.6, and the estimated PDFs can be found in Appendix G. In Figure 5.6 two characteristic PDFs are shown.

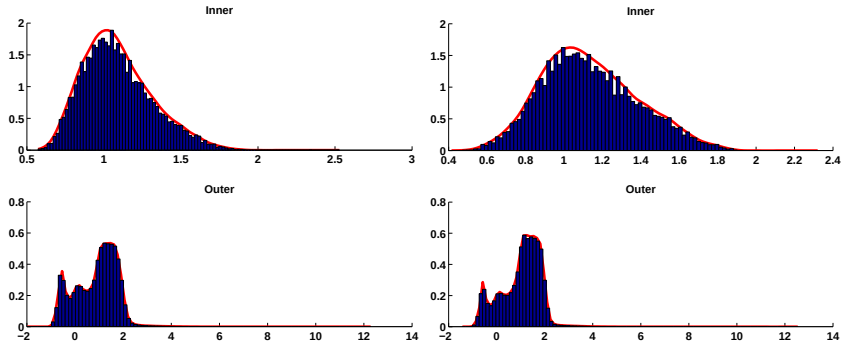

Figure 5.6: *Left: Region 3 - Enthorinal Cortex Left Right: Region 8 - Parahippocampal Cortex Right*

Figure 5.7:

Here it is seen that the inner regions are quite similar and rather homogenous which was expected as the figures in Section 4.3 hinted. The outer regions are characterized by three peaks which is expected as the different tissue types in the surroundings of the regions. This looks rather good, but the problem is that these distributions are not unique in the area of the MTL, and many local minima exists which makes the segmentation very challenging. To visualize this problem, a good way is to fix all parameters except for two, and plot so-called

energy maps based on the ground truth.

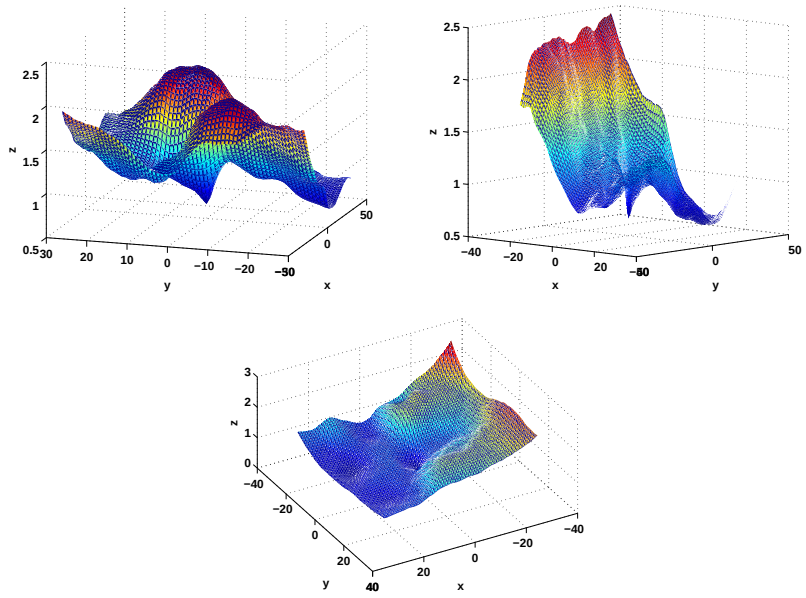

Figure 5.8: *Energy maps for ground truth shape in grid around true position. Scaling and Rotation parameters are fixed. Translation parameters X, Y and Z moves. From left to right the moving parameters are XY,YZ,XZ*

The energy maps are created by changing two parameters at the time. The present Figure 5.8 is of hippocampus in test-person f1377 where the translational parameters are moved and the rest are fixed. The function is clearly far from convex, thus making it a difficult optimization problem and prone to descend into a spurious minimum. The peak in the middle of all maps is the correct position, which we are searching for. Armed with this knowledge the model is validated in the next section.

5.3.2 Validity of the model-theory

To verify that the theory actually works as it is supposed to, a simple 2D synthetic dataset is created consisting of 20 shapes, Figure 5.9. The shape model is created as described in Section 4.1, after the shapes have been aligned. Naturally this shape-model can only be used to locate nove elliptical shapes such as Figure 5.11 since all training data are of such character. Two cases are

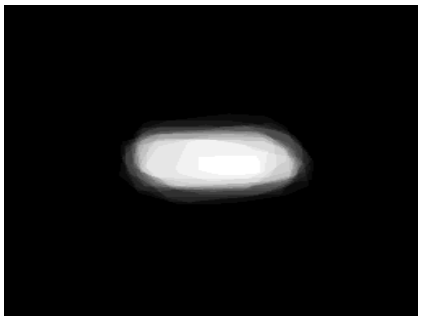

Figure 5.9: *Training set of shapes used to verify the model*

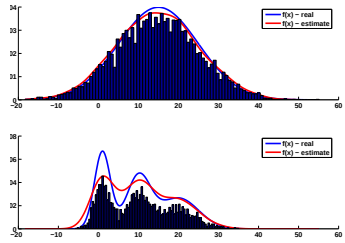

Figure 5.10: *Estimated PDFs of the synthetic regions*

demonstrated;
The first presents the segmentation model with a shape from the training set, perturbed with gaussian noise in the inner region, and noise created as a mixture of three gaussians in the remaining outside region. The PDF's for the model is estimated using a parzen window, Section 3.6. The PDF's are seen in Figure 5.10, together with a histogram of the samples and the real distributions, used to create the perturbations. It is seen that the Parzen window is biased some, especially in the outside region. This error is unavoidable when using a non-parametric method for PDF-estimation, and will of course have influence on the accuracy of the segmentation. The reason for perturbing the outside region with a mixture of gaussians is to make the example more like the regions in the MTL, Section 4.3. Certainly this does far from represent the real optimization problem, but it makes for demonstration.
The second case is a novel shape, in the sense that it is not part of set of shapes that were used to train the shape model, i.e. Figure 5.9. This shape was perturbed in the same manner. In both cases the diameter of the outer region, r was set to 10, and the model seems to fit the perturbed shapes fairly well. It is seen that the shape from the training set is fitted better than the novel shape, which was expected. Especially the deformable contour has problems imitating the small dent in the top of the novel shape for obvious reasons.
As already mentioned, the initial guess has tremendous impact on the result of segmentation, as this may cause the snake to go towards some unwanted local minima. Furthermore the radius of the outer region also plays a very significant role in the segmentation performance. In the current implementation this parameter should be chosen empirically, which is a rather big drawback.

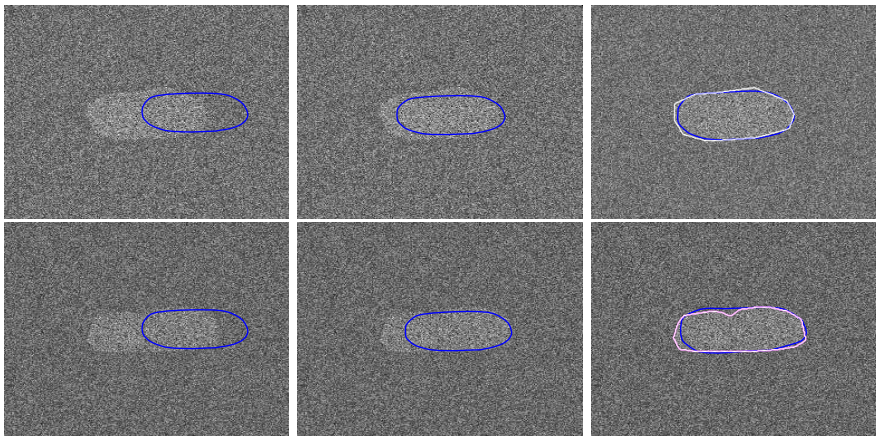

Figure 5.11: *Synthetic segmentation example of ellipse. Top row shows the segmentation of a shape known from the training-set. Bottom row shows the segmentation of a novel shape. The columns show initial, midway and final fit. The white contour in the final plots is the ground truth shape*

5.3.3 Individual Segmentations

Moving on to a more challenging segmentation problem, each region of the MTL is segmented separately, i.e. no coupling between the regions. Furthermore a generalization-error calculated using the DICE-measure and LOOCV will be calculated. Naturally the diameter of the outer region will be fixed to make for a more fair comparison. To perform this test it is clear that individual shape models should be created for each region, in order to make a meaningful segmentation. However, as the region-based model generalizes from single to multiple regions, this is straightforward.
In Figure 5.12 plots of the generalization error is seen for the initial guess(the mean shapes) and the segmentation results. It is seen that the model performs different on each of the regions, which is of course natural. In region 9 and 10, the hippocampal areas show good overlaps with up to 85%. Unfortunately some results such as regions 1 and 11, the temporopolar cortex and amygdala are seen to fail completely with an overlap of 0%. This is due to the initial guess, which as mentioned have high impact; generally it is seen from the plot that bad initial overlaps also often causes bad segmentation results. The variance of many of the shapes is high, which shows that the model is not so robust. To illustrate some scenarios from the above experiment, Figure 5.15 shows two different examples.
It is clear from these figures how the initial guess has affected the result. Furthermore the region of the hippocampus is more distinct than that of the temporopo-

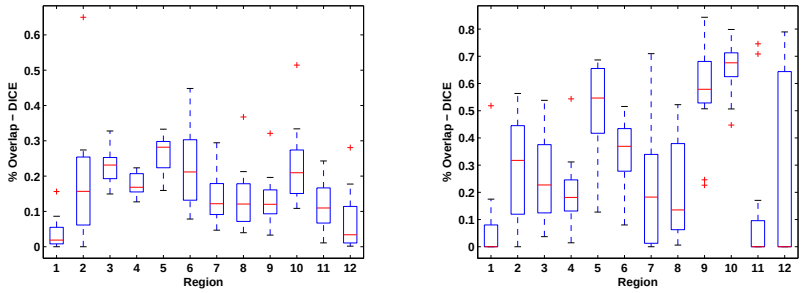

Figure 5.12: *Boxplot of the generalization error of each shape in each of the 13 test-persons. Left box-plot shows the error of the initial guess compared to the ground truth, while the right shows the error of the segmentation result compared to the ground truth.*

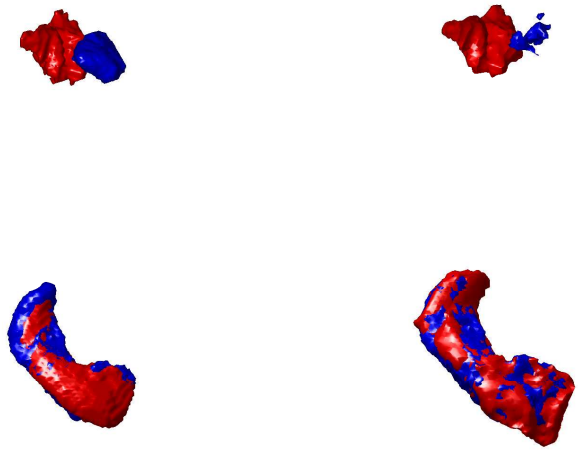

Figure 5.13: *Two opposite segmentation scenarios: The red shapes are the initial guesses, and the blue are the final results. In the top is a segmentation gone wrong. In the bottom is a more successful segmentation. From top-left to bottom-right the DICE score is 0.05, 0, 0.25 and 0.84.*

lar areas, as can be seen in Appendix B, which makes it an easier segmentation challenge. So to sum up, based on this experiment, regions such as 1,11 and

12 are difficult to find with this method in this setting. The two hippocampal areas, region 9 and 10 shows pretty good results however, so it might be a good idea to guide the segmentation after these regions. There are of course different ways of doing this, and the method explored here, is the strongly coupled model, Section 4.1.2.2.

5.3.4 Simultaneous Segmentation of all regions

Like in the previous section the model is presented with novel images, by using LOOCV. This is can be seen if the coupling of the model actually enhances the performance. Similar to Figure 5.15, Figure 5.14 shows a boxplot, where each box represents a region. This model certainly outperforms the approach of

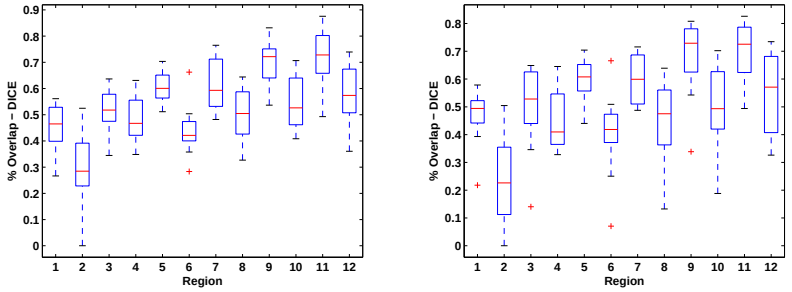

Figure 5.14: *Generalization errors of the coupled segmentation model. In the left plot the diameter of the outer region was set to 5 in all shapes. In the right, the diameter was set to 10.*

segmenting each region separately. The overlaps of each shape is in general much higher, and the variance is lower as well. The initializations are the same as when shapes were segmented individually, so this gives a good basis for comparison. The model has been tested with two different values of R, 5 and 10. For optimal performance it should be tested systematically, and the model with the lowest generalization error should be chosen. An empirically set $R = 5$ seems however to show that this model seems to be working as intended. Figure 5.15 shows the energy as a function of the iterations. The energy descents in a natural manner, and converges at some point after 100 iterations because the change in function value falls below an empirical set threshold. Figure 5.15 shows the volumetric view of the segmentation result, while a 2D axial slice view can be seen in Figure 5.16 showing five out of the six regions.

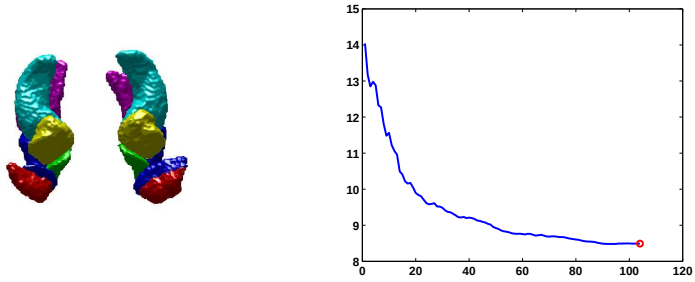

Figure 5.15: *Left: Result of the segmentation with R=5, Right: The decay of energy as a function of time*

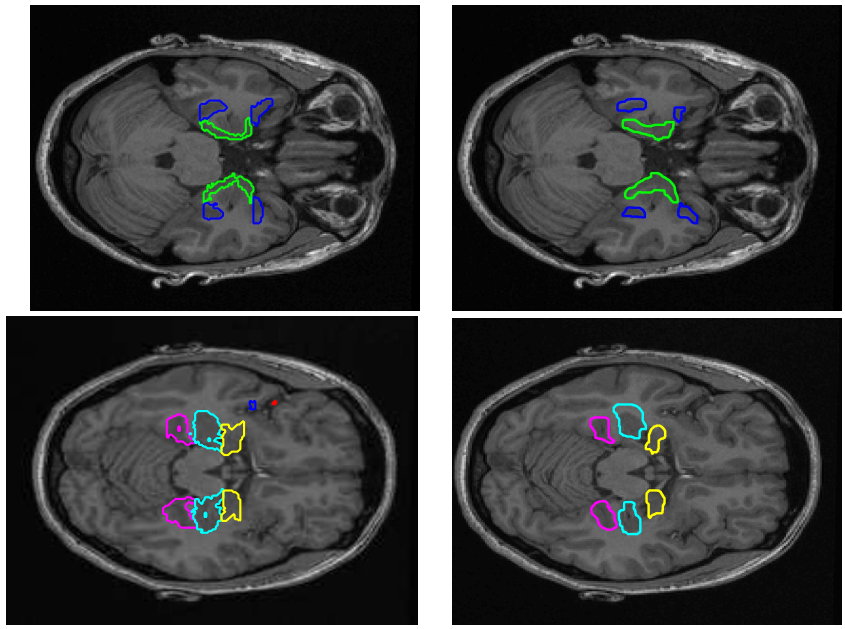

Figure 5.16: *Axial view of segmentation results for f1778 in slice 50 and 60. - Regions: HC(cyan), PHC(magenta), AD(yellow), EC(green), PRC(blue) Left: Ground Truth, Right: Segmentation results*

Chapter 6

Discussion and Conclusion

6.1 Discussion and Future Work

This section provides a discussion of the implemented model, what is good, what is bad and what can be done in other ways. It is clear from the results that the coupled shape model works, and is able to locate the regions in the MTL. The results however, are not convincing, and this needs to be investigated. So what are the reasons for this lack of performance? This is a difficult question to answer, but the following sums up ideas to be improved in the different phases of the model. In the alignment phase the shapes were aligned individually. This of course is a bit inconsistent with the fact that a coupled model is built. When the shapes are aligned individually in each shape class, some spatial information might be lost. (Tsai et al.; 2003) makes a joined binary registration by minimizing a common overlap measure for all shapes in all shape classes. This seems reasonable, and it might be a good idea to formulate a similar energy functional working on the LSF' registration scheme presented in Section 4.2. A such energy-functional could be calculated as

$$\sum_{\substack{i,j=1 \\ i \neq j}}^{n} \sum_{k=1}^{m} \left(\Phi_i^k(\mathbf{x}) - \Phi_j^k(\mathbf{W}(\mathbf{x};\mathbf{p})) \right)^2 \tag{6.1}$$

There is no doubt that this would be a rather heavy registration procedure depending on the size and dimensionality of the images. A narrowband technique would help to solve this, and in combination with the inverse compositional algorithm (Baker and Matthews; 2002) it should be feasible.

It would also be interesting to investigate whether the removal of scale in the alignment procedure would allow for other features to come forward in the shape model. As we saw in Section 5.2, the first mode of variation was very much dominated by the scaling of the volumes. If this factor disappeared it might allow the shape model to deform in other more specific directions. This was however not chosen in this work.

Another problem with scaling is that when an SDM is scaled, it is no longer an SDM, see Section 4.1. So a reinitialization of a shape would have to be performed after each warp, which would be very costly.

As for the shape model, and how this is coupled, there might also be different approaches than the one used here. Certainly a coupling is necessary, which is seen by the results of the individual segmentations in Section 5.3.4. But the problem with this strongly coupled shape model is that it might be to strong coupled. In this model, the degree of freedom is seven plus the number of modes in the shape model. This means that all shapes moves as a single unit. It would be a good idea to incorporate the possibility to let each shape move individually to a certain limit. A sequential coupling is used in (Hansen; 2005), where an initial shape is found using a single shape model. When this shape has been found the pose parameters of the remaining shapes are updated by sampling from an

estimated gauss distribution. Although this is a nice approach, each shape in the MTL is very hard to locate individually, as demonstrated in Section 5.3.4. Therefore it might be difficult to get good initializations using this procedure. Finally, the heuristic method of avoiding overlap performed here should be done in a more clever and elegant way, such as a preventive approach where a constraint is put on $\mathbf{w}$ so that an overlap will never occur.
Looking at different approaches, the log odds framework was mentioned in Section 4.1. This is a method that utilizes probabilistic atlases, and a completely novel approach called *Active Mean Fields*(AMF)(Pohl et al.; 2007) is being presented at the IPMI[1] conference. The AMF is a further development of the log-odds framework previously mentioned in Section 4.1. This means that the shape model is viewed as a probabilistic atlas, which might be difficult to utilize in the setting used in this project, so an implementation of an algorithm similar to might yield rather interesting results. Using the Logodds framework in the AMF model implicitly avoids the problem of overlapping regions due to the fact that the region-ownership of each voxel is handled by a maximum aposteriori probability.

Finally, another example of a method to do the segmentation would be by using deformation-based morphometry. This was shortly mentioned in the end of the introductory Chapter 2. A specific method is by utilizing diffusion registration, in a demon-based registration. Thirion

6.2 Conclusion

The objectives set forth in the beginning of this project was the following:

- Investigate the possibilities of making an intelligent system which is able to localize the regions of interest.
- Elaborate and analyze a sensible approach to an automatic or semi-automatic labeling of these regions
- Develop a prototype which demonstrates the approach and can carry out a labeling of the regions of interest

These objectives have been achieved to an extent as now discussed.

The investigation of the possibilities of making a system capable of helping Thomas Zöega Ramsøy ended up with the choice of using a level set method

[1] Information Processing in Medical Imaging 2007 in the Netherlands

with a shape prior, which has been used to solve similar problems earlier in (Tsai et al.; 2004) and (Hansen; 2005) with good results. An extensive amount of literature have been studied to investigate this method which covers a broad range of different theory. In the final period of the thesis it has come to the attention of the author that a recent different approach, (Pohl et al.; 2007), would be very interesting to investigate as an alternative to the method explored here. However, the LSF method with a shape based prior have been investigated to a great extent here, and found adequate to suit the needs for segmenting regions in the Medial Temporal Lobe. Improvements to the model should however be made. An alternative method, deformation-based morphometry, specifically diffusion registration, has also been investigated to an extent but with unsuccessful results.

Finding and testing a suiting energy function to drive the region-based segmentation model in the MTL have been a challenge and took a good portion of time. The coupling of the shape model might have been done in a different manner, although the one used is an elegant way of capturing intra variation among the regions in the MTL.

Finally, the developed prototype have been tested and generalization errors have been visually inspected, which showed relatively promising results. The coupled model was compared to segmentation of the shapes individually and showed improvement. However, the model is not perfect, and more work can be performed.

Appendix A

Volumetric view of all shapes in the 13 patients

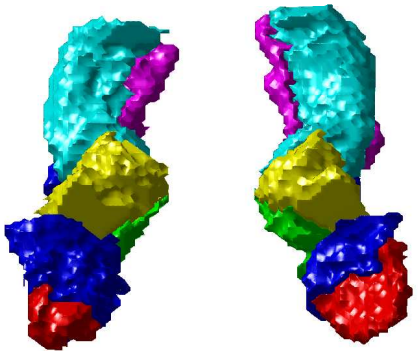

Figure A.1: *MTL of test person f1371*

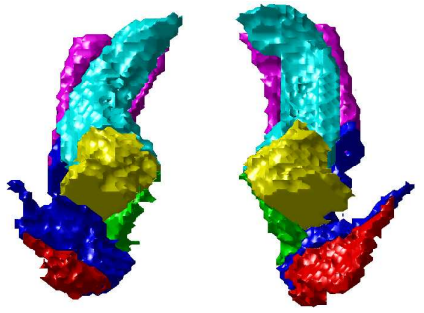

Figure A.2: *MTL of test person f1374*

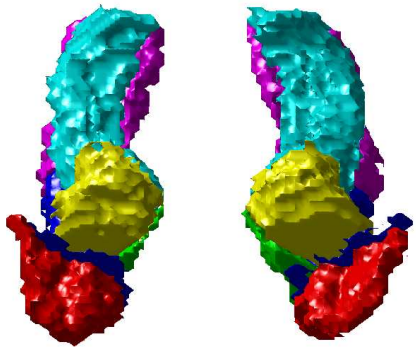

Figure A.3: *MTL of test person f1387*

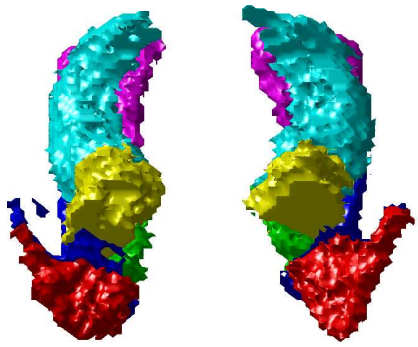

Figure A.4: *MTL of test person f1388*

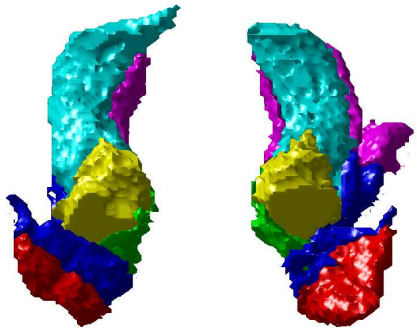

Figure A.5: *MTL of test person f1512*

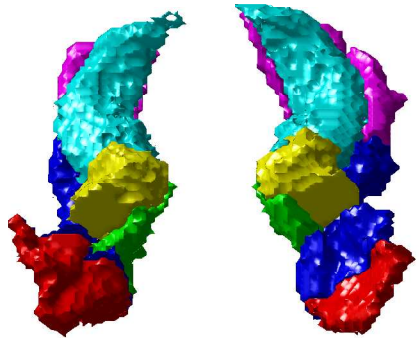

Figure A.6: *MTL of test person f1577*

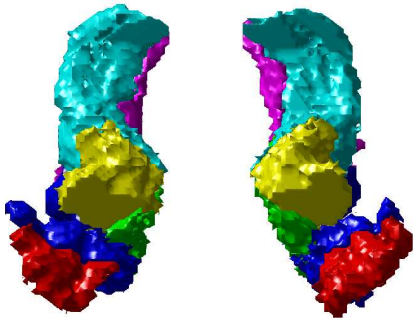

Figure A.7: *MTL of test person f1593*

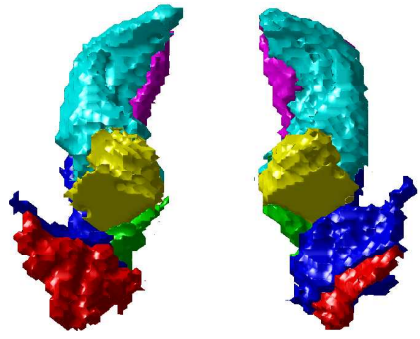

Figure A.8: *MTL of test person f1736*

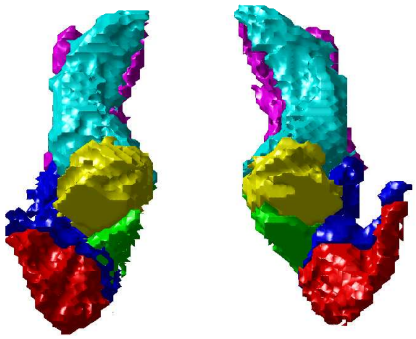

Figure A.9: *MTL of test person f1737*

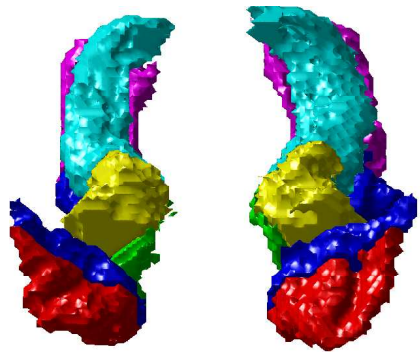

Figure A.10: *MTL of test person f1740*

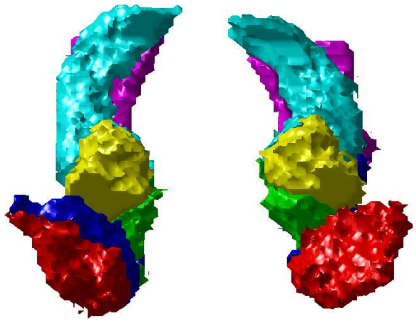

Figure A.11: *MTL of test person f1777*

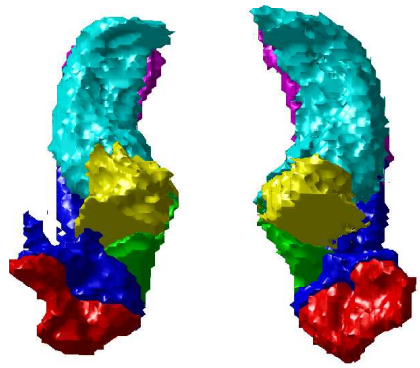

Figure A.12: *MTL of test person f1778*

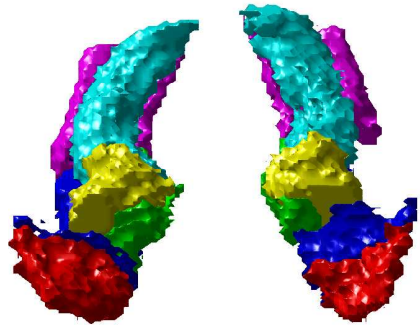

Figure A.13: *MTL of test person f1830*

Appendix B

Overview of all regions in the MTL

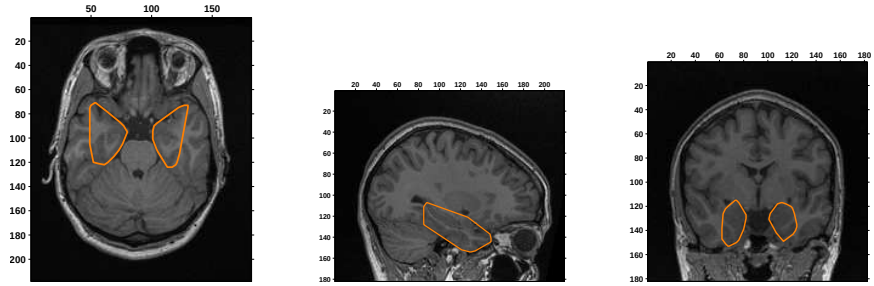

Figure B.1: *Convex hull of all the supplied regions seen in Axial, Coronal and Sagittal view*

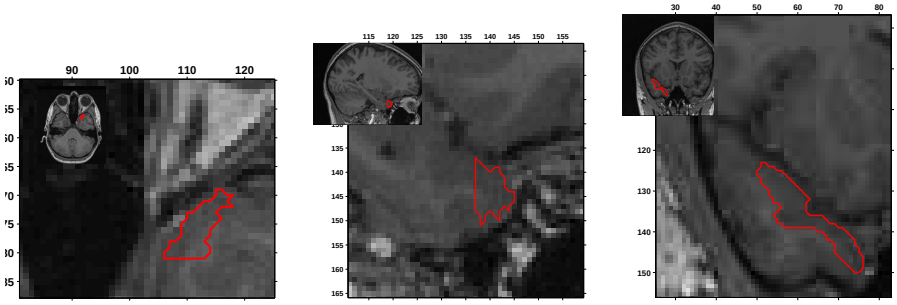

Figure B.2: *Region 1 - Temporopolar cortex left*

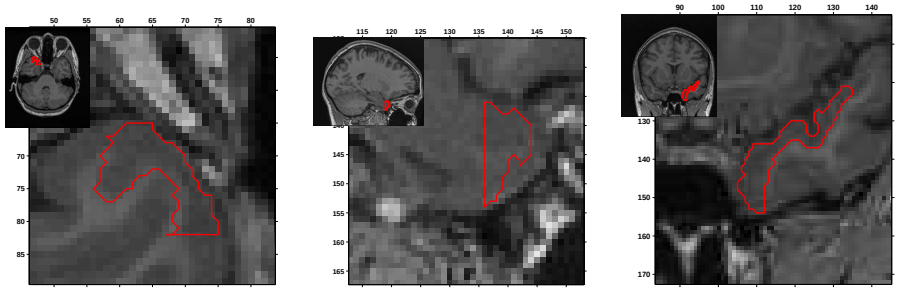

Figure B.3: *Region 2 - Temporopolar cortex right*

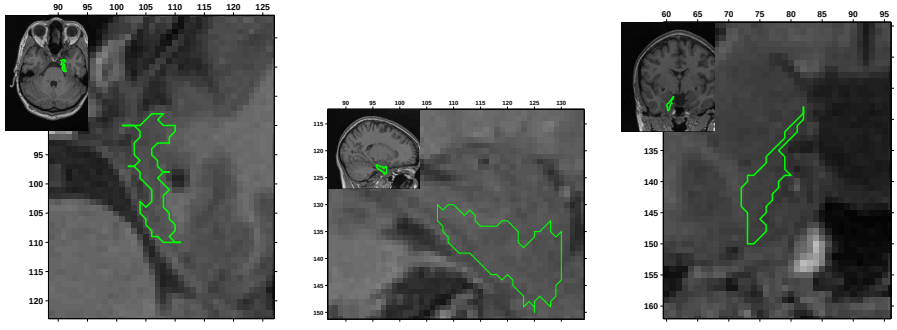

Figure B.4: *Region 3 - Entorhinal cortex left*

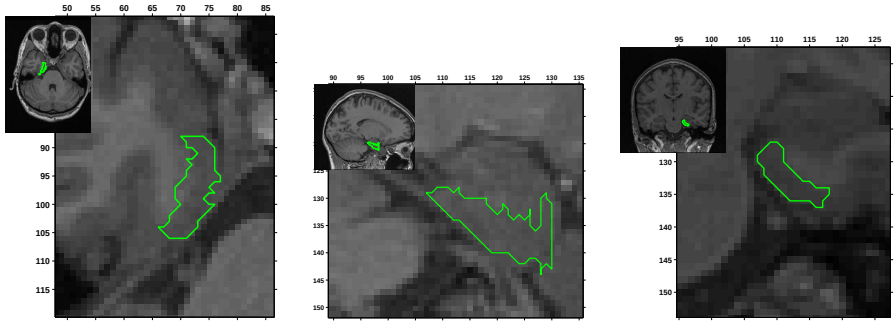

Figure B.5: *Region 4 - Entorhinal cortex right*

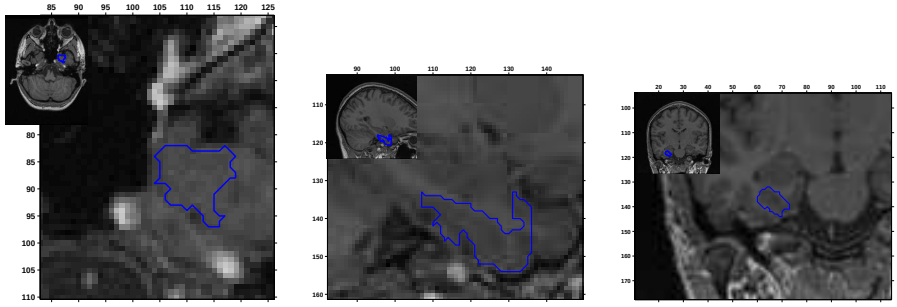

Figure B.6: *Region 5 - Perirhinal cortex left*

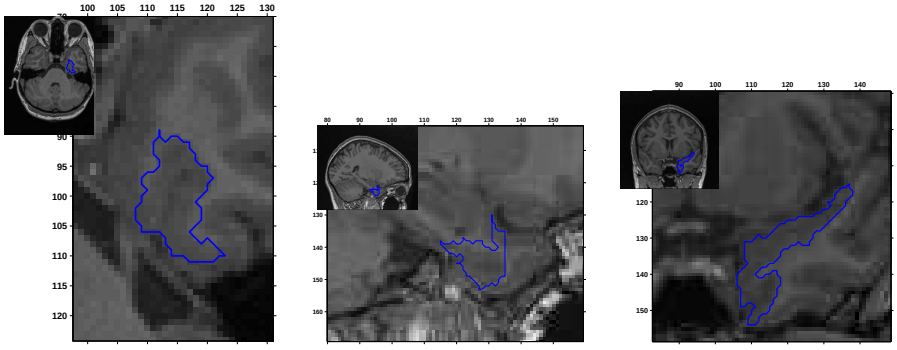

Figure B.7: *Region 6 - Perirhinal cortex right*

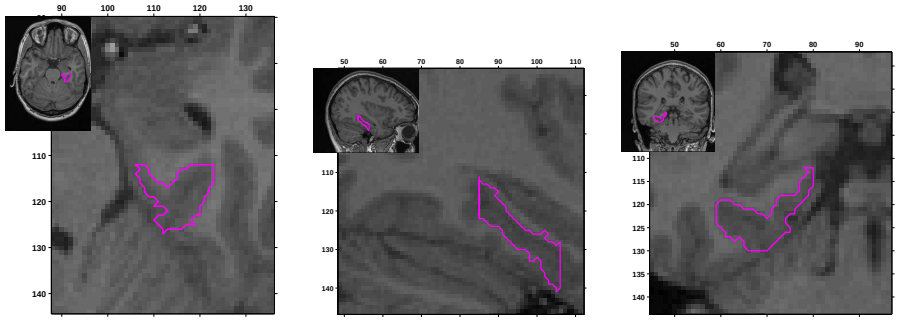

Figure B.8: *Region 7 - Parahippocampal Cortex left*

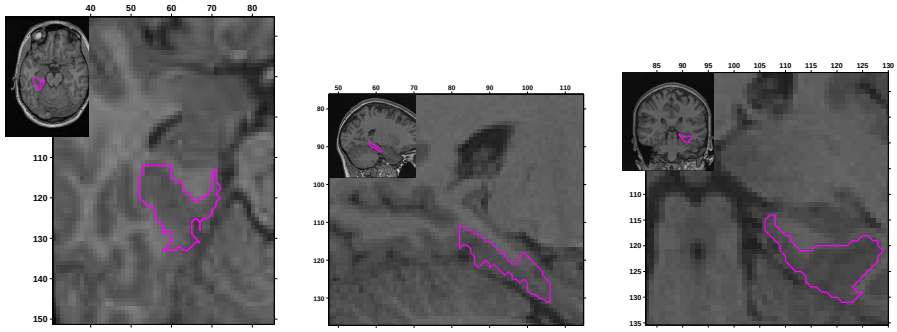

Figure B.9: *Region 8 - Parahippocampal Cortex right*

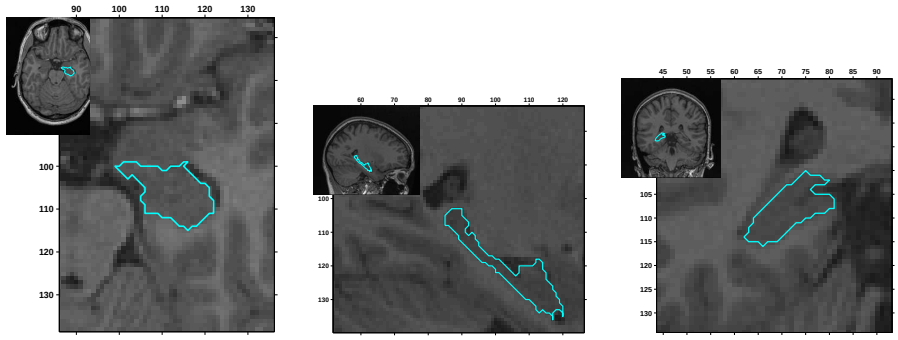

Figure B.10: *Region 9 - Hippocampus left*

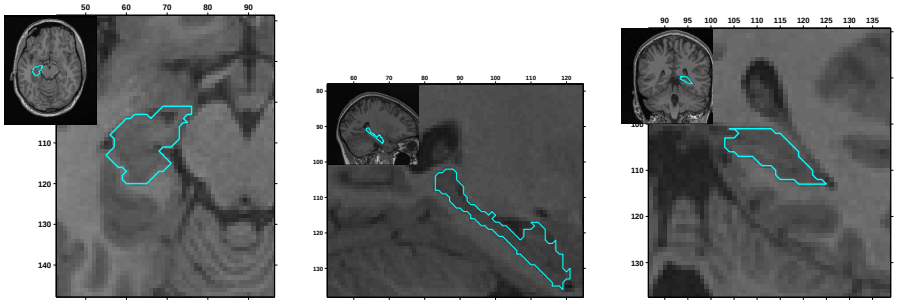

Figure B.11: *Region 10 - Hippocampus right*

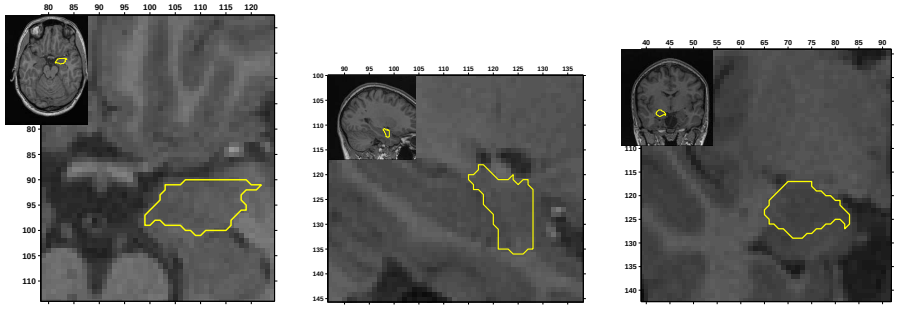

Figure B.12: *Region 11 - Amygdalar left*

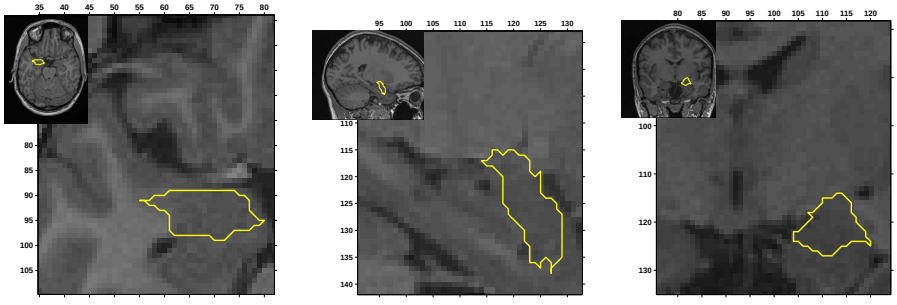

Figure B.13: *Region 12 - Amygdalar left*

Appendix C

Volume sizes of the 12 shapes

Person	Reg. 1	Reg. 2	Reg. 3	Reg. 4	Reg. 5	Reg. 6	Reg. 7	Reg.8
1	624	1261	1200	1355	5595	4401	3396	3663
2	599	1567	1417	2004	5314	2922	3361	3009
3	1915	1732	696	1110	2651	2613	3857	3018
4	2380	2685	1884	1721	4240	4296	3441	3570
5	1114	2014	1720	1329	5312	4724	3339	2918
6	2793	1413	2268	1449	4447	6294	2490	3123
7	1439	1970	1800	1402	4161	4539	3199	3130
8	2474	583	1534	1438	3539	6315	2630	3055
9	2303	2615	2272	1888	4691	4622	3271	3843
10	2172	2477	1552	1441	4992	4943	3449	3134
11	1891	2884	1417	1206	3750	2069	2868	2677
12	1626	2085	1688	1741	6358	5323	2597	3566
13	2247	1559	1914	1298	3618	3279	3152	3190
Mean	1813.62	1911.15	1643.23	1490.92	4512.92	4333.85	3157.69	3897.92
Std	697.83	652.85	424.83	267.64	1004.89	1310.11	400.24	352.35

Person	Reg. 9	Reg. 10	Reg. 11	Reg. 12
1	4563	4602	2934	2460
2	5185	4853	2669	2650
3	4501	4307	2720	2957
4	5336	5444	3318	2434
5	4920	4503	3412	2545
6	4661	4584	2648	2378
7	4988	4788	2888	2751
8	4450	4311	2412	2155
9	4711	4601	3134	2829
10	4455	5063	2602	2667
11	4380	4445	2346	2249
12	4932	5521	2584	1995
13	3944	3632	2345	2236
Mean	4694.31	4665.69	2770.15	2485.08
Std	374.27	496.51	349.37	282.43

Appendix D

Probability Distribution Functions for 13 images

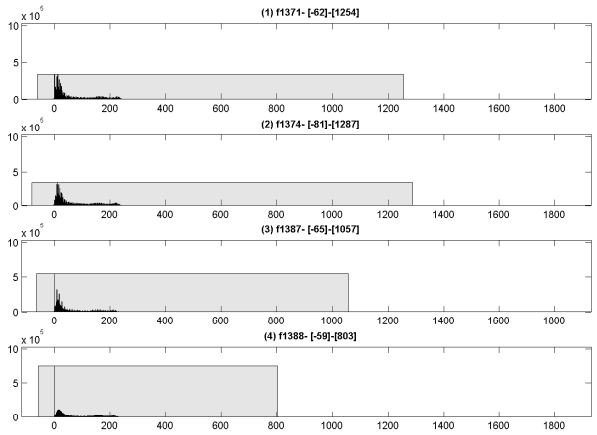

Figure D.1: *From top to bottom, PDF for: f1371,f1374,f1387,f1388*

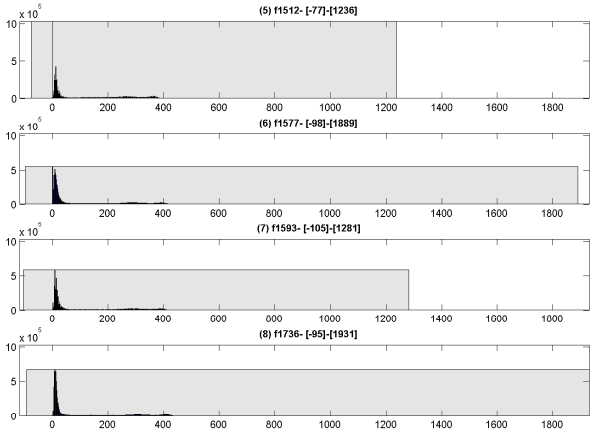

Figure D.2: *From top to bottom, PDF for: f1512,f1577,f1593,f1736*

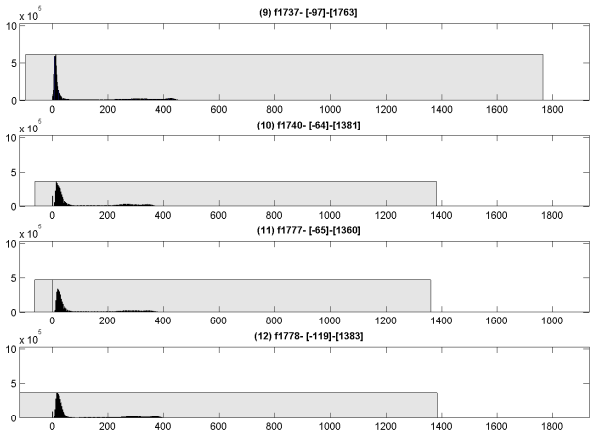

Figure D.3: *From top to bottom, PDF for: f1737,f1740,f1777,f1778*

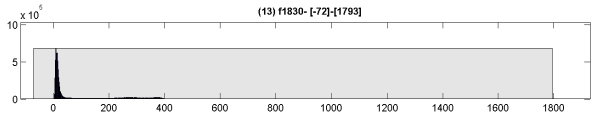

Figure D.4: *PDF for: f1830*

Appendix E

Transformation matrices

$$
\begin{aligned}
\underline{\underline{T}} &= \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\underline{\underline{R_\Omega}} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \cos(\Omega) & \sin(\Omega) & 0 \\ 0 & -\sin(\Omega) & \cos(\Omega) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\underline{\underline{R_\Phi}} &= \begin{bmatrix} \cos(\Phi) & 0 & -\sin(\Phi) & 0 \\ 0 & 0 & 0 & 0 \\ \sin(\Phi) & 0 & \cos(\Phi) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\underline{\underline{R_\kappa}} &= \begin{bmatrix} \cos(\kappa) & \sin(\kappa) & 0 & 0 \\ -\sin(\kappa) & \cos(\kappa) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\mathcal{T}[P] &= \underline{\underline{R_\kappa}} \cdot \underline{\underline{R_\Phi}} \cdot \underline{\underline{R_\Omega}} \cdot \underline{\underline{T}}
\end{aligned}
\tag{E.1}
$$

$$
\begin{aligned}
\frac{\partial T}{\partial t_x} &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\frac{\partial T}{\partial t_y} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\frac{\partial T}{\partial t_z} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
\frac{\partial R_\Omega}{\partial \Omega} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\sin(\Omega) & \cos(\Omega) & 0 \\ 0 & -\cos(\Omega) & -\sin(\Omega) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\frac{\partial R_\Phi}{\partial \Phi} &= \begin{bmatrix} -\sin(\Phi) & 0 & -\cos(\Phi) & 0 \\ 0 & 1 & 0 & 0 \\ \cos(\Phi) & 0 & -\sin(\Phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\frac{\partial R_\kappa}{\partial \kappa} &= \begin{bmatrix} -\sin(\kappa) & \cos(\kappa) & 0 & 0 \\ -\cos(\kappa) & -\sin(\kappa) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{E.2}
$$

Appendix F

Aligned Shapes

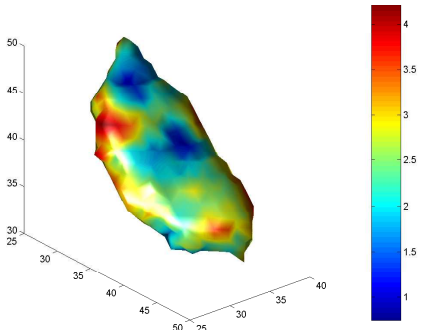

Figure F.1: *Region 1 Unaligned*

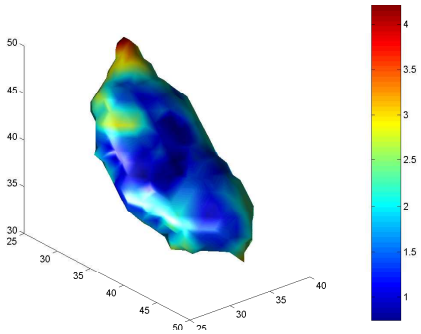

Figure F.2: *Region 1 Aligned*

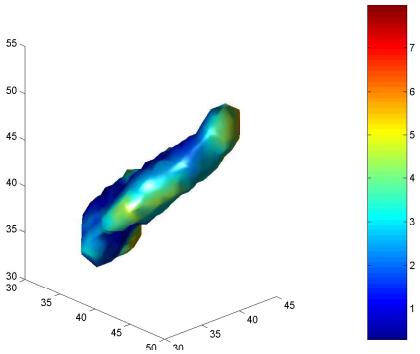

Figure F.3: *Region 2 Unaligned*

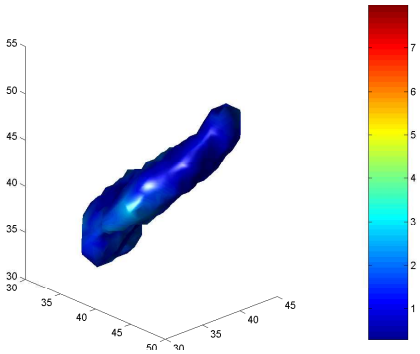

Figure F.4: *Region 2 Aligned*

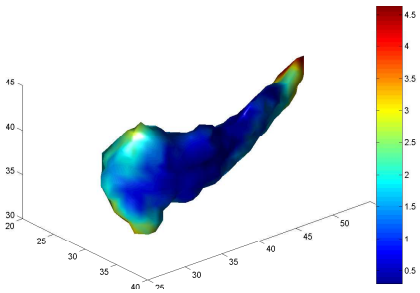

Figure F.5: *Region 3 Unaligned*

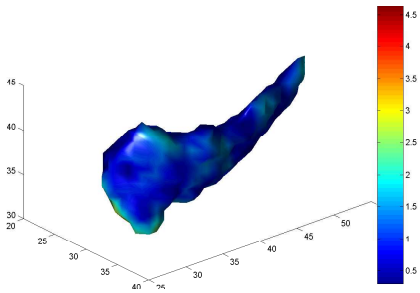

Figure F.6: *Region 3 Aligned*

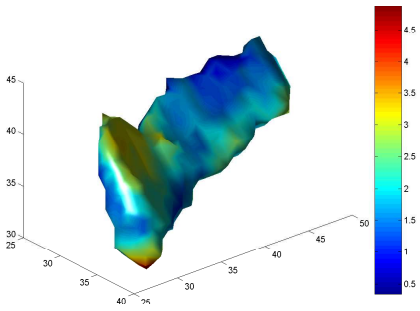

Figure F.7: *Region 4 Unaligned*

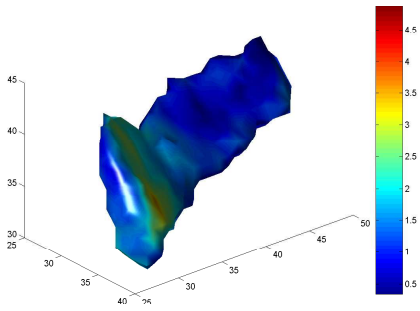

Figure F.8: *Region 4 Aligned*

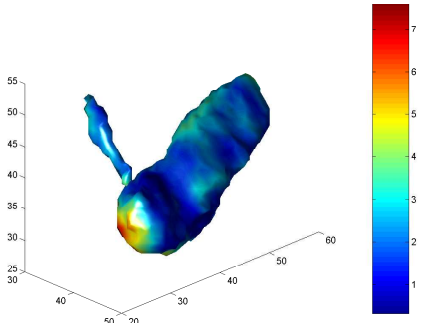

Figure F.9: *Region 5 Unaligned*

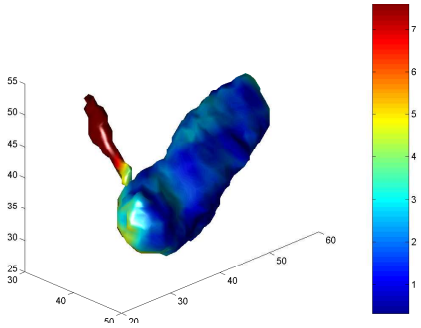

Figure F.10: *Region 5 Aligned*

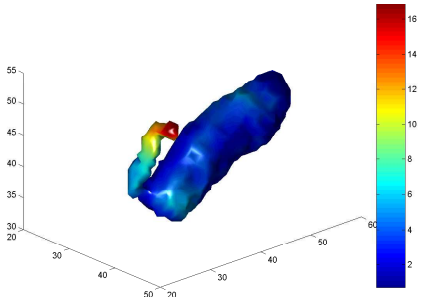

Figure F.11: *Region 6 Unaligned*

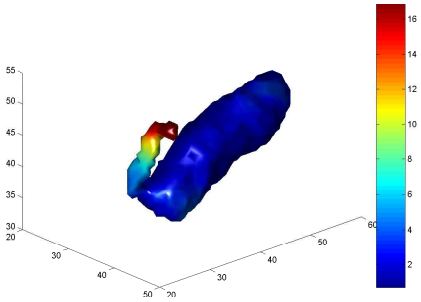

Figure F.12: *Region 6 Aligned*

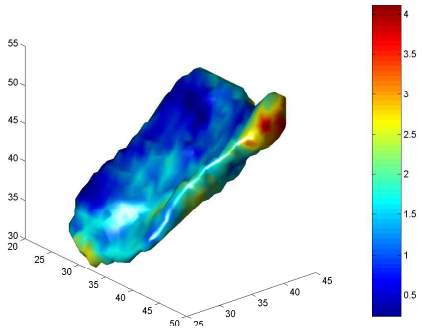

Figure F.13: *Region 7 Unaligned*

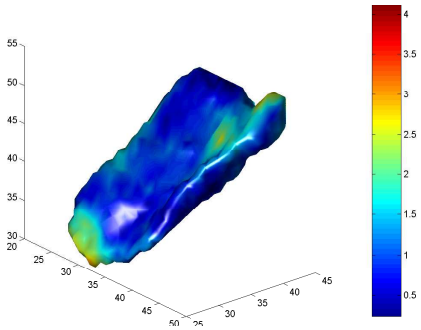

Figure F.14: *Region 7 Aligned*

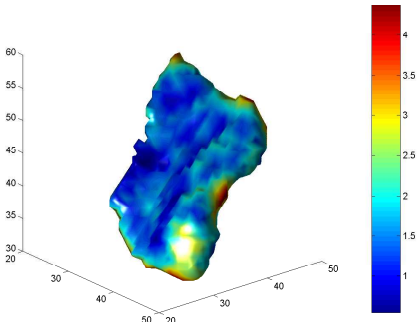

Figure F.15: *Region 8 Unaligned*

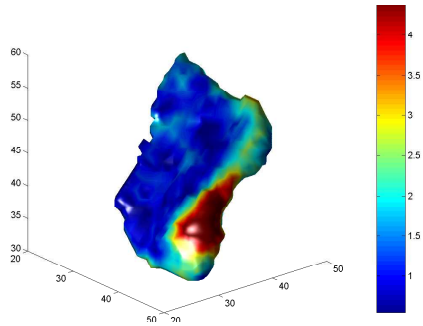

Figure F.16: *Region 8 Aligned*

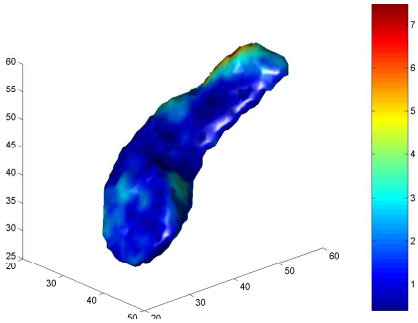

Figure F.17: *Region 9 Unaligned*

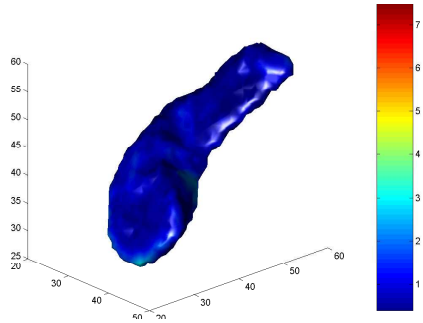

Figure F.18: *Region 9 Aligned*

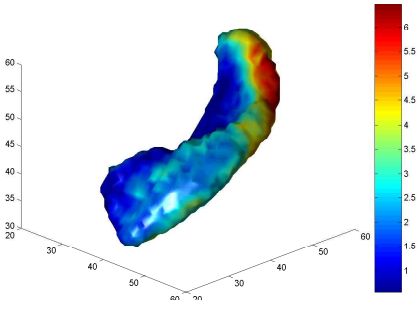

Figure F.19: *Region 10 Unaligned*

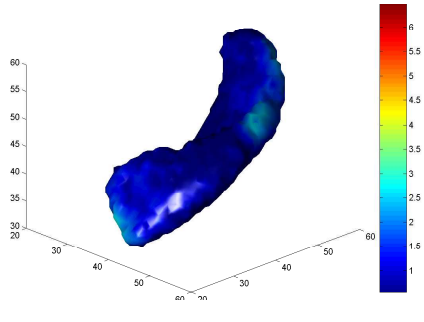

Figure F.20: *Region 10 Aligned*

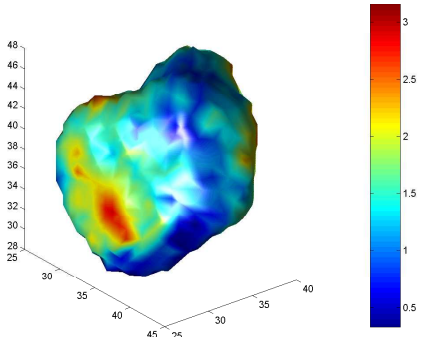

Figure F.21: *Region 11 Unaligned*

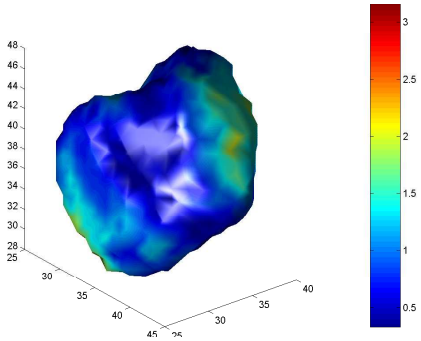

Figure F.22: *Region 11 Aligned*

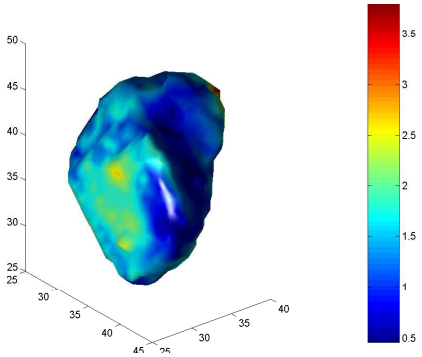

Figure F.23: *Region 12 Unaligned*

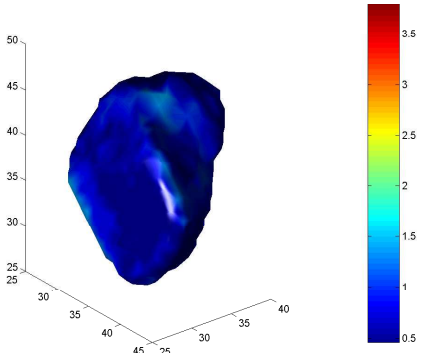

Figure F.24: *Region 12 Aligned*

Appendix G

Probability Distribution Functions

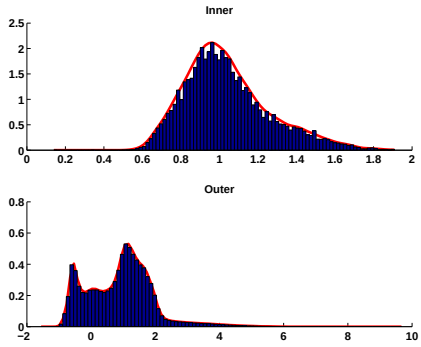

Figure G.1: *Region 1 - Temporopolar Cortex Left*

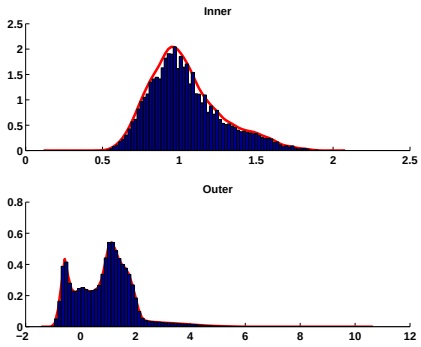

Figure G.2: *Region 2 - Temporopolar Cortex Right*

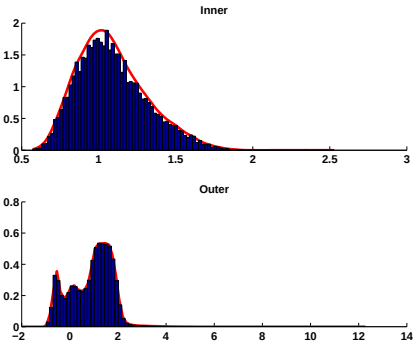

Figure G.3: *Region 3 - Entorhinal Cortex Left*

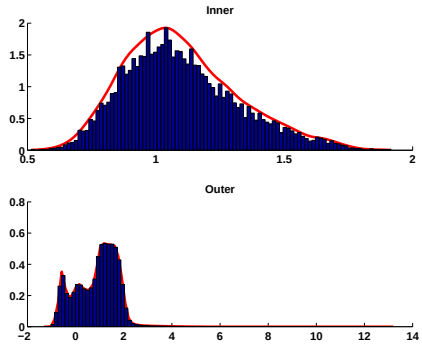

Figure G.4: *Region 4 - Entorhinal Cortex Right*

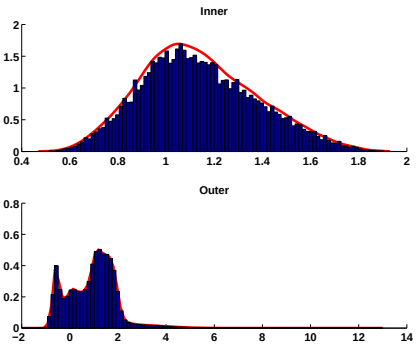

Figure G.5: *Region 5 - Perirhinal Cortex Left*

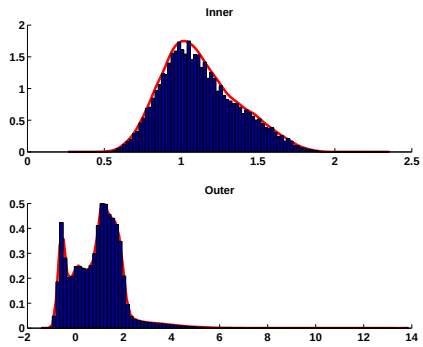

Figure G.6: *Region 6 - Perirhinal Cortex Right*

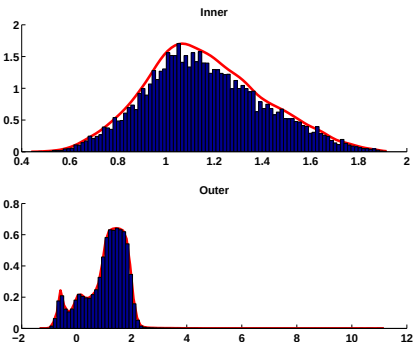

Figure G.7: *Region 7 - Parahippocampal Cortex Left*

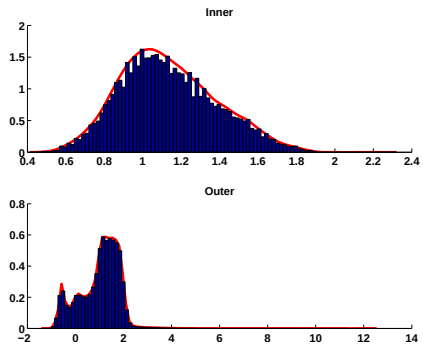

Figure G.8: *Region 8 - Parahippocampal Cortex Right*

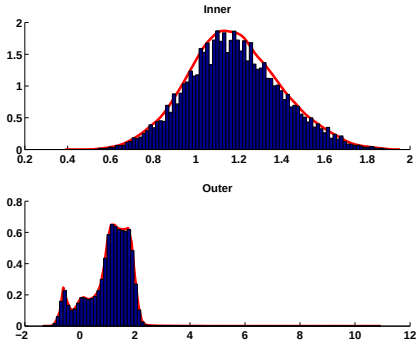

Figure G.9: *Region 9 - Hippocampus Left*

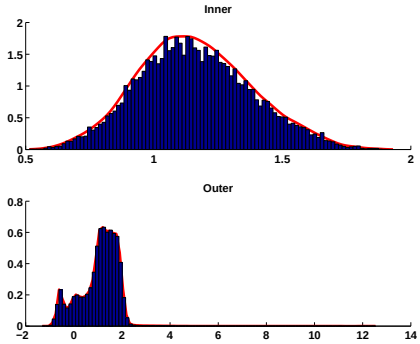

Figure G.10: *Region 10 - Hippocampus Right*

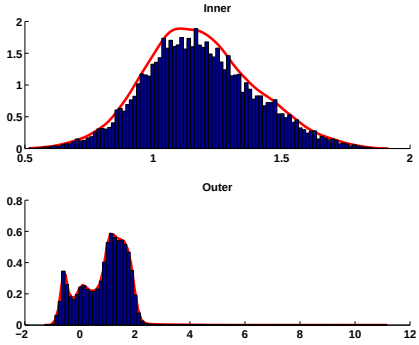

Figure G.11: *Region 11 - Amygdalar Left*

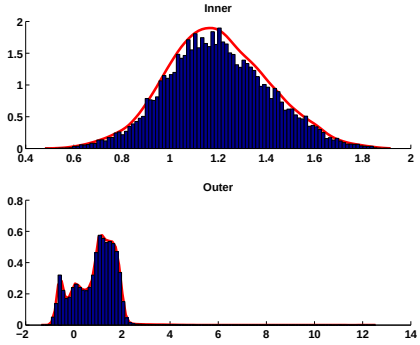

Figure G.12: *Region 12 - Amygdalar Right*

Appendix H

Implementation

On the included CD is the matlab implementation of the different processes in the theses. For reasons of size only two test persons are included on the CD, together with the calculated distance maps. Furthermore is the spm2 toolbox included, to read to ANALYZE images. The code is separated into folders concerning each their area of responsibility. The main-files are

- alignShapes.m - Aligns a set of shapes
- Step1.m - Standardizes data and creates LSFs
- Step2.m - Estimates PDFs
- Step3.m - Aligns all shapes
- make3DModesNB.m - Creates a coupled shape model
- fullscaleRun.m - Runs the segmentation algorithm
- Viewers - Cpp source code for developed viewers - requires VTK

Bibliography

Baker, S. and Matthews, I. (2002). Lucas-kanade 20 years on: A unifying framework: Part 1, tech. report CMU-RI-TR **02**(16).

Ballard, D. H. (1981). Generalizing the hough transform to detect arbitrary shapes, Pattern Recognition **13**(2).

Brodmann, K. (1909). Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellet auf Grund des Zellenbaues.

Caselles, V., Catte, F., Coll, T. and Dibos, F. (1993). A geometric model for actrive contours in image processing, Numerische Mathematik (66): 1–31.

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

Chan, T. F. and Vese, L. A. (2001). Active contours without edges, IEEE Conference on Image Processing **10**(2).

Cohen, L. D. (1991). On active contour models and balloons, CVGIP: Image understanding pp. 211–218.

Cootes, T., Cooper, D., Taylor, C. and Graham, J. (1995). Active shape models - their training and application, Computer Vision and Image Understanding **61**(1): 38–59.

Cootes, T. and Taylor, C. (2004). Statistical models of appearance for computer vision.

Danielsson, P. (1980). Euclidean distance mapping, Computer Graphics and Image Processing (14): 227–248.

Dice, L. (1945). Measure of the amount of ecological association between species, Ecology **26**: 297–302.

Dryden, I. L. and Mardia, K. V. (1998). Statistical Shape Analysis, John Wiley and Sons.

Fisker, R. (2000). Making Deformable Template Models Operational, PhD thesis, Technical University Of Denmark, Institute Of Mathematical Modeling.

Glasbey, C. and Mardia, K. (1998). A review of image warping methods, Journal of Applied Statistics (25): 155–171.

Golland, P., Grimson, W., Shenton, M. and Kikins, R. (2005). Detection and analysis of statistical differences in anatomical shape, Medical Image Analysis (9): 69–86.

Gramkow, C. (1996). Registration of 2d and 3d medical images, Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

Hansen, M. F. (2005). Quality estimation and segmentation of pig backs, Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). The Elements of Statistical Learning, Springer.

Hornak, J. P. (1996-2006). The Basics of MRI, http://www.cis.rit.edu/htbooks/mri/index.html.

Insausti, R., Juottonen, K., Soininen, H., Insausti, A. M., Partanen, K., Vainio, P., Laakso, M. P. and Pitkänen, A. (1998). Mr volumetric analysis of the human entorhinal, perirhinal, and temporopolar cortices, American Society Of Neuroradiology (19): 659–671.

Kass, M., Witkin, A. and Terzopoulos, D. (1988). Snakes: Active contour models, International Journal Of Computer Vision pp. 321–331.

Kim, J., III, J. F., Jr., A. Y., Cetin, M. and Willsky, A. (2002). Nonparametric methods for image segmentation using information theory and curve evolution, IEEE International Conference on Image Processing .

Kotcheff, M. and Taylor, C. (1998). Automatic construction of eigenshape models by direct optimisation, Med. Image Anal. **2**: 303–314.

Leventon, M. E. (2000). Statistical models in medical image analysis, PhD thesis. Supervisor-W. Eric Grimson and Supervisor-Olivier D. Faugeras.

Malladi, R., Sethian, J. A. and Vemuri, B. C. (1995). Shape modeling with front propagation:a level set approach, IEEE transactions on pattern analysis and machine intelligence **17**(2): 158–175.

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

Modersitzki, J. (2004). Numerical Methods For Image Registration, Oxford Science Publications.

Moos, T. and Møller, M. (2006). Centralnervesystemets anatomi, FADL's forlag.

Mumford, D. and Shah, J. (1989). Optimal approximations by piecewise smooth functions and associated variational problems, Commun. Pure Appl. Math **42**: 577–685.

Pitman, J. (1993). Probability, Springer.

Pluim, J., Maintz, J. and Viergever, M. (2003). Mutual information based registration of medical images: a survey, IEEE transactions on medical imaging .

Pohl, K., Fisher, J., Kikinis, R., Grimson, W. and Wells, W. (2005). Shape based segmentation of anatomical structures in magnetic resonance images, In Proc. ICCV 2005: Computer Vision for Biomedical Image Applications **3765**.

Pohl, K., Fisher, J., Shenton, M., McCarley, R., Grimson, W., R.Kikinis and Wells, W. (2006). Logarithm odds maps for shape representation, MICCAI 2006 - Proceedings .

Pohl, K., Kikinis, R. and Wells, W. (2007). Active mean fields: Solving the mean field approximation in the level set framework, Information Processing in Medical Imaging, Lecture Notes in Computer Science, Springer-Verlag **4584**: 26–37.

Pruessner, J. C., Köhler, S., Crane, J., Pruessner, M., Lord, C., Byrne, A., Kabani, N., Collins, D. L. and Evans, A. C. (2002). Volumetry of temporopolar, perirhinal, entorhinal and parahippocampal cortex from high-reolution mr images: Considering the variability of the collateral sulcus, Cerebral Cortex **12**(12): 1342–1353.

Pruessner, J. C., Li, L., Serles, W., Pruessner, M., Collins, D. L., Lupien, S. and Evans, A. C. (2000). Volumetry of hippocampus and amygdala with high-reolution mri and three-dimensional analysis software: Minimizing the discrepancies between laboratories, Cerebral Cortex **10**: 433–442.

Sethian, J. (1999). Level Set Methods and Fast Marching Methods, Cambridge University Press.

Shenten, M., Gerig, G., McCarley, R., Szekely, G. and Kikinis, R. (2002). Amygdala-hippocampus shape differences in schzophrenia: the application of 3d shape models to volumetric mr data., Psychiatry Research Neuroimaging (115): 15–35.

Silverman, B. (1986). Density Estimation for Statistical and Data Analysis, Chapman and Hall.

Sled, J., Zijdenbos, A. and Evans, A. (17). A nonparametric method for automatic correction of intensity nonuniformity in mri data, IEEE Trans. Med. Imaging pp. 87–97.

Sonka, M. and Fitzpatrick, J. M. (2000). Handbook of Medical Imaging Volume 2. Medical Image Processing And Analysis, SPIE PRESS.

T.F.Cootes, Edwards, G. and C.J.Taylor (1998). Active appearance models, Proc. European Conference on Computer Vision 1998 **2**: 484–498.

Tsai, A., Anthony Yezzi, J., Wells, W., Tempany, C., Tucker, D., Fan, A. and Willsky, W. E. G. A. (2003). A shape-based approach to the segmentation of medical imagery using level sets, IEEE transactions on medical imaging **22**(2): 137–154.

Tsai, A., Wells, W., Tempany, C., Grimson, E. and Willsky, A. (2004). Mutual information in coupled multi-shape model for medical image segmentation, Elsevier Science .

Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A. and E Grimson, A. W. (2001). Model-based curve evolution technique for image segmentation, IEEE Conference on Computer Vision and Pattern Recognition .

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition, Journal of Cognitive Neuroscience **3**(1): 71–86.

Wyvill, B., McPheeters, C. and Wyvill, G. (1986). Data structure for soft object, The Visual Computer **2**(4): 227–234.

Index