

Shape Analysis of the Human Hippocampus
Using Spherical Harmonics:
An Application to Alzheimer's Disease

Heike Jänicke

July 31, 2006

Acknowledgements

The work for this thesis was carried out at three institutions, the Max Planck Institute of Human Cognitive and Brain Science (Leipzig), the Karolinska Institute (Stockholm, Sweden) and the Department of Computer Science (University Leipzig), which had a significant influence on the quality of this work. I wish to express my gratitude to a number of people who became involved in one way or another. First, I want to thank Marc Tittgemeyer, who arouse my interest in brain science and to whom I owe the interesting topic. Moreover, I want to thank him for inspiring my ambition. My stay at the Karolinska Institute would not have been possible without Lars-Olof Wahlund, who did not only offer a place to work to me, but also showed great interest in my work and gave me the feeling that I was about to do something important. For many fruitful discussions and lots of motivation I would like to thank Gerek Scheuermann. Many thanks go to the team at the Karolinska, who gave me a warm welcome and showed me many interesting aspects of their daily work. Special thanks go to Eva-Lena Engman, Elin Lundström, Leszek Stawiarz and Susanne Müller, who made my time in Sweden so much more pleasant. Tack so mycket! For his persistent explanations concerning programming and Linux, I would like to thank Gert Wollny. Furthermore, I want to thank the team at the Department of Computer Science for the wonderful atmosphere and the support throughout the work. Thank you Alex, Christian, Julia, Mario and Tobi. Special thanks go to Alexander Wiebel who shared his office with me and cheered me up every day. My time at university would not have been the same without Daniel Müller. Thank you for the wonderful time, the constant interest and encouragement, and for being a perfect friend. Last but not least, I want to thank my family for all the love and support they gave to me.

Abstract

Every year a higher life expectancy is reported for people living in industrial countries. With increasing age, the risk of getting Alzheimer's Disease increases as well. Alzheimer's Disease is a neurodegenerative disease that is characterised by progressive deterioration of brain tissue. One of the first regions in the brain to be affected is the hippocampus. A common method to quantify the deterioration of the hippocampus is to measure its volume. However, decreasing volume is no specific marker of Alzheimer's Disease, but can be caused by other diseases as well. Thus, the deformation has to be observed in more detail, which can be done using shape analysis. A powerful shape analysis technique is the approximation of the surface by means of spherical harmonics. A process chain computing such an approximation is explained in this thesis. Therefore, the data is triangulated, forming a closed manifold. Afterwards, a homogeneous mapping of the surface to the unit sphere is computed in two steps. First an initial spherical parametrisation is computed, which is optimised afterwards to resemble the properties of the initial surface. The optimisation is mandatory, to allow for inter-subject comparability. The optimised parametrisation defines a function on the sphere, that can be approximated by spherical harmonics, a set of basis functions on the unit sphere. This procedure results in a mathematical description of the surface that can be analysed statistically. The method is applied to data of Alzheimer's Disease patients.

Contents

1	Introduction	2
1.1	Motivation and Objectives	2
1.2	Thesis Overview	2
2	Medical Background	4
2.1	Overview	4
2.2	Dementia	4
2.3	Alzheimer’s Disease	5
2.3.1	Discovery	6
2.3.2	Causes	6
2.3.3	Diagnosis	7
2.3.4	Preclinical Detection	7
2.4	The Human Hippocampus	8
2.5	Shape Analysis of the Hippocampus	9
3	Related Work	10
3.1	Overview	10
3.2	Analysis of Volume and Shape	10
3.2.1	Manual Outline	10
3.2.2	Comparison of Intensities	11
3.2.3	Transformation Fields	12
3.2.4	Skeletal Descriptions	13
3.2.5	Spherical Harmonics	13
3.3	Comparison of the Different Methods	14
4	Proposed Solution	17
5	Implementation	19
5.1	Triangulation of Point Clouds	19
5.1.1	Overview	19
5.1.2	Open-Source Software	19

5.1.3	Triangulation of Planar Slices	21
5.1.4	Applied Triangulation Method	23
5.2	Surface Simplification	26
5.2.1	Overview	26
5.2.2	Topological Operator	26
5.2.3	Fairness Criterion	27
5.2.4	Computation of a Mesh-Hierarchy	32
5.3	Spherical Parametrisation	33
5.3.1	Overview	33
5.3.2	Initial Parametrisation	35
5.4	Spherical Optimisation	38
5.4.1	Overview	38
5.4.2	Optimisation	40
5.4.3	Applied Optimisation Methods	44
5.4.4	Spherical Trigonometry	47
5.4.5	Global Approach - Constrained Optimisation	48
5.4.6	Local Approach - Simulated Annealing	53
5.5	Approximation Using Spherical Harmonics	57
5.5.1	Overview	57
5.5.2	Theoretical Background	57
5.5.3	Computation of the Coefficients	61
5.5.4	Rotation Independent Descriptors	61
6	Realisation	63
7	Conclusion	70
	Bibliography	72
A	Definition of Shape	81
B	Spherical Harmonics	82
C	Illustrations	84
D	Pseudocode	94

*“For the harmony of the world is made manifest
in Form and Number, and the heart and soul and
all poetry of Natural Philosophy are embodied in
the concept of mathematical beauty.”*

- D’Arcy Thompson (“On Growth and Form”, 1917)

Chapter 1

Introduction

1.1 Motivation and Objectives

If it were true that the whole world follows the concepts of mathematical beauty (Thompson, 1917), that every object and every phenomenon can be described by a set of functions, then even the most chaotic and most extraordinary processes would have to follow some basic rules. So far, nobody knows, whether there is a general concept behind everything. Nevertheless, trying to understand processes is one of the driving forces of the evolution of mankind.

In medicine, many diseases cause an ongoing change in the body. If this change can be described and quantified mathematically, it would be a promising extension to the way diagnosis are made. One of these diseases that is hard to diagnose from the outside or by laboratory tests is Alzheimer's Disease, a disease that slowly destroys the human brain and that cannot be cured, yet. If the decline can be quantified in early stages of the diseases, the causes and the progress could be analysed more precisely, providing the chance to find a treatment.

1.2 Thesis Overview

Thus, a program was implemented that aims for mathematical descriptions of regions in the brain that are affected by AD. These descriptions are to be used to find an early marker of the disease. To clarify the medical background, Chapter 2 gives an introduction to AD, its progress in the brain and techniques to quantify the progress. Chapter 3 explains and compares different approaches to analyse magnetic resonance (MR) images that have been

used to find an early marker of AD. The solution proposed to find such a marker is given in Chapter 4. The details on the implementation provided in Chapter 5, explain a process consisting of seven steps that results in a mathematical description of the surface of the affected region in the brain. This description can be used to analyse the structure statistically, which might reveal an early marker of AD. The results of the work are summarised in Chapter 6, and Chapter 7 draws a conclusion and explains the future work.

Chapter 2

Medical Background

2.1 Overview

Exactly 100 years ago, Alois Alzheimer described a disease that changed the patients social and cognitive abilities completely. Ever since, scientists tried to discover the causes and a distinct marker of this disease, called Alzheimer’s Disease (AD). But even in times of sophisticated medical examination techniques, these questions remain still unanswered. Answers might be found, when making more extensive use of the comprehensive possibilities of these techniques, e.g. a more precise analysis of MR images by means of shape analysis.

In the following, the medical background of the thesis is explained, starting with an introduction to dementia (Section 2.2) and to AD in particular (Section 2.3). One of the first regions in the brain to be affected is the hippocampus, a structure explained in Section 2.4. Shape analysis of the hippocampus is one of the promising techniques to find a marker of early AD. Thus, an introduction to the basic concepts of shape analysis is given in Section 2.5.

2.2 Dementia

Dementia (from Latin *demens*, ‘out of one’s mind’) is a “decline of intellectual function in comparison with the patient’s previous level of function” (McKhann et al., 1984). This decline is usually associated with changes in behaviour and impairment of social and professional activities.

“Dementia, in its contemporary usage, is an irreversible condition, and is

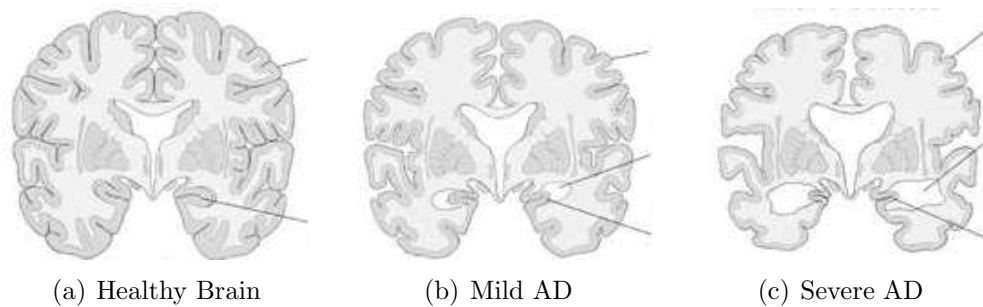


Figure 2.1: Different stages of Alzheimer's disease: (a) A healthy brain, showing no gaps inside the sulci (fissure in the surface of the brain). The lower line marks the hippocampus. (b) Under AD, the brain starts to deteriorate, enlarging the sulci and ventricles (cavity in the brain, filled with fluid). (c) In the final stages, large cavities are visible in the brain tissue (American Health Assistance Foundation, 2006).

not applied to states of mental deterioration that may be overcome, such as delirium. The condition is generally caused by deterioration of brain tissue, though it can occasionally be traced to deterioration of the circulatory system. Major characteristics include short- and long-term memory loss, impaired judgement, slovenly appearance, and poor hygiene. Dementia disrupts personal relationships and the ability to function occupationally" (Heston and White, 1991).

"About 5–10 per cent of the population over 65 have some kind of cognitive decline, which is considered to be abnormal for this age group" (Gauthier, 1996, p. 35).

2.3 Alzheimer's Disease

The most common form of dementia among people older than 65 is Alzheimer's Disease (AD), which "is an irreversible, progressive brain disease that slowly destroys memory and thinking skills, eventually even the ability to carry out the simplest tasks" (Rodgers, 2002). The disease is characterised by an ongoing deterioration of brain tissue, which is illustrated in Figure 2.1. From the onset of symptoms, AD progresses from two to twenty, in average seven, years and always ends in death. At present, the only way to accurately diagnose AD, is to physically examine the brain of a probable AD sufferer after the patient has died.

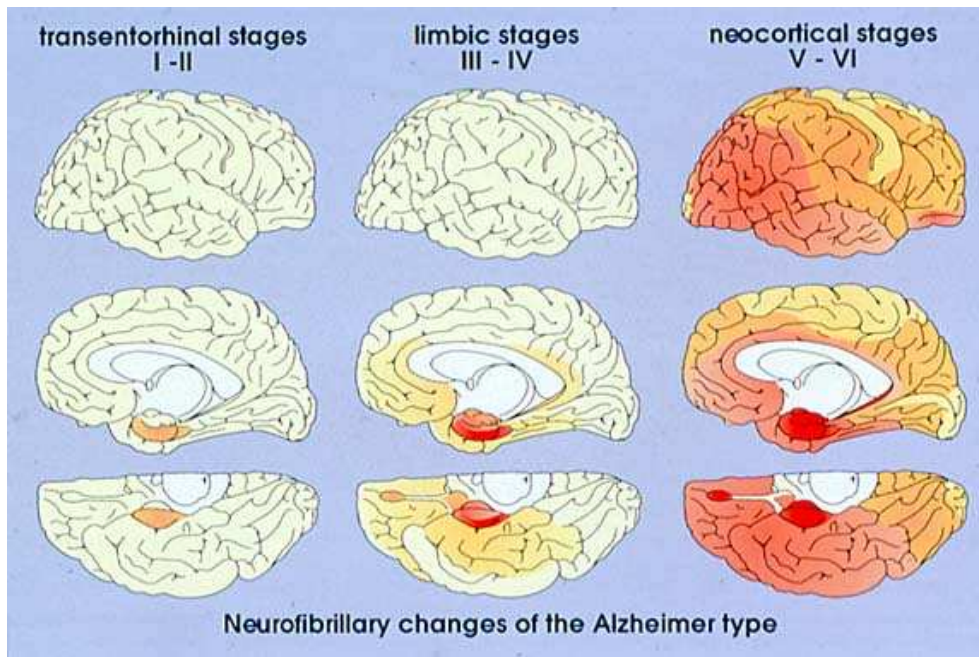


Figure 2.2: The image shows the broadening of neurofibrillary tangles, which are a marker for the deterioration of the brain (Braak and Braak, 2006).

2.3.1 Discovery

AD was firstly described in 1906, when Alois Alzheimer gave a lecture at the 37th Conference of South-West German Psychiatrists in Tübingen, presenting a new form of dementia, he had observed at the 51-year-old patient Auguste D. She showed symptoms of “progressive cognitive impairment, focal symptoms, hallucinations, delusions, and psychosocial incompetence” (Maurer et al., 1997). Necropsy revealed amyloid plaques, neurofibrillary tangles and arteriosclerotic changes.

2.3.2 Causes

Although the disease was discovered nearly 100 years ago, its causes are not yet fully understood. There is probably not a single factor causing AD, but it is a complex, multifactorial process (Mann, 1997; Kamboh, 2004; George-Hyslop and Petit, 2005). The most important risk factor is age.

The likelihood of developing AD increases exponentially after the sixth decade of life (Brayne et al., 1995), such that “by age 85 more than 40% of survivors may be affected” (Terry et al., 1994, p. 179).

Due to declining birthrates and growing life expectations, the proportion of US-citizens 65 years of age and older will increase from currently 12.4% to 18.2% in 2025 and is supposed to have reached 20.7% by 2050 (U.S. Bureau of the Census). Figures for other industrial countries are similar. Thus, the population in industrial countries will consist of more and more people with a high risk of developing AD, which so far cannot be cured.

2.3.3 Diagnosis

A first step towards finding a treatment is diagnosing the disease as early as possible. At present, AD can only be positively diagnosed by performing an autopsy, revealing the density and distribution of characteristic neurofibrillary tangles (NFT) and amyloid plaques. "The diagnosis of 'probable Alzheimer's disease' depends on clinical criteria, that is, a characteristic history, compatible findings on physical and mental status examination, and the exclusion of other disorders that mimic AD by appropriate information from history, examination, and laboratory tests" (Terry et al., 1994, p. 9). To define the disease's severity, clinical instruments such as the Mini-Mental State Examination (MMSE) (Folstein et al., 1975) and Clinical Dementia Rating Scale (CDR) (Morris, 1993) are used. To improve sensitivity and specificity of the antemortem diagnosis, further investigations are needed to reveal more specific markers of AD.

Braak and Braak (1991) propose that the progress of AD can be divided into several stages, as illustrated in Figure 2.2. Their "[...] criterion is based on the distribution of NFT, which start in the transentorhinal area (Stage I), spread to the entorhinal region (II), extend to the hippocampus proper (III), increase in number there (IV), and involve the association neocortex (V) and finally the primary cortex (VI). These stages are divided into the entorhinal (I & II), limbic (III & IV), and neocortical (V & VI) phase, corresponding, respectively, to cognitively normal, cognitively impaired, and dementia" (Murayama and Saito, 2004). Magnetic resonance imaging (MRI) allows in vivo assessment of the macroscopic effects of this pathology, namely changes in shape of the affected brain structures.

2.3.4 Preclinical Detection

In the preclinical stage of AD the subjects exhibit no detectable cognitive changes despite an ongoing, underlying disease process that can be fortified

by pathologic evidence, i.e. neurofibrillary tangles and β -amyloid plaques, (Price, 2003; Csernansky et al., 2005).

In the earliest stages, this pathology is located in the entorhinal cortex and the hippocampus (Braak and Braak, 1991; Arnold et al., 1991). Increased atrophy of these regions has been proven in several volumetric MRI studies (Csernansky et al., 2005; Chan et al., 2001; Convit et al., 1997), even in patients with presymptomatic familial AD (Fox et al., 1996). Correlation between antemortem hippocampal volumes and both dementia severity and the density of hippocampal neurofibrillary tangles at autopsy have been shown (Jack et al., 2002; Csernansky et al., 2004). To assure specificity, distinctness of early AD from both elderly and younger control subjects is needed. This was verified by Csernansky et al. (2000) and Wang et al. (2003).

These results suggest that shape analysis of the hippocampus can be used to detect AD in predementia stage.

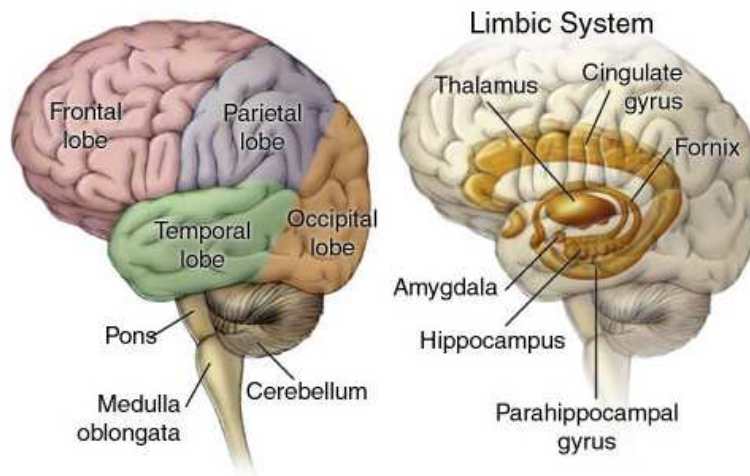


Figure 2.3: Gross anatomy of the human brain (left) and the limbic system (right) (American Health Assistance Foundation, 2006).

2.4 The Human Hippocampus

One of the first regions to be affected by AD is the hippocampus, which is a part of the brain. It is situated inside the temporal lobe and belongs to the limbic system. Formation of new memories and processing of spatial

information are essentially influenced by the hippocampus. Its name derives from the Greek word hippocampus, which means seahorse, referring to the rolled shape, which resembles the appearance of the sea dwellers.

2.5 Shape Analysis of the Hippocampus

Volumetric changes of the hippocampus under AD have already been observed successfully. Unfortunately, decreasing volume is not a specific marker of AD, as this pathology can be observed in other diseases such as Epilepsy (Garcia-Finana et al., 2006), Schizophrenia (McDonald et al., 2006) and Parkinson's disease (Bruck et al., 2004) as well. Thus, the structure of the hippocampus has to be observed in more detail.

A promising approach of in-vivo investigation is shape analysis of the structure. Commonly, shape stands for the surface of an object or creature, an outline or a contour. It is supposed to be "something distinguish from its surroundings by its outline" (The American Heritage Dictionary of the English Language, 3rd Edition). A more mathematical definition of shape is given in Appendix A.

As shape analysis is only concerned with an object's surface and is independent of size, it is well suited to investigate not only intra-subject changes, but gives the opportunity to compare different subjects. To fulfill this task, a mathematical description of the shape is needed, providing characteristic parameters, that can be compared and analysed.

In the following chapter, five different approaches are explained, that can be used, to quantify changes of the hippocampus.

Chapter 3

Related Work

3.1 Overview

Several methods have been proposed, that relate certain parameters to the progress of AD. Section 2.3.4 shows that analysis of the structure of the brain can be used to diagnose AD. A short introduction to the different methods along with their advantages and disadvantages is given in Section 3.2, while Section 3.3 compares the different methods.

3.2 Analysis of Volume and Shape

3.2.1 Manual Outline

Several techniques have been applied to give a mathematical description of the structure of the hippocampus. A first intuitive approach towards evaluation of MRI-structures is the measurement of volume, which might explain global atrophy or dilation due to illness.

Currently, the gold standard in measuring volumetric changes is manual segmentation of the hippocampus. For this purpose an expert rater outlines the hippocampus' boundary on each slice of the MR-scan manually. Counting the voxels inside the border on each slice results in the approximate hippocampal volume. This volume is normalised, i.e. divided by the intracranial volume, to correct for individual differences in brain size (Lehericy et al., 1994; Convit et al., 1997; Insausti et al., 1998; Jack et al., 2000; Laakso et al., 2000). This method links only one parameter, volume loss, to the progress of AD, providing an intuitive interpretation. Local changes over time can be investigated by comparing the deformation of the outlines of the same regions at

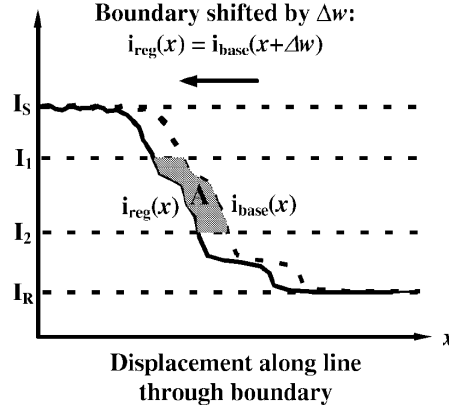


Figure 3.1: A one-dimensional representation of a boundary shift between a baseline scan, $i_{base}(x)$, and a registered repeat scan, $i_{reg}(x)$: An estimate of the shift, $\Delta\omega$, may be obtained as the area A divided by $|I_1 - I_2|$ (the intensity window) (Freeborough and Fox, 1997).

different timesteps (Laakso et al., 2000), but do not provide full insight into deformation, since only two dimensions are used. Furthermore, this method needs extensive time and human training, which still bears the risk of poor intra-rater reproducibility and inter-rater reliability.

3.2.2 Comparison of Intensities

These problems are overcome by semiautomated segmentation using brain boundary shift integrals (BBSI) or deformable shape models, measuring changes over time.

The BBSI compares pixel intensities on the border of the region of interest. Thus, correspondance of pixels to be compared has to be established first. Using the BBSI as introduced by Freeborough and Fox (1997), manual outline is performed only in the initial scan. Further intrasubject images are normalised with respect to intensity distributions and rigidly registered to this baseline scan to ensure correspondance. The boarder region of the hippocampus is computed, using morphological operations. Finally, changes in voxel intensities are added up throughout the boundary region, i.e. the differences of intensities between the baseline and the follow-up image are accumulated (Fox and Freeborough, 1997; Barnes et al., 2004).

A further improvement was proposed by Ezekiel et al. (2004), taking tissue

classification and a modified bias field correction into account. Like the manual outline, BSI is a promising, easy to interpret marker of the progress of AD and moreover eliminates the problems of the former method. However, there are a couple of limitations. Firstly, BBSI needs manual classification of voxels belonging to brain tissue, which are used for normalisation. This limits the ability to compare different raters and laboratories. Secondly, effects of between-scan differences in voxel intensity may bias the results, as well as, mis-registration and nonlinear changes, which lead to under-report of changes.

Another approach using differences in intensities was taken by Chen et al. (2004, Principal Component Analysis (PCA)), who plotted intensities of corresponding voxels from two scans in a two dimensional graph and analysed it statistically to find outliers, that are voxels that changed tissue affiliation. PCA achieves very good results in detecting small volumetric changes and moreover, is robust with respect to intensity shifts. Nevertheless, this shape description technique reflects only global changes and lacks information about particular locations.

Improving the ideas of BBSI and an earlier approach, which used digital subtraction of serially acquired coregistered MR scans, Freeborough and Fox (1998) proposed that fluid registration in combination with the comparison of intensities may permit better localisation of structural change (see also Kubicki et al., 2002).

3.2.3 Transformation Fields

Along with the estimation of differences in voxel intensities, elastic- or visco-elastic registration (Wollny and Kruggel, 2002) was also used to analyse the deformation fields resulting from the registration.

Assuming that an atrophic brain holds the physical properties of a fluid, i.e. gradual and continuous structural changes, a subset of nonlinear warping techniques was proposed to model changes of the brain over time (Christensen et al., 1996; Freeborough and Fox, 1998). Fluid registration results in a transformation vector field describing three-dimensional voxel displacements, which encode voxel-level compression or expansion. Due to its high dimensionality, the resulting matrix is hard to analyse.

Different techniques have been used to find characteristic parameters. Firstly, the volumetric change can be evaluated by integrating the Jacobian determi-

nant derived from the deformation field over the structure of interest (Crum et al., 2001; Janssen et al., 2005). Secondly, the high dimensionality of the result can be reduced by calculating the complete set of eigenvectors of a pooled covariance matrix from the transformation vector fields. A linear combination of a subset of eigenvectors serves as discriminator between patients and control group (Csernansky et al., 1998; Tepest et al., 2003). A standard technique is to use a colour overlay on the region of interest to visualise volumetric changes, which provides a good overview of the results.

All these methods compute a fast and even more important a deterministic result. However, details in the baseline segmentation will be propagated forward to the repeat scan, keeping wrongly labeled voxels. Susceptibility to significant topological changes and noisy or artifactual images, and sensitivity to the initial position are further weak points of this approach.

3.2.4 Skeletal Descriptions

Pizer et al. (1999) described a technique to model object surfaces using a skeletal representation. Applying this technique to shape analysis, a model medial shape description (M-Rep) derived from a set of training datasets is fitted to the objects to be investigated. Thus, each skeleton consists of the same number of components, differing just in lengths and angles. These two features are used to analyse changes statistically (Golland et al., 1999; Joshi et al., 2002; Gerig et al., 2003; Styner et al., 2004). A similar approach was chosen by Narr et al. (2004) and Thompson et al. (2004), who derive the individual skeletons from the objects' surface meshes. Comparability of the different skeletons is provided using anatomical mesh modeling methods.

As the parameters are very intuitive, the interpretation of both methods is straightforward. However, like in the analysis of vector fields, changes are only described locally. Holistic analysis requires a large set of skeletal compounds, where coherence can only be analysed if all the different features are combined in a large model. Moreover, skeletonisation schemes are known to be unstable and sensitive to boundary noise. A solution to this problem was proposed by (Styner and Gerig, 2000), who combined M-Reps and spherical harmonic descriptions.

3.2.5 Spherical Harmonics

A complementary representation is given by the use of spherical harmonics (spharms), which provides a parametric boundary description. With this

method the surface of an object of spherical topology is approximated using a set of orthonormal functions defined on the unit sphere. Using a linear combination of these basis functions, any function on the sphere can be approximated. The more functions are used, the finer details can be represented. (Brechtbühler et al., 1995; Quicken et al., 2000; Gerig et al., 2001; Shen et al., 2003)

Advantages of this approach are the small number of parameters that are needed to describe a complicated surface, and the fact that the level of detail can be increased easily by taking additional parameters into account. Thus, the representation can focus at major and subtle changes at the same time. Unfortunately, these parameters are hard to interpret and do not provide a simple image for human vision.

3.3 Comparison of the Different Methods

The big advantage of the first four methods is the fact that they stick close to human vision and can therefore be interpreted easily. Unfortunately, these straightforward approaches have a number of limitations. Although, volume measurement via manual outline achieves good results, it can hardly be generally applied as it is very time consuming. Moreover, volumetric changes are no unique marker of AD, but can also be caused by other diseases or nutrition. Just like the comparison of intensities, volumetric analysis neglects a big part of the information provided by form, i.e. changes in shape. This drawback is overcome by the analysis of deformation fields, which also provide a fast and deterministic result. However, in order to calculate deformations or intensity differences, the images have to be registered beforehand, including all associated difficulties. Besides, deformation fields have to cope with their high dimensionality. The most promising techniques to analyse shape are descriptions using skeletons or spharms, which use complementary approaches. While the former is more concerned with changes at a certain shape position, the parametric description of the latter focuses on global changes. Thus, skeletons are suited very well, if a certain part of an object is to be investigated, while interaction of the whole surface can be described easily using spharms. Another advantage of spharms is the easily achieved change of level of detail, just by taking more parameters into account. The major disadvantage of spharms is the interpretation of the parameters, which are beyond human experience.

As is summarised in Table 3.1, all of these methods imply advantages and

disadvantages. Thus, no method can be considered to be optimal. The goal of this work is to analyse the changes of the human hippocampus during the progress of AD. Thus, spharms were chosen, as they reflect holistic changes of the hippocampus, which are supposed to discriminate the deformations under AD from those of other diseases best.

	Deterministic	Low Number of Parameters	Easy to Interpret	Time Consuming	Needs Registration	References
Manual Outline		X	X	X		Laakso et al. (2000) Jack et al. (2000)
BSI		X	X	(X)	X	Fox and Freeborough (1997) Etzeziel et al. (2004)
Deformation Fields	X		(X)		X	Freeborough and Fox (1998) Csernansky et al. (1998)
Skeletal Descriptions	(X)	(X)	X	(X)		Gerig et al. (2003) Thompson et al. (2004)
Spherical Harmonics	(X)	X		(X)		Brechtbühler et al. (1996) Gerig et al. (2001)

Table 3.1: Comparison of the described analysis methods with respect to desired (green) and undesired (red) properties. X denotes, that the method holds the property, and (X) states, that the property is partly given, depending on the used procedure, e.g. BBSI, Skeletal Descriptiona and Spherical Harmonics are time consuming, if manual outlines have to be drawn first.

Chapter 4

Proposed Solution

Spharms are a powerful surface description technique. They provide a holistic approximation of the surface, that can be represented easily on different levels of details. Thus, information concerning major and subtle changes are captured by the same description.

In the following work the analysis of changes of the hippocampi under AD will be accomplished using statistical analysis of the coefficients of the spharms series expansion. Although these parameters are hard to interpret it is a promising approach, since all the variety of holistic shape change is covered by a few numbers.

The analytic procedure consists of seven steps:

1. **Preprocessing of the data:** The hippocampal outlines are extracted from MRI scans. Thus, the patients have to be scanned, and an expert rater has to outline the hippocampi in each dataset. The outlines are refined using spline interpolation.
2. **Triangulation of coplanar slices:** Shape analysis needs the surfaces of the hippocampi. Therefore, the outlines are triangulated, forming a manifold surface mesh.
3. **Computation of a mesh-hierarchy:** Quicken et al. (2000) proposed a multi-level parametrisation approach, to map surfaces on the unit sphere. Thus, the surfaces have to be provided at different levels of detail.
4. **Spherical parametrisation** gives an initial mapping from the mesh in Cartesian space to the unit sphere.

5. **Spherical optimisation** is used to receive a uniform distribution of the mesh on the unit sphere, which is mandatory to ensure a homogeneous representation of the surface.
6. **Spherical harmonics representation** is used to approximate the surface function on the unit sphere.
7. **Statistical analysis** of the coefficients of the spharms series expansion.

The first six steps are explained in the following chapter. Due to temporal restrictions, the spherical optimisation could not be finished. Thus, a statistical analysis of the coefficients is not possible, as the coefficients would be biased.

Chapter 5

Implementation

5.1 Triangulation of Point Clouds

5.1.1 Overview

Commonly, shape analysis needs an outline shape, which is a contour or a surface (Section 2.5). However, manually segmented data consists only of single points. Thus, the data points P in Cartesian space have to be triangulated, i.e. a set of triangles has to be found forming the convex hull of P , with all points from P being among the vertices of the triangulation. Moreover, the triangles have to form a closed manifold, i.e. a surface without ‘holes’, on which every point has a neighbourhood, which resembles Euclidean space.

Many algorithms have been proposed in the field of surface reconstruction. Some of the most popular methods have been implemented in open-source software. Four of these programs have been applied to the given data. Section 5.1.2 summarises the results of the tested open-source triangulation programs.

As none of these techniques created satisfying meshes, Section 5.1.3 provides an overview over the results of the research, concerned with triangulation of planar slices. The implemented method is explained in Section 5.1.4.

5.1.2 Open-Source Software

A large variety of triangulation algorithms have been implemented and are available as open-source software. Owen (2006) gives an overview over “Triangle Mesh Generation Software” on his homepage, classifying the programs

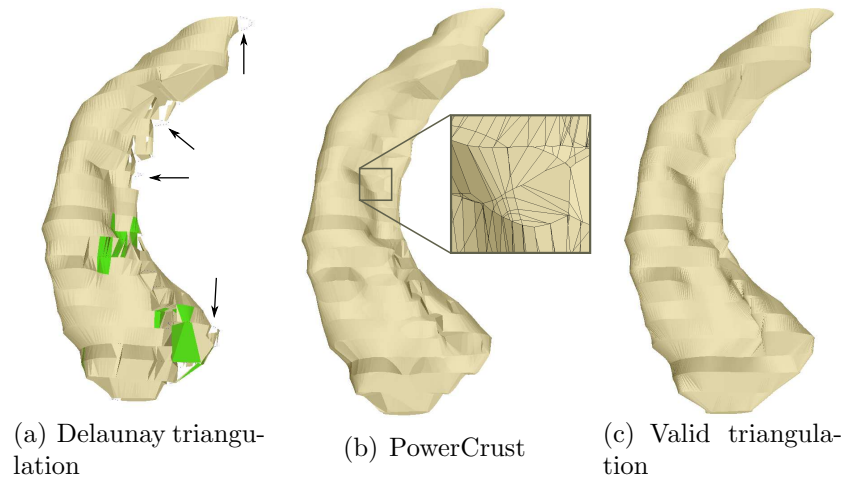


Figure 5.1: Reconstruction of the hippocampal surface from a point cloud: (a) The program using Delaunay triangulation generates invalid triangles (green) and omits points (arrows). (b) The power crust algorithm introduces additional points. (c) A triangulation computed by the algorithm explained in Section 5.1.4

according to availability, supported platform, the applied method, and several other features. A second collection is given by Schneiders (2006), describing the basic idea of the different programs in one sentence, and linking to their homepage.

Suitable programs were selected according to the following criteria:

- The program has to run under Linux.
- The resulting surface has to be a (orientable) closed manifold.
- All given points have to be used as vertices.
- No additional points are allowed to be inserted. (This feature was in some programs hard to verify beforehand.)

Several programs were chosen for testing, amongst others

- “Alpha Shapes” by N. Akkiraju et al., using alpha shapes,
- “PowerCrust” by Amenta et al. (2001), based on the Power Crust algorithm,
- “Tight Cocone” by Dey and Goswami (2003), using Delaunay triangulation, and

- an inhouse-tool from the “Max-Planck-Institute of Human Cognitive and Brain Sciences”.

Most programs failed at computing a valid triangulation, which is mainly caused by the fact, that they require more or less equally distant points. The data was obtained by manual outline of the structure, given by magnetic resonance imaging (MRI), and spline interpolation. Thus, the points on each plane are very dense, whereas points on different planes exhibit great distances. Problems, that occurred during triangulation are displayed in Figure 5.1. The first triangulation contains holes (marked by arrows) and invalid triangles (colored in green), a common problem, when applying Delaunay triangulation to noisy or inhomogeneous data. The second image shows a triangulation introducing new vertices, which falsifies the data. Although the in-house tool produced a valid triangulation, it could not be used, as the triangles were poorly conditioned, causing frequent self-intersections in the simplification process explained in Section 5.2. As none of the investigated programs was able to construct satisfying surfaces, a new algorithm had to be implemented.

5.1.3 Triangulation of Planar Slices

Extensive research has been undertaken in the field of 3D-surface reconstruction, including the special case of triangulation of planar contours (Meyers et al., 1992), as present in the given data. Ekoule et al. (1991) gives a good summary of the different techniques used to triangulate planar slices. Later research in this field focuses mainly on more complicated problems, like the triangulation of slices with several unconnected outlines.

Solutions to the triangulation problem can be divided into two groups: optimal and heuristic approaches. While optimal solutions provide the best triangulation, heuristic methods are computationally less expensive. For the following work, a heuristic approach was chosen, using a local decision criterion.

The first approach is based on the idea of Christiansen and Sederberg (1978), where triangle strips connecting two slices are created iteratively. First, two start vertices, one on each slice, are chosen, with minimal distance. These vertices are now considered to be the end vertices of the triangulation. In each iteration the shortest edge between an end vertex and the neighbour of the other end vertex is used to form a new triangle (compare Figure 5.2 (c), (d)). This procedure is continued, until the start vertices are reached.

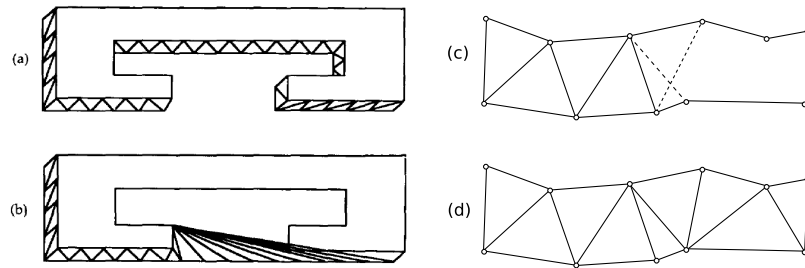


Figure 5.2: Example of links provided by Christiansen's algorithm. (a) Correct results for two contours having a similar shape. (b) Incorrect results for two dissimilar contours. (Ekoule et al., 1991) (c) Selection of the closest vertex. (d) Final triangulation.

Ekoule et al. (1991) had shown that the triangulation might result in non-manifold surfaces, if the two slices have dissimilar shape (Figure 5.2 (b)). Nevertheless, this method was implemented, as a visual analysis of some data sets revealed only tolerable differences in successive slices. The application of the implemented algorithm showed that self-intersecting surfaces occurred in the present data as well. In one third of the data sets, some of the successive slices showed big differences in shape, size and the position of the centre of mass. Ekoule et al. (1991) proposed to overcome this problem by transforming each contour into a convex one. Afterwards the contour with fewer points was selected, and each of its points was linked to the closest point on the other slice.

This method overcame some of the previous problems, but was still not able to triangulate two slices correctly, that diverged strongly in size and the position of the centre of mass (Figure 5.3(a)). As most of the problems occurred when points were linked to the opposite side of the succeeding slice, the next step was to align the centres of mass along one axis. Though this technique computed correct manifold meshes, the program needed further improvement, as distorted triangles emerged due to the translation of the slices. Distortions had already shown to cause problems in further computations, when using the meshes of the in-house tool.

As this straightforward approach does not work, a method is needed to determine, whether two points are on the same side of the surface or not. Commonly, the direction of a surface element is determined using its normal. In the present case, the bisectors through each vertex are used as normals.

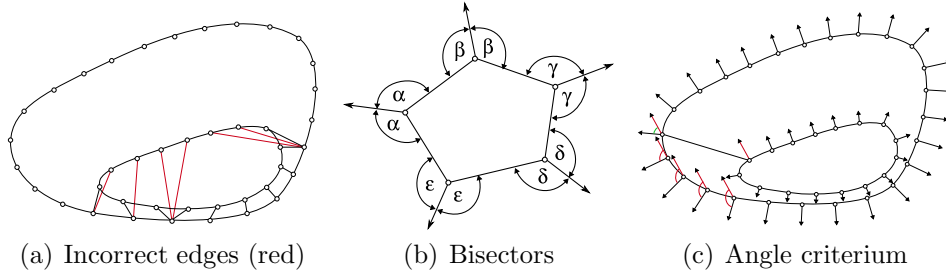


Figure 5.3: Application of the Angle Criterion: (a) Shortest edges between the vertices. Incorrect links between opposite sides of the surface are marked in red. (b) Bisector of the outline. (c) Application of the angle criterion, which enforces links between the same sides.

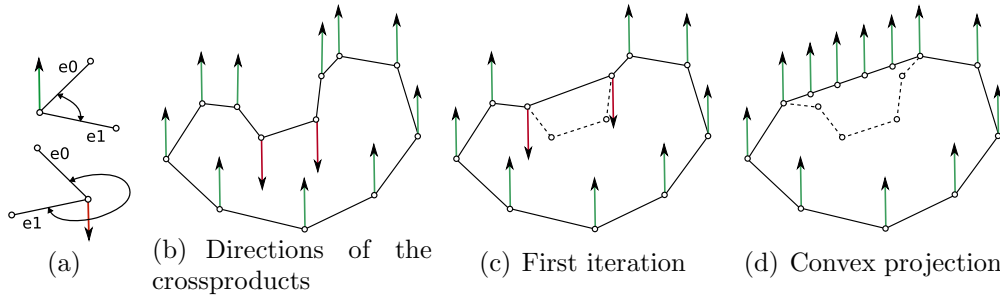


Figure 5.4: Computation of a convex projection: (a) The right-hand rule for convex (top) and concave (bottom) regions. (b) The cross product belonging to each vertex. (c) The first iteration of the elimination of concave parts. As can be seen, the inversion has not been fully excluded, yet. (d) Final state of the elimination process.

The bisector through point p lies in the same plane as the outline and bisects the outer angle between the two edges adjacent to p (Figure 5.3(b)). These vectors are used to decide, whether two points are on the same side. If the angle between the two bisectors is smaller than a defined bound, then the vertices can be linked (Figure 5.3(c)).

Starting from these initial links, the triangle strip is generated by connecting the remaining vertices. The three different configurations that can occur and their triangulations are illustrated in Figure 5.5.

5.1.4 Applied Triangulation Method

Combining all steps explained in the previous section, the algorithm consists of the following five steps (Figure 5.6):

Rearrangement of the outlines: The points of the preprocessed data are ordered only with respect to slices, i.e. points belonging to one slice are stored successively, whereas the slices can be in arbitrary order and the points can be sorted either clockwise or counterclockwise. In this first step, all slices are ordered according to increasing height levels (y-coordinate), and the points are stored in clockwise order.

Convex projection of the outlines: The convex hull is created by first eliminating all vertices, being located on concave parts of the outline, and by inserting them afterwards into the convex outline (Figure 5.4).

Convexity is determined using the cross product of the two adjacent edges. The cross product can be represented graphically, with respect to a right-handed coordinate system. If the inner angle between the two edges is smaller than 180° , i.e. the region is convex, then the y-coordinate of the crossproduct is positive, and the new vector points upwards (Figure 5.4(a) (top)), otherwise negative and the vector points downwards (Figure 5.4(a) (bottom)). Using this method, all vertices belonging to concave regions are eliminated iteratively. An iterative approach is needed to detect inversions completely, as their boundaries are convex (Figure 5.4(b)). The algorithm propagates the new front, until the inversion is fully excluded (Figure 5.4(c) and 5.4(d)). Afterwards, the excluded vertices are distributed equally on the straight line between their convex neighbours.

Computation of bisectors: Starting from this convex hull, the bisectors are computed, as described in the previous section.

Connection of the closest vertices: The shortest edges are computed for each pair of successive slices separately. In the outline with fewer vertices, each vertex is linked to the closest vertex in the other outline, that does not violate the angle criterion. The angle criterion ensures, that the bisectors of two vertices enclose an angle, that is smaller than a predefined bound, e.g. 90° (Figure 5.4(d)).

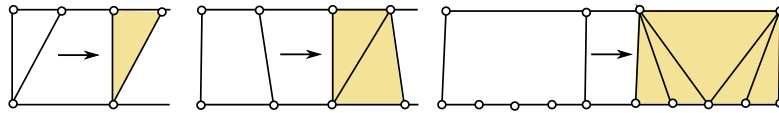


Figure 5.5: Triangulation of linking edges.

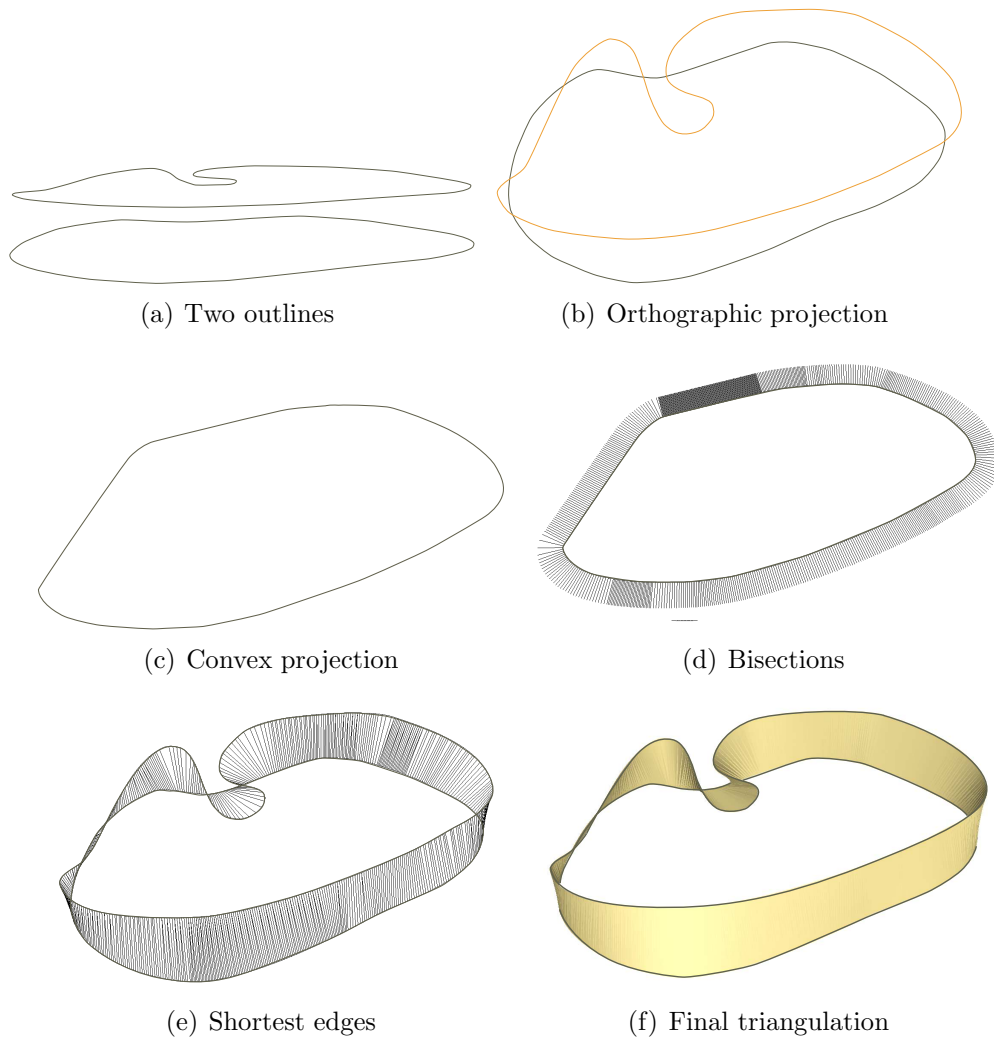


Figure 5.6: Steps of the triangulation process: (a) Two successive outlines of a hippocampus. (b) Orthographic projection of the two outlines, showing the differences in shape. (c) Convex projection of the upper outline. (d) Bisections of the outline. Each bisection divides the outer angle between two neighbouring edges in the outline. (e) The shortest edges between the two outlines. (f) The final triangulation of the two outlines.

Triangulation of the surface: The links are triangulated as illustrated in Figure 5.5.

This algorithm was applied to all present data sets. The triangulated meshes of the left and right hippocampi (left and right) of six different patients are depicted in Figure 6.3.

5.2 Surface Simplification

5.2.1 Overview

The triangular meshes of the hippocampi are to be described using spherical harmonics (spharms). Besides, a spherical parametrisation of the surface meshes is required. A solution to this problem was given by Brechbühler et al. (1995), who solved the parametrisation problem by means of constrained optimisation. Unfortunately, this approach is only applicable to quadrilateral meshes consisting of no more than a few thousand vertices, as proven by Quicken et al. (2000). Quicken et al. (2000) proposed a more stable algorithm using multi-resolution meshes. As the implemented algorithm will follow this approach, the meshes of the hippocampi have to be reduced iteratively first, defining a mesh hierarchy.

Many concepts for polygonal surface simplification algorithms have been proposed, mainly in the field of computer graphics (Heckbert and Garland, 1997). In general, these methods consist of two parts. First an algorithm chooses the next vertex to be eliminated and secondly, a topological operator is applied, which removes the chosen vertex and remeshes the affected patch. If these two methods are applied alternatingly, the mesh can be reduced to an arbitrary number of vertices.

In the following, the two methods required by the reduction algorithm, i.e. the topological operator (Section 5.2.2) and the applied fairness criterion (Section 5.2.3), are described. The final section explains how these two methods are used to reduce the surface mesh.

5.2.2 Topological Operator

A topological operator is a function modifying the topology, i.e. convergence, connectedness, and continuity of a structure. In the following, topological operators are methods to modify manifold meshes. They are used to delete

or insert a vertex and to retriangulate the affected patch, so that the mesh is a manifold again.

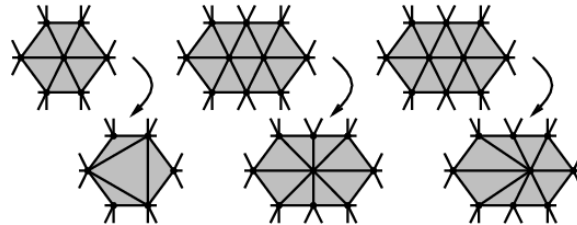


Figure 5.7: Vertex-Removal, Edge-Collapse and Half-Edge-Collapse

Kobbelt et al. (1998) compared three different topological operators called vertex-removal, edge-collapse, and half-edge-collapse, which are visualised in Figure 5.7. The vertex-removal operator deletes a single vertex and retriangulates its crown (Figure 5.7 - left). To guarantee a reasonable quality of the retriangulated patch when using vertex-removal, local optimisation is required. The collapse of two vertices into a single one is called edge-collapse (Figure 5.7 - middle), where the position of the resulting vertex has to be found using local optimisation. A special case of the second operator is the half-edge-collapse, as the resulting vertex has the same position as either one of the original vertices, depending on the direction of the half-edge.

Kobbelt et al. (1998) state that the underlying topological operator, on which an algorithm is based, does not affect the results significantly. Thus, they suggest to use the simplest one, i.e. half-edge-collapse, as it eliminates all geometric degrees of freedom. In the following, this operator is used, as it is easy to use and, what is even more important, does not introduce new vertices. More important than the type of the operator is the decision of where to apply it, to resolve a good mesh quality.

5.2.3 Fairness Criterion

This decision is made applying a fairness criterion, which assigns a predicate to each contemplable half-edge. The half-edge with the best predicate, i.e. least costs, is chosen for the next half-edge collapse. The criterion is described in the following section.

As the topological operator affects the surface only locally, the fairness criteria is based on local surface properties as well. “From differential geometry we

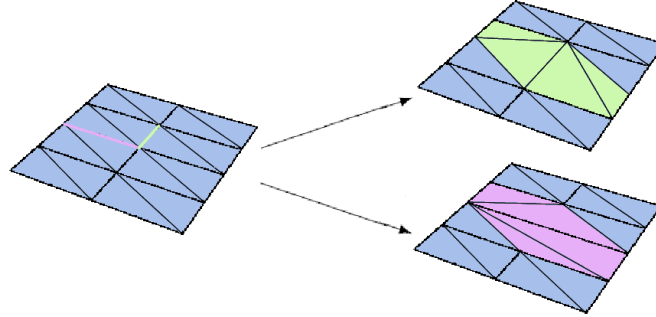


Figure 5.8: Local Distortion: (left) The original part of the mesh. (right-top) The less distorted patch after merging the green edge. (right-bottom) The more distorted patch after merging the purple edge.

know that the first and second fundamental form characterize the behavior of a surface sufficiently well for most applications. Here, the first fundamental form accounts for the local distortion within a parameterized surface, i.e. the mapping of lengths and angles, while the second fundamental form provides complete information about local curvature.” (Kobbelt et al., 1998) Transferring these concepts to the discrete setting of triangular meshes, Kobbelt et al. (1998) propose the following two measures :

- Local Distortion: can be measured using the roundness of the triangle, i.e. the ratio of the inner circle radius to the longest edge.
- Local Curvature: can be estimated by adding up the dihedral angles between adjacent faces.

Quicken et al. (2000) modified the measurement of local curvature in order to decrease computational costs:

- Local Curvature: can be estimated computing the reciprocal of the sum of all angles between normals of adjacent faces in the affected patch.

Local Distortion

The goal of the measure for local distortion ($c_{round_{j,k}}$) is to improve the condition of the triangles after changing topology. The condition of a triangle is supposed to be good, if it resembles an equilateral triangle (= ‘roundest’

triangle), as these kinds of triangles cause less difficulties in the optimisation process, e.g. less flipped triangles. Thus, in the reduction process edge-merges are preferred that create ‘round’ triangles, when applying the topological operator. Figure 5.8 shows the resulting patches of two different merging directions of the same vertex. The resulting triangles in the lower case are much more distorted than the ones in the upper case, which would be penalised by the local distortion measure (Equation (5.1)).

$$c_{round_{j,k}} = \sum_{i \in \Delta_{j,k}} \frac{\max(\|\vec{a}_i\|, \|\vec{b}_i\|, \|\vec{c}_i\|) \cdot \sqrt{s}}{\sqrt{(s - \|\vec{a}_i\|)(s - \|\vec{b}_i\|)(s - \|\vec{c}_i\|)}} \quad (5.1)$$

$$s = \frac{1}{2} (\|\vec{a}_i\| + \|\vec{b}_i\| + \|\vec{c}_i\|),$$

where \vec{a}_i , \vec{b}_i and \vec{c}_i are the edges, forming triangle i . Equation (5.1) measures the roundness of the triangles in $\Delta_{j,k}$, i.e. the set of triangles i , which remain in the affected patch after collapsing vertex j in vertex k (purple/green triangles in Figure 5.8). The more distorted triangles emerge in the affected patch after the collapse, the higher the costs for this operation, worsening the predicate for the current half-edge get.

Local Curvature

Unlike distortion local curvature is measured before applying the operator, as this measure is used to ensure outer fairness, i.e. characteristic shape formations are to be preserved. As shown in Figure 5.9, two different surfaces can result in identical simplified patches. Whereas the local curvature is preserved in the upper case, i.e. both patches are planar, it changes in the lower one. The measure for distortion would treat both of them equally, since they give the same result after the topological operation. In order to keep characteristic surface properties and increase the accuracy of the goal areas, the upper one is to be preferred. Thus, the second measure preserving local curvature was introduced, which penalises the merger of vertices belonging to a patch of high curvature.

The curvature of a patch can be measured using Equation (5.2), which computes the angles between the normals $\vec{n}_{j,l}$ of adjacent faces l surrounding vertex j . The parameter $\beta > 1$ controls the sensitivity of the measure.

$$c_{curv_j} = \sum_{l=0}^{l_{max}} \frac{1}{(\vec{n}_{j,l} \cdot \vec{n}_{j,(l+1) \bmod (l_{max}+1)}) + \beta} \quad (5.2)$$

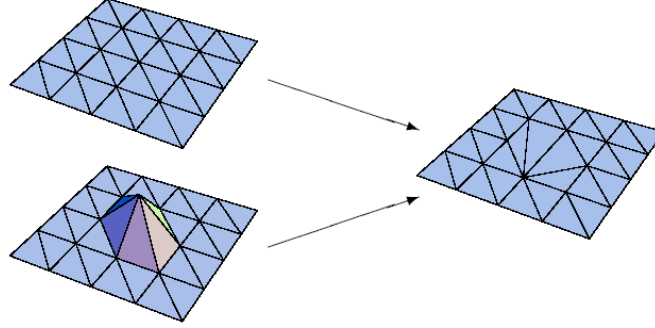


Figure 5.9: Local Curvature: Different three-dimensional surface patches can result in identical plain simplifications, despite having different surface areas (Quicken et al., 2000).

Combination of the two Measures

Combining the measure for local distortion and curvature results in the cost function (5.3) proposed by Quicken et al. (2000).

$$c_{j,k} = c_{round_{j,k}} + \alpha \cdot c_{curv_j} \quad (5.3)$$

When starting with well conditioned meshes, i.e. meshes consisting mainly of ‘round’ triangles, this measure achieves good results, as was shown by Kobbelt et al. (1998) and Quicken et al. (2000). In the present case, the triangles of the reconstructed surface are distorted as can be seen in Figure 5.10(b), which shows a surface patch of an input data set (Figure 5.10(a)).

When applying the described simplification operations to the given data unwanted face formations emerged, as is visualised in Figure 5.11 (right, top). After several reduction steps using Equation (5.3) as fairness criterion a fan-fold structure, embracing the whole surface, occurred. Figure 5.11 presents a typical situation, in which this phenomenon arose. The fairness criterion is computed for each half-edge, starting from the central vertex marked with red. As the measure for local curvature is calculated before the application of the topological operator, this part of the predicate is the same for all outgoing half-edges. Thus, the fact influencing the predicate is the condition of the triangles emerging after the half-edge-collapse. ‘Best results’, in terms of the fairness criterion, can be achieved when merging an edge connecting two different layers, as it results in two triangles, the ones connecting the two layers, which are much ‘rounder’ than the original ones in the mesh.

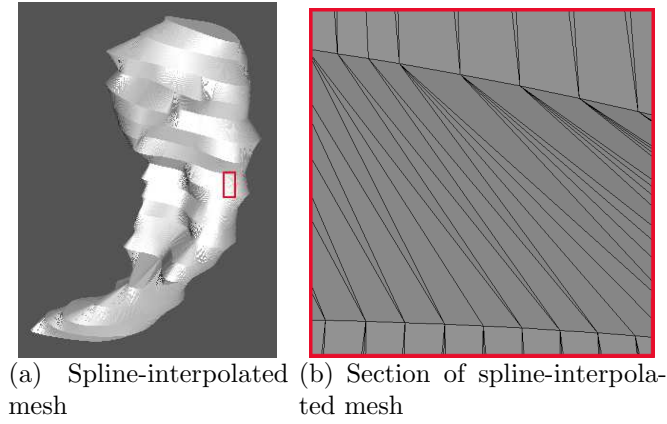


Figure 5.10: An example of a mesh of a hippocampus showing the poorly conditioned triangles

Consequently, not the desired case as shown on the bottom of Figure 5.11 is chosen, but the one resulting in a fanfold structure. In order to prohibit these collapses the measure defined in Equation (5.3) had to be modified.

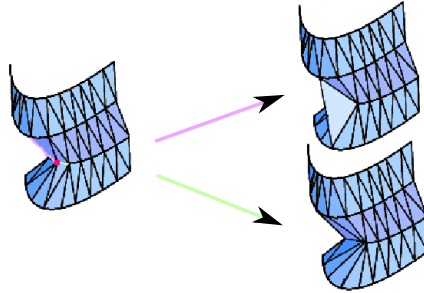


Figure 5.11: Surface simplification of the given data: (top) If edges are merged that link two slices, ‘round’ triangles emerge, and a fanfold structure is created. (bottom) The fairness criterion of the desired case is worse, as the condition of the triangles is worse.

Modifications

The following changes were tested:

- Local Curvature 1: Evaluation of the local curvature *after* the application of the topological operator.

- **Local Curvature 2:** Evaluation of the difference between the local curvatures before and after the application of the topological operator.
- **Additional Measure:** The weighted length of the edge to be merged is added to the cost function.

Local Curvature 1: Applying the topological operator before evaluating the cost-function for local curvature achieved the desired effect, but at the same time eliminated the positive effects of the previous measure, i.e. preservation of characteristic shape formations and increase of accuracy of the goal areas in the coarsest optimisation steps.

Local Curvature 2: As intended this function resulted in half-edge collapses that preserved the given surface curvature in the affected patch. Similar to the previously described curvature measures this one caused the fairness criteria to concentrate on certain regions. Thus, the mesh was not equally simplified, but consisted in some regions of very large and in others of very small triangles.

Additional Measure: Extending Equation (5.3) to

$$c_{j,k} = c_{round_{j,k}} + \alpha \cdot c_{curv_j} + \gamma \cdot |e_{j,k}^{\vec{}}|, \quad (5.4)$$

where $|e_{j,k}^{\vec{}}|$ denotes the length of the half-edge to be merged, gives the best results. The additional term ensures that short edges are preferred, resulting in equally sized triangles in the beginning of the reduction process. At the end the curvature measure became the guiding aspect of the process, which was also desirable.

5.2.4 Computation of a Mesh-Hierarchy

If half-edge collapse is used to simplify the surface, an appropriate data structure, to store the hippocampus surfaces, is needed. Methods commonly used in the collapse are deleting the faces bordering the edge and updating the faces which shared the vertices at end points of the edge. These operations illustrate, that easy access to adjacent elements is a basic data structure requirement. Moreover, the data structure has to be flexible, providing functionalities for reorganisation of the mesh.

A suitable library is provided by the Harvard Graphics Archive *Mesh Library*, which can be found at <http://people.deas.harvard.edu/~xgu/mesh>. The *Mesh Library* implements the half-edge structure and provides all required surface operations.

The half-edge structure owes its name to the way edges are stored, i.e. each edge consists of two half-edges of opposite direction. All vertices, half-edges and faces are stored in doubly-linked lists. Each vertex, half-edge, edge and face has access to all other elements adjacent to it. Thus, all neighbourhood information can be accessed within constant time. For a more detailed description of the structure see http://www.flipcode.com/articles/article_halfedge.shtml.

An advantage of the *Mesh Library* is that it already implements most functions required by the simplification process. The reduction is enclosed by an outer loop, counting the number of eliminated edges. In each iteration, the half-edge with the lowest costs is chosen for reduction. The half-edge collapse method is already implemented in the library. It eliminates the current half-edge and retriangulates the affected patch. After the collapse, the cost function has to be evaluated for all half-edges adjacent to the affected patch, using equation (5.4). Edges in the affected patch are blocked, i.e. cannot be collapsed, to ensure a valid insertion in the optimisation process. If the ratio ‘number of blocked to total number of half-edges’ exceeds a certain limit, e.g. 50 %, all edges are unblocked and a new level is started. Thus, when reinserting the collapsed edges, all half-edges of a single level can be inserted at a time.

An example of the simplification process is given in Figure 6.4, exhibiting the desired features creation of round triangles and preservation of shape.

5.3 Spherical Parametrisation

5.3.1 Overview

Spherical harmonics can be used to approximate functions of the sphere. The hippocampus data consists of surface meshes with vertices in Cartesian coordinates. In order to approximate these meshes, each vertex is to be assigned a parameter vector (θ_i, ϕ_i) . For surfaces of spherical topology, the unit sphere with polar coordinates is the natural parameter space. A homogeneous distribution of the parameter space is essential for the computation

of the coefficients of the spharms. This parametrisation is a key problem, affecting the results of the algorithm tremendously.

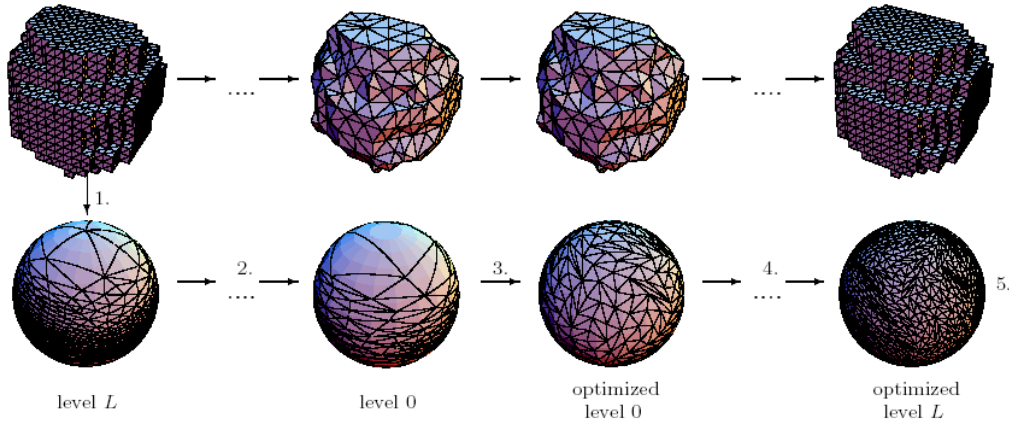


Figure 5.12: Hierarchical optimisation proposed by Quicken et al. (2000)

A good parametrisation is achieved in three steps, as is illustrated in Figure 5.12:

1. A bijective mapping of the surface to the unit sphere is computed, guaranteeing that every point on the surface maps to exactly one point on the sphere, and vice versa (Figure 5.12 - Step 1.). This procedure is described in Section 5.3.2.
2. As this initial parametrisation does not succeed in a homogenous distribution, the values have to be optimised in a second step (Figure 5.12 - Steps 2.–4.). The optimisation is explained in Section 5.4, while Section 5.2 describes the mesh hierarchy required to make the optimisation more robust.
3. In order to receive comparable descriptions the spharm representations need to be independent from (a) scaling, (b) rotation and (c) translation. These requirements are guaranteed by using (a) a mapping to the unit sphere, (b) reorientation of the first order ellipsoid and (c) by neglecting the coefficients of order 0. A description can be found in Section 5.5.4.

5.3.2 Initial Parametrisation

The goal of the parametrisation is to find a bijective mapping of a surface from cartesian space to the unit sphere parameterised by spherical coordinates. If the surfaces to be analysed were convex, the parametrisation could be simply obtained by projection, as shown in 2D in Figure 5.13(a). Unfortunately, many structures in the brain are not convex, causing ambiguities in the projection. Figure 5.13(b) demonstrates a case in which two points of the surface would be assigned the same spherical coordinates. Brechbühler et al. (1995) proposed an algorithm for the mapping that is analogue to heat conduction, compare Figure 5.13(c). Assuming that two points on the surface are heated and cooled respectively, all other positions on the surface take a temperature resembling their distance from these poles. Together with the topology of the mesh, this gives a bijective mapping (compare Figure 5.14 and 5.15). Mathematically the values of the parameter vectors are obtained by solving two Laplace's equations with Dirichlet conditions, one for latitude θ and the other one for longitude ϕ . In the discrete case, these equations correspond to a system of linear equations $Ax = b$. The initialisation of the two matrices (A_θ, A_ϕ) and the vectors (b_θ, b_ϕ) are explained in the following paragraphs.

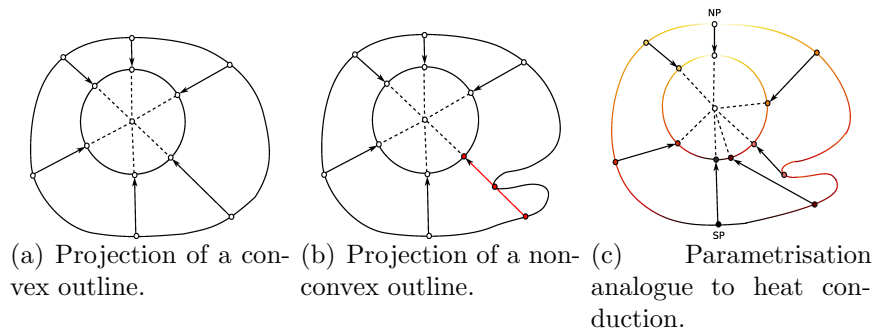


Figure 5.13: Different projection techniques.

Set up of the Matrices

Before setting up the matrices, two vertices have to be chosen as coolest and hottest point in the mesh. In order to ease the optimisation two points furthest apart are chosen. Any other selection would have the same result, but would need more iteration steps in the optimisation. These vertices are assigned to the poles of the unit sphere.

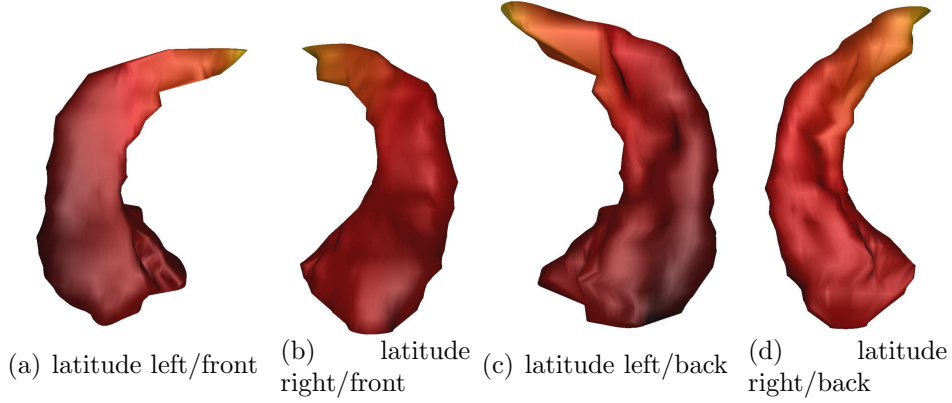


Figure 5.14: Latitudinal parametrisation of the right and left hippocampus of the same subject, front- and backside.

Latitude θ : Subsequently a Laplace equation with Dirichlet conditions is solved for θ :

$$\Delta\theta = 0 \quad \text{except at the poles, where} \quad (5.5)$$

$$\theta_{north} = 0 \quad (5.6)$$

$$\theta_{south} = \pi \quad (5.7)$$

In the discrete case the Laplace operator is approximated using second order finite differences, i.e. each latitude shall be equal to the mean of the latitudes of its n neighbours Nb_i :

$$\theta_i = \frac{1}{n} \sum_{j \in Nb_i} \theta_j \quad (5.8)$$

This equation can be rewritten as

$$n\theta_i - \sum_{j \in Nb_i^*} \theta_j = \pi \quad \text{if southpole} \in Nb_i \quad (5.9)$$

$$n\theta_i - \sum_{j \in Nb_i} \theta_j = 0 \quad \text{otherwise} \quad (5.10)$$

where Nb_i^* denotes Nb_i without the southpole. Let n_V be the number of vertices in the mesh. Evaluating Equation (5.8) for all n_V vertices except for the poles results in $n_V - 2$ linear equations. The pseudocode for the initialisation is given in Algorithm 1 in Appendix D.

A sample distribution of a pair of hippocampi is given in Figure 5.14.

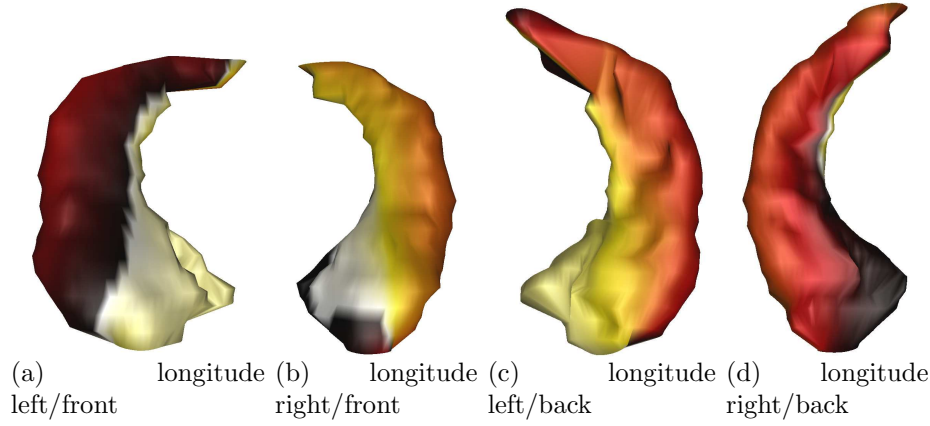


Figure 5.15: Longitudinal parametrisation of the right and left hippocampus of the same subject, front- and backside

Longitude ϕ : Unlike latitude, longitude is a cyclic parameter, which has a discontinuous line connecting the two poles. Whenever crossing the line, longitude has to be incremented or decremented by 2π , depending on the direction of the crossing. At the poles ϕ has no sensible values. Taking this into account, a date line is imposed and all values crossing it from west to east are decremented, values propagated to the west are incremented. For further computations, the poles and its linking edges are removed from the net, resulting in a mesh topologically equivalent to a cylinder. Due to the cyclic boundary condition, the system is singular. To overcome this, $2\phi_1 = 0$ is arbitrarily set and added to the first equation and the Laplace equation

$$\Delta\phi = 0 \quad (5.11)$$

is solved. Minor modifications have to be applied to the matrix and vector for latitude (compare Algorithm 2).

The longitudinal parametrisation of a pair of hippocampi is visualised in 5.15.

Solving Laplace Equations

As stated in the last two paragraphs the systems of linear equations consist of $n_V - 2$ equations. Thus, a square matrix of dimension $n_V - 2$ has to be handled. As the number of non-zero matrix entries in each row equals the number of neighbours of the according vertex, only a few values need to be

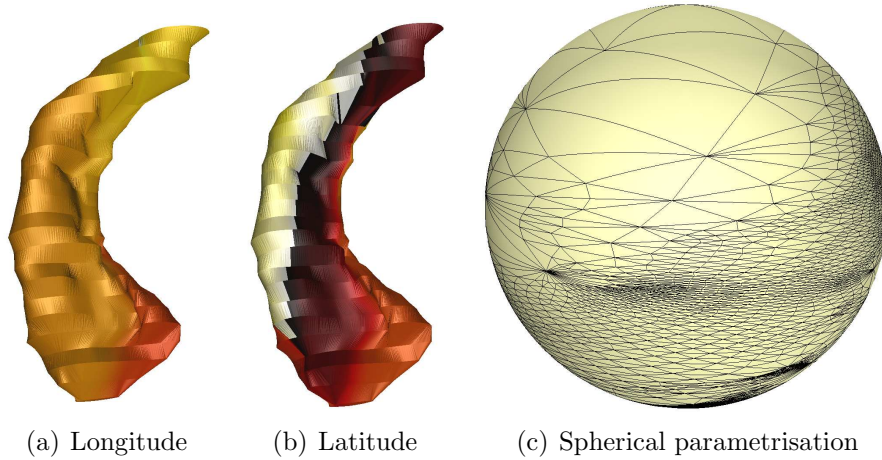


Figure 5.16: Initial parametrisation of a right hippocampus: (a) Longitude, (b) Latitude and (c) the parametrisation on the unit sphere.

stored. This fact is crucial for further computations, as it provides the opportunity to get results at reasonable costs (time and memory), using software packages implementing solvers for systems of sparse linear equations. Don-garra (2004) and Eijkhout (1997) compared several freely available packages.

Tested packages had to provide the following functionalities:

- Compatibility to C/C++ Programs
- Support for sparse matrices
- Sufficient matrix operations
- Iterative solvers with included preconditioners

Two packages, meeting these requirements, were chosen for testing: *Laspack* (implemented in C) and *GMM++* (implemented in C++). Both packages gave equivalent results. *Laspack* was chosen, as it is easy to use, well documented and written in C as the whole program implemented in the context of this thesis.

5.4 Spherical Optimisation

5.4.1 Overview

The parametrisation computed so far is valid, but unfortunately results in major distortions of the spherical mesh (Figure 5.16). Thus, the initial

parametrisation has to be corrected. This step is necessary to obtain a homogeneous distribution of the parameterspace over the surface, which is important to represent the shape properly, as was shown by Brechbühler et al. (1995). Two criteria are used to measure distortion: Firstly, every object region must map to a region of proportional size, and secondly, the inner angles must be proportional as well. In an optimisation process, these two features have to be controlled.

Brechbühler et al. (1995) used constrained optimisation considering area preservation and minimal distortion to optimise meshes consisting of quadrilaterals. Thus, each quadrilateral on the surface is assigned a spherical quadrilateral of proportional area on the unit sphere. As optimisation is accomplished for all vertices at a time, the dimensionality of the problem increases linearly with the number of vertices. For objects consisting of several thousand vertices the optimisation becomes unstable. Moreover, it is restricted to quadrilateral meshes, as this method, applied to triangular meshes, leaves only two degrees of freedom for optimisation, which was shown by Quicken et al. (2000). Furthermore, quadrilaterals meshes are supported to a lesser extent by mesh-software compared to triangular meshes, which would make the manipulation of the meshes far more challenging.

To overcome these limitations, two approaches following different concepts were implemented. The first one was introduced by Quicken et al. (2000) using the concept of hierarchical optimisation, which optimises the parametrisation iteratively at increasing levels of detail in three steps:

- Simplification of the triangular mesh (Section 5.2).
- Iterative optimisation on each level of detail.
- Final optimisation on the finest level.

The second method is based on simulated annealing, which locates a good approximation to the global minimum in a large search space, utilising the probabilistic approach of this technique.

The following sections are structured as follows. Section 5.4.2 gives an introduction to the basic concepts of optimisation. Three different optimisation algorithms used in the following are explained in Section 5.4.3. As the optimisation processes make extensive use of spherical trigonometry, the formulas used in the following are explained in Section 5.4.4. Finally, Section 5.4.5 and 5.4.6 explain the two methods used for optimisation, i.e. constrained optimisation and simulated annealing respectively.

5.4.2 Optimisation

Often results are influenced by many different, partly controversial parameters, e.g. the quality of a medical operation depends on the experience of the surgeon and his team, the equipment of the hospital, the time the patient can be narcotised, the state of the patient, and many more factors. Naturally, every patient wants the best treatment available, whereas it is impossible that the best surgeon conducts all operations. Thus, a compromise has to be found, guaranteeing sufficient treatment and keeping down costs at the same time, while meeting certain prerequisites.

The aim of the optimisation process is to minimise a cost function. The function depends on variables, which can be constrained to certain intervals by equalities or inequalities. A visualisation of an optimisation problem in one and two variables is given in Figure 5.17. Mathematically, this problem can be stated as follows. Find x to

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && g_i(x) \geq 0 \quad i = 1, \dots, m \\ & && h_j(x) = 0 \quad j = 1, \dots, p \end{aligned} \tag{5.12}$$

where f , g_i , and h_j are general functions of the parameter $x \in \mathbb{R}^n$ (Papadimitriou and Steiglitz, 1982).

In the case of the operation f can be chosen as a combination of expected quality of the operation and its costs, while the constraints limit duration of the operation and the number of assistants and demand a certain method to be used.

Classification

Depending on the type of the functions f , g_i , and h_j , the optimisation problems are classified. For each type special optimisation techniques are available, making use of the individual structure of the problem. Major subfields used in the following are: Linear, quadratic, and non-linear programming.

The problem is considered to be *linear*, if f , g_i , and h_j are linear. The

previous operation example can be formulated as:

$$\begin{aligned} f(\text{expQuality}, \text{costs}, \text{nSurgeons}) &= 2 \cdot \text{costs} - \text{expQuality} \\ g_1(\text{expQuality}, \text{costs}, \text{nSurgeons}) &: \text{expQuality} \geq 4 \\ g_2(\text{expQuality}, \text{costs}, \text{nSurgeons}) &: \text{costs} \leq 62 \\ g_3(\text{expQuality}, \text{costs}, \text{nSurgeons}) &: \text{nSurgeons} = 1 \end{aligned}$$

where expQuality is the expected quality of the operation, costs are the costs caused by operation and aftercare, and nSurgeons is the number of surgeons involved in the operation. In this case f is a linear combination of the costs and the expected quality of the operation, in which costs are twice as important as the expected quality. The quality is subtracted as it shall be maximised, which is the same as minimising the negative. g_1 demands a minimum quality and g_2 limits the costs, while g_3 ensures, that only a single surgeon takes part in the operation.

If a quality level qL , giving good recovery rates and keeping down costs at the same time, was known, deviation from this value should be controlled as given in 5.13

$$f(\text{expQuality}, \text{costs}, \text{nSurgeons}) = 2 \cdot \text{costs} + (qL - \text{expQuality})^2 \quad (5.13)$$

As f includes a quadratic term, the program is referred to as *quadratic programming*.

The general case in which both, objective function and constraints can contain nonlinear parts, e.g. products of variables or trigonometric functions, is referred to as *nonlinear programming*.

An introduction to combinatorial optimisation can be found in Papadimitriou and Steiglitz (1982) and to nonlinear optimisation in Spellucci (1993).

Local and Global Optima

The aim of optimisation is to find the minimum of the function. Finding a global minimum can be extremely difficult, whereas it is in general comparatively easy to determine a solution of $f(x)$ that is optimal in its neighbourhood.

Definition 5.1. An *instance of an optimization problem* is a pair (F, c) , where F is any set, the domain of feasible points; c is the cost function, a mapping

$$c : F \longrightarrow \mathbb{R}^1.$$

The problem is to find an $f \in F$ for which

$$c(f) \leq c(y) \text{ for all } y \in F$$

Such a point f is called a **globally optimal** solution to the given instance, or, when no confusion can arise, simply an **optimal** solution. (Papadimitriou and Steiglitz, 1982)

Definition 5.2. Given an optimization problem with instances (F, c) , a **neighbourhood** is a mapping

$$N : F \longrightarrow 2^F$$

(Papadimitriou and Steiglitz, 1982)

Definition 5.3. Given an instance (F, c) of an optimisation problem and a neighbourhood N , a feasible solution $f \in F$ is called **locally optimal with respect to N** (or simply **locally optimal** whenever N is understood by context) if

$$c(f) \leq c(g) \text{ for all } g \in N(f)$$

(Papadimitriou and Steiglitz, 1982)

Figure 5.17 shows an example of a one-dimensional Euclidean optimisation problem. If the neighbourhood is suitably small, the points A , B , and C are local minima, while only B is also the global minimum.

Techniques

Depending on the cost function and the constraints, different techniques can be applied. A further aspect influencing the choice of a method is the type of the minimum searched for. Some methods are only capable of finding the closest (local) minimum, whereas others detect the global minimum. If a good start position close to the global minimum is known, or if only one minimum exists, the first group of methods is preferred, as computational costs are lower in general.

Many optimisation techniques need twice-differentiable functions, as in this case the local minima can be found, where the gradient of the cost function is zero (stationary points). In order to classify the type of the minimum, the Hessian (second derivative matrix) is computed. If the matrix is positive definite, the point is a local minimum, if it is negative definite, a local maximum, and if it is indefinite it is some kind of saddle point. Common iterative methods used to find the stationary points, are *Newton's method*,

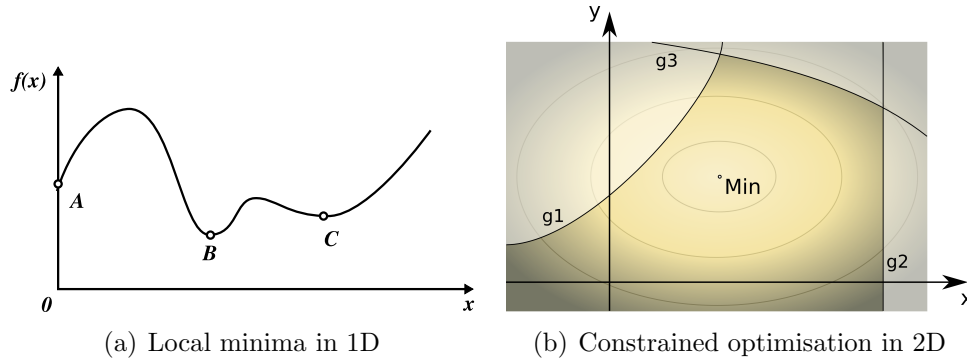


Figure 5.17: Different Euclidean optimisation problems: (a) A one-dimensional optimisation problem with several minima. (b) A two-dimensional optimisation problem with a single minimum and three constraints g_i . The function values are colour-coded. Beige stands for low values, green for high values. Transparent regions indicate regions excluded by the constraints.

line search and *conjugate gradient*. All three methods can be applied to nonlinear optimisation problems.

As these techniques do not take constraints into account, a method is needed to convert a constrained optimisation problem into an unconstrained one. Lagrange-multipliers are such a method, transforming an optimisation problem in n variables into an unconstrained problem in $n + 1$ variables.

Definition 5.4. The function $L : \mathbf{R}^n \times \mathbf{R}^m \times \mathbf{R}^p \rightarrow \mathbf{R}$

$$L(x, \lambda, \mu) := f(x) - \sum_{i=1}^m \lambda_i g_i(x) - \sum_{j=1}^p \mu_j h_j(x)$$

is called the **Lagrange function** assigned to the nonlinear program. (Spellucci, 1993)

Spellucci (1993) proves: If $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a saddle point of the Lagrange function, i.e. $L(\bar{x}, \lambda, \mu) \leq L(\bar{x}, \bar{\lambda}, \bar{\mu}) \leq L(x, \bar{\lambda}, \bar{\mu})$ for all $(x, \lambda, \mu) \in \mathbf{R}^n \times \mathbf{R}_+^m \times \mathbf{R}^p$, then \bar{x} is a solution to the nonlinear program. Thus, instead of optimising the constrained program, the solutions to the Lagrange function can be used to find local minima.

If the derivatives of the cost function are not available different methods, like *hill climbing*, *simulated annealing* or *genetic algorithms*, have to be used. Simulated annealing will be explained in the next section.

5.4.3 Applied Optimisation Methods

In the following the three optimisation methods used in the algorithm are explained.

Non-Linear Least-Squares Fitting - The Levenberg-Marquardt Algorithm

The description is extracted from the GSL-Dokumentation (GNU Scientific Library, Galassi et al., 2005).

The problem of multidimensional nonlinear least-squares fitting requires the minimization of the squared residuals of n functions, f_i , in p parameters, x_i ,

$$\Phi(x) = \frac{1}{2} \|F(x)\|^2 = \frac{1}{2} \sum_{i=1}^n f_i(x_1, \dots, x_p)^2$$

All algorithms proceed from an initial guess using the linearization,

$$\psi(p) = \|F(x + p)\| \approx \|F(x) + Jp\| \quad (5.14)$$

where x is the initial point, p is the proposed step and J is the Jacobian matrix $J_{ij} = df_i/dx_j$. Additional strategies are used to enlarge the region of convergence. These include requiring a decrease in the norm $\|F\|$ on each step or using a trust region to avoid steps which fall outside the linear regime.

To perform a weighted least-squares fit of a nonlinear model $Y(x, t)$ to data (t_i, y_i) with independent gaussian errors σ_i , use function components of the following form,

$$f_i = \frac{Y(x, t_i) - y_i}{\sigma_i} \quad (5.15)$$

Note that the model parameters are denoted by x in this chapter since the non-linear least-squares algorithms are described geometrically (i.e. finding the minimum of a surface). The independent variable of any data to be fitted is denoted by t .

With the definition above the Jacobian is

$$J_{ij} = \frac{1}{\sigma_i} \cdot \frac{dY_i}{dx_j}, \quad (5.16)$$

where $Y_i = Y(x, t_i)$.

Interior Point Method

The following explanation of primal-dual nonlinear interior methods is extracted from Nocedal et al. (2005).

The problem under consideration will be written as

$$\min_x f(x) \quad (5.17)$$

$$s.t. \quad c(x) = 0 \quad (5.18)$$

$$x \geq 0, \quad (5.19)$$

where $f : \mathbf{R}^n \rightarrow \mathbf{R}$ and $c : \mathbf{R}^n \rightarrow \mathbf{R}^m$ are twice continuously differentiable functions. For conciseness we will refer to interior-point methods for nonlinear programming as “nonlinear interior methods”. [...]

We associate with the nonlinear program (5.17) the barrier problem

$$\min_x \varphi_\mu(x) \equiv f(x) - \mu \sum_{i=1}^n \ln x_i \quad (5.20)$$

$$s.t. \quad c(x) = 0, \quad (5.21)$$

where $\mu > 0$ is the barrier parameter. As is well known, the KKT conditions of the barrier problem (5.20) can be written as

$$\nabla f(x) - A(x)^T y - z = 0 \quad (5.22)$$

$$Xz - \mu e = 0 \quad (5.23)$$

$$c(x) = 0 \quad (5.24)$$

together with

$$x \geq 0, \quad z \geq 0. \quad (5.25)$$

Here $A(x)$ denotes the Jacobian matrix of the constraint functions $c(x)$.

Applying Newton's method to (5.22), in the variables (x, y, z) , gives the *primal-dual* system

$$\begin{bmatrix} \nabla_{xx}^2 \mathcal{L} & -A(x)^T & -I \\ Z & 0 & X \\ A(x) & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} \nabla f(x) - A(x)^T y - z \\ Xz - \mu e \\ c(x) \end{bmatrix}, \quad (5.26)$$

where \mathcal{L} denotes the Lagrangian of the nonlinear program, that is,

$$\mathcal{L}(x, y, z) = f(x) - y^T c(x) - z^T x. \quad (5.27)$$

After the step $\Delta = (\Delta x, \Delta y, \Delta z)$ has been determined, we compute primal and dual steplengths, α_p and α_d , and define the new iterate (x^+, y^+, z^+) as

$$x^+ = x + \alpha_p \Delta x, \quad y^+ = y + \alpha_p \Delta y, \quad z^+ = z + \alpha_p \Delta z. \quad (5.28)$$

The steplength are computed in two stages. First we compute

$$\alpha_x^{max} = \max \alpha \in (0, 1] : x + \alpha \Delta x \geq (1 - \tau)x \quad (5.29)$$

$$\alpha_z^{max} = \max \alpha \in (0, 1] : z + \alpha \Delta z \geq (1 - \tau)z \quad (5.30)$$

with $\tau \in (0, 1)$ (e.g. $\tau = 0.995$). Next, we perform a backtracing line search that computes the final steplengths

$$\alpha_p \in (0, \alpha - x^{max}], \quad \alpha_d \in (0, \alpha - z^{max}], \quad (5.31)$$

providing sufficient decrease of a merit function or ensuring acceptability by a filter.

Simulated Annealing

From metallurgy it is well known, that annealing a substance, i.e. slowly cooling a material, can relieve stresses and support the formation of a perfect crystal lattice. The initial high temperature allows the atoms to wander randomly through states of higher kinetic energy. As temperature cools down, the configuration gets stiffer, allowing mainly changes towards states of less energy. In this process the atoms organise themselves in an optimal way, i.e. in a configuration with least kinetic energy. This is exactly, what is to be achieved in optimisation.

In the early 1980's Kirkpatrick et al. (1983) and Cerny (1985) invented the Simulated Annealing (SA) algorithm indepently. It is a meta-heuristic algorithm, simulating thermodynamic systems, which can be used to solve combinatorial and continuous variable problems.

By analogy to the physical process, SA starts at a certain initial state and a high temperature. The goal is to minimise the energy function, measuring deviation from the desired state. Therefore, a random walk through the configuration space is taken. In the beginning, transitions to a neighbouring parameter set occur almost randomly, independent of the differences in the energy level. The lower the temperature gets, the less likely are changes towards configurations with a higher kinetic energy, enforcing a “downhill” strategy. The transition probability from state i with energy E_i to state $i+1$ with energy E_{i+1} can be given by a Boltzman distribution:

$$p = \begin{cases} e^{-\frac{E_{i+1}-E_i}{k \cdot T}} & \text{if } E_{i+1} > E_i \\ 1 & \text{if } E_{i+1} \leq E_i \end{cases} \quad (5.32)$$

where T is the current temperature, and k is the Boltzmann constant. Other distributions are possible. Thus, the probability of a transition to a state of higher energy is proportional to the temperature and inversely proportional to the energy difference. This slight probability of taking a step towards a configuration of higher energy is what allows simulated annealing to frequently get out of local minima and, hence, has a greater chance of finding the global minimum.

5.4.4 Spherical Trigonometry

A short introduction to spherical trigonometry shall be given in the following, as the optimisation methods explained will make extensive use of some of the given interrelations.

So far, each vertex is given by a pair of spherical coordinates. Basic quantities needed are the length of edges, inner angles and the area of the triangles on the unit sphere. In order to compute inner angles the lengths of the edges of the spherical triangles are needed, which can be computed using equation 5.33. Let $p_0 = (\theta_0, \phi_0)$ and $p_1 = (\theta_1, \phi_1)$ be the spherical coordinates of the vertices of the edge, then the distance of the points is given by

$$|p_0 - p_1| = \arccos(\cos \theta_0 \cdot \cos \theta_1 + \sin \theta_0 \cdot \sin \theta_1 \cdot \sin(\phi_0 - \phi_1)) \quad (5.33)$$

Using the law of cosine, the inner angles can be computed from the three edges a , b , and c .

$$\cos a = \cos b \cdot \cos c + \sin b \cdot \sin c \cdot \cos \alpha \quad (5.34)$$

The analogue equation for edges is called cosine rule for sides:

$$\cos \alpha = -\cos \beta \cdot \cos \gamma + \sin \beta \cdot \sin \gamma \cdot \cos a \quad (5.35)$$

The area of the triangle A_T can be computed using the spherical excess ε .

$$A_T = \varepsilon \cdot R^2 \quad (5.36)$$

This can either be done according to equation 5.37 with inner angles

$$\varepsilon = \alpha + \beta + \gamma - \pi \quad (5.37)$$

or by application of l'Huiliers equation (5.38), which utilises the edges

$$\tan \frac{\varepsilon}{4} = \sqrt{\tan \frac{s}{2} \cdot \tan \frac{s-a}{2} \cdot \tan \frac{s-b}{2} \cdot \tan \frac{s-c}{2}} \quad (5.38)$$

$$s = \frac{1}{2}(a + b + c) \quad (5.39)$$

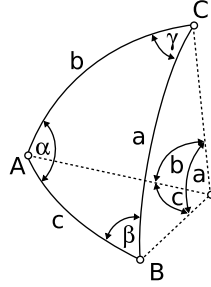


Figure 5.18: A spherical triangle.

5.4.5 Global Approach - Constrained Optimisation

The aim of the spherical optimisation is to minimise distortions of areas of triangles and their inner angles. The algorithm proposed by Quicken et al. (2000) tries to minimise all distortions at a time on different levels of detail, using constrained optimisation. The hierarchical structure is applied to make the method more stable. Hence, a mesh hierarchy has to be established first, which was explained in Section 5.2.

After computing the hierarchy, each of these levels has to be optimised successively, starting with the coarsest. The optimisation criteria, cost function and constraints, proposed by Brechbühler et al. (1995) are not applicable to triangular meshes, as they leave too few degrees of freedom. Quicken et al. (2000) changed the constraints to overcome this problem:

1. The constraints for area preservation are dropped. Inequalities are only established to forbid flipped faces.
2. The new objective function aims for area preservation and distortion minimisation.

Thus, the cost function (5.40) consists of two terms. The first one measures area preservation by summing the differences of relative areas of the spherical triangle and the triangle in object space. The relative area of a polygon is its area divided by the total area of the surface. The second term quantifies distortion of triangles. Quicken et al. (2000) exploit the fact that their meshes were formed from volumes, consisting of of equally shaped cubes. Hence, all triangles are similar, i.e. are orthogonal isosceles triangles. These properties are utilised to measure distortion. $\underline{a}_i \cdot \underline{b}_i$ is zero, if the edges are orthogonal and the term after β is minimal, if both edges have same length.

$$\min_f \sum_{i \in \Delta} \left[\left(\frac{A_i}{4\pi} - \frac{A_i^*}{\sum_{i \in \Delta} A_i^*} \right)^2 + \alpha \left(\left(\frac{\underline{a}_i \cdot \underline{b}_i}{\|\underline{a}_i\| \cdot \|\underline{b}_i\|} \right)^2 + \beta \left| \frac{\|\underline{a}_i\| \cdot \|\underline{a}_i\| - \|\underline{b}_i\| \cdot \|\underline{b}_i\|}{\|\underline{a}_i\| \cdot \|\underline{a}_i\| + \|\underline{b}_i\| \cdot \|\underline{b}_i\|} \right| \right) \right], \quad (5.40)$$

where \underline{a}_i and \underline{b}_i are the edges of same length in the triangle, A_i is the area of the spherical triangle and A_i^* is the area of the triangle in object space.

As the data to be analysed in this thesis consists of arbitrary triangles this measure had to be modified. The basic idea of the modified approach was to use spherical angles for optimisation instead of cartesian coordinates, as both properties, area preservation and distortion of angles can be measured easily. Moreover, derivatives of the cost function can be computed easily.

Objective Function

An objective function, which considers only inner angles, is used to optimise general triangles:

$$\min_f \sum_{i \in \Delta} \left(\frac{A_i}{4\pi} - \frac{A_i^*}{\sum_{i \in \Delta} A_i^*} \right)^2 + \quad (5.41)$$

$$\lambda \left(\left(\frac{\alpha_i}{\sigma_i} - \frac{\alpha_i^*}{\pi} \right)^2 + \left(\frac{\beta_i}{\sigma_i} - \frac{\beta_i^*}{\pi} \right)^2 + \left(\frac{\gamma_i}{\sigma_i} - \frac{\gamma_i^*}{\pi} \right)^2 \right)$$

$$\sigma_i = \alpha_i + \beta_i + \gamma_i \quad (5.42)$$

$$A_i = \alpha_i + \beta_i + \gamma_i - \pi \quad (5.43)$$

where Δ denotes the triangles of the mesh, A_i and A_i^* are the areas of a spherical and its corresponding triangle in object space, α_i , α_i^* , β_i , β_i^* , γ_i and γ_i^* denote the inner angles of the current triangles on the sphere and in object space. Like the cost function by Quicken et al. (2000), equation (5.41) penalises differences in relative face areas. The second term, which prevents distorted triangles, was modified. Now, each relative inner angle on the sphere has to have the same size as its counterpart in the surface mesh. The linear combination is controlled by the parameter λ .

Constraints

Constraints are only established to ensure a valid triangulation, i.e. angles are within the range $(0, \pi)$, and edges must have the same length in both adjacent triangles. Using the cosine rule for sides, this gives

$$\frac{\cos \alpha_{i,0} + \cos \beta_{i,0} \cdot \cos \gamma_{i,0}}{\sin \beta_{i,0} \cdot \sin \gamma_{i,0}} - \frac{\cos \alpha_{i,1} + \cos \beta_{i,1} \cdot \cos \gamma_{i,1}}{\sin \beta_{i,1} \cdot \sin \gamma_{i,1}} = 0 \quad (5.44)$$

where 0 and 1 indicate, whether the variable belongs to the first or second triangle. This requirement is established for all edges, resulting in n_{edges} constraints.

In a first implementation, further constraints were formulated to ensure that all angles surrounding a vertex sum to 2π , i.e. the mesh covers the whole surface of the unit sphere. This left too few degrees of freedom for the optimiser to find a solution. Thus, the constraints for surface covering were dropped, as they are implicitly included in the cost function. If a minimum is found, the relative areas of the spherical triangles sum to 1, i.e. the whole surface is covered, as do the planar ones, meaning, that the inner angles form a valid triangulation on the unit sphere.

After the optimisation the new parametrisation has to be computed. Therefore, the one vertex is assigned to the northpole. Starting from this vertex all other positions are computed iteratively. Therefore, all vertices having neighbours that are not inserted yet, are stored in a list. In each iteration the first vertex of the list is rotated to the northpole and the parametrisation for its neighbours is computed. Afterwards, the new vertices are inserted into the list, and the old vertex is eliminated from the list. When the list is empty, the parameters have been computed for all vertices.

Implementation

The interior-point method explained in section 5.4.3 is used to optimise the inner angles, as it is capable of handling large-scale nonlinear problems. The basic idea of this approach is to achieve optimisation by going through the middle of the solid defined by the problem, rather than around its surface.

The IPOPT-library, developed by Wächter and Biegler (2006), implements the interior-point method and provides all functionalities to conduct the optimisation. The library and its documentation can be found on <https://projects.coin-or.org/Ipopt>.

The following functions have to be specified for the optimiser:

- Objective function, $f(x_k)$
- Gradient of the objective function, $\nabla f(x_k)$
- Constraint residuals, $g(x_k)$
- Jacobian of the constraints, $\nabla g(x_k)$
- Hessian of the Lagrangian, $\sigma_f \nabla^2 f(x_k) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(x_k)$

The Lagrangian is given by $f(x) + g(x)^T \lambda$ (see Lagrange function in section 5.4.2) and the Hessian of the Lagrangian by $\nabla^2 f(x_k) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(x_k)$. The library needs to ask for the Hessian of the objective function and the constraints separately. Thus, a factor σ_f was introduced, which can be set to zero to get the Hessian of the constraints.

The implementation of the objective function and the constraints are a straightforward realisation of equation (5.41) and (5.44) respectively. The first one returns the error summed for all triangles, and the second one is a vector of size n_{edges} , giving the difference of edge lengths for each edge.

One reason to choose inner angles for optimisation, was the fact that first and second derivatives reduce nicely. The cost function sums over all triangles, considering only the inner angles of a single triangle in each summand. When computing partial derivatives, all other variables are considered to be constant. Thus, only the summand containing the angle has to be differentiated, as all others drop out. The following examples illustrate this property.

Let the angles α_i, β_i , and γ_i belong to the i 'th triangle, $relArea_i^*$ be the relative area of the i 'th triangle in the mesh, and $rel\alpha_i^*, rel\beta_i^*$, and $rel\gamma_i^*$ be the relative angles in the surface mesh. Then, the first and second derivative of f with respect to α_i are given by

$$\begin{aligned} \frac{\partial f}{\partial \alpha_i} = & \frac{1}{2\pi} \left(\frac{\alpha_i + \beta_i + \gamma_i - \pi}{4\pi} - relArea_i^* \right) - \\ & 2\lambda \left[\frac{(\beta_i + \gamma_i)((\alpha_i + \beta_i + \gamma_i)rel\alpha_i^* - \alpha_i)}{1} \right] + \\ & 2\lambda \left[\frac{\beta_i((\alpha_i + \beta_i + \gamma_i)rel\beta_i^* - \beta_i)}{1} \right] + \\ & 2\lambda \left[\frac{\gamma_i((\alpha_i + \beta_i + \gamma_i)rel\gamma_i^* - \gamma_i)}{1} \right] \end{aligned} \quad (5.45)$$

and

$$\begin{aligned} \frac{\partial^2 f}{\partial \alpha_i^2} = & 6\lambda \frac{\alpha_i^2 + \beta_i^2 + \gamma_i^2}{(\alpha_i + \beta_i + \gamma_i)^4} + 2\lambda \frac{1 + 2rel\alpha_i^*}{(\alpha_i + \beta_i + \gamma_i)^2} + \frac{1}{8\pi} - \\ & 4\lambda \frac{\alpha_i(2 + rel\alpha_i^*) + \beta_i rel\beta_i^* + \gamma_i rel\gamma_i^*}{(\alpha_i + \beta_i + \gamma_i)^3} \end{aligned} \quad (5.46)$$

The derivatives of the constraints, which include the inner angles of two triangles, e.g. for the edge adjacent to face i and j , are

$$\begin{aligned} \frac{\partial g}{\partial \alpha_i} &= -\frac{\sin \alpha_i}{\sin \beta_i \cdot \sin \gamma_i} \\ \frac{\partial g}{\partial \beta_i} &= -\frac{1}{\tan \gamma_i} - \frac{\cos \alpha_i + \cos \beta_i \cdot \cos \gamma_i}{\tan \beta_i \cdot \sin \beta_i \cdot \sin \gamma_i} \\ \frac{\partial g}{\partial \gamma_i} &= -\frac{1}{\tan \beta_i} - \frac{\cos \alpha_i + \cos \beta_i \cdot \cos \gamma_i}{\tan \gamma_i \cdot \sin \beta_i \cdot \sin \gamma_i} \end{aligned} \quad (5.47)$$

These partial derivatives for m constraints in n angles are organised in the Jacobian as follows

$$\nabla g(x) = \begin{pmatrix} \frac{\partial g_0}{\partial \alpha_0} & \frac{\partial g_0}{\partial \beta_0} & \frac{\partial g_0}{\partial \gamma_0} & \cdots & \frac{\partial g_0}{\partial \alpha_n} & \frac{\partial g_0}{\partial \beta_n} & \frac{\partial g_0}{\partial \gamma_n} \\ \vdots & \ddots & & & & & \vdots \\ \frac{\partial g_m}{\partial \alpha_0} & \frac{\partial g_m}{\partial \beta_0} & \frac{\partial g_m}{\partial \gamma_0} & \cdots & \frac{\partial g_m}{\partial \alpha_n} & \frac{\partial g_m}{\partial \beta_n} & \frac{\partial g_m}{\partial \gamma_n} \end{pmatrix} \quad (5.48)$$

Likewise, the Hessian of the Lagrangian is given by:

$$\sigma_f \nabla^2 f(x) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(x) = \quad (5.49)$$

$$\sigma_f \begin{pmatrix} \frac{\partial^2 f}{\partial \alpha_0^2} & \cdots & \frac{\partial^2 f}{\partial \alpha_0 \gamma_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial \alpha_0 \gamma_n} & \cdots & \frac{\partial^2 f}{\partial \gamma_n^2} \end{pmatrix} + \sum_{i=1}^m \lambda_i \begin{pmatrix} \frac{\partial^2 g_i}{\partial \alpha_0^2} & \cdots & \frac{\partial^2 g_i}{\partial \alpha_0 \gamma_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 g_i}{\partial \alpha_0 \gamma_n} & \cdots & \frac{\partial^2 g_i}{\partial \gamma_n^2} \end{pmatrix}$$

As each constraint includes only six angles, each row in the Jacobian has only six non-zero entries, which can be determined according to equation (5.47). IPOPT stores these matrices in a compressed mode, using only non-zero entries. Thus, the matrix can be reduced from a $m \times n$ to a $m \times 6$ system, which is quite important, as several thousand faces, i.e. three times as many angles, are to be handled. In the case of the Hessian, a further compression can be achieved, since the matrices are symmetric. Thus, only the upper or lower triangular matrix has to be stored.

The initialisation of the optimiser follows the instructions given in the tutorial at <http://www.coin-or.org/Ipopt/documentation/>. Two parameters were set:

- AddIpoptNumOption(nlp, "tol", 1e-9); and
- AddIpoptStrOption(nlp, "mu_strategy", "adaptive");

The first option specifies the convergence tolerance of the algorithm, and the second one tells the optimiser to use an adaptive update strategy for the barrier parameters.

The optimisation is invoked by a function call, providing the optimiser with the required functions and options, and the initial point, i.e. the inner angles from the initial parametrisation, as parameters.

Results of the optimisation process are given in Section 6.

5.4.6 Local Approach - Simulated Annealing

Simulated annealing (SA) is a heuristic method to solve complex optimisation problems in many variables. The constrained optimisation approach explained in the last section uses the gradient of the cost function and the

constraints, to conduct the walk through the configuration space. If such information cannot be accessed, as the structure of the space is not well understood or not smooth, stochastic search techniques like SA have to be used. The basic idea of SA is to take a random walk through the space of valid configurations and choose the one with the lowest energy. Steps to a point of higher energy can only be taken with a certain probability (Equation 5.32), which depends on the energy differences and the state of the optimisation, i.e. the temperature. See Section 5.4.3 for more details on SA.

The walk in the configuration space taken by SA can be influenced in two ways: Firstly, an algorithm is needed, returning the next configuration, and secondly, an energy function has to be specified, telling whether this new parameter set is closer to the optimum than the last one.

Step Function

The step function modifies the current configuration using a random step in parameter space. Although an arbitrary number of parameters could be modified, only a single vertex is chosen for optimisation, to ease controll. This vertex is chosen randomly from all vertices, having both, adjacent faces that are too small, and those, that are too large. This restriction was made to accelerate convergence.

After choosing a vertex v , its surrounding patch, i.e. all triangles adjacent to v , are projected onto the plane tangent to the unit sphere in v . The projection is done to ease optimisation of the patch. Lambert azimuthal equal-area projection was chosen as projection method, since it preserves areas and directions from the central point.

The forward transformation is given by

$$\begin{aligned} x &= k' \cos \phi \sin (\lambda - \lambda_0) \\ y &= k' [\cos \phi_1 \sin \phi - \sin \phi_1 \cos \phi \cos (\lambda - \lambda_0)] \\ k' &= \sqrt{\frac{1}{2} [1 - \sin \phi_1 \sin \phi + \cos \phi_1 \cos \phi \cos (\lambda - \lambda_0)]}, \end{aligned} \tag{5.50}$$

where ϕ_1 is the standard parallel and λ_0 is the central longitude. The inverse

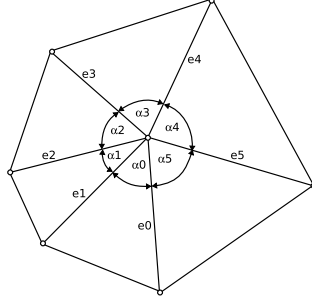


Figure 5.19: Projected triangle patch.

formulas are

$$\begin{aligned}
 \phi &= \arcsin \left(\cos c \sin \phi_1 + \frac{y \sin c \cos \phi_1}{\rho} \right) \\
 \lambda &= \lambda_0 + \arctan \left(\frac{x \sin c}{\rho \cos \phi_1 \cos c - y \sin \phi_1 \sin c} \right) \\
 \rho &= \sqrt{x^2 + y^2} \\
 c &= 2 \arcsin \left(\frac{1}{2} \rho \right).
 \end{aligned} \tag{5.51}$$

(Weisstein, 2004)

In 2D, a position of the central vertex has to be found that minimises distortions in relative areas and angles around the vertex, which is an independent optimisation problem inside SA. Again, optimisation requires a cost function, given by

$$f(\vec{x}, \vec{y}) = \sum_{i \in \Delta} \left[\left(\frac{\arccos \left(\frac{e_{i,0} \cdot e_{i,1}}{|e_{i,0}| \cdot |e_{i,1}|} \right)}{\sigma} - \alpha_i^* \right)^2 + \beta \left(\frac{\frac{1}{2} |e_{i,0} \times e_{i,1}|}{s} - A_i^* \right)^2 \right]$$

$$\sigma = \sum_{i \in \Delta} \arccos \left(\frac{e_{i,0} \cdot e_{i,1}}{|e_{i,0}| \cdot |e_{i,1}|} \right); \quad s = \sum_{i \in \Delta} \frac{1}{2} |e_{i,0} \times e_{i,1}|$$

where Δ are the faces adjacent to the central vertex in clockwise order and $e_{i,0}$, and $e_{i,1}$ are the two edges of each face adjacent to the vertex (compare Figure 5.19). α_i^* and A_i^* denote relative angles and areas in the patch in the mesh in object space (not on the sphere). The cost function sums the distortion of each triangle, using squared differences in inner angles and

relative areas. The linear combination is controlled by β , defining, which aspect, optimal angles or areas, is more important.

The Levenberg-Marquardt algorithm, a method to minimise a sum of squares of several, generally nonlinear functions, is used for optimisation. The algorithm, also known as nonlinear least squares fit, requires a cost function and its derivative to be specified by the user.

The optimised vertex is projected back onto the surface of the unit sphere. This new configuration, with the parameters of one vertex modified, serves as candidate for the next step.

Energy Function

Before taking the step, the energy function E_i has to be evaluated. E_i is given by

$$E_i(\vec{\theta}, \vec{\phi}) = \sum_{j=1}^n \left(\frac{A_j}{4\pi} - \frac{A_j^*}{\sum_{i \in \Delta} A_j^*} \right)^2, \quad (5.52)$$

where A_j and A_j^* denote the area of the spherical triangle and its corresponding mesh triangle. Spherical areas are computed using l'Huiliers equation (5.38). The energy function aims only for area preservation and neglects distortion of angles, as this was already taken into account in the step function.

Implementation

The GNU Scientific Library (GSL) was used for optimisation. The GSL is a numerical library for C and C++ programmers. It is free software under the GNU General Public License and can be found at <http://www.gnu.org/software/gsl/>. The library provides a wide range of mathematical routines amongst others optimisation routines as Simulated Annealing and Nonlinear Least-Squares Fitting. The functions are well documented and examples ease the application.

As explained earlier, SA needs a step and a cost function. The methods are declared in the GSL header and have to be specified by the user. Both algorithms are a straightforward implementation of equation (5.52) for the energy function and the procedure explained in the paragraph Step Function. The same is true for the optimisation of the patch, using least-squares fitting in 2D.

5.5 Approximation Using Spherical Harmonics

5.5.1 Overview

Now, that the surfaces can be represented on the sphere, spharms can be used to approximate this function. Thus, a linear combination of spherical harmonics basis functions is used to describe a complex surface. The coefficients of the series expansion can be used for a statistical analysis of the surface properties.

The theoretical background of the shape description using spharms is explained in Section 5.5.2. Section 5.5.3 provides information on the implementation, and Section 5.5.4 describes methods to resolve rotation independent descriptors.

5.5.2 Theoretical Background

All the previous work has been done to allow for surface descriptions using spherical harmonics. Spherical harmonics (spharms) are the angular portion of the solution to Laplace's equation in spherical coordinates where azimuthal symmetry is not present and form an orthonormal set of basis functions on the sphere. Thus, they can be used to approximate functions on the unit sphere. For more information on the definition of spharms, see Appendix B. Using a linear combination of these basis functions, any function on the sphere can be approximated up to a predefined accuracy. The more basis functions or harmonics are included, the more details can be depicted.

As spharms operate on the sphere, the mesh had to be projected onto the unit sphere first (Section 5.3). Optimisation of the triangles was performed, to ensure comparability and allow a uniform representation of the surface, as shown in Section 5.4. So far a bijective mapping between the mesh on the unit sphere and the one in Cartesian space is given. In order to apply spharms, function values on the sphere are required, characterising the object. Two approaches are possible: The first one uses the scalar distances of the vertices to an object related center, whereas the second one defines vector values on the sphere, giving the Cartesian coordinates of the vertices. The advantage of the second approximation is that the series expansion can be made independent from the position of the object in Cartesian space (Section 5.5.4). Thus, the second method was implemented. Thus, the function on

the sphere is given by

$$f(\theta, \phi) = \begin{pmatrix} x(\theta, \phi) \\ y(\theta, \phi) \\ z(\theta, \phi) \end{pmatrix}. \quad (5.53)$$

A further advantage of this function is the straightforward reconstruction of the approximated surface. As the free parameters θ and ϕ run over whole the sphere, e.g. $\theta = 0 \dots \pi, \phi = 0 \dots 2\pi$, the function runs over the whole surface in object space.

f is approximated by spharms using

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l c_l^m Y_l^m(\theta, \phi), \quad (5.54)$$

where Y_l^m denotes the spharm with parameters l, m and c_l^m its associated coefficient. Thus, a set of coefficients c_l^m has to be found, fulfilling equation (5.54). Formally, this is achieved by forming the inner product

$$c_l^m = \langle f(\theta, \phi), Y_l^m \rangle = \int_0^\pi \int_0^{2\pi} f(\theta, \phi) Y_l^m d\phi \sin \theta d\theta. \quad (5.55)$$

Let n_{verts} denote the number of vertices, \vec{v}_i the Cartesian coordinates of the i 'th vertex, and i be the index with $0 \leq i < n_{verts}$. As the bijection of the surface points to parameter space is a discrete function, the function $f(\theta, \phi)$ is also discrete, defined only for parameters associated with a vertex, where it takes the value $f(\theta_i, \phi_i) = \vec{v}_i$. In order to evaluate the integral (5.55), f needs to be defined on the intervals $\theta \in [0, \pi]$ and $\phi \in [0, 2\pi]$, requiring an interpolation function inside the triangles. Several approaches were compared by Praun and Hoppe (2003), stating that each function bears disadvantages, and thus introduces further computational errors. Moreover, it provides an artificial resolution, which has no basis on the given data. The opposite extreme, the straightforward discretisation of the integral,

$$c_l^m \approx \frac{4\pi}{n_{verts}} \sum_{i=0}^{n_{verts}-1} v_i Y_l^m(\theta_i, \phi_i), \quad (5.56)$$

does not give the required result, as the values of Y_l^m evaluated at some arbitrary sets of parameters (θ_i, ϕ_i) , will in general not form an orthonormal basis.

Striking a balance between these two extremes is finding a set of coefficients, that form an orthonormal basis and minimise the approximation error at the

given data points. Thus, coefficients c_l^m have to be found that minimise the error-term e in

$$f(\theta_i, \phi_i) = \sum_{l=0}^{l_{max}} \sum_{m=-l}^l c_l^m Y_l^m(\theta_i, \phi_i) + e_i. \quad (5.57)$$

Truncating the sum by an upper bound l_{max} for l , as depicted in Equation (5.57), allows computer based evaluation. The greater l_{max} , the more details can be captured by the series expansion. In matrix notation, the following system of linear equations is given

$$f = Yc + e \quad (5.58)$$

with

$$Y \in \mathbb{C}^{n_{verts} \times n_{harms}}, \quad c \in \mathbb{C}^{n_{harms}}, \quad f \in \mathbb{R}^{n_{verts}} \quad \text{and} \quad e \in (\mathbb{R}^+)^{n_{verts}}$$

where $n_{harms} = (l_{max} + 1)^2$,

$$Y = \begin{pmatrix} Y_0^0(\theta_0, \phi_0) & Y_1^{-1}(\theta_0, \phi_0) & \cdots & Y_{l_{max}}^{l_{max}}(\theta_0, \phi_0) \\ Y_0^0(\theta_1, \phi_1) & Y_1^{-1}(\theta_1, \phi_1) & \cdots & Y_{l_{max}}^{l_{max}}(\theta_1, \phi_1) \\ \vdots & \vdots & \ddots & \vdots \\ Y_0^0(\theta_{n_V-1}, \phi_{n_V-1}) & Y_1^{-1}(\theta_{n_V-1}, \phi_{n_V-1}) & \cdots & Y_{l_{max}}^{l_{max}}(\theta_{n_V-1}, \phi_{n_V-1}) \end{pmatrix},$$

$c = (c_0^0, c_1^{-1}, c_1^0, \dots, c_{l_{max}}^{l_{max}})^T$, $f = (f(\theta_0, \phi_0), f(\theta_1, \phi_1), \dots, f(\theta_{n_V}, \phi_{n_V}))^T$ and $e = (e_0, e_1, \dots, e_{n_V})^T$, $n_V = n_{verts}$.

As the system has far more equations (≈ 20.000) than unknown coefficients (= number of harmonies, equivalent to the desired resolution), it is usually overdetermined and cannot be solved, so that e is the zero vector. Thus, a solution has to be found that minimises the Fröbenius-norm of the error term $e = Yc - f$, which means finding a \hat{c} with

$$\|Y\hat{c} - f\|_F = \min_c \|Yc - f\|_F = \min_c \|e\|_F = \min_c \sum_{i=0}^{n_V} e_i^2(c). \quad (5.59)$$

As indicated by the last term, this is a problem that can be solved using least squares algorithms. The Fröbenius-norm is the squared norm of a vector v , which is given by $\|v\|^2 = v^T v$. Thus, equation (5.59) can be rewritten as

$$\begin{aligned} \|Y\hat{c} - f\|_F &= \|Y\hat{c} - f\|^2 \\ &= (Y\hat{c} - f)^T (Y\hat{c} - f) \\ &= (Y\hat{c})^T (Y\hat{c}) - f^T Y\hat{c} - (Y\hat{c})^T f + f^T f. \end{aligned} \quad (5.60)$$

The two middle terms are equal, and the minimum is found at the zero of the derivative with respect to \hat{c} ,

$$2Y^TY\hat{c} - 2Y^Tf = 0. \quad (5.61)$$

Therefor the minimising vector \hat{c} is a solution of the normal equation

$$Y^TY\hat{c} = Y^Tf \quad (5.62)$$

If Y has full rank, Y^TY is invertible and the unique solution to the minimisation problem is given by

$$\hat{c} = (Y^TY)^{-1}Y^Tf. \quad (5.63)$$

The matrix $Y^+ = (Y^TY)^{-1}Y^T$ is called the pseudo inverse of Y . Several methods to determine the pseudoinverse have been proposed:

- Straightforward computation of the pseudoinverse, which is often the fastest but also the least stable method.
- QR decomposition is a good compromise, but is more difficult to implement than the other two.
- Singular value decomposition (SVD) is the slowest, but also the most robust method.

Due to its robustness and simplicity of implementation, SVD was chosen to compute the pseudo inverse. For a complex matrix A , the singular value decomposition is a decomposition into the form

$$A = U \cdot \Sigma \cdot V^H \quad (5.64)$$

where U and V are unitary matrices, V^H is the conjugate transpose of V , and Σ is a diagonal matrix whose elements are the singular values of the original matrix. If A is a complex matrix, then there always exists such a decomposition with positive singular values (Golub and Van Loan 1996, pp. 70 and 73).

After the pseudo inverse has been computed, a simple matrix vector multiplication is needed to determine the coefficients \hat{c} . All surface information of the hippocampus is now embedded in this small set of numbers, that can be analysed statistically.

5.5.3 Computation of the Coefficients

In order to compute the pseudo inverse Y^+ , Y has to be specified first. Y is a $n_{verts} \times n_{harms}$ complex matrix, which is important to note, as the computation of the pseudo inverse is the most challenging part concerning memory. Each row of Y stores the spharms evaluated at the parameters of the vertex, associated to that row, i.e. $y_i = (Y_0^0(\theta_i, \phi_i), Y_1^{-1}(\theta_i, \phi_i), Y_1^0(\theta_i, \phi_i), Y_1^1(\theta_i, \phi_i), \dots)$, where (θ_i, ϕ_i) are the parameters of v_i . The set up of the required matrix Y and vector f is given in Algorithm 3 in the appendix.

For matrix manipulation, the *clapack*- and *cblas*-libraries are used. Thus, matrices have to be stored in an array of complex numbers in row- or column-major order. Column-major order was chosen. Spharms are computed using the GSL (Gnu Scientific Library, Galassi et al., 2005), and are afterwards stored in the matrix.

The pseudo inverse is computed in three steps using matrix operations of the *clapack*- and *cblas*-library. First the SVD of $Y = U \cdot \Sigma \cdot V^H$ is computed using the *cgesvd*-routine of the *cblas*-library, which returns the three matrices U , Σ and V^H . Afterwards, Σ , which is a diagonal matrix, is inverted by computing the reciprocals of the diagonal entries. In a last step, the three matrices U , Σ^{-1} and V are multiplied using *cblas_cgemm* (*cblas*), giving $Y^+ = U \cdot \Sigma^{-1} \cdot V$. The conjugate transposed of V^H is computed internally, using the *CblasConjTrans* option.

In the final step, Y^+ and f are multiplied, giving the coefficients of the series expansion.

5.5.4 Rotation Independent Descriptors

So far, the surface descriptions still depend on translation, rotation and scaling of the original object. Brechbühler et al. (1995) proposed three methods to overcome these dependencies.

Translation invariance can be achieved by ignoring c_0^0 , which adds a constant term to each coordinate x, y and z independently, if

$$f(\theta, \phi) = (x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))^T$$

is chosen as surface function.

Any three real valued linear combinations of the first order harmonics Y_1^{-1} , Y_1^0 , and Y_1^1 , interpreted as Cartesian coordinates in the object space, will always describe an ellipsoid. Rotation invariance is achieved by rotating this ellipsoid to a standard position and applying this rotation to the coefficients.

Scaling invariance can be achieved by dividing all coefficients by the length of the longest main axis of the ellipsoid.

These transformations have not been implemented yet.

Chapter 6

Realisation

The goal of this work was to implement a process chain allowing for shape analysis of human hippocampi under Alzheimer's Disease (AD) using spherical harmonics. The original intention was to use open-source software such as FreeSurfer (spherical parametrisation; <http://surfer.nmr.mgh.harvard.edu>) and the SpharmonicKit (spherical harmonic transforms; <http://www.cs.dartmouth.edu/~geelong/sphere>) to ease implementation. The version of FreeSurfer, that was available in spring 2005, was not capable of handling the given data, i.e. anatomical substructures of the brain. Thus, the spherical parametrisation had to be implemented from scratch, following the ideas of Brechbühler et al. (1995). The spherical harmonics transformation, implemented in the SpharmonicKit could not be applied, as it needs the function on the sphere to be sampled at equally distant positions. Besides, an interpolation function on the sphere is required, of which Praun and Hoppe (2003) had already shown to cause errors. Moreover, interpolation provides an artificial resolution having no basis on the given data.

Thus, the process chain illustrated in Figure 6.1 was implemented, which consists of five major subproblems: triangulation of coplanar outlines, simplification of triangular meshes, spherical parametrisation, spherical optimisation and approximation of the function on the unit sphere by spherical harmonics. Except for the optimisation all subproblems have been solved. Due to temporal restrictions of the thesis, the optimisation algorithm could not be finished. In order to analyse the coefficients of the spherical harmonics series expansion statistically, all subprograms have to work correctly. Hence, an evaluation of the present data cannot be accomplished.

In the following the results of the different subproblems are described.

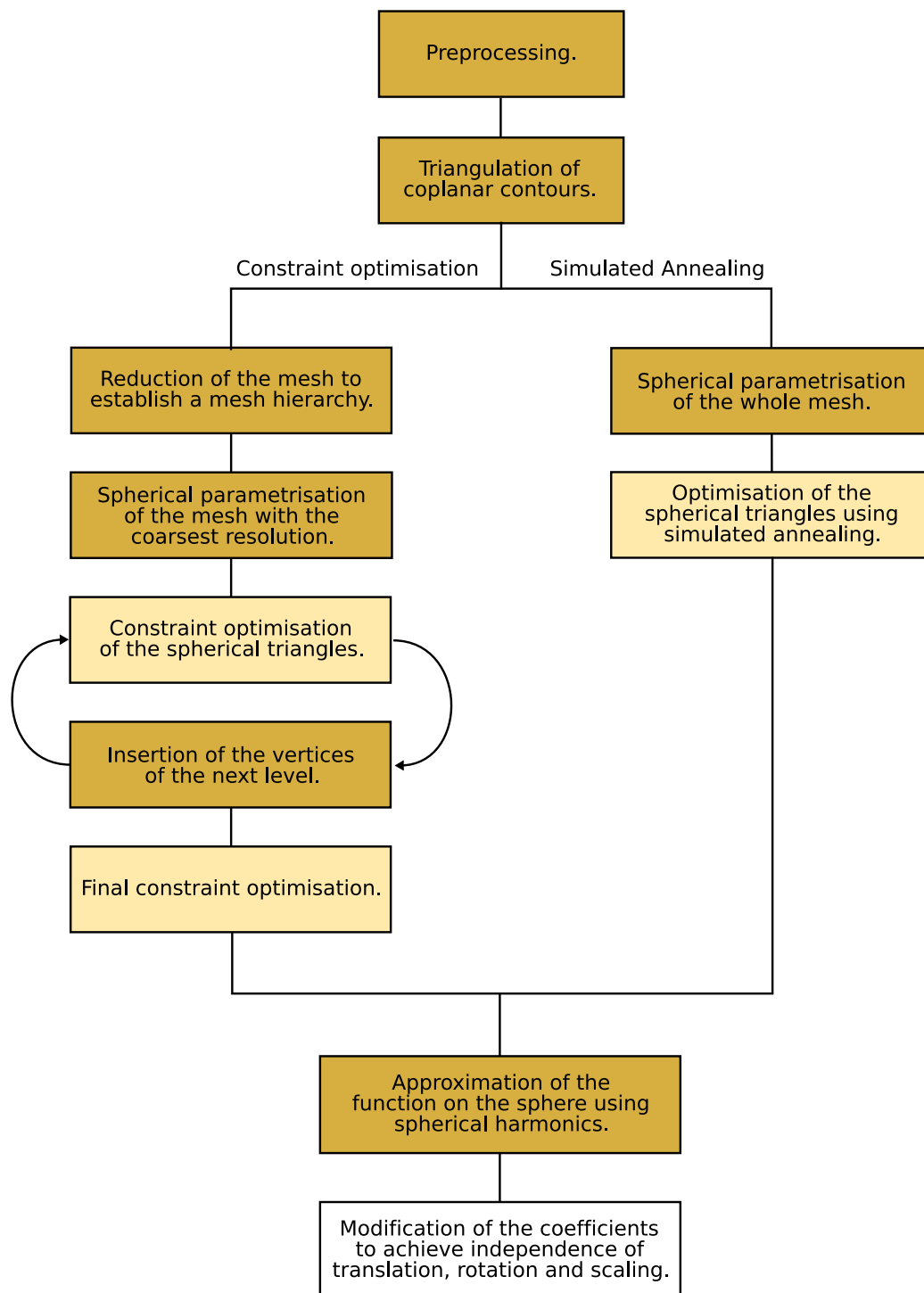


Figure 6.1: State of the implementation: Dark boxes indicate finished sub-problems, light boxes need further improvement and white boxes have not been implemented yet.

Triangulation of Coplanar Outlines

The present data was taken from 35 patients, resulting in the outlines of 47 left and 47 right hippocampi (including follow up examinations). All data sets were triangulated using the algorithm explained in section 5.1.3. In 46 cases the new method, using convex outlines and bisectors, was able to triangulate the surfaces correctly, as demonstrated by the example of six hippocampi in Figure 6.3. The illustration reveals the great differences in the shape of the hippocampi. Most problems were caused by the lower regions (foldings) and the outlines on top (great deviations of the centres of mass). Problems occurred just in one dataset, which exhibits major foldings in the hippocampus (Figure 6.2).

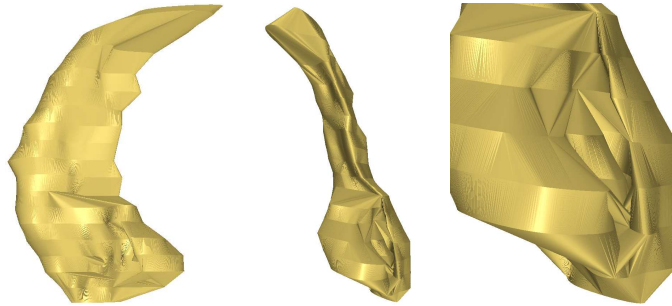


Figure 6.2: Different views of one hippocampus that could not be triangulated correctly. The lower levels exhibit self-intersections (left), which are caused by ambiguities in triangulation of the folded region (middle and right). In the fold the upper outline is closer to the opposite side, causing wrong links at the tips of the fold, where the angle criterion fails.

Simplification of Triangular Meshes

If constrained optimisation is used as optimisation method, a mesh hierarchy has to be computed first to ease convergence towards the global minimum. The triangulated surfaces consist in average of 10,000 vertices, 30,000 edges and 20,000 faces. The reduction process from 30,000 to 2,000 edges takes at most 90 seconds. Different stages of the reduction process of a mesh consisting of 3325 edges are depicted in Figure 6.4. The parameters for the fairness criterion were chosen as follows: $\alpha = 50$, $\beta = 1.01$, $ratio = 0.5$ and $\gamma = 100$.

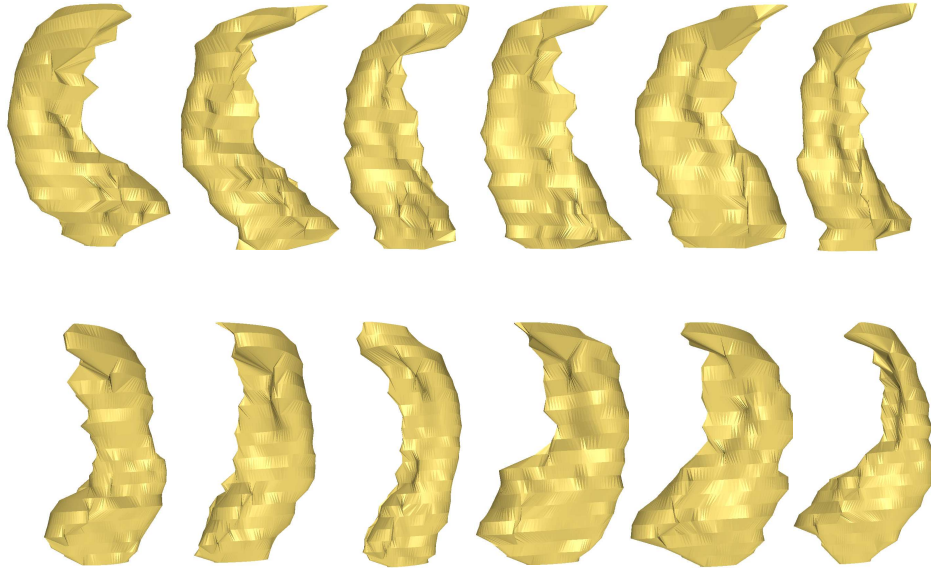


Figure 6.3: Triangular meshes of six hippocampus pairs. The top row shows the right hippocampus of six different patients. The bottom row shows the associated left hippocampi. As can be seen the hippocampi exhibit great differences in shape.

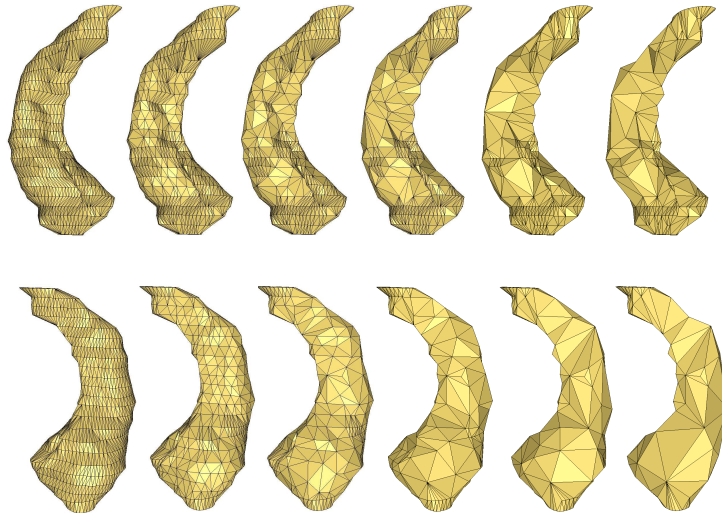


Figure 6.4: Mesh simplification of a hippocampus. The top and bottom row show the mesh from front and back at different levels of detail: (number of edges from left to right) 3325, 2232, 1551, 1098, 804 and 594.

Spherical Parametrisation

The spherical parametrisation was computed for all hippocampi. Figure C.1 gives an example of the different illustrations using one hippocampus pair. For each hippocampus three graphics are given: The two colormaps on the mesh in Cartesian space show the distribution of latitude and longitude. The third image shows the spherical mesh, where each vertex of the mesh in Cartesian space is positioned at its spherical coordinates given by the parametrisation. The parametrisations of six pairs of hippocampi (left and right) are shown in Figure C.2. Figure C.3 illustrates the corresponding spherical meshes of the twelve hippocampi.

The distribution of latitude and longitude is visualised in Figure C.2, which reveals the major differences in the parametrisation of different hippocampi. Latitude parameterises the hippocampus from top to bottom. The more homogeneous the color of the whole hippocampus appears, the closer the values are. Thus, many small triangles are concentrated on a small part of the sphere, whereas few large triangles cover the big remaining part of the sphere. The same is true for longitude, giving the parametrisation ‘around’ the sphere. The dateline is the line connecting the two poles, where the parametrisation returns from 2π to 0. In Figure C.2 (c) and (g), this line is indicated by the transition from black to white. As can be seen, the dateline has no fix position, causing differences with regard to a rotation around the y-axis (‘top to bottom’-axis).

Figure C.3 shows the spherical meshes of the hippocampi. All meshes are rotated to the same position, to allow for the comparison of the different parametrisations. The illustrations verify the impression received in the individual images for latitude and longitude. The parametrisations exhibit large differences in the size of the triangles and the distribution of the triangles. The clustering of small triangles occurs at different positions. Thus, the different parametrisations are not comparable.

Spherical Optimisation

To allow for smooth approximations and comparability, the parametrisations have to be corrected, i.e. distortions of spherical angles and areas of triangles have to be minimised. Two optimisation techniques were implemented and tested: constrained optimisation and simulated annealing.

An example of the constrained optimisation process is given in Figure C.4.

The example mesh consists of 1580 vertices. The optimisation process takes 57 iterations (< 1 min) and reduces the objective function from $4.6749943\text{e-}03$ to $3.5063960\text{e-}04$. The parametrisation was only optimised with respect to distortions of areas.

The triangles of the surface mesh have approximately same size. Thus, the relative area of each triangle ought to be 1.0 after optimisation. In the initial parametrisation (Figure C.4 (b) - blue curve) the relative areas range from 0.008 to 3.092, i.e. the smallest triangle takes only $1/125$ of the desired area and the largest triangle is three times as large as it ought to be. After the optimisation (Figure C.4 (b) - pink curve) the relative areas oscillate around 1, ranging from 0.8 to 1.27. In image Figure C.4(d) the areas are sorted according to relative areas. Before optimisation (blue) the values diverge greatly from 1. After optimisation (pink) most triangles have a relative size of 1, which was the aim of the optimisation process.

The optimisation process modifies the spherical angles of the mesh on the sphere, giving a set of optimal angles. After the optimisation the new spherical coordinates have to be computed from these angles. Already in this small example, the reconstruction of the spherical mesh causes problems, i.e. numerical errors become too large. This problem gets obvious when the position of a new vertex is computed from two different adjacent faces, and the parameters do not match. Thus, this approach cannot be used for optimisation, but has to be modified, which will be discussed in Section 7.

The second optimisation method uses simulated annealing. Two examples of the optimisation are given in Figure C.5 and C.7. The first mesh consists of 36 and second of 217 vertices. In the first case, the energy decreased from 2.31 to 0.12 (< 1 min), and in the second, from 0.21 to 0.19 (< 2 min). For each example, the surface mesh (a), the initial parametrisation (b), and the optimised parametrisation (c) are shown.

The parameters for the first optimisation were chosen as follows: $\beta = 100,000$, initial temperature 2000, damping factor 1.02, steps on each temperature level 15, minimal temperature 0.5. Thus, the triangles are optimised with regard to proportional areas, i.e. angles are neglected. The triangles of the resulting parametrisation have equal size, but exhibit still major distortions C.5(c). When considering preservation of angles and areas at the same time, different parameters have to be chosen for SA and the optimisation of the patch. Due to the large number of parameters and the resulting complexity of the problem, this could not be tested yet.

The same is true for the second optimisation of a simplified hippocampus mesh. Here the parameters are chosen as follows: $\beta = 100,000$, initial temperature 2000, damping factor 1.02, steps on each temperature level 100, minimal temperature 0.5. This parameter set is not optimal, as can be seen in Figure C.8, showing the progress of the energy function. The energy decreases within six iteration from 0.21 to 0.19, and remains after 27 iterations at approximately 0.185. This process confirms the visual impression of Figure C.7(c), showing mainly modifications at the poles, which appear during the first optimisation steps. From iteration 27 to 418, the parametrisation remains static. Thus, further testing of the parameters is needed to resolve a more uniform parametrisation.

Figure C.6 and C.8 display the decrease of the energy function. In the first case, the energy function reaches very quickly a low level. The areas of the spherical triangles are compared in Figure C.9(a). As the low energy indicates, the areas have approximately the same size after optimisation. Figure C.9(b) illustrates the differences in areas of the second mesh. The high energy and the large differences in triangular areas, indicate that the optimisation was not successful. An improvement of the simulated annealing optimisation can be achieved by modifying the parameters, which was not possible due to temporal restrictions.

Approximation Using Spherical Harmonics

An example of the reconstruction of a mesh from the spherical harmonics series expansion is given in Figure C.10 and C.11. Both images show the reconstruction process, where the coefficients of the series expansion are used up to different degrees. The first image shows the surfaces that are resolved, if no optimisation is used in the approximation process. In the second image, the parametrisation is optimised with respect to distortions of triangular areas. As can be seen, the optimisation affects the reconstruction tremendously. With the optimisation, the surfaces are smoother than without, which becomes obvious in the reconstructions of degree seven. The optimised parametrisation results in a perfect reconstruction of the mesh vertices, which can be seen in Figure C.11(f). The reconstruction can be further improved by taking distortion of angles into account in the optimisation process, as shown by Brechbühler et al. (1995).

Chapter 7

Conclusion

A process chain consisting of seven steps was explained and implemented to approximate surface meshes by spherical harmonics. Several modifications had to be made in order to cope with the given data, i.e. surfaces of human hippocampi.

A new triangulation algorithm was presented to triangulate nonconvex coplanar slices that exhibit major differences in the distance between single points. The algorithm was able to compute valid surfaces, that include all given vertices and have no holes. Currently, the algorithm is only based on a local decision criterion. Thus, the surface properties are solely controlled locally. Although, the algorithm gives good results, the surfaces might be further improved by taking more holistic information like the skeleton of the surface into account, to ensure that prominent features are directly connected, even though other vertices might be closer.

In the second step, a surface simplification algorithm was presented that was able to cope with the special structure of the data resolved from MR images, i.e. surfaces composed of poorly conditioned triangles. Whether the simplification is appropriate to improve the spherical optimisation could not be tested yet, as the process chain is not finished and the optimisation is mandatory.

The initial spherical parametrisation followed the ideas of Brechbühler et al. (1995) and was directly applicable to the present data.

The major challenge of the whole process was the spherical optimisation. Two different approaches were implemented and compared: constrained optimisation and simulated annealing. In the constrained optimisation, the

inner angles of the spherical triangles were chosen as variables, as they characterise both quantities that were to be controlled, i.e. distortion of the areas of the triangles and distortion of their inner angles. Though the optimisation gave good results, it could not be used, as the new spherical coordinates could not be computed from the angles due to numerical errors. Two different improvements are possible. Either a numerically more stable approach is chosen for the computation of the parameters, or the Cartesian coordinates of the vertices on the sphere are used for optimisation. In the second case, the improved parametrisation would be accessible directly after optimisation. The spherical parameters should not be used for optimisation, as the parametrisation causes problems at the poles where discontinuities appear. Thus, partial derivatives cannot be computed.

The simulated annealing gave good results for the small mesh, where the influence of the parameters could be tested easily. The larger the meshes got, the harder the estimation of the influence of a single parameter got. Extensive testing is necessary to determine an optimal parameter set. Moreover, simulated annealing takes longer, as it makes no use of the information given about the parameter space. Thus, it takes SA longer to find an optimal configuration.

The implementation of the optimisation process became such complicated, that it could not be realised within the temporal restrictions. A solution to this problem remains part of future work.

In the last step, the mesh on the sphere was approximated by spherical harmonics. The implementation was a straightforward realisation of the explanation given by Brechbühler et al. (1995). As the optimisation has not been finished yet, a statistical analysis of the data was not possible.

On the whole, the approximation of surfaces using spherical harmonics is a very powerful technique, that poses many challenges in the implementation. If these difficulties can be solved a mathematical surface description for arbitrary objects is available, that might help gain further insight into the deformation of objects. This remains subject of journal publications.

Bibliography

- N. Amenta, S. Choi, and R. Kolluri. The power crust. In *6th ACM Symposium on Solid Modeling and Applications*, pages 249–260, 2001.
- American Health Assistance Foundation, 2006. URL <http://www.ahaf.org/alzdis/about/AnatomyBrain.htm>.
- S. E. Arnold, B. T. Hyman, J. Flory, A. R. Damasio, and G. W. Van Hoesen. The topographical and neuroanatomical distribution of neurofibrillary tangles and neuritic plaques in the cerebral cortex of patients with Alzheimer’s disease. *Cereb. Cortex*, 1(1):103–116, 1991.
- J. Barnes, R. I. Scatell, R. G. Boyes, C. Frost, E. B. Lewis, C. L. Rossor, M. N. Rossor, and N. C. Fox. Differentiating AD from aging using semi-automated measurement of hippocampal atrophy rates. *NeuroImage*, 23: 574–581, 2004.
- H. Braak and E. Braak. Neuropathological staging of Alzheimer-related changes. *Acta. Neuropathol.*, 82:239–259, 1991.
- H. Braak and E. Braak. Evolution of alzheimer’s disease related intraneuronal changes, 2006. URL <http://www.alzforum.org/res/for/vir/osaka/braak/seminar/7-8.asp>.
- C. G. Brayne, C. Huppert, F. A. Huppert, C. Barkley, E. Gehlhaar, D. M. Girling, D. W. O’Connor, and E. S. Paykel. Incidence of clinically diagnosed subtypes of dementia in an elderly population. cambridge project for later life. *The British Journal of Psychiatry*, 167:255–262, 1995.
- C. Brechbühler, G. Gerig, and O. Kübler. Parametrization of closed surfaces for 3-D shape description. *Comput. Vis. Image Underst.*, 61(2):154–170, 1995. ISSN 1077-3142.
- A. Bruck, T. Kurki, and V. Kaasinen T. Vahlberg J. O. Rinne. Hippocampal and prefrontal atrophy in patients with early non-demented Parkinson’s

- disease is related to cognitive impairment. *J. Neurol. Neurosurg. Psychiatry*, 75(10):1467–9, 2004.
- W. E. Byerly. *An elementary treatise on Fourier's series and spherical, cylindrical and ellipsoidal harmonics*. Ginn & Company, 1893.
- V. Cerny. A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications*, pages 41–51, 1985.
- D. Chan, N. C. Fox, R. I. Scahill, W. R. Crum, J. L. Whitwell, G. Leschziner, A. M. Rossor, J. M. Stevens, L. Cipolotti, and M. N. Rossor. Patterns of temporal lobe atrophy in semantic dementia and Alzheimer's disease. *Ann. Neurol.*, 49(4):433–442, 2001.
- K. Chen, E. M. Reiman, G. E. Alexander, D. Bandy, R. Renaut, W. R. Crum, N. C. Fox, and M. N. Rossor. An automated algorithm for the computation of brain volume change from sequential MRIs using an iterative principal component analysis and its evaluation for the assessment of whole-brain atrophy rates in patients with probable Alzheimer's disease. *Neuroimage*, 22(1):134–143, 2004.
- G. E. Christensen, R. D. Rabbitt, and M. I. Miller. Deformable templates using large deformation kinematics. *Image Processing, IEEE Transactions on*, 5(10):1435–1447, 1996.
- H. N. Christiansen and T. W. Sederberg. Conversion of complex contour line definition into polygonal element mosaics. *Comput. Graph.*, 12 (3): 187–192, 1978.
- A. Convit, M. J. De Leon, C. Tarshish, S. De Santi, W. Tsui, H. Rusinek, and A. George. Specific hippocampal volume reductions in individuals at risk for Alzheimer's disease. *Neurobiol. Aging*, 18(2):131–138, 1997.
- W. R. Crum, R. I Scahill, and N. C. Fox. Automated hippocampal segmentation by regional fluid registration of serial MRI: Validation and application in Alzheimer's disease. *NeuroImage*, 13:847–855, 2001.
- J. G. Csernansky, J. Hamstra, L. Wang, D. McKeel, J. L. Price, M. Gado, and J. C. Morris. Correlations between antemortem hippocampal volume and postmortem neuropathology in AD subjects. *Alzheimer Disease and Associated Disorders*, 18(4):190–195, 2004.

- J. G. Csernansky, S. Joshi, L. Wang, J. W. Haller, M. Gado, J. P. Miller, U. Grenader, and M. I. Miller. Hippocampal morphometry in schizophrenia by high dimensional brain mapping. *Neurobiology*, 95:11406–11411, 1998.
- J. G. Csernansky, L. Wang, S. Joshi, J. P. Miller, M. Gado, D. Kido, D. McKeel, J. C. Morris, and M. I. Miller. Early DAT is distinguished from aging by high-dimensional mapping of the hippocampus. *Neurology*, 55:1636–1643, 2000.
- J. G. Csernansky, L. Wang, J. Swank, J. P. Miller, M. Gado, D. McKeel, M. I. Miller, and J. C. Morris. Preclinical detection of Alzheimer’s disease: hippocampal shape and volume predict dementia onset in the elderly. *NeuroImage*, 25:783–792, 2005.
- T. K. Dey and S. Goswami. Tight cocone: A water tight surface reconstructor. In *Proc. 8th ACM Symposium on Solid Modeling and Applications*, pages 127–134, 2003.
- J. Dongarra. Freely available software for linear algebra on the web, 2004. <http://www.netlib.org/utk/people/JackDongarra/la-sw.html>.
- I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. John Wiley & Sons, 1998.
- V. Eijkhout. Overview of iterative linear system solver packages, 1997. <http://www.netlib.org/utk/papers/iterative-survey>.
- A. B. Ekoule, F. C. Peyrin, and C. L. Odet. A triangulation algorithm from arbitrary shaped multiple planar contours. *ACM Transactions on Graphics*, 10 (2):182–199, 1991.
- F. Ezekiel, L. Chao, J. Kornak, A.-T. Du, V. Cardenas, D. Truran, W. Jagust, H. Chui, B. Miller, K. Yaffe, N. Schuff, and M. Weiner. Comparisons between global and focal brain atrophy rates in normal aging and Alzheimer disease: Boundary shift integral versus tracing of the entorhinal cortex and hippocampus. *Alzheimer Disease and Associated Disorders*, 18(4):196–201, 2004.
- M. F. Folstein, S. E. Folstein, and P. R. McHugh. Mini-mental state: a practical method for grading cognitive state of patients for the clinicians. *J. Psychiatr. Res.*, 12:189–198, 1975.
- N. C. Fox and P. A. Freeborough. Brain atrophy progression measured from registered serial MRI: validation and application to Alzheimer’s disease. *J. Magn. Reson. Imaging*, 7(6):1069–1075, 1997.

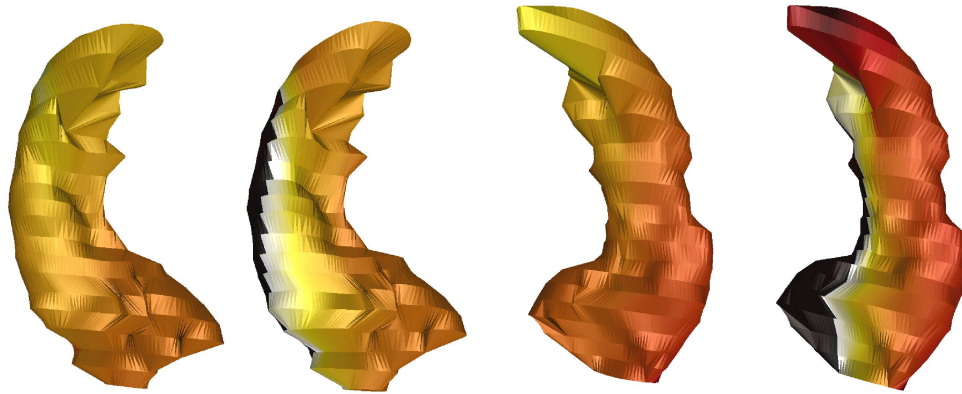
- N. C. Fox, P. A. Freeborough, and M. N. Rossor. Visualisation and quantification of rates of atrophy in alzheimer's disease. *Lancet*, 348(9020):94–97, 1996.
- P. A. Freeborough and N. C. Fox. The boundary shift integral: an accurate and robust measure of cerebral volume changes from registered repeat MRI. *IEEE Trans. Med. Imaging*, 16:623–629, 1997.
- P. A. Freeborough and N. C. Fox. Modeling brain deformations in Alzheimer disease by fluid registration of serial 3d mr images. *Journal of Computer Assisted Tomography*, 22(5):838–843, 1998.
- M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, M. Booth, and F. Rossi. *GNU Scientific Library Reference Manual*, 2005. URL http://www.gnu.org/software/gsl/manual/html_node/index.html.
- M. Garcia-Finana, C. E. Denby, S. S. Keller, U. C. Wieshmann, and N. Roberts. Degree of hippocampal atrophy is related to side of seizure onset in temporal lobe epilepsy. *AJNR Am. J. Neuroradiol*, 27(5):1046–52, 2006.
- S. Gauthier, editor. *Clinical Diagnosis and Management of Alzheimer's Disease*. Martin Dunitz, 1996.
- S. George-Hyslop and A. Petit. Molecular biology and genetics of Alzheimer's disease. *C. R. Biol.*, 328(2):119–30, 2005.
- G. Gerig, K. E. Muller, E. O. Kistner, Y.-Y. Chi, M. Chakos, M. Styner, and J. A. Lieberman. Age and treatment related local hippocampal changes in schizophrenia explained by a novel shape analysis method. *Proc. in Medical Image Computing and Computer-Assisted Intervention*, pages 653–660, 2003.
- G. Gerig, M. Styner, D. Jones, D. Weinberger, and J. Lieberman. Shape analysis of brain ventricles using spharm. *Proc. IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 171–178, 2001.
- P. Golland, W. E. L. Grimson, and R. Kikinis. Statistical shape analysis using fixed topology skeletons: Corpus callosum study. *In Proc. IPMI'99: 16th International Conference on Information Processing and Medical Imaging*, 1613:382–387, 1999.

- H. Groemer. *Geometric applications of Fourier series and spherical harmonics*. Cambridge University Press, 1996.
- P. S. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms. Technical report, 1997. URL <http://www.graphics.cs.uiuc.edu/~garland/papers/simp.pdf>.
- L. L. Heston and J. A. White. *Vanishing Mind: A Practical Guide to Alzheimer's Disease and Other Dementias (Series of Books in Psychology)*. W.H. Freeman & Company, 1991.
- R. Insausti, K. Juottonen, H. Soininen, A. M. Insausti, K. Partanen, P. Vainio, M. P. Laakso, and A. Pitkanen. Mr volumetric analysis of the human entorhinal, perirhinal, and temporopolar cortices. *Am. J. Neuroradiol*, 19(4):659–671, 1998.
- C. R. Jr Jack, D. W. Dickson, J. E. Parisi, Y. C. Xu, R. H. Cha, P. C. O'Brien, S. D. Edland, G. E. Smith, B. F. Boeve, E. G. Tangalos, E. Kokmen, and R. C. Petersen. Antemortem MRI findings correlate with hippocampal neuropathology in typical aging and dementia. *Neurology*, 58(5):750–757, 2002.
- C. R. Jr Jack, R. C. Petersen, Y. Xu, P. C. O'Brien, G. E. Smith, R. J. Ivnik, B. F. Boeve, E. G. Tangalos, and E. Kokmen. Rates of hippocampal atrophy correlate with change in clinical status in aging and AD. *Neurology*, 55(4):484–489, 2000.
- J. C. Janssen, J. M. Schott, L. Cipolotti, N. C. Fox, R. I. Scahill, K. A. Josephs, J. M. Stevens, and M. N. Rossor. Mapping the onset and progression of atrophy in familial frontotemporal lobar degeneration. *J. Neurol. Neurosurg. Psychiatry*, 76(2):162–168, 2005.
- S. Joshi, S. Pizer, P. T. Fletcher, P. Yushkevich, A. Thall, and J. S. Marron. Multiscale deformable model segmentation and statistical analysis using medial descriptions. *IEEE Transactions on Medical Imaging*, 21 (5), 2002.
- M.I. Kamboh. Molecular genetics of late-onset Alzheimer's disease. *Ann. Hum. Genet.*, 68(Pt 4):381–404, 2004.
- S. Kirkpatrick, Jr. C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220, 1983.
- L. Kobbelt, S. Campagna, and H.-P. Seidel. A general framework for mesh decimation. In *In Proceedings Graphics Interface '98*, pages 43–50, 1998.

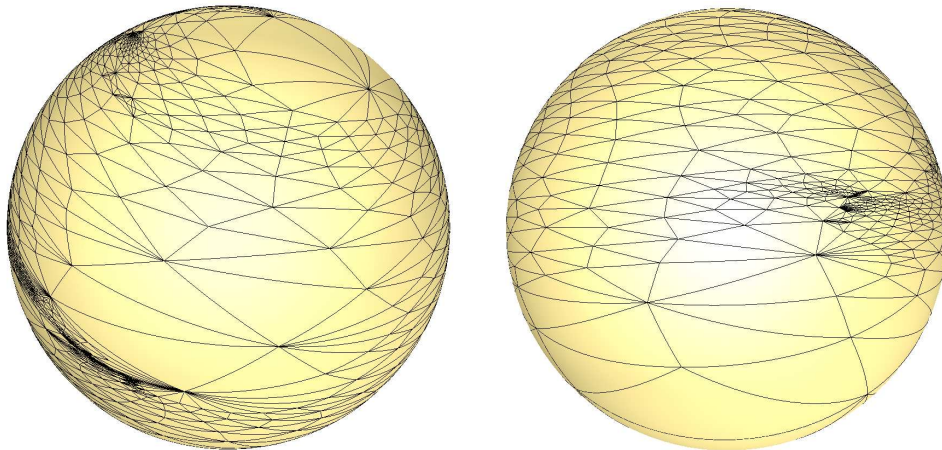
- M. Kubicki, M. E. Shenton, D. F. Salisbury, Y. Hirayasu, K. Kasai, R. Kikinis, F. A. Jolesz, and R. W. McCarley. Voxel-based morphometric analysis of gray matter in first episode schizophrenia. *NeuroImage*, 17(4):1711–1719, 2002.
- M. P. Laakso, M. Lehtovirta, K. Partanen, P. J. Riekkinen, and H. Soininen. Hippocampus in Alzheimer’s disease: a 3-year follow-up MRI study. *Biol. Psychiatry*, 47(6):557–561, 2000.
- S. Lehericy, M. Baulac, J. Chiras, L. Pierot, N. Martin, B. Pillon, B. Deweer, B. Dubois, and C. Marsault. Amygdalohippocampal MR volume measurements in the early stages of Alzheimer disease. *Am. J. Neuroradiol*, 15(5):929–937, 1994.
- D. M. A. Mann. Molecular biology’s impact on our understanding of aging. *BMJ*, 315:1078–1081, 1997.
- K. Maurer, S. Volk, and H. Gerbaldo. Auguste D and Alzheimer’s Disease. *The Lancet*, 349:1546–49, 1997.
- C. McDonald, N. Marshall, P. C. Sham, E. T. Bullmore, K. Schulze, B. Chapple, E. Bramon, F. Filbey S. Quraishi, M. Walshe, and R. M. Murray. Regional brain morphometry in patients with schizophrenia or bipolar disorder and their unaffected relatives. *Am J Psychiatry*, 163(3):478–87, 2006.
- G. McKhann, D. Drachman, and M. Folstein. Clinical diagnosis of alzheimer’s disease: report of the NINCDS-ADRDA work group under the auspices of department of health and human services task force on alzheimer’s disease. *Neurology*, 34:939–944, 1984.
- D. Meyers, S. Skinner, and K. Sloan. Surfaces from contours. *ACM Transactions on Graphics*, 11 (3):228–258, 1992.
- J. C. Morris. The clinical dementia rating (CDR): current version and scoring rules. *Neurology*, 43:2412–2414, 1993.
- S. Murayama and Y. Saito. Neuropathological diagnostic criteria for Alzheimer ’ disease. *Neuropathology*, 24:254–260, 2004.
- K. L. Narr, P. M. Thompson, P. Szeszko, D. Robinson, S. Jang, R. P. Woods, S. Kim, K. M. Hayashi, D. Asuncion, A. W. Toga, and R. M. Bilder. Regional specificity of hippocampal volume reductions in first-episode schizophrenia. *Neuroimage*, 21(4):1563–1575, 2004.

- J. Nocedal, A. Wächter, and R. A. Waltz. Adaptive barrier strategies for nonlinear interior methods. *Research Report RC23563, IBM T. J. Watson Research Center, Yorktown, USA*, 2005.
- S. J. Owen, 2006. URL <http://www.andrew.cmu.edu/user/sowen/software/triangle.html>.
- C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Inc., 1982.
- S. M. Pizer, D. S. Fritsch, P. A. Yushkevich, V. E. Johnson, and E. L. Chaney. Segmentation, registration, and measurement of shape variation via image object shape. *IEEE Transactions on Medical Imaging*, 18(10):851–865, 1999.
- E. Praun and H. Hoppe. Spherical parametrization and remeshing. *ACM Trans. Graph.*, 22(3):340–349, 2003.
- J. L. Price. Aging, preclinical Alzheimer disease, and early detection. *Alzheimer Disease and Associated Disorders*, 17:S60–S62, 2003.
- M. Quicken, G. Gerig, and O. Kübler. Parametrization of closed surfaces for parametric surface descriptions. In *CVPR 2000*, 2000.
- A. B. Rodgers. *Alzheimer’s Disease: Unraveling the Mystery*. National Institutes of Health, 2002. www.alzheimers.org/unraveling.
- R. Schneiders, 2006. URL <http://www-users.informatik.rwth-aachen.de/~roberts/software.html>.
- L. Shen, F. Ford, F. Makedon, and A. Saykin. Hippocampal shape analysis: surface-based representation and classification. *SPIE Medical Imaging 2003: Image Processing*, Proc. 5032, 2003.
- P. Spellucci. *Numerische Verfahren der nichtlinearen Optimierung*. Birkhäuser Verlag, 1993.
- M. Styner and G. Gerig. Hybrid boundary-medial shape description for biologically variable shapes. *Proc. of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 235–242, 2000.
- M. Styner, J. A. Lieberman, D. Pantazis, and G. Gerig. Boundary and medial shape analysis of the hippocampus in schizophrenia. *Medical Image Analysis*, 8:197–203, 2004.

- R. Tepest, L. Wang, M. I. Miller, P. Falkai, and J. G. Csernansky. Hippocampal deformities in the unaffected siblings of schizophrenia subjects. *Biol. Psychiatry*, 54:1234–1240, 2003.
- R. D. Terry, R. Katzman, and K. L. Bick, editors. *Alzheimer Disease*. Raven Press, 1994.
- D’Arcy W. Thompson. *On Growth and Form*. Cambridge University Press, 1917.
- P. M. Thompson, K. M. Hayashi, G. I. De Zubicaray, A. L. Janke, S. E. Rose, J. Semple, M. S. Hong, D. H. Herman, D. Gravano, D. M. Doddrell, and A. W. Toga. Mapping hippocampal and ventricular change in Alzheimer disease. *Neuroimage*, 22(4):1754–66, 2004.
- A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.
- L. Wang, J. S. Swank, I. E. Glick, M. H. Gado, M. I. Miller, J. C. Morris, and J. G. Csernansky. Changes in hippocampal volume and shape across time distinguish dementia of the Alzheimer type from healthy aging. *Neuroimage*, 20:667–682, 2003.
- E. W. Weisstein. “Lambert Azimuthal Equal-Area Projection.” from mathworld—a wolfram web resource. (February 18, 2004), 2004. URL <http://mathworld.wolfram.com/LambertAzimuthalEqual-AreaProjection.html>.
- E. W. Weisstein. “Spherical Harmonic.” from mathworld—a wolfram web resource. (february 17, 2005), 2005. URL <http://mathworld.wolfram.com/SphericalHarmonic.html>.
- G. Wollny and F. Kruggel. Computational cost of nonrigid registration algorithms based on fluid dynamics. *IEEE Trans. Medical Imaging*, 21:946–952, 2002.



(a) Parametrisation of the right hippocampus. (b) Parametrisation of the left hippocampus.



(c) Mesh on the sphere of the right hippocampus. (d) Mesh on the sphere of the left hippocampus.

Figure 1: Spherical parametrisation of the hippocampi of one patient. The top row shows the latitudinal and longitudinal parametrisation of the (b) left and (a) right hippocampus. Below the spherical parametrisations of the same two hippocampi are given ((c) right hippocampus; (d) left hippocampus).

Appendix A

Definition of Shape

Definition A.1. *Shape is all the geometrical information that remains when location, scale and rotational effects are filtered out from an object. (Dryden and Mardia, 1998)*

In other words, the term shape is invariant to Euclidean transformations, as shown in Figure A.1.

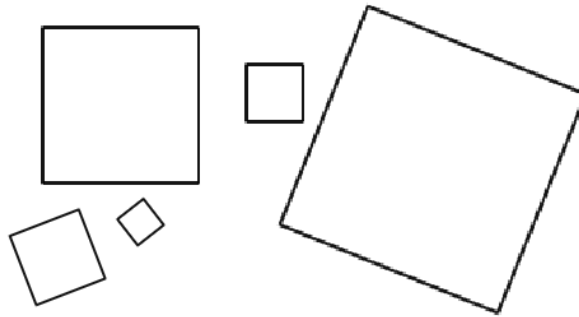


Figure A.1: Five exact copies of the same square under different Euclidian transformations.

Appendix B

Spherical Harmonics

Like Fourier transforms in 1D and 2D, spherical harmonics are a compact representation of functions on the unit sphere by means of a set of orthogonal basis functions. The spherical harmonics $Y_l^m(\theta, \phi)$ are the angular portion of the solution to Laplace's equation in spherical coordinates where azimuthal symmetry is not present.

The following explanation is extracted from Weisstein (2005). For a more detailed explanation of the formulas see Byerly (1893) or Groemer (1996).

Spherical harmonics satisfy the spherical harmonic differential equation, which is given by the angular part of Laplace's equation in spherical coordinates. Writing $F = \Phi(\phi)\Theta(\theta)$ in this equation gives

$$\frac{\Phi(\phi)}{\sin \theta} \frac{d}{d\theta} \left(\sin \theta \frac{d\Theta}{d\theta} \right) + \frac{\Theta(\theta)}{\sin^2 \theta} \frac{d^2 \Phi(\phi)}{d\phi^2} + l(l+1)\Theta(\theta)\Phi(\phi) = 0. \quad (\text{B.1})$$

Multiplying by $\sin^2 \theta / (\Theta\Phi)$ gives

$$\left[\frac{\sin \theta}{\Theta(\theta)} \frac{d}{d\theta} \left(\sin \theta \frac{d\Theta}{d\theta} \right) + l(l+1) \sin^2 \theta \right] + \frac{1}{\Phi(\phi)} \frac{d^2 \Phi(\phi)}{d\phi^2} = 0. \quad (\text{B.2})$$

Using separation of variables by equating the ϕ -dependent portion to a constant gives

$$\frac{1}{\Phi(\phi)} \frac{d^2 \Phi(\phi)}{d\phi^2} = -m^2, \quad (\text{B.3})$$

which has solutions

$$\Phi(\phi) = Ae^{-im\phi} + Be^{im\phi}. \quad (\text{B.4})$$

Combining the information gives the equation for the ϕ -dependent portion, whose solution is

$$\Theta(\theta) = P_l^m(\cos \theta), \quad (\text{B.5})$$

where $m = -l, -(l-1), \dots, l-1, l$ and $P_l^m(z)$ is an associated Legendre polynomial. The spherical harmonics are then defined by combining $\Phi(\phi)$ and $\Theta(\theta)$,

$$Y_l^m(\theta, \phi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos \theta) e^{im\phi}, \quad (\text{B.6})$$

where the normalisation is chosen such that

$$\int_0^{2\pi} \int_0^\pi Y_l^m(\theta, \phi) \overline{Y_{l'}^{m'}}(\theta, \phi) \sin \theta d\theta d\phi = \quad (\text{B.7})$$

$$\int_0^{2\pi} \int_{-1}^1 Y_l^m(\theta, \phi) \overline{Y_{l'}^{m'}}(\theta, \phi) d(\cos \theta) d\phi = \delta_{mm'} \delta_{ll'} \quad (\text{B.8})$$

(Arfken 1985, p. 681). Here, \bar{z} denotes the complex conjugate and δ_{mn} is the Kronecker delta. [...]

The spherical harmonics are sometimes separated into their real and imaginary parts,

$$Y_l^{ms}(\theta, \phi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos \theta) \sin(m\phi) \quad (\text{B.9})$$

$$Y_l^{mc}(\theta, \phi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos \theta) \cos(m\phi). \quad (\text{B.10})$$

The spherical harmonics obey

$$Y_l^{-l}(\theta, \phi) = \frac{1}{2^l l!} \sqrt{\frac{(2l+1)!}{4\pi}} \sin^l \theta e^{-il\phi} \quad (\text{B.11})$$

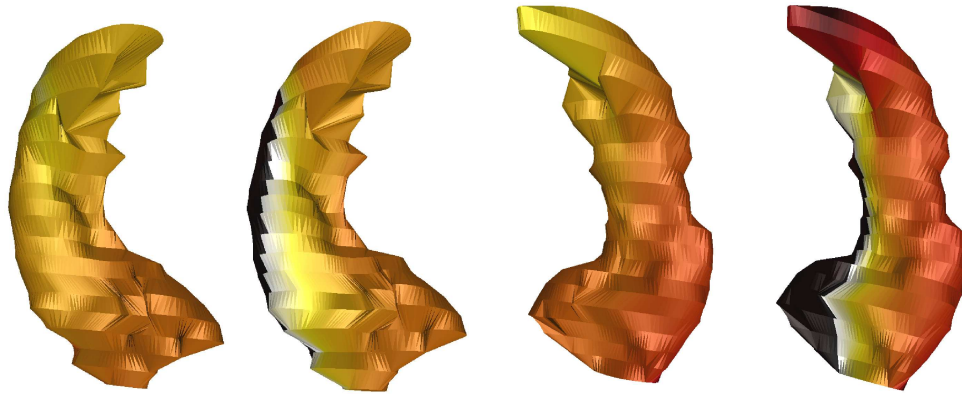
$$Y_l^0(\theta, \phi) = \frac{2l+1}{4\pi} P_l(\cos \theta) \quad (\text{B.12})$$

$$Y_l^{-m}(\theta, \phi) = (-1)^m \overline{Y_l^m(\theta, \phi)}, \quad (\text{B.13})$$

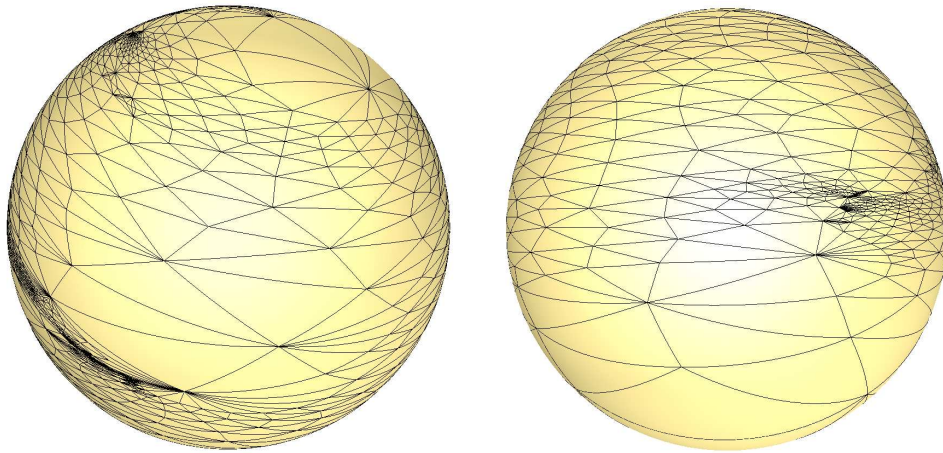
where $P_l(x)$ is a Legendre polynomial.

Appendix C

Illustrations

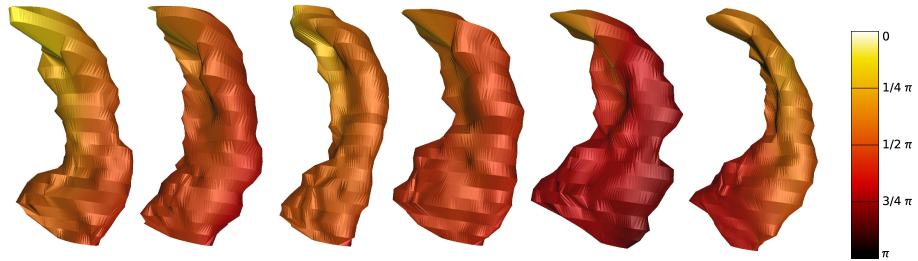


(a) Parametrisation of the right hippocampus. (b) Parametrisation of the left hippocampus.

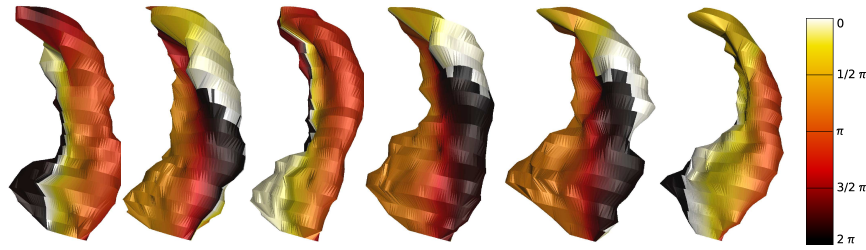


(c) Mesh on the sphere of the right hippocampus. (d) Mesh on the sphere of the left hippocampus.

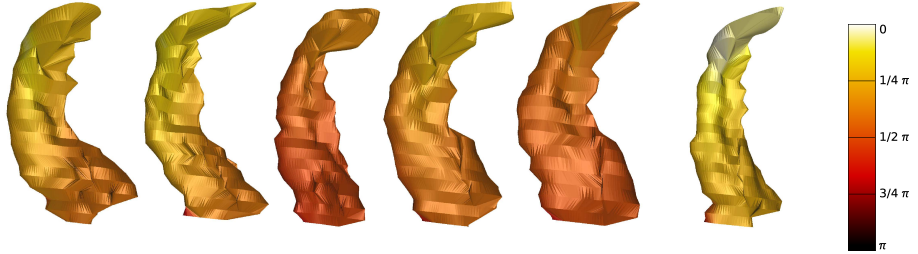
Figure C.1: Spherical parametrisation of the hippocampi of one patient. The top row shows the latitudinal and longitudinal parametrisation of the (b) left and (a) right hippocampus. Below the spherical parametrisations of the same two hippocampi are given ((c) right hippocampus; (d) left hippocampus).



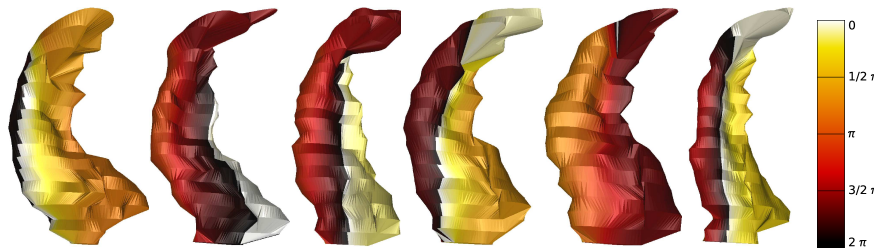
(a) Latitude of right hippocampi.



(c) Longitude of right hippocampi.

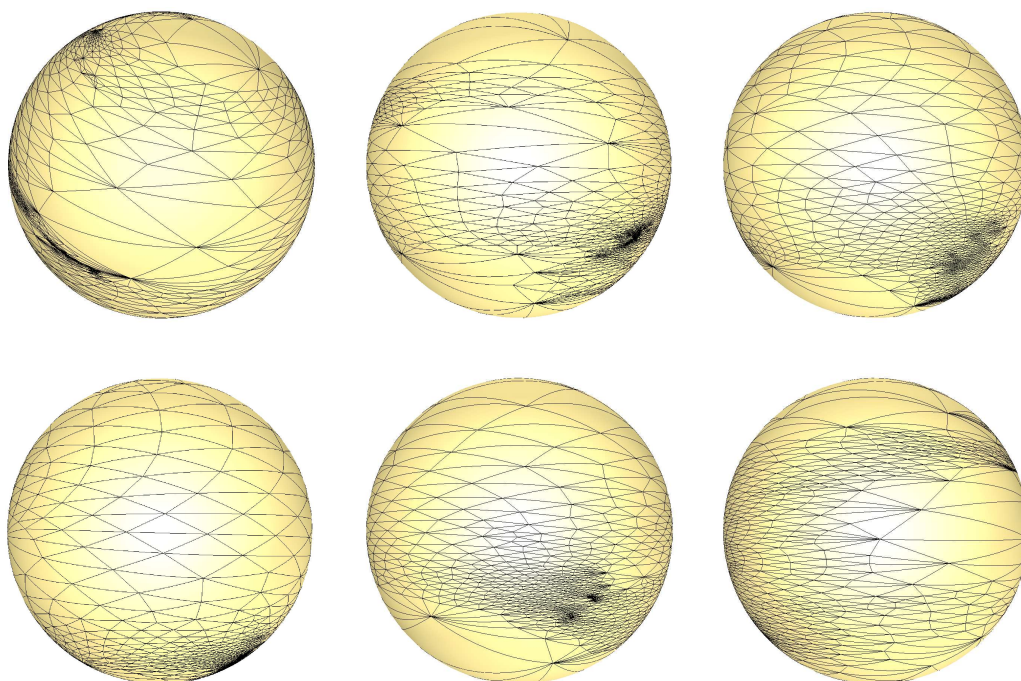


(e) Latitude of left hippocampi.

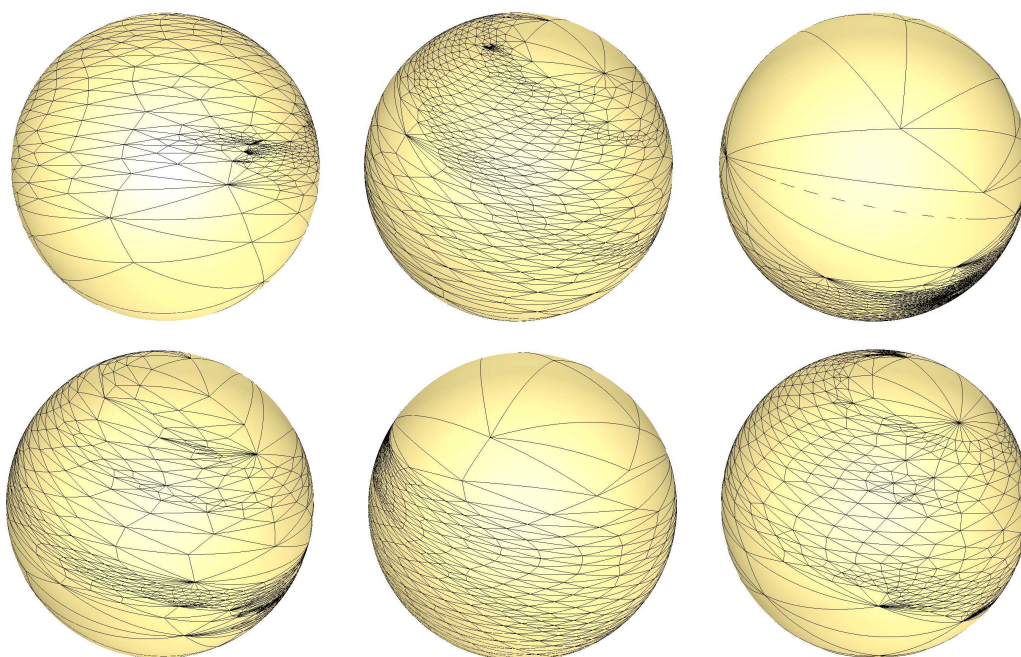


(g) Longitude of left hippocampi.

Figure C.2: Spherical parametrisation of the six pairs of hippocampi.



(a) Spherical parametrisation of the left hippocampi.



(b) Spherical parametrisation of the right hippocampi.

Figure C.3: Spherical parametrisation of the hippocampi depicted in Figure C.2 (top row: hippocampus 1–3, bottom row: hippocampus 4–6).

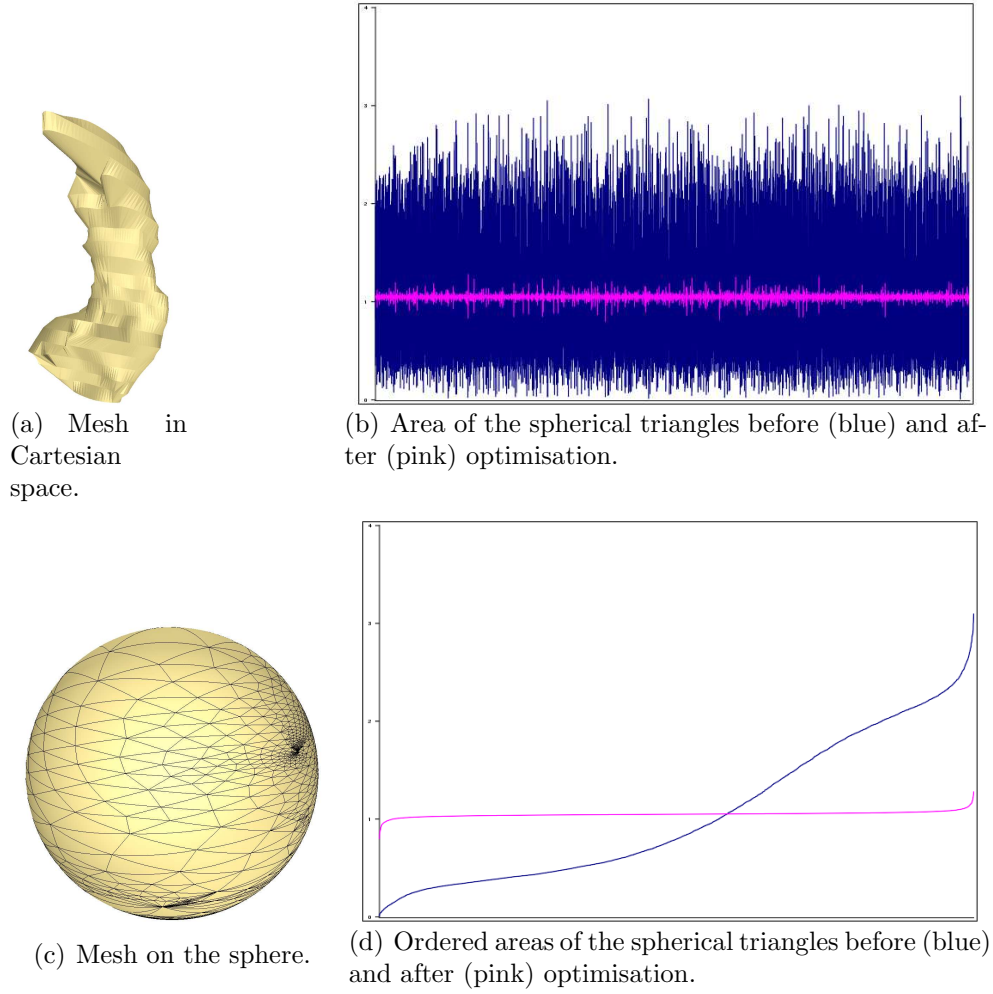


Figure C.4: Optimisation of a hippocampus using constrained optimisation: (a) The surface mesh in Cartesian space. (c) The initial (not optimised) spherical mesh. The relative areas ($\rho = \text{area}/\text{surface area}$) of the spherical triangles before (blue) and after (pink) optimisation are given in Figure (b). Figure (d) shows the same plot, where the areas were ordered according to size.

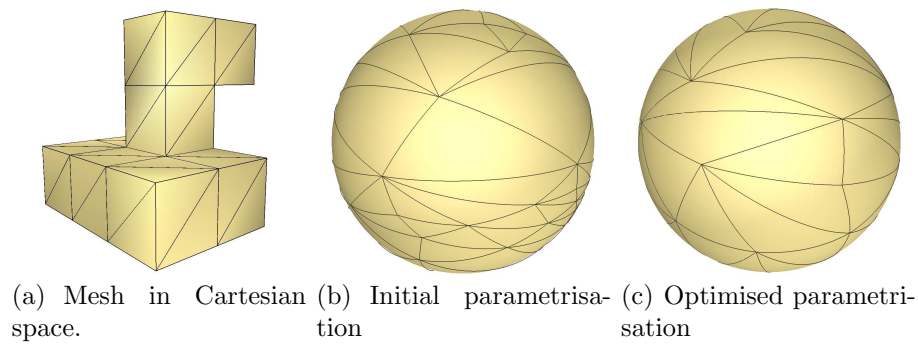


Figure C.5: Simulated annealing optimisation of a mesh consisting of 36 vertices (a). The initial parametrisation (b) shows great differences in the sizes of the triangles. After the optimisation (c) the distortions of the areas of the triangles are eliminated.

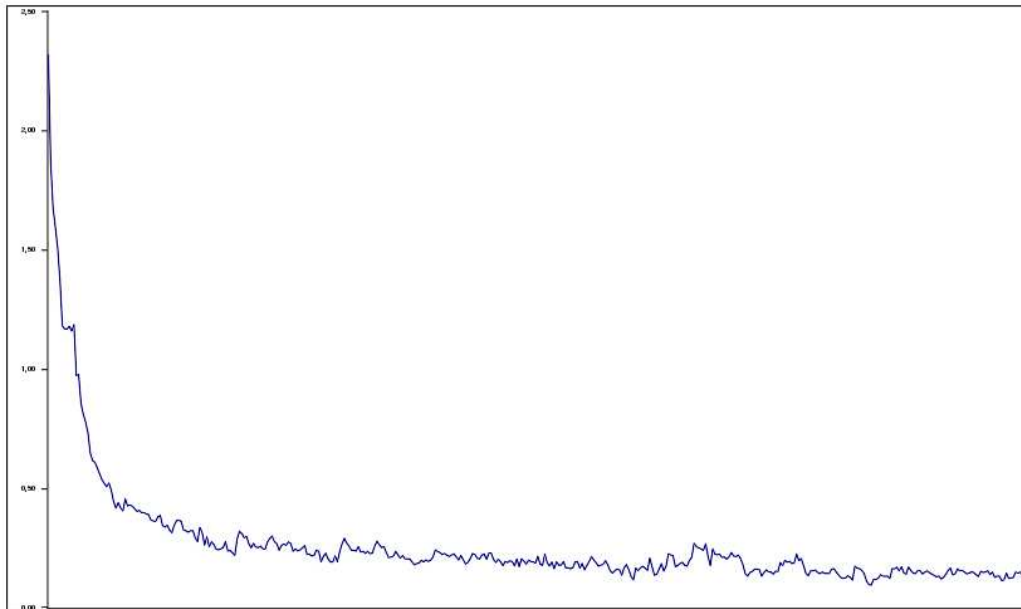


Figure C.6: Energy function of the simulated annealing process of the parametrisation given in C.5(b)

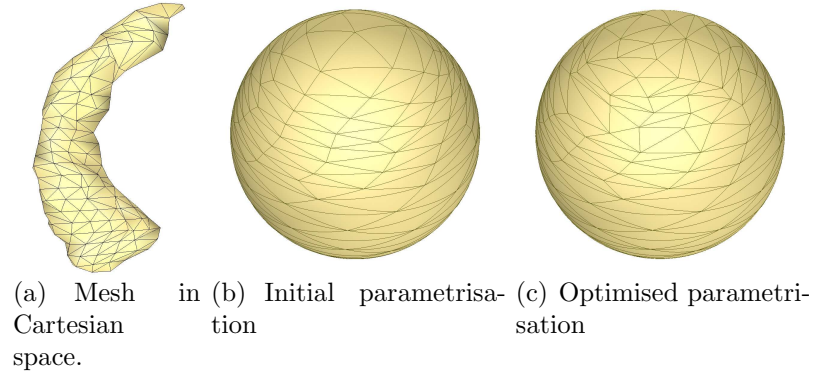


Figure C.7: Simulated annealing optimisation of a mesh consisting of 217 vertices (a). At the poles, the initial parametrisation (b) shows great differences in the sizes of the triangles. After the optimisation (c) these distortions are eliminated. Nevertheless, the triangles exhibit still distortions with respect to triangular areas.

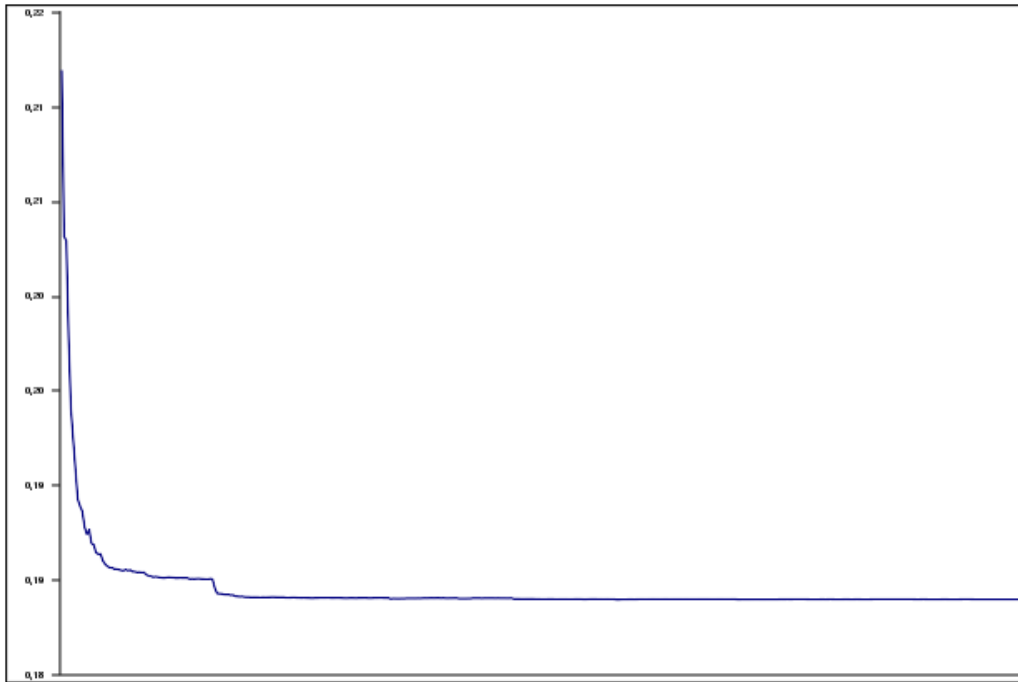
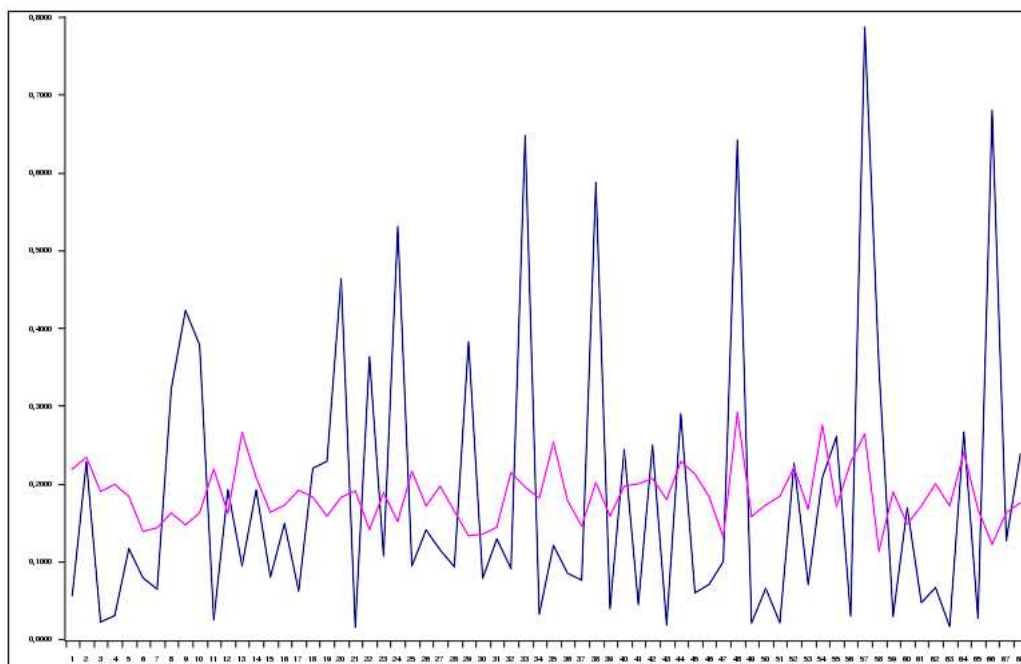
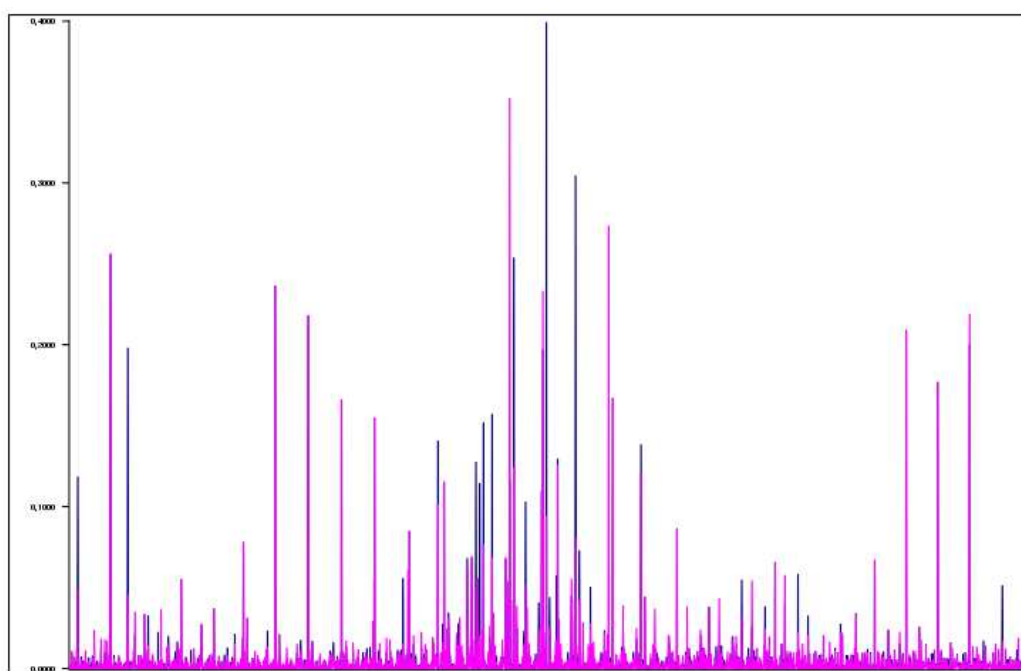


Figure C.8: Energy function of the simulated annealing process of the parametrisation given in C.7(b).



(a) Areas of the mesh shown in Figure C.5(a).



(b) Areas of the mesh shown in Figure C.7(a).

Figure C.9: The areas of the triangles before (blue) and after (pink) optimisation.

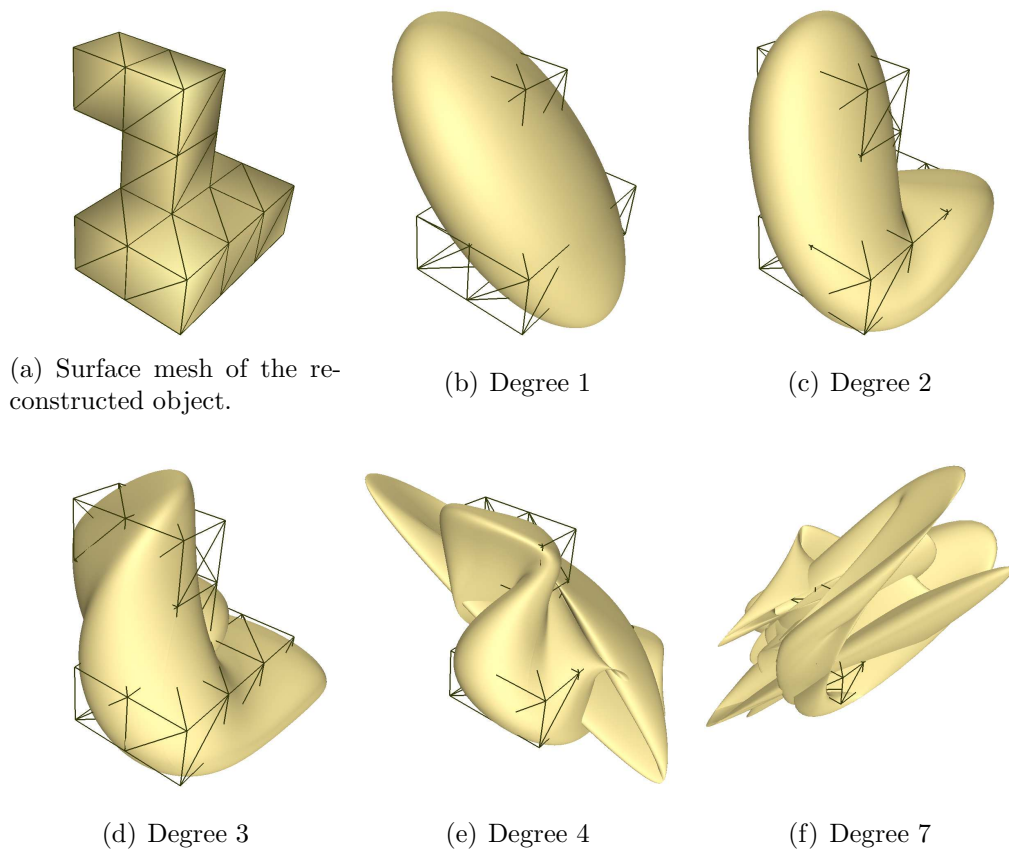


Figure C.10: Reconstruction of the surface mesh shown in (a) using spherical harmonics, **before** the parametrisation was optimised. Partial series with coefficients up to degree (b) 1, (c) 2, (d) 3, (e) 4 and (f) 7 are used to compute the surface.

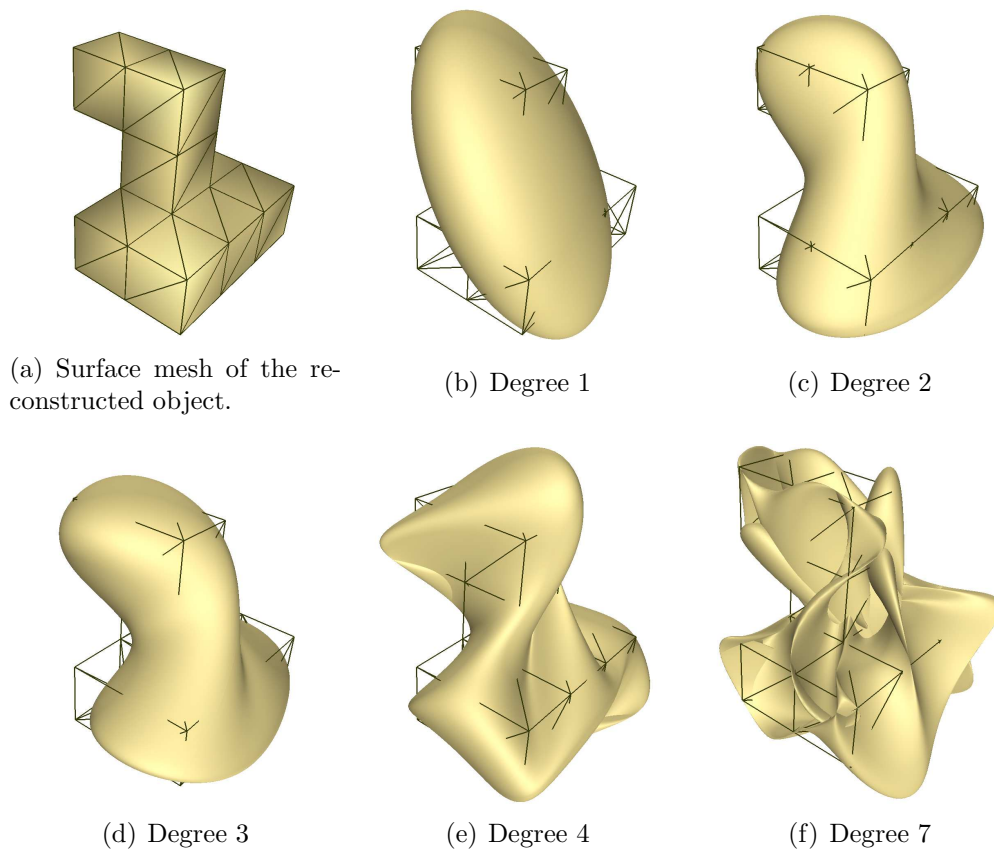


Figure C.11: Reconstruction of the surface mesh shown in (a) using spherical harmonics, **after** the parametrisation was optimised. Partial series with coefficients up to degree (b) 1, (c) 2, (d) 3, (e) 4 and (f) 7 are used to compute the surface.

Appendix D

Pseudocode

Algorithm 1 Set up Laplace equation for latitude (Brechtbühler et al., 1995)

Set up matrix A:

for $vertex = 1 \dots n$ **do**

$a_{vertex,vertex} :=$ number of direct neighbours;

for the direct *neighbors* of $vertex$ **do**

if $neighbour$ is not a pol **then**

$a_{vertex,neighbour} = -1$;

end if

end for

end for

Set up vector b:

Set all entries of b to 0;

for the direct *neighbours* of $south_pole$ **do**

$b_{neighbour} := \pi$

end for

Algorithm 2 Set up Laplace equation for longitude (Brechtbühler et al., 1995)

Modify matrix A:

for both *poles* **do**

for the direct *neighbors* of *pole* **do**

$a_{\text{neighbour}, \text{neighbour}} = -1;$

end for

end for

$a_{0,0} = 2;$

Set up vector b:

for $row = 1 \dots n$ **do**

$b_{row} := 0;$

end for

$previous := north_pole;$

$here := 1;$

$maximum := 0.0;$

while $here \neq south_pole$ **do**

for the direct *neighbours* of *here* **do**

if $\theta_{\text{neighbour}} > maximum$ **then**

$maximum := \theta_{\text{neighbour}};$

$nextpos := \text{position of neighbor};$

end if

if $\text{neighbour} == previous$ **then**

$prevpos := \text{position of neighbour}$

end if

end for

for the direct *neighbours* clockwise between $prevpos$ and $nextpos$ **do**

 add 2π to $b_{\text{neighbour}};$

 subtract 2π from $b_{\text{here}};$

end for

$previous := here;$

$here := \text{neighbour of here indicated by nextpos};$

end while

Algorithm 3 Set up of f and Y

```
for  $i = 0$  to  $n_{verts}$  do  
   $f[i][j] = v_{i,j}$  with  $j \in \{x, y, z\}$   
  for  $l = 0$  to  $order_{harms}$  do  
    for  $m = -l$  to  $l$  do  
       $Y[i][l^2 + l + m] = Y_l^m(\theta_i, \phi_i);$   
    end for  
  end for  
end for
```
