3 Dimensional modelling of the muscles of the human tongue and implications to surgical management of cancer

Konstantinos Pataridis

Wolfson College, Oxford



A Thesis submitted for the degree of
Doctor of Philosophy

Nuffield Department of Surgery
Radiology Group
University of Oxford
January 2007

Abstract

Imaging has an important role in demonstrating the tongue for clinical purposes, especially in staging cancer. However present staging schemes have limitations, particularly in their relation to criteria for surgical resection. Progressively, image processing has been used to obtain more information from image data than is possible from conventional display. One form that is particularly appropriate for surgical purposes is three dimensional image modelling.

This project has constructed a template three dimensional model of the extrinsic muscles of the tongue to assist surgical management of cancer. This is achieved by non-rigid registration of a template constructed from a cryophotography-based series to the MRI volume acquired from the patient. The registration is performed in 2.5 and 3 dimensions. The structure of the tongue on the MR images is segmented by using automated and semi-automated methods based on active contours. Validation is achieved at each stage by a combination of procedural data and image data acquired from volunteers and clinical subjects.

Successful construction of the model describing tongue architecture and its relation to test cases was achieved. The end model is demonstrated in three dimensional volume rendering to facilitate understanding of the spatial correlation of tumour and myoarchitecture. Depiction of significant muscles and the margins of tumours has been demonstrated to a level within surgical tolerance. The method is readily applicable without a priori specific knowledge of the intrinsic computer algorithms.

This work has made significant contribution to developing approaches to computer-assisted surgery of the tongue. Clinical trials of this system are now indicated and it also appears adaptable to other tumour sites. It offers a valuable technique for further research in oral physiology and pathology, including the science of articulation and phonetics.

Dedicated to

the rest of ASD
my beloved friends
Amoivikos, Ch3, aMUSiC, Leviathan

Contents

1 Introduction **2**

1.1 Motivation 2

1.2 Scope and significance of thesis 3

1.2.1 Implementation details 4

1.3 Organization of thesis 5

2 Background **7**

2.1 Introduction 7

2.1.1 Organization of sections 7

2.2 Tongue 8

2.2.1 Basic anatomy - physical appearance of the tongue . . 8

2.2.2 Extrinsic attachments of the tongue muscles 9

2.2.3 Extrinsic muscles 10

2.2.4 Intrinsic muscles 12

2.3 Biomedical Imaging Modalities: an overview 13

2.3.1 X-ray Radiography and Cineradiography 14

2.3.2 Ultrasound Imaging 15

2.3.3 Computed Tomography 17

2.3.4 Magnetic Resonance Imaging: overview 19

2.4 MRI: technical basis and applications 20

2.4.1 The MR Phenomenon - spin and excitation 20

Relaxation - the biological parameters T_1 and T_2 in resonance 21

T_1 Effect 21
T_2 Effect 21
2.4.2 Spatial encoding 22
Slice selection 22
Frequency and phase selection 23
K-space and image reconstruction 23
2.4.3 Image characteristics 24
Weighting factors that affect image brightness 25
Factors affecting Signal to Noise ratio 27
Contrast agents for MRI 28
2.4.4 Ability of MRI to display the tongue 29
Variability of tongue shape on MR images 30
Presentation of tongue on different weighted images . . 31
2.5 Cancer 31
2.5.1 Causes, Symptoms, Diagnosis and Treatment 31
Diagnosis 32
Biopsy 32
Treatment 33
2.6 Cancer of the oral cavity 33
2.6.1 Etiology and risk factors 33
2.6.2 Clinical presentation 34
2.6.3 Clinical presentation of tongue cancer in MRI 35
2.6.4 Treatment 35
2.7 Disease staging: the TNM System 36
2.7.1 Mandatory parameters 'T', 'N' and 'M' 36
2.7.2 Other parameters 37
2.7.3 Prefix modifiers 37
2.7.4 TNM classification of tongue cancer 37
2.7.5 Disadvantages of TNM system for tongue cancer 38
2.7.6 Relation to this experiment 38
2.8 Summary 38

3 General experimental procedures and generation of data volumes **40**

3.1 Introduction 40
3.1.1 Organization of sections 41
3.2 Survey of tongue modelling approaches 42
3.2.1 Introduction 42
3.2.2 2D articulatory modelling 42
3.2.3 3D tongue modelling 44
3.2.4 Geometrical modelling of the tongue based on the MRI modality 46
3.2.5 Geometrical modelling of the tongue based on photography 49
3.2.6 Summary 52
3.3 Geometric modelling of the tongue and extrinsic muscles . . . 53
3.3.1 The Visible Human Project 54
3.3.2 The Female Visible Human Project dataset 56
3.3.3 Preparation of the Visible Human Project data 56
Reading the images - conversion 56
Selection of axial slices containing the oral cavity . . 56
Cropping of the selected axial slices 57
3.3.4 Geometric modelling 58
3.3.5 Validation through data visualization 59
3.3.6 Summary 63
3.4 Retrieval of tongue from MR scans 63
3.4.1 Description of the MR sequences 63
3.4.2 Shortcomings of the MR images 64
3.4.3 Experiment of image enhancement using Gadolinium . 65
The use of Gadolinium in pectin 66
Experimental procedures: scanning sequences and subject 68
3.4.4 Findings and evaluation of the experiment 69

3.5 Significance of findings 70

4 Coplanar anisotropic volumes fusion 73

4.1 Introduction 73
4.1.1 Requirement for isotropy 73
4.1.2 Fusion of coronal and sagittal volumes 74
4.1.3 Organization of sections 77
4.2 Image processing of optical flow 77
4.2.1 Optical flow constraint equations 79
4.3 Lucas-Kanade Flow 80
4.4 Morphing along optical flow 82
4.5 Fusion of multi-planar anisotropic volumes 83
4.5.1 1D Magnification filtering 84
4.6 Single modality rigid registration 86
4.6.1 Rigid registration using cross-correlation 87
4.6.2 Rigid registration using phase-correlation 88
4.7 Implementation and results 89
4.7.1 Software implementation 89
4.7.2 Results 90
Scanning procedure 90
Fusion of the two volumes 90
Efficiency of software 92
4.8 Discussion of Results 92
4.9 Conclusions 94

5 Automated extraction of tongue model 97

5.1 Introduction 97
5.2 Feature extraction mechanisms 98
5.2.1 Segmentation based on geometric features 99
5.2.2 Edge detection: the Canny edge detector 100
Edge detectors 100
Canny edge detector 101

Noise reduction 102
Determination of intensity gradient 102
Tracing of edges through the image 103
5.2.3 Extraction of contour 105
Radial Sweep algorithm 106
5.2.4 Encoding of contour using Freeman chain code 107
5.2.5 Active Contours - Snakes 109
Snakes 110
The Greedy Algorithm for Snakes 113
Summary 119
5.2.6 Straight line detection: Hough transformation 119
5.3 Image Restoration (Preprocessing) 122
5.3.1 Anisotropic diffusion 123
Definition 123
Discrete implementation in 2D 125
5.4 Implementation 127
5.5 Extraction of contours in VHP volume 128
5.6 Extraction of contours in MRI volume 128
5.6.1 Preprocessing 129
5.6.2 Midsagittal line calculations 129
5.6.3 Use of midsagittal line $\overline{n}$ 131
5.6.4 Use of a snake to segment the tongue 132
Initialization: problem of initial conditions - requirements for a bounding ellipse 133
Problem of initial conditions - alternative methods . . 134
Summary of method 137
Iteration of snake 138
Evaluation of snake 138
5.7 Results 139
5.7.1 VHP domain contours 139
5.7.2 MRI domain contours 140

Generation of in-between contours by interpolation . . 142
5.8 Discussion 143
5.9 Conclusions 146

6 Soft body dynamics in registration **148**
6.1 Introduction 148
6.1.1 Organization of sections 149
6.2 The problem of registering the two domains 149
6.2.1 Definition of a relation between coordinate systems . . 151
6.2.2 Segmentation of reference features and definition of a disparity or similarity function between extracted features 151
Segmented feature-based techniques 152
Image-based techniques 152
6.2.3 Optimization of the similarity or disparity function . . 152
6.3 Elastic registration based on Boundary mapping 155
6.3.1 Overview 155
6.3.2 Z-axis contour alignment 156
6.3.3 Coordinate system transformation 157
6.3.4 Homothetic mapping of the contours 158
Spline representation 159
Selection of number of control points 161
Creating a constant speed spline 161
Homothetic mapping of the splines 164
Correspondence of control point: use of curvature as a similarity metric 164
Determination of circular shift O 165
6.3.5 Elastic deformation: Spring-Mass-Damper systems . . 168
Spring-Mass-Damper unit 169
Spring-Mass-Damper network 169
Explicit Integration - Fourth order Runge-Kutta 171
Dynamic modelling of the SMD network 173

6.3.6 Construction of SMD mesh in 2D 174
Mesh generation in detail 177
6.3.7 Mapping of UV coordinates 178
6.4 Result images 181
6.5 Evaluation of methods and discussion 183
6.5.1 Bounding box alignment 183
6.5.2 Contour matching 185
Evaluation of Homothetic mapping 186
Conclusions on Homothetic mapping evaluation 188
6.5.3 Elastic deformation based on SMD network 189
Evaluation of the Elastic deformation step 190
6.5.4 UV texture mapping 194
6.5.5 Conclusion 194

7 Extensions in 3D **197**
7.1 Introduction 197
7.1.1 3D boundary mapping 199
7.1.2 Overview 199
7.2 Generation of planar B-Spline curve 199
7.2.1 Bicubic uniform B-spline surface representation 200
7.2.2 Approximation of general shape of object with a B-Spline surface 201
Construction of volumetric object 204
Initialization of 3D active contour 204
Iteration of the 3D active contour 205
Post-processing: bounding box alignment 207
7.3 Homothetic mapping of the bounding surface 208
7.3.1 Selection of number of control points 208
Creating a constant speed B-Spline 209
Curvature as a similarity metric 210
Homothetic mapping - determination of circular shift . 211
7.4 Spring-mass-damper system 212

7.4.1 Construction of 3D Spring-mass-damper network . . . 212
7.4.2 The Spring-mass-damper system in action 213
7.4.3 Mapping of UV coordinates 213
7.4.4 Result images 215
Comparison test to model generated using the 2D registration scheme 216
7.4.5 Discussion and conclusion 216

8 Validation and Results **222**
8.1 Introduction 222
8.2 Validation using a phantom 223
8.2.1 Objective 223
8.2.2 Materials 224
8.2.3 Experimental procedure 224
8.2.4 Results 226
8.2.5 Conclusion 226
8.3 Validation using clinical data 227
8.3.1 Case 1 228
8.3.2 Case 2 229
8.3.3 Case 3 230
8.4 Significance of findings 231

9 General Discussion and Conclusion **242**
9.1 Contributions of this thesis 242
9.1.1 Construction of a 3D atlas of tongue musculature . . . 243
9.1.2 Cavitatory enhancement with Gadolinium contrast agent in solution 243
9.1.3 Construction of the 3D model of the tongue based on MRI 244
9.1.4 Fusion of anisotropically sampled MR volumes into a single isotropic 244
9.1.5 Registration of VHP atlas tongue on MRI derived tongue 245

9.1.6 Construction of a system to identify tongue musculature on MR images 245
9.2 Limitations of the system 246
9.2.1 The VHP atlas model 246
9.2.2 Limitations of MR images in segmentation 247
9.3 Further clinical applications of the model and system 248
9.3.1 Articulation disorders 248
9.3.2 Post-operative evaluation of partial glossectomy 249
9.3.3 Swallowing Disorders 249
9.3.4 Other areas 249
9.4 Future plans 250
9.4.1 Diffusion Tensor MRI 250
9.4.2 Direction of muscle fibres: implications in terms of staging and treatment 251
9.4.3 Image Guided Surgery and surgery simulation 252
9.5 Conclusion 254

List of Figures

2.1 3D Rendering of the skeletal framework for the tongue (from CT scan). 9
2.2 Extrinsic muscles of the tongue in para-sagittal section. From (http:// face-and-emotion.com/ dataface/ anatomy/ tongue-muscles.jsp) 11
2.3 3D Rendering of the Intrinsic and Extrinsic muscles of the tongue, taken from the Upper airway project website [SIA02]. 13
2.4 Sagittal X-ray photograph of the tongue. 14
2.5 Ultrasound imaging of the tongue in sagittal plane [Wat99]. . 16
2.6 Axial image of the tongue acquired with CT. 18
2.7 Representation of pixels on image and voxels in volume. 24
2.8 T_1 and T_2 weighted MRI sagittal images of the same subject. . 25
2.9 Normal tongue (left) and tongue that bulges due to missing teeth (right) on T_1. 30
2.10 Presentation of large tumour on T_2 image. Tumour has higher signal intensity than surrounding tissue. 34

3.1 Posterior Genioglossus and Styloglossus muscles as depicted using the model by Sanguinet [SVY97]. 43
3.2 The early 2D model suggested by Perkell [J.74]. 44
3.3 The 3D finite element model by Kiritani. [ea76] 45
3.4 The geometrical model of the tongue suggest by Honda, as an aid in phonetics experiments [DH02]. 47

3.5 A shaded 3D render of a tongue model calculated by Evangelou using MRI images [Eva04]. 48
3.6 A representation of the model developed by Tricarico, with the different muscles depicted in different colour [WT06b]. . . 50
3.7 3D shaded models of the tongue muscles, as depicted in Upper airway project website [SIA02]. 52
3.8 Different views of Upper airway project models (Upper airway website) [SIA02]. 53
3.9 The Visible Human Project in original axial form. 55
3.10 The Visible Human Project: different views of the tongue in grayscale, generated through Multiplanar reconstruction (MPR). 57
3.11 Muscle tagging using the annotation tool. 59
3.12 MIP rendering with shading of the Visible Human Project based tongue and extrinsic muscles atlas. 60
3.13 The 3D rendered models of tongue and extrinsic muscles superimposed. Top image in coronal plane, bottom in sagittal. . 62
3.14 16 colour-coded images of the Visible Human Project based tongue atlas in coronal plane. 64
3.15 16 Coronal MRI images of the head of a subject (Lucy), scanned with T_2 weighting. 65
3.16 Effect of contrast agent on contrast of dorsum. Left image is T_1 weighted (Matthew), right image is T_2 weighted (Lucy). . . 67
3.17 T_1 weighted scan of Lucy before and after contrast enhancement. 68
3.18 Overview of the work-flow of chapter 3. 72

4.1 High resolution slices in sagittal domain (Matthew dataset, T_1 weighted). 75
4.2 High resolution slices in coronal domain. 76
4.3 Sequential images in a sagittaly scanned volume. 78
4.4 Optical flow: normalized displacement magnitude. 79

4.5 Magnification in 1 dimension. Top row: the 3 original pixels. Middle row: after nearest neighbour interpolation. Bottom row: after bilinear interpolation. 85
4.6 Use of phase-correlation to determine relative translation. Rightmost image shows the phase-correlation spectrum with peak. . 89
4.7 A slice of an anisotropically sampled volume. Low and high spatial resolution domains. 91
4.8 Fusion of the two images based on optical flow calculations. . . 92
4.9 Overview of the work-flow of chapter 4. 96

5.1 The effect of using edge detectors: original image, Laplace, Sobel, Canny. 104
5.2 Contour tracing: the algorithm traces iteratively the contour and terminates when it visits the starting point again. 106
5.3 Termination criterion problem in other algorithms (left) and the solution provided by the Radial Sweep. 107
5.4 Freeman chain encoding of a curve. 108
5.5 A Snake in action: convergence is achieved after many iterations. 114
5.6 Hough transformation: original image, and image with superimposed lines. 120
5.7 Flow function plotted as a function of image gradient. 124
5.8 Anisotropic diffusion filtering: original image. 126
5.9 Anisotropic diffusion filtering: Gaussian and anisotropically filtered images. 127
5.10 Use of Hough transformation to detect midsagittal fissure line. Example uses Lucy's dataset. 130
5.11 Initialization step of Active Contour: different approaches. . . 136
5.12 VHP domain contour calculation: 3 cases, visualized before and after the operation. 140
5.13 MRI domain contour calculation: the same 3 cases, visualized before and after the operation. 141

5.14 Interpolation of contours. Red contour (middle) is the result of interpolating the contours on left and right. 143
5.15 A sagittal view of the dataset. This view can potentially assist the segmentation performed by the user. 144
5.16 Overview of the work-flow of chapter 5. 147

6.1 A registration example between two MRI images. 150
6.2 Z-axis alignment of the datasets. Blue set aligns with red by relative scaling and offset in the Z axis. 156
6.3 Alignment of the bounding boxes of two objects. Blue bounding box aligns with red by scaling and translation. 159
6.4 Polygon (cyan) and Spline (red), defined by 6 control points. . 161
6.5 Example of a spline with control points at unequal distances (top). Bottom version is the constant speed spline. 163
6.6 Homothetic mapping. Top: the two contours. Bottom: a graph showing the global error as a function of circular shift. . 167
6.7 Homothetic mapping. The mapping of control points between the two contours. 168
6.8 Illustration of a Spring-Mass-Damper unit. 169
6.9 The Spring-Mass-Damper network configuration: interconnections of mass units through springs and dampers. 170
6.10 Construction of SMD network. 179
6.11 Rasterization of texture mapped quads. Top: point-masses constructing a sparse quad grid. Bottom: texture mapping applied for each quad. 180
6.12 A VHP contour over a grid of potential SMD units. 181
6.13 Highlighted SMD units that will form an SMD network. . . . 182
6.14 The SMD network, with highlighted interconnections between the SMD units. The left image shows the initial state, and the right the final state after mapping. 183
6.15 Extrinsic muscles denoted with red colour. Mapping from VHP onto MRI contour. 184

6.16 Estimation of extrinsic muscles blended with the original MRI image. 185
6.17 The 6 splines derived from the VHP contour set. Left to right, top to bottom: slice no.220, 200, 186, 142, 114, 100. 188
6.18 The original spline and its distorted version. 189
6.19 The different interpolated versions visualized as a surface of revolution. 190
6.20 A plot of Homothetic mapping confidence level over filter strength, for 6 selected VHP contours. 191
6.21 Elastic deformation evaluation. Original model (top), and solutions after rotation: elastic deformation (left), transformation of image (right). 192
6.22 Overview of the work-flow of chapter 6. 196

7.1 Coronal dissection of two objects. The bottom object produces two separated components. 198
7.2 A spherified cube, with the different faces of the original cube colour-coded. 205
7.3 The resulting mesh after the iteration of a coarse tessellated 3D snake on the VHP set of contours. 207
7.4 Displacement of control point to the average of its neighbouring points due to elasticity forces. 209
7.5 Cross-section of the VHP atlas in voxel-space. The extrinsic muscle voxels are colour-coded with red. 214
7.6 Two different views of the VHP atlas rendered in voxel-space. 218
7.7 Two different views after registration of the VHP atlas into the MRI scan of Lucy, rendered in voxel-space. 219
7.8 Overview of the work-flow of chapter 7. 220
7.9 The work-flow of the system described in this thesis. The red numbers indicate the number of the relevant chapter. Question-marks denote non-essential units. 221

8.1 Photographs of the ox tongue slices before and after deformation. 233
8.2 Results for case 1: image before and after deformation - middle left shows estimated deformation. Bottom image shows error map between estimation and reality. 234
8.3 Results for case 2: image before and after deformation - middle left shows estimated deformation. Bottom image shows error map between estimation and reality. 235
8.4 Case 1: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom. 236
8.5 Case 1: volume rendering of cancer (blue) and extrinsic muscles (red). 237
8.6 Case 2: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom. 238
8.7 Case 2: volume rendering of cancer (blue) and extrinsic muscles (red). 239
8.8 Case 3: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom. 240
8.9 Case 3: volume rendering of cancer (blue) and extrinsic muscles (red). 241

9.1 3D multislice diffusion tensor MRI of the bovine tongue. Coronal slices comprising the bovine tongue obtained by diffusion tensor imaging, with muscle components identified and rendered with octahedra [VJWG01]. 251

List of Tables

2.1 Influence of different sequence characteristics on Signal to Noise ratio, spatial resolution and acquisition time. 28

3.1 Scanning sequence characteristics for scan of Matthew. 68
3.2 Scanning sequence characteristics for scan of Lucy. 69

8.1 Scanning sequence characteristics for experimental procedure in 8.2.3. 225
8.2 Scanning sequence characteristics for case number 1. 228
8.3 Scanning sequence characteristics for case number 2. 229
8.4 Scanning sequence characteristics for case number 3. 230

Acknowledgments

There are many people I would like to thank for a variety of reasons.

Firstly I would like to thank the one person who guided me from the beginning of the course to the end: my supervisor Dr. Stephen Golding. He is the best mentor I could possibly have and without his sense and knowledge I would never have made it. This one is for you.

Similarly I owe a big thank you to Mr Steve Watt-Smith for his ideas and support, and to our academic radiographer Chris Alvey whose input was invaluable. It has been a pleasure working with you and I hope we will collaborate at some point in the nearest future.

Then there are two people without whom I would have not even started this program: my friend Dr. Tom Bowles who suggested that I could undertake a PhD successfully, and Lisa who believed in me all these years. They both provided the encouragement I needed throughout the three years. I love you both, albeit in a different way (sorry Tom!).

There are also the people who helped me morally and financially. From Empire Interactive and Sick Puppies: Gregg Barnett, Dave Hunt and Dave Pringles. From Razorworks: Kevin Bezant and Dave Proctor. Thank you for all your help and I forgive you for misspelling my name in 3 different products we made together. I also appreciate all the assistance from my

volunteers Matthew Watt-Smith and Lucy Butler, and from my friends who read the first drafts of this thesis: Kostas Sambanis and Nikos Mpatalas.

Finally many thanks to my generous family back in Greece, especially to my mother, my sister and my grandparents. And Malu. What a trip this has been! 11 years and counting.
I miss you all very much...

Chapter 1

Introduction

1.1 Motivation

Since its inception, medical imaging has been an important aid to clinicians in the diagnosis of pathology and therapy planning. Images acquired in different modalities, such as X-rays, Computed Tomography (CT), Magnetic Resonance imaging (MRI) and Ultrasound, are routinely used to visualize anatomical features and functions of the human body for clinical and research purposes. More specifically, in surgery, these images are used to assist the surgeon in planning and performing the operation, as they provide the means to understand the spatial correlation of the underlying organs without the need for an invasive, pre-operative step.

Recent technological developments such as cross-sectional imaging and image processing have vastly improved the potential of medical imaging; the latter can be thought of as the set of tools that is used to improve the diagnostic potential of investigation. Image processing is routinely applied to data coming out of scanners in order to clarify, extract and classify the objects depicted, through metrics of shape, texture and intensity. This is a difficult task for a person, mainly due to the complexity of the images and the time required. It is also a difficult task for a machine, because of variability of the subjects.

Because of that there is no golden rule or perfect solution in medical image processing, and each problem may be solved through a different path.

The problem that this thesis investigates combines the two elements together, the need for medical images in pathology and surgical planning and the use of image processing as a bridge between the images and the clinician.

1.2 Scope and significance of thesis

This thesis focuses on the potential of processing MRI images to assist the staging of cancer of the tongue, a malignancy that every year affects more than 4000 people in the UK alone [ONS02]. In general, the cancer staging mechanism provides the clinicians with a standard description of the size of the tumour, local extent and metastatic spread.
More specifically, this thesis looks into providing the necessary methods and tools to assist the clinician in determining the surgical stage of tongue cancer, which is partly based on tumour invasion into intrinsic and extrinsic muscles of the tongue. In order to do so, a model atlas of the tongue muscularity was first built and then co-registered with the subject's tongue, as viewed in coronal MRI images. The distinction between extrinsic and intrinsic muscles on the images of the tongue of the subject is then made possible. As a result, a metric of penetration of cancer into the extrinsic muscles of the tongue can be established and used in quantifying the stage of cancer. Additionally, the information can be visually presented in a more tangible manner, and used in treatment and operative planning, aiding precise tumour excision.

This is the main objective of the thesis, which is split into the following specific tasks:

1. To develop a technique for using colour photography images from the Visible Human Project in order to construct an atlas model of the human tongue, with the tongue muscles identified as either extrinsic or

not extrinsic (e.g. excluding fat).

2. To extract the contours of the tongue in any coronal MRI scan of the head in an automatic or semi-automatic fashion, using an Active Contour model as a segmentation method.

3. To co-register the two, that is to map the atlas model and the MRI scan into spatial correspondence in order to determine which muscles of the scan images should be classified as extrinsic or not extrinsic.

4. To visualize results in an intuitive fashion.

It can be seen that the scope of the thesis is not to determine whether the tongue cancer exists in the first place, or to build a model of the tongue for generic experimental use in medicine. The objectives are very specifically designed to address practical inability to determine the tongue muscularity in any set of normal MRI images. The evaluation of clinical results, and their potential impact on the understanding of the finer details of tongue cancer staging is beyond the scope of this work, but provides targets for further research.

1.2.1 Implementation details

The implementation of the above tasks is based on theory derived from the existing literature, personal experience in image processing and computer graphics, and experimentation. The tools that I used were:

- Software: Microsoft Visual Studio C++ .NET 2003 for all code. Intel OpenCV library for auxiliary image processing functions. OpenGL and Nvidia Cg for 3D graphics. Discreet 3D Studio MAX 4.0 and Wings3D 0.98 for some 3D Visualization and Mesh manipulation. Dicom Works and eFilm for handling and anonymizing of DICOM images.

- Hardware: 4 channel 1.5 Tesla MRI scanner (General Electric Medical Systems) in clinical use. My development and test machine was an

AMD Athlon XP 3000+ with NVidia 5600 Ultra running Microsoft Windows 2000 Pro. All photography was performed using a Canon S50 5 MPixels digital camera.

1.3 Organization of thesis

This thesis is organized in chapters as follows:

- **Chapter 2** is a background review of the anatomy of the tongue and the imaging modalities used in clinical and experimental studies with an emphasis given to MRI. The chapter continues with a description of cancer, and more specifically cancer of the tongue, the TNM cancer staging system and its correlation with the scope of my project.

- **Chapter 3** starts by describing the tongue modelling approaches in the literature. It then presents the theory of the methods I used to retrieve the 2 volumes of data that this experiment is based upon: the Visible Human Project anatomical atlas and the extracted MRI volume of the subject under investigation.

- **Chapter 4** is an extension of Chapter 3, in which I describe a method I developed to make the extracted MRI volume isotropic.

- **Chapter 5** introduces the concept of segmentation in image processing. It presents the development of a system for automated extraction of the tongue structure from both volumes as a series of contours, a format that is appropriate for further use in Chapter 6.

- **Chapter 6** is the core chapter, which describes the methodology for registering the atlas contours over the MRI derived tongue contours. In this chapter I propose and describe the construction of a 2D method to register slices to slices.

- **Chapter 7** is an extension of the experimental approach to full 3D.

- **Chapter 8** presents the experiments performed to validate the work and the results together with an evaluation of findings and their significance.

- **Chapter 9** is a summary of the work. I examine the applications of the model developed, aspects of this work and possibilities for further research.

When possible I have tried to follow a similar format in each chapter. This normally consists of a general background, in which the existing literature is reviewed, followed by the algorithms and experimental methods developed for the set objective. At the end of each chapter there is a discussion section, where several issues regarding the results and theory in the chapter are addressed.

Additionally, each chapter is complemented by a general work-flow diagram on the final page. As such, each chapter can be viewed as a part of the whole thesis, or an entity of its own.

The chapters are allocated in a work-flow sequence, so that, for example, the "output" data of chapter 4 are "input" in chapter 5. There is also a general overview of the different components of the system described in this thesis (figure 7.9) on the last page of chapter 7.

Chapter 2

Background

2.1 Introduction

The scope of this chapter is to present the general medical background of this thesis and by doing so establish its clinical importance. The description follows a sequence: it starts with the anatomy of tongue, then moves into imaging modalities and expands into the complex phenomenon of MRI. Cancer of the tongue and the concept of cancer staging are finally described as they are central to the clinical objective of this thesis.

2.1.1 Organization of sections

- Section 2 describes the anatomy of the tongue, starting with the extrinsic attachment of the tongue muscles and moving into extrinsic and intrinsic muscles.
- Section 3 is an overview of current imaging modalities employed in imaging. X-ray radiography and cineradiography , ultrasound, CT and MRI are reviewed with emphasis on tongue visualization.
- Section 4 describes the MRI phenomenon and applications in tongue imaging in more detail. In this section I assert the advantages and

disadvantages of using MR imaging for the purpose of the experiment as a whole.

- Sections 5 and 6 describe cancer in general and of the tongue specifically.
- Section 7 overviews the TNM cancer staging mechanism that is referred in this text as a clinical standard in categorizing cancer of the tongue, and how this can be extended to provide information related to surgical management of cancer.
- The chapter concludes with section 8, where the clinical importance of the experiment is established.

2.2 Tongue

The tongue is a fleshy, muscular organ, an entangled network of muscles attached to the floor of the mouth. The purpose of the tongue is to provide the sense of taste, to aid the chewing and swallowing of food and, in humans, to assist forming sounds of speech. In 1985 a theory describing the tongue as a muscular hydrostat was set forth by Kier et al. [KS85]. A muscular hydrostat organ is a biological structure of cylindrical shape that is mainly made up of muscle and retains a constant volume throughout movement. As such, the tongue exhibits bio-mechanical properties akin to hydraulic devices, in contrast to the skeletal muscle system that is based on the principle of mechanical levers. The term was coined originally to characterize the tentacles and arms of squids and octopuses. Other common muscular hydrostats are the bodies of worms and trunks of elephants.

2.2.1 Basic anatomy - physical appearance of the tongue

Several studies have attempted to describe the muscular anatomy of the tongue. An early, and often quoted study by Malek [Mal39], provides a

detailed definition of the muscles based on dissections and experimental work. A more thorough description of the dissected muscles was provided by Miyawaki [K.74], Doran [Dor75] and more recently by Takemoto [Tak01]. Four of them originate outside the tongue body ("extrinsic") and four originate and form within the tongue body ("intrinsic").
The following 3 subsections review the human tongue in more detail, starting with the attachments of the extrinsic muscles and then the morphology of extrinsic and intrinsic muscles.

2.2.2 Extrinsic attachments of the tongue muscles

The extrinsic muscles of the tongue are attached to the skeleton at the Mandible, the Styloid process of the Temporal bone and the Hyoid bone. In detail:

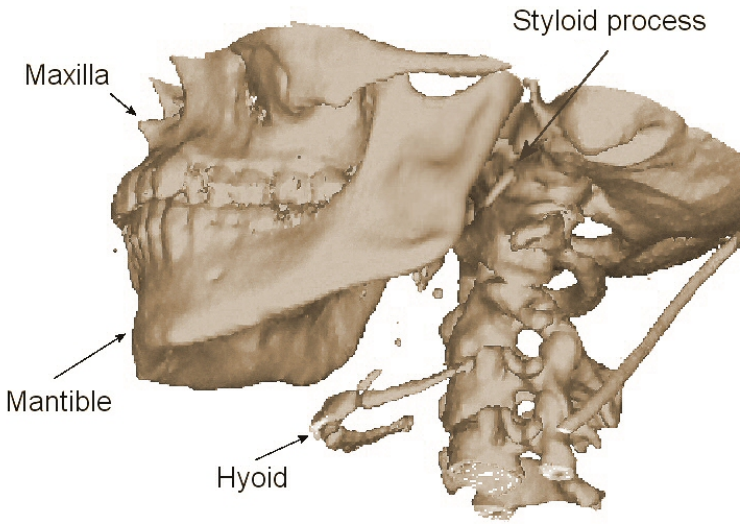


Figure 2.1: 3D Rendering of the skeletal framework for the tongue (from CT scan).

- The Mandible is the largest bone of the face. It forms the lower jaw and holds the lower teeth. The body of the bone is U-shaped, and

has two surfaces and two borders. The internal surface has a pair of laterally placed spines, the mental spines, to which the Genioglossus muscle is attached. The Mandible articulates with the temporal bones, which are situated at the sides and base of the skull, and the Maxilla, the large paired facial bones that contain the upper dentition and form the upper jaw.

- The Styloid process is a long slender bony process of the temporal bone of the skull, attached to the base of the skull just below the ear, that projects downward and forward and provides attachment to the Styloglossus muscle.

- The Hyoid bone is a horseshoe-shaped bone at the base of the tongue. The Hyoid bone is unique in the face in that it is not articulated to any other bone. It is suspended above the larynx by the muscles of the neck.

The skeletal framework for the tongue is shown in figure 2.1.

2.2.3 Extrinsic muscles

- **Genioglossus muscle**

  is fan-shaped and forms the bulk of the central and inferior portion of the tongue. It arises as a short, narrow tendon at the mental spine of the mandible. Its insertions are the dorsum and body of the tongue. The function of the muscle is to depress and protrude the tongue.

- **Hyoglossus muscle**

  has quadrilateral shape and originates from the whole length of the greater cornu of the hyoid bone. It continues vertically upwards and enters the side of the tongue between the Inferior Longitudinal and the Styloglossus muscles. The function is to depress the tongue and draw the root anteriorly.

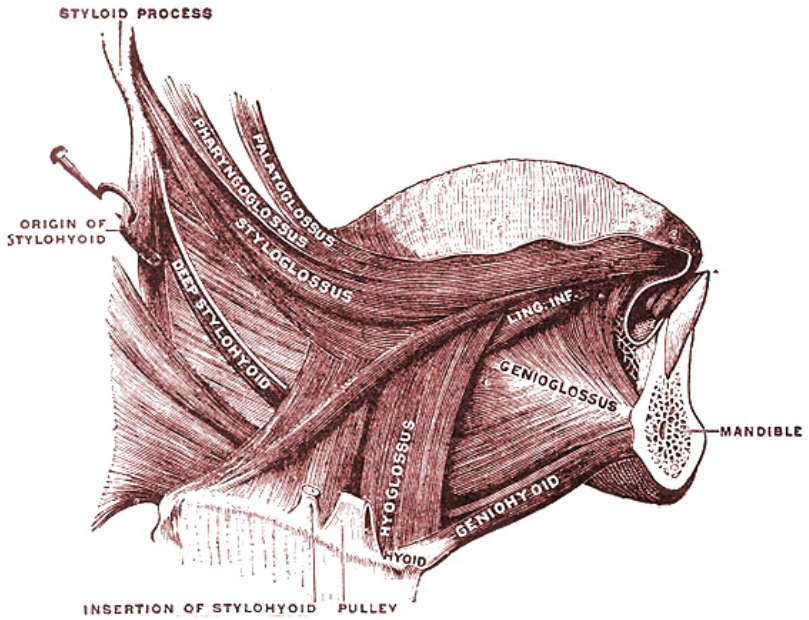


Figure 2.2: Extrinsic muscles of the tongue in para-sagittal section. From (http:// face-and-emotion.com/ dataface/ anatomy/ tonguemuscles.jsp)

- **Styloglossus muscle**

  is the smallest of the three styloid muscles. It originates from the anterior surface of the styloid process of the mastoid bone. It passes downwards and forwards, splits on the side of the tongue near its dorsal surface and its fibres blend with the Inferior Longitudinal muscle in front of the Hyoglossus muscle. Additionally, the upper limit of the Styloglossus serves as part of the attachment for the Palatoglossus muscle. The muscle is mainly responsible for retracting the tongue and aiding swallowing by creating a hollow in the centre of the tongue.

- **Palatoglossus**

  is one of the muscles of the soft palate. It arises from the anterior surface of the velum, and passes downward and forward to mix with the fibres of the dorsum of the tongue. The Palatoglossus acts to raise the dorsum and draw the soft palate inferiorly. The two actions together

are important for the initiation of swallowing.

Figure 2.2 shows the extrinsic muscles of the tongue on a para-sagittal section.

2.2.4 Intrinsic muscles

The intrinsic muscles run along the length of the tongue, and have no attachments outside its body. They are divided into:

- **Inferior Longitudinal muscle**

  is situated between the Genioglossus muscle and Hyoglossus muscle, under the surface of the tongue. It originates from the fibrous tissue beneath the mucous membrane at the tip of the tongue, and lines the sides longitudinally back to the hyoid bone. This muscle assists in curling the tip of the tongue inferiorly and making the surface convex in shape.

- **Superior Longitudinal muscle**

  runs along the superior surface and is responsible for assisting the movement and retraction of the tip of the tongue. The muscle originates near the hyoid bone and the epiglottis but is attached to neither.

- **Verticalis muscle**

  together with Transversus muscle this forms the bulk of the tongue. The muscle is located in the middle of the tongue and joins the Superior and Inferior Longitudinal muscles.

- **Transversus muscle**

  is attached to the mucous membranes that line the sides of the tongue, and to the septa inside the tongue. The Transversus divides the tongue in the middle, and lies superior to the Inferior Longitudinal and inferior to the Superior Longitudinal muscle. Its function is to contract the

tongue laterally by decreasing the lateral diameter. The fibres of the Transversus muscle interdigitate with those of Verticalis.

An overview of the tongue muscularity is represented in figure 2.3.

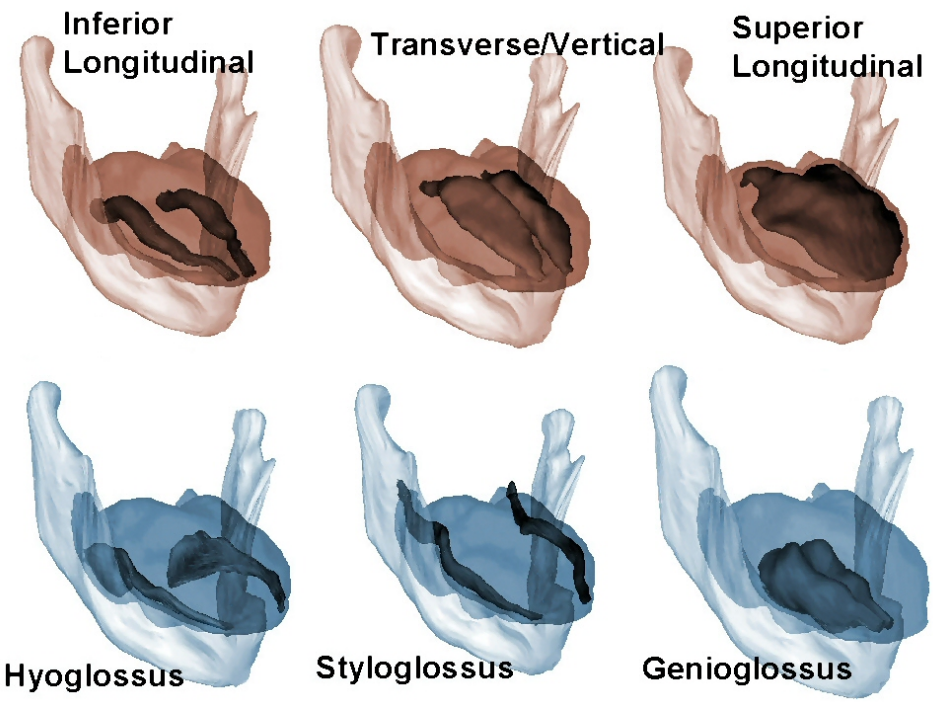


Figure 2.3: 3D Rendering of the Intrinsic and Extrinsic muscles of the tongue, taken from the Upper airway project website [SIA02].

2.3 Biomedical Imaging Modalities: an overview

This section presents the basic non-invasive methods used to visualize the internal anatomical structures of a patient, with an emphasis on tongue and mouth. The section describes the imaging technologies of X-rays, Ultrasound imaging, Computed Tomography (CT) and finally MRI, which is relevant to my work and expanded in more detail in the following section.

2.3.1 X-ray Radiography and Cineradiography

In conventional radiography the objective is to create images by exposing a photographic film to X-rays, usually with the aid of a fluorescent plate. X-rays have the ability to penetrate solid objects and as they do so their energy is attenuated depending on the object's composition. The result is an image that reveals the structure of the object that stood between the X-ray source and the film. X-ray imaging was the first non-invasive method of revealing the structure of the body in history, and its invention by Wilhelm Rontgen was responsible for the development of the science of radiology.
X-rays are a very commonly used medical test and mainly used for bone

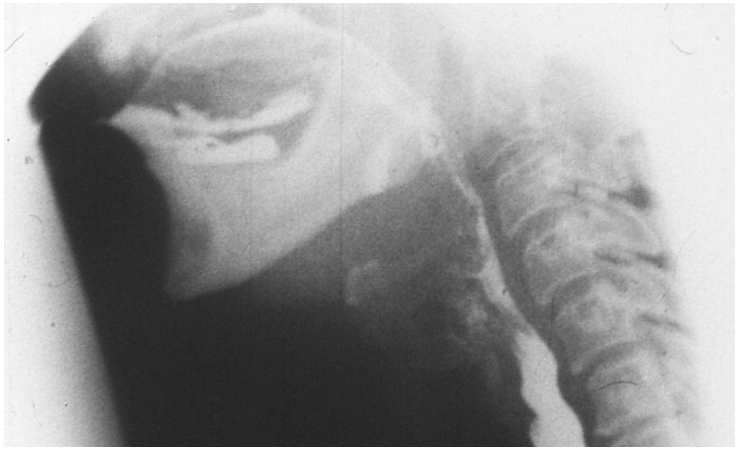


Figure 2.4: Sagittal X-ray photograph of the tongue.

imaging (to detect fractures, tumors or degenerative conditions of the bone), in dental examination, mammography and lung imaging. X-rays offer superior resolution and contrast for bones, but little soft tissue discrimination. Figure 2.4 shows a sagittal X-ray photograph of the tongue with its surface only faintly visible. This picture holds little information about the internal structure of the soft tissue.
In X-ray cineradiography, flash X-ray sources are triggered simultaneously with a gated camera, so that displacement is captured in a sequence of discrete steps. Since the process is relatively inexpensive and it is possible to

record in high frame-rate, with sufficient spatial resolution, cineradiography has been the most practical and widely used method for investigating tongue movement in speech and swallowing [AG55]. In the past, X-rays and optical recordings were the only sources of quantitative information on the physiology of articulation [J.69].
The accuracy of measurements has increased over time, as technology evolved and new methods were introduced. [Hou67] suggested the use of metallic particles attached to points on the surface of the tongue, or even painting the tongue with a contrast compound containing Barium, which generates a frame of reference for subsequent (manual or automated) image processing. But while cineradiography is, in theory, adequate for time-based modelling of the tongue, it cannot give detailed insight into the anatomical features of the soft tissue or the oral floor, because of its inherit inability to separate muscle groups. Also, the problem of radiation dose associated with ionizing radiation exposure prohibits the use of in-vivo X-rays with subjects in a research environment for reasons of patient protection.

2.3.2 Ultrasound Imaging

Medical ultrasonography (sonography) is a diagnostic imaging method for visualizing internal organs, including muscles, and for distinguishing pathology. In sonography an ultrasound source produces a sequence of short soundwave pulses that travel through the tissue. Based on the acoustic impedance of the tissue, and planes between different tissues, the sound waves are reflected back to a transducer, which turns them into electrical pulses. The interpretation of the echo signals into an image is based on identifying the transducer elements that received the signals, their strength, and how long it took for the echo to be transmitted back to the transducer [JT02]. Sonography can create 2D and 3D images ([Wat99]) by acquiring a sequence of 2D images, as well as velocity maps, in which the relative speed and direction of a volume of moving particles to the plane of the probe is determined through Doppler effect measurements (Doppler Sonography).

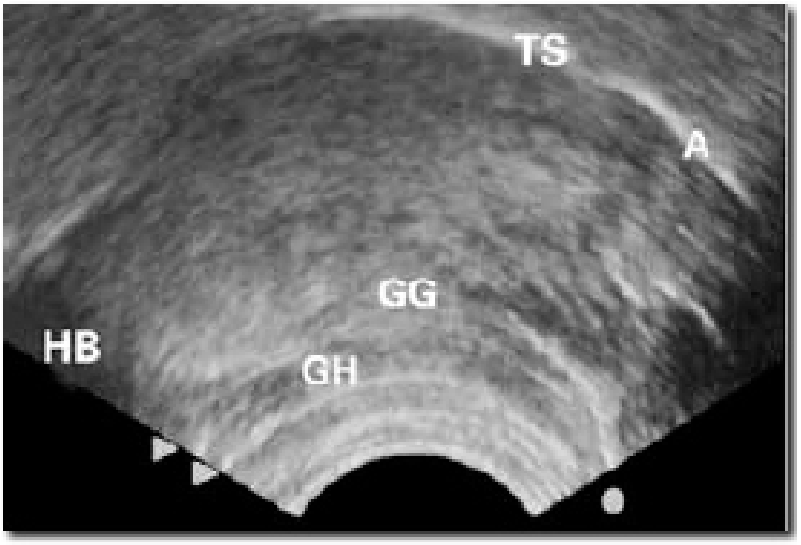


Figure 2.5: Ultrasound imaging of the tongue in sagittal plane [Wat99].

The strengths of sonography lie in visualizing muscle and soft tissue adequately, with sufficient delineation between structures with different water and collagen content. Ultrasound renders images in real-time, has no known long-term side effects and, compared to CT and MRI, is inexpensive. However ultrasound imaging is limited by poor performance in the presence of air between the scanner and the organ of interest, which is due to differences in acoustical impedance. Ultrasound also has difficulties penetrating bone, because of reflection and acoustical impedance difference. Hence the use of sonography in the mouth is limited by access and presence of air pockets above and beneath the tongue. Ultrasound imaging of the mouth is usually achieved through the submental tissues ([Sto91]) and may be degraded by artefacts causing poor image quality for the palate and the jaw and hyoid bone. Figure 2.5 shows a typical ultrasound image of the mouth taken from a time-series based phonetics experiment.
Sonography can be employed to visualize large tumours in the tongue. However, due to the limitations of the physiology and the technology of ultra-

sound, it is difficult to delineate and display the individual components that constitute the tongue [LM96]. As the images acquired are noisy, do not allow the use of scout-image guide (hence it is difficult to determine a much needed frame of spatial reference) and do not provide a clear display of the zonal anatomy of the tongue, sonography was not considered an adequate resource for the scope of this experiment.

2.3.3 Computed Tomography

In CT, a large series of 2D X-ray measurements is taken around a single axis of rotation using sensors positioned opposite the X-ray source. The subject is positioned inside the cylindrical shaped scanner, and moves axially through the gantry, while scans are taken sequentially. They are combined by a mathematical procedure called tomographic reconstruction in order to produce multi-planar images, allowing reconstruction of the 3-Dimensional volume of the internal structures of the subject. For the principles of tomographic reconstruction consult the work of Kak et al. [KS88].

As with X-ray radiography, the operation is based on tissue attenuation of the beams as they penetrate tissue and bone. Therefore, CT images show the radiographic density of matter within the subject. The main characteristics of CT are the inherent high-contrast and high-spatial resolution, which allows excellent delineation between tissues that differ in physical density, such as soft tissue and bone. As such CT is an important tool in medical imaging, and although the hardware is still quite expensive it is used extensively in the diagnosis of a large number of diseases, including stroke, bone fractures, diseases of the airspace and for delineation of tumors. More specifically for the mouth area, cranial CT scanning has a role in detecting bone involvement by disease, and is mostly used in surgical planning for craniofacial and dentofacial deformities, dental implant reconstruction planning and cancer staging in the jaw, sinuses and nasal area.

Before the advent of MRI CT was used as a technique for staging oral cancer and cervical lymph node metastasis [KYS00], as it provides detailed, high

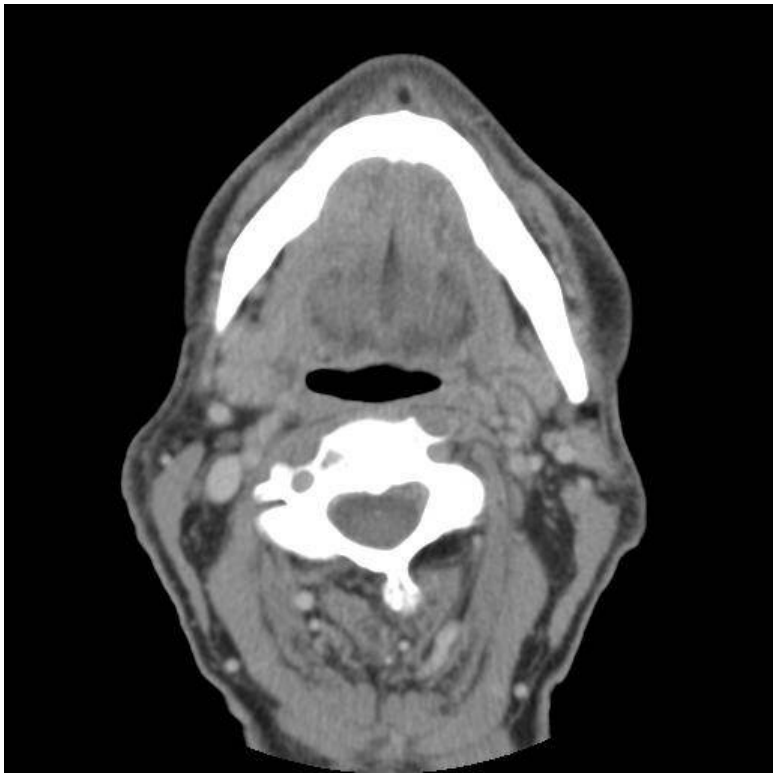


Figure 2.6: Axial image of the tongue acquired with CT.

resolution cross sections of the face and neck. Also, due to the very short exposure time, tongue displacement artefacts that could degrade the quality of images are largely prevented. Nevertheless, CT imaging is prone to a type of artefact that is not present in MRI: streak artefact is a result of a phenomenon called "beam hardening". The phenomenon is observed when high X-ray absorption structures are in the field of view and attenuate the X-ray polychromatic spectrum in a non-linear fashion: as X-rays transverse an object, the higher frequency portion of the spectrum increases as the lower frequency attenuates. This effect depends on the composition of the material and, if it is not compensated for, results in cupping artefacts and star-like patterns around or spreading out of a dense object. Streaking artefacts are introduced into CT images by high density materials such as metals. Therefore, the presence of metallic objects in the oral cavity, such as dental fillings,

non-removable metallic joint replacements and surgical clips, usually introduces severe artefacts that undermine the quality of the image. The artefacts are can be treated, to an extent, with beam-hardening correction algorithms [JS78] and equipment-related technologies (i.e. heavy beam filtration prior to entering the body). In addition to beam hardening, there is another, less prominent image artefact, unique to CT, that degrades the image quality and potentially hides areas of pathology, called ring artefact. Ring artefacts are due to misalignment of the detectors in the detector array, and can be avoided through hardware recalibration. For further information on artefacts consult Ameen [Ame04], and for an extensive description of other minor errors with CT imaging refer to Barrett et al. [JB03].

CT is not an adequate solution for visualizing the internal structure of the tongue. While the tongue margins can be well delineated, the different muscle groups can only be visible when there is an adequate amount of fat between them, which is not the case with the majority of subjects (especially children and thin adults), where the tongue appears as a homogeneous soft tissue body [BR94]. Compared to MRI, CT has poor tissue contrast even despite the use of contrast agents. These can be source of media-related complications, such as allergic reactions and renal toxicity. Additionally, the problem of ionizing radiation exposure is present, particularly in the radiosensitive organs in the face, such as the optic lens and the thyroid gland, something which limits the use of CT in a experimental environment.

A typical CT image of the tongue is represented in figure 2.6: while the mandible and vertebrae have high intensity levels and are well delineated to the surrounding soft tissue, the tongue is viewed as an almost homogeneous body.

2.3.4 Magnetic Resonance Imaging: overview

Magnetic Resonance Imaging (MRI) is another modality used to visualize the structure of living organisms and represents a significant innovation in the technology of medical imaging. MRI has the ability to image in any plane,

unlike CT which acquires axial data and reformats (i.e. interpolates) into other plane views. Additionally MRI provides superior soft tissue contrast resolution and freedom from ionizing radiation, hence it is safe to be used on humans in a clinical research context or, at least, there is no evidence yet of any harmful consequences. MRI has become the technique of choice for most facial examinations. However since the imaging sequences are longer than CT it is more prone to motion artefacts. Also, because MRI employs strong magnetic fields it is not applicable to patients with pacemakers, internal hearing devices or other ferromagnetic bodies. A further disadvantage of MRI in the face is that it is also susceptible to artefacts generated by metallic dental fillings and crowns, although these are not as severe as those in CT. Since MRI is being used extensively throughout the project, a full description of the complex operation of MRI is provided below.

2.4 MRI: technical basis and applications

2.4.1 The MR Phenomenon - spin and excitation

MRI utilizes the relaxation properties of excited hydrogen nuclei protons in tissue water [ea03a]. A superconducting magnet produces a powerful uniform field B_0 which slowly aligns the spins (angular momentum) of atomic nuclei of the object in a direction either parallel or anti-parallel to the magnetic field. Without loss of generality we assume this to be the Z direction. The number of nuclei aligned is small (of the order of one in a million) yet a change in magnetic field is detectable because of the large number of atoms in the volume under consideration. A sequence of electromagnetic pulses, in a plane perpendicular to the magnetic field and having a frequency corresponding to the Larmor[1] frequency, is then generated by a radio transmitter and radiated

[1]The Larmor equation is the resonance condition, as it dictates the relationship between the resonant frequency of the precessed protons in the nuclei and the strength of their magnetic field. It is given by $f_{Larmour} = \gamma \cdot B_0$, with B_0 expressed in Tesla and γ in Mhz per Tesla.

into the object. The energy transfered by the wave causes the direction of spin and therefore the longitudinal magnetization to deflect from the aligned state, a process which is referred to as "excitation" of the spin system. With the correct power and duration, deflection of exactly 90 ° can be achieved. As the magnetic field tries to revert the inclination of the spins into the Z direction, the protons eventually precess around the Z axis, and turn within the perpendicular, transverse XY plane. This motion induces an alternating voltage with a frequency equal to the Larmor frequency. It is then the task of sensitive EM receivers and computers to collect this information and process it into an image.

Relaxation - the biological parameters T_1 and T_2 in resonance

As the nuclei realign, they emit energy which is received and analyzed so as to provide information about their environment. Following excitation, all spins and their magnetization rotate in the XY plane, a process called "transversal magnetization". The magnitude of the transverse magnetization reduces and returns to the state before excitation through two processes: spin-lattice and spin-spin interaction, dictated by two independent biological relaxation parameters T_1 and T_2.

T_1 Effect

"Time 1" or T_1 "recovery" is the time required for 63 % of the tissue protons to return to the initial low energy state. In T_1, the high energy protons exchange energy with the neighbouring nuclei and dissipate energy in the form of heat. As the potential to exchange energy is dependent upon intrinsic physical properties of the matter the value T_1 can be used to distinguish the different types of tissue present.

T_2 Effect

T_2 follows the transverse relaxation time: after excitation, all spins precess in phase [oDD]. However, due to magnetic field inhomogeneity, which is caused

by the external stationary field B_0 and the self-generated fields of the neighbouring protons, the protons start running out of phase with respect to each other. The transverse magnetization, the net result of the magnetic moment in the XY plane, decreases until it finally disappears. The time taken for the transverse magnetization value to go from its maximum value to zero is T_2. T_2 time is a descriptor of the mobility of the protons of the tissue, as larger mobility values result in smaller average magnetic field variations, giving rise to long T_2 times. T_2 is independent of the strength of the magnetic field B_0.

T_1 and T_2 stages, while they proceed concurrently, are independent phenomena. However, T_2 relaxation decays the signal within the first 300 ms - long before the return, due to T_1, to the state of longitudinal magnetization (0.5-5.0 sec).

2.4.2 Spatial encoding

Slice selection

The next step in the sequence of events that leads to the creation of the image is spatial encoding, in which the excitation pulses are directed towards the slice under investigation rather than the entire body. This is achieved by generating gradient magnetic fields in the 3 directions X,Y and Z by means of adding a pair of extra coils along each direction. The gradients have different names: Z gradient for slice selection, X and Y for frequency and phase encoding respectively. The coils alter the magnetic field (B_0 for slice selection) and make it non-uniform. The precession of the Larmor frequency is proportional to the strength of the magnetic field, which is now set to be slightly above and below B_0 at the two extremes of Z. Since the gradient generated is continuous there is exactly one slice corresponding to a selected frequency, the thickness of which can be selected by changing the gradient's strength.

Frequency and phase selection

Once the slice is selected, the MR signal is further decoded through the assistance of the additional X and Y coils that generate a gradient in their direction. The phase encoding gradient results in a phase shift between the already excited spins, as those at one extreme will spin faster than the ones at the other extreme. The phase shift depends on the strength and duration of the gradient as well as the in-plane position of each spin. The phase gain or loss is retained even if the gradient is switched off after some time, as the spins precess with equal speed. Thus, each line on a given slice on Z can be spatially localized by its phase, as the spins are sorted according to their phase along Y. The third spatial variable on X is encoded through the frequency encoding. The gradient along X makes the spins on one end precess faster than on the other. The MR signal that this line is transmitting back is a frequency spectrum rather than a single frequency and thus each column on X can be identified by its frequency. Combining frequency and phase distribution (X and Y for a given slice on Z) generates a number of unknown quantities for which as many different equations are needed. Therefore, the measurement sequence must be repeated as many times, by altering the phase encoding and calculating the phase distribution. The sampled information is saved into a special map called "k-space", for further processing.

K-space and image reconstruction

K-space is a "temporary image space" in which the digitized MR signals are stored during the sequence [ea03b]. The K-space image holds information about the phase and spatial frequencies that constitute the image or volume of the scanned object. Phase is encoded in the vertical direction of K-space, frequency in the horizontal, hence every horizontal line is a measurement for a fixed phase gradient. The k-space is reconstructed into the real space image through 2D Fourier transform, and while k-space and the real-space images have the same number of rows and columns, there is no correlation

between pixels that occupy the same position in both images. Every pixel in the real-space image is a function of all of k-space pixels, with the middle of k-space containing more contrast information, while the outer regions (i.e. the pixels further from the center) more spatial information.

2.4.3 Image characteristics

The background theory, up to this point, has dealt with the production of the MR signal, spatial encoding and image reconstruction. This section looks into the factors affecting the characteristics of the end image and how it is possible to influence them. First it is important to clarify certain concepts: The reconstructed image is digital and consists of a matrix, a two-dimensional array of pixels (picture elements) that determine the scan resolution. Each pixel is associated with a signal intensity. However, a pixel contains information about an element of volume (voxel, a 3-Dimensional pixel), as MRI is a cross-sectional imaging modality. Figure 2.7 shows the two concepts of pixels and voxels.

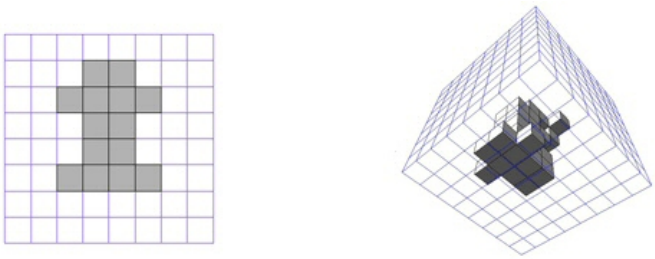


Figure 2.7: Representation of pixels on image and voxels in volume.

The physical dimensions of voxels are determined by the matrix size, the field of view and slice thickness:

- The matrix size is the number of pixels in each direction that represent the end image.
- The field of view (FOV) is the size of the region of interest, the area that encapsulates the imaged structure. It is measured in cm.

- The slice thickness is also measured in cm and shows the size of the voxels in the Z direction.

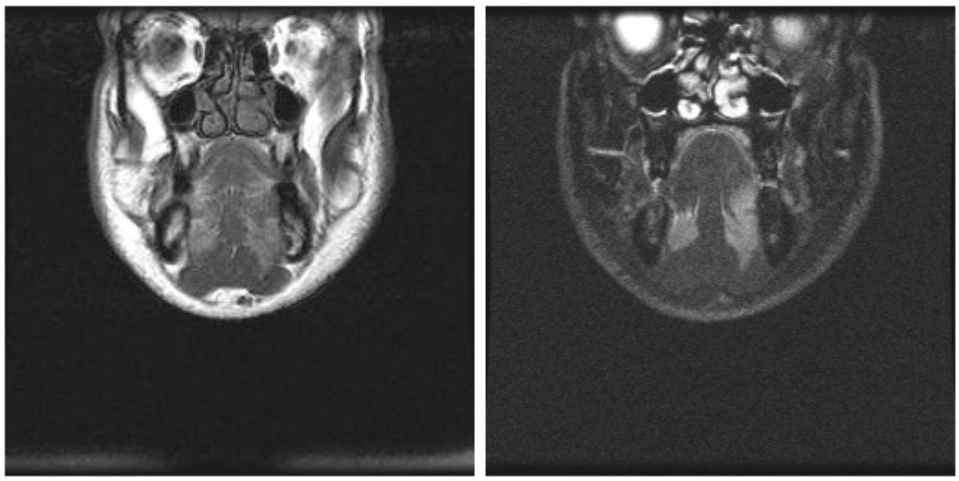


Figure 2.8: T_1 and T_2 weighted MRI sagittal images of the same subject.

Weighting factors that affect image brightness

The brightness levels of the pixels of the end image is affected by three parameters. Depending on the tissue type that needs to be highlighted, different acquisition time parameters are used, and the images are said to be weighted by T_1, T_2 or Proton Density. The image is always affected by the combination of these parameters; however, depending on the weighting, the brightness levels are affected relatively more by the contribution of the T_1, T_2 or Proton Density phenomenon.

In a clinical environment, the selection of image weighting parameters is determined by the type of tissue and pathology under examination. For example, under T_1-weighting examination of the brain, the nerve connections are highlighted while the cerebrospinal fluid appears dark. In T_2 imaging, the reverse is true for the contrast of white matter and cerebrospinal fluid. The following list summarizes the use of the three different weighting factors:

- Proton Density weighted images (also simply called "Proton Density imaging") are determined by the concentration of excitable spins per unit volume and an indication of the signal strength emitted by the sample tissue. The higher the proton density, the brighter the signal, and conversely, a darker signal is an indication of lower number of protons in a given unit of tissue. Solids, for instance in cortical bone, or gaseous regions (eg. sinuses) produce no detectable signal. Proton density imaging is not used routinely, as more information on the tissue and pathology is possible by the combination of T_1 and T_2 weighting alone.

- T_1 imaging is used to highlight fat, and the contrast of fat with other tissues. It is therefore valuable for demonstrating tissue morphology. T_1 imaging is also used to highlight blood products such as methoglobium, and is used in conjunction with contrast media (enhanced imaging). A parameter that influences T_1 contrast is "Repetition Time" (TR), which determines the time available between successive pulse sequences for the spins to recover from previous excitations. By varying TR, the image contrast characteristics can be controlled - a long TR (of the order of 1500 ms) will allow enough time for the tissue to relax and emit strong signals, with the drawback of increased total scan time (i.e. the total time required for the completion of all procedures related to one scan).

- T_2 is more sensitive to the existence of water within the cells, and so T_2 images can give insight into contrast between individual tissues. T_2 is usually valuable for giving insight into pathology of tissue. The parameter that affects the influence of T_2 on the image contrast is "Echo Time" (TE), which is the period between excitation and measurement of the MR signal, measured in ms. A long TE results in strong T_2 weighting, while a short TE results in weak T_2 weighting.

Figure 2.8 shows two sagittal MR images of the same subject acquired with different weighting - T_1 for the left and T_2 for the right - highlighting fat and

water content respectively.

Factors affecting Signal to Noise ratio

As previously mentioned, the reconstructed image is digital and consists of a matrix, a two-dimensional array of pixels that determine the scan resolution. As the resolution increases, so does detail - the measure of the ability to discriminate between neighbouring anatomical features, to the expense of increased scan time [ea03a]. It was also mentioned that MRI is used to scan a volume, and hence pixels carry information about a volume element (voxel), the physical dimensions of which are determined by the matrix size, the field of view and slice thickness. Ideally slice thickness should be as thin as possible in order to increase the detail. However, the drawback of decreasing slice size is reduction in Signal to Noise ratio (SNR), with the introduction of cross talk between neighbouring slices. This can be counterbalanced by increasing the number of measurements per slice. Thicker slices give rise to partial-volume effect, which is the loss of contrast between adjacent tissues as a result of insufficient spatial resolution caused by two or more tissue types being present in the same voxel.
The field of view (FOV) is also closely related to SNR. FOV is the size of the region of interest, the square area that encapsulates the imaged structure: if the matrix size (measured in number of pixels) is given, then the FOV can determine the pixel size in mm^2. Hence, if the matrix size is fixed, a reduction of FOV will increase the spatial resolution and detail. However, smaller FOV requires high gradient strength and produces low SNR signal, as well as possible phase wrapping artefacts: portions of the image that lie on the outside of the FOV reappear in the opposite margin of the image in the phase encoding direction.
It can be seen that the correct balancing of the elements that affect image quality is important to achieve maximum efficiency for a given application. Table 2.1 summarizes the influence of different image and sequence characteristics with regards to SNR, effect on spatial resolution and image acquisition

time:

Characteristic	SNR	Spatial resolution	Acquisition time
Increased slice thickness	*Increases*	*Decreases*	*Decreases*
Increased field of view	*Increases*	*Decreases*	
Larger matrix		*Increases*	*Increases*
Increased TR	*Increases*		*Increases*
Increased TE	*Increases*		*Increases*
Increased magnetic field strength	*Increases*		
Employing local coils	*Increases*		

Table 2.1: Influence of different sequence characteristics on Signal to Noise ratio, spatial resolution and acquisition time.

Contrast agents for MRI

One commonly used method of influencing image characteristics is the employment of contrast agents. Contrast agents for MRI and MRA[2] are compounds that improve the resolution of MRI images by increasing the visibility of the structures of the body where the agent resides. As a result contrast agents usually provide better diagnostic information than MRI without contrast enhancement.
Contrast agents are classified into two categories, according to application:

- Intravascular, which highlight perfused tissue according to its blood supply.
- Cavitatory, which is not common in MRI but common in Radiology (eg. Barium meal). The current project has tried cavitatory enhancement of the mouth to outline the tongue.

[2]Magnetic Resonance Angiography, a special category of MR based imaging used to study blood vessels.

Most contrast agents are paramagnetic[3] species which affect T_1 and T_2 such that an increase in signal intensity is observed [Fri05]. There is a "sweet spot" for the effect of the agent on the intensity observed, as a function of the concentration of agent [ea84]. At low concentrations, an increase in contrast agent results in an increase in signal intensity. After an optimal point, the effects are reversed, and the signal levels are reduced.
Not all substances with paramagnetic properties exhibit strong enough properties to be used for practical applications. An example is molecular oxygen [MMD83]. Paramagnetic metal ions, such as Cu, Eu, Gd and Dy, however, do have a suitable effect that depends on the amount of unpaired electrons in the ion and the matching of the electron spin-relaxation time to the Larmor frequency. This condition mainly satisfies Fe3+, Mn2+ and Gd3+ ions.

This work focuses on cavitatory application of contrast agent using Gadolinium tried on both T_1 and T_2 weighted sequences. Gadolinium is a rare earth paramagnetic that reacts slowly with water and is soluble in dilute acid. Gadolinium compounds have strong paramagnetic properties as the atomic magnetic dipoles exhibit a tendency to align with an applied external field. Hence Gadolinium solutions can be used as a contrast agent in MRI to enhance tissue delineation and characterization [ea99]. Gadolinium has been used for many years, and considered safe for use in MRI in 1988 [SJ02].

2.4.4 Ability of MRI to display the tongue

In pathology, due to the multi-planar operation, contrast resolution and lack of radiation, MRI has been used to diagnose lesions of the face and neck, the sublingual and submandibular spaces, vascular malformations, different inflammation types (e.g. cellulitis, abscess) and tumours [LF95].

In terms of using MRI to reveal the internal anatomy of the tongue in more detail, it has the clear advantage over other non-invasive methods, and

[3]Paramagnetic species are substances with unpaired electrons. They can be simple substances (i.e. oxygen), stable radicals, nitroxide radicals or metal ions.

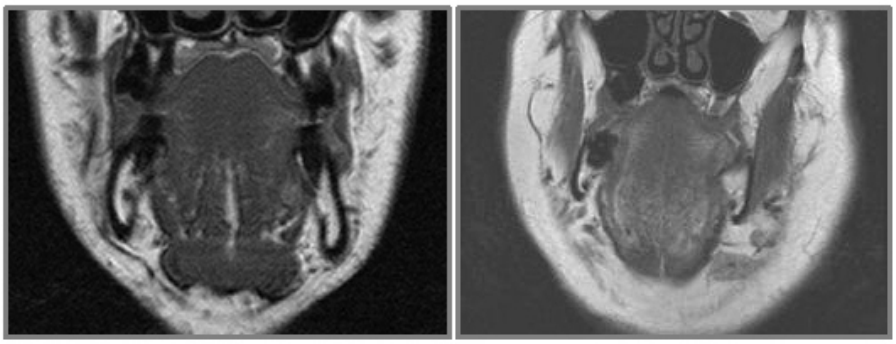


Figure 2.9: Normal tongue (left) and tongue that bulges due to missing teeth (right) on T_1.

while MRI is still not an optimal imaging technique for the tongue, it is the best. In 1985, Unger [Ung85] used MRI and identified Genioglossus and Geniohyoid muscles as discrete (albeit with very coarse resolution) structures, while all the intrinsic muscles were grouped together. Lufkin et al. [ea86a] performed a study of subjects with improved resolution (4 mm thickness of slice and 256 $\times$ 256 pixels matrix) in coronal and sagittal imaging planes. With the advent of improved technology and sequences the visualization of intrinsic tongue musculature was more clear than in earlier experiments, and it was possible to differentiate the superior Longitudinal, Transversus and Genioglossus muscles on the sagittal plane. The study also demonstrated that for the scope of the experiment, MR imaging was superior to CT, in terms of delineation and sensitivity to artefacts such as beam-hardening from the mandible.

Variability of tongue shape on MR images

Recent studies have shown that tongue appearance on MRI is a variable between subjects [H.W03]. While the shape of the tongue follows a generic pattern, there are factors that affect its perceived shape such as width of mouth or absence of teeth. 2.9 shows an example on coronal plane of a normal tongue (left) and a tongue that bulges laterally due to missing teeth.

Presentation of tongue on different weighted images

On T_1 weighted images, the delineation of muscles depends mainly on the presence of fat between the muscle fibres, as fat signal is much more pronounced. On T_2, muscles have low intensity signal regardless of fat presence. On both sequences, bone (mandible) and teeth have low signal intensities, and artefacts due to dental fillings appear (depending on their size) as a dark spherical region of the size of a few millimeters wide, that clouds the surrounding tissue and tooth.

2.5 Cancer

Cancer is a disorder characterized by the development of abnormal cells that divide uncontrollably and may invade and destroy normal body tissue, either by direct growth onto adjacent sites through invasion or by spreading to other places of the body through the process of metastasis. As the abnormal cells group together and multiply, a tumour grows. [ea04] Tumours are classified as being benign or malignant. Benign tumours do not invade surrounding tissues or metastasize, so even if left untreated or treated with symptomatic therapy, may not become life threatening. Malignant tumours are life threatening, as they invade nearby tissues and organs, causing damage. They can metastasize and form secondary tumours.
There are many types of cancer, classified in terms of the type of tissue and the site of the disease. Causes, symptoms, character of the malignancy, and survival rate are widely varied, and treatments are becoming more specific for each type of cancer [Kle05].

2.5.1 Causes, Symptoms, Diagnosis and Treatment

Cancer can affect people of any age, sex or race. In the developing world, cancer is the second leading cause of death after heart disease. It is responsible for about 1 in 4 deaths, and 0.5% of the population is diagnosed with the

disease every year [AJ05]. While the cause of most cancers remains unknown, some forms are associated with environmental factors such as overexposure to UV light from the sun, ionizing radiation and consumption of carcinogens - agents that promote cancer, such as chemicals found in cigarette smoke. Diet and other lifestyle factors can affect the risk of cancer. Obesity, lack of exercise and alcohol intake can all contribute.
Cancer does not give exclusive symptoms or signs that indicate the disease [eHl05], and a definitive diagnosis for most malignancies can only be confirmed by histological examination.

Diagnosis

Through the diagnosis of cancer an attempt is made to identify accurately the origin of the malignancy and the type of cells involved. Cancer can arise in any organ or tissue in the body and there is no single test that can accurately diagnose the disease. Diagnostic procedures include tumor biopsy, endoscopic examination, laboratory tests and medical imaging [oV06]. In each case, there are benefits and risks involved with the diagnostic procedures. While imaging is useful for detection of the malignancy, it cannot provide characterization. For this a biopsy is required.

Biopsy

Since a radiologic study is not sufficient to establish a diagnosis of cancer, the clinician must prove the presence of cancer cells as seen under a microscope. Biopsy is a test that involves the removal of a sample of tissue from the body for examination. The tissue is examined under microscopy. Usually very small samples are needed (incisional biopsy), but certain tests involve the removal of an entire lump of suspicious area (excisional biopsy) or removal of tissue or fluid with a needle (needle aspiration biopsy[4]). Biopsy specimens are

[4]An indication of the extensive use of medical imaging is the fact that clinicians often have to rely on CT and ultrasound imaging in order to guide the long, thin needles deep inside the body to biopsy organs. This negates the need for open surgical procedures.

acquired in cases where the cause, extent or type of malignancy is uncertain.

Treatment

Cancer is treated with surgery, radiation therapy (irradiation), chemotherapy, immunotherapy, transplant (bone marrow), hormonal suppression and other methods. Treatment can be targeted either towards controlling symptoms and relieving distress, or towards a complete cure. The choice of treatment depends upon the location, the extent of the tumour and a measure of the patients' general well being. Where disease is localized, treatment by excursion or radiotherapy is possible, with surgery usually being the best chance of cure. Treatment therefore depends on disease delineation, which is an important application of imaging.

2.6 Cancer of the oral cavity

Oral or mouth cancer involves the growth of malignant tumour in the oral cavity [Gru05]. The tumour might involve tissue of the lips, the tongue, cheek lining, gums and palate. Most oral cancers, usually called squamous cell carcinomas, have the same physical appearance under the microscope. They are malignant and can metastasize rapidly depending on the primary site.

2.6.1 Etiology and risk factors

Smoking (both tobacco and marijuana) is associated with 70-80% of all tumor cases. Smoke and heat from cigarettes aggravates the mucous membranes found in the oral cavity. In the case of tobacco chewing, the irritation is caused from direct contact with the membranes. Alcohol, while on its own not clearly linked to the malignancy, is believed to have a synergistic role to the damage done by heavy smoking. Other risk factors include poor orodental hygiene, chronic irritation from rough teeth, fillings or dentures,

vitamin deficiencies and, in the case of carcinomas of the lip, sun exposure [DJ82]. Hereditary factors have not been identified.

2.6.2 Clinical presentation

Symptoms associated with this disease are mainly unexplained bleeding, non-healing ulcers, erythroplakia, small skin lesions and lumps found on the tongue, the lips and surrounding areas. The carcinoma may be completely painless at the beginning but in advanced stages can develop a burning, painful sensation, leading to swallowing and eating difficulties. Misfitting dentures are another sign of oral cancer, and a dentist may spot the disease during a routine dental check.

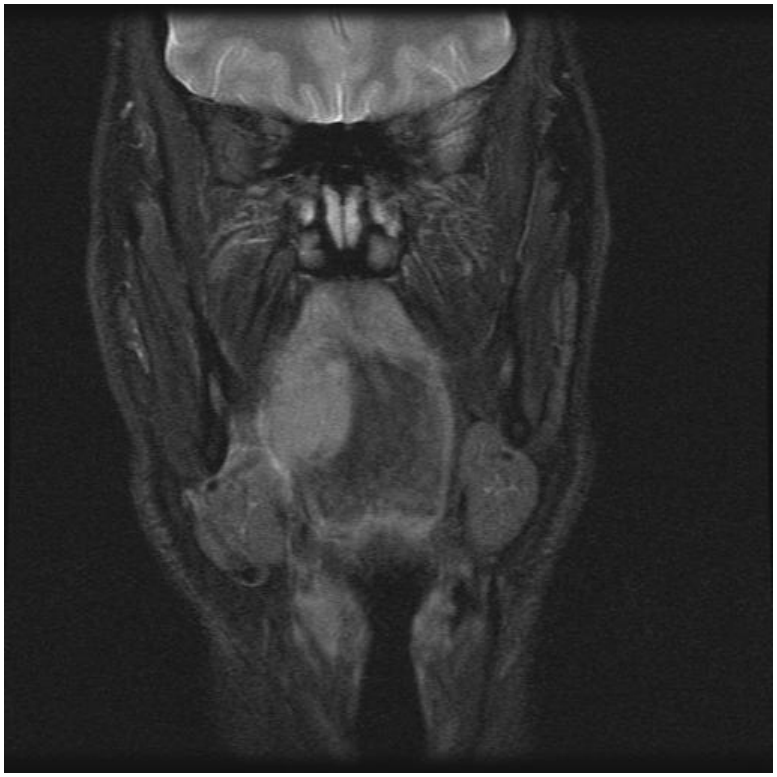


Figure 2.10: Presentation of large tumour on T_2 image. Tumour has higher signal intensity than surrounding tissue.

2.6.3 Clinical presentation of tongue cancer in MRI

On T_1 weighted images tumours of the tongue appear as a mass of intermediate signal intensity. Muscles have a similar intensity and therefore the outline of the tumour may not be visible unless it distorts the tongue outline or fat between normal muscle fibres allows the outline of the tumour to be detected.

On T_2 weighted images the carcinoma generally has a higher signal than the surrounding muscles and is therefore easier to define (figure 2.10 shows an example). Local tumour staging therefore depends more on T_2 weighted imaging. The evaluation of the relationship between the tumour and normal muscles depends on the signal difference between the tissues.

In general the tumour relation to large extrinsic muscles can be determined when the tumour is large or extensive but smaller degrees of disease may be more difficult to evaluate. This aspect of MRI is used in establishing tumour staging, which is described in section 2.7.

More insight into the value of MR imaging for the demonstration of the masses in the tongue is given by Ozturk et al. ([MO00]). Further information on the MR sequences that best delineate the margins and extent of tumours can be found in the work of Yasumoto et al. ([MY95]) and Campbell et al. ([RCM95]).

2.6.4 Treatment

Oral cancer is best treated in the early stages, with a combination of surgery and radiation therapy. Radiation therapy alone may be applicable for Stages I and II. In advanced form (Stages III and IV), surgery can be used for removing the carcinoma and for further reconstructive procedures.

2.7 Disease staging: the TNM System

The TNM Classification of malignant tumours ([LH02]) is a cancer staging system used globally and recognized as a standard in categorizing cancer. The TNM system was developed by the International Union Against Cancer (UICC), and since 1987 is compatible with the staging systems used by the American Joint Committee on Cancer (AJCC). The TNM format has been adopted to assist in prognosis and evaluation of a treatment. Since it is a universal standard, clinical and research facilities around the world can gather data and group them in an efficient and productive manner with fewer logistical problems. In the TNM classification system, almost all tumors are classified. T describes the tumour, N the regional lymph nodes and M the distant metastases.
In more detail:

2.7.1 Mandatory parameters 'T', 'N' and 'M'

- **T**x x being the size of tumor and taking values between 0-4. With **T0** there is no evidence of primary tumor. **TX** means that the tumor cannot be assessed.

- **N**$(0-3)$ spreading into lymphatic nodes. The number indicates the size, location and/or the number of nodes involved. With **N0** the nearby lymph nodes do not contain cancer cells, while with **NX** the nearby lymph nodes cannot be measured or found (perhaps if previously removed).

- **M**$(0-1)$ spreading into other organs (metastasis). **M0** : no known distant metastases. **MX** : metastasis cannot be measured (or found) and **M1** means that distant metastases are present.

2.7.2 Other parameters

- **G**$(1-4)$ the grade of the cancer cells. A high grade implies that cancer cells appear different to normal cells, a low grade if they appear similar to normal cells.
- **R**$(0-2)$ the completeness of the operation.
- **L**$(0-1)$ invasion into lymphatic vessels.
- **V**$(0-1)$ invasion into veins.
- **C**$(0-1)$ the certainty of the last parameter **V**. It can be assessed from whether the evidence was retrieved by standard diagnostic means (inspection, endoscopy, radiography), by special diagnostic means (tomography, MRI), or by surgical exploration and autopsy, in ascending order.

2.7.3 Prefix modifiers

- **c** stage is the clinical classification. **c** prefix is implicit in the absence of a **p** prefix.
- **p** stage given by microscopic evaluation using surgical specimen.
- **y** stage assessed after neoadjuvant therapy specimen.

2.7.4 TNM classification of tongue cancer

With respect to the tongue, cancer is staged similarly in the TNM system. The following **T** combinations are used to describe cancer of the tongue:

- **TX** Primary tumour cannot be assessed.
- **T0** No evidence of primary tumour.
- **T1** Tumour 2 cm or less in greatest dimension.

- **T2** Tumour between 2 and 4 cm in greatest dimension.
- **T3** Tumour more than 4 cm in greatest dimension.
- **T4** Tumour invades adjacent structures (through cortical bone, into deep/extrinsic muscles of tongue, maxillary sinus, skin).

2.7.5 Disadvantages of TNM system for tongue cancer

The TNM system is based on clinical evaluation, not depth of penetration. However, depth of penetration is the information that matters to the surgeon. For example, the system dictates that tumour invasion into extrinsic muscles of the tongue is classified as T3 or T4. It can be seen that there is no additional information about the depth of penetration into the extrinsic muscles and the relation of tumour to local tissue. Hence there is a need to find a method of delineation that relates to surgical parameters, i.e. the muscles.

2.7.6 Relation to this experiment

It is therefore of clinical importance to establish a tangible metric of the presence of cancer cells in the area occupied by the extrinsic muscles of the tongue. This defines the objective of this thesis: to provide the methods to map the extrinsic muscles of the tongue in order to assist surgical staging. The technique employed to achieve this objective is to create an atlas model of the interlingual anatomy and relate that to any given MRI examination of the tongue.

2.8 Summary

The chapter started with a detailed description of the interlingual anatomy, then moved into the different imaging modality technologies, expanding on

MRI which is employed as the primary source of data in this text. A connection was made with MR imaging of the tongue, and its importance in visualizing cancer, with emphasis on tongue cancer and the TNM classification system employed to describe it. In doing so, the purpose of this thesis was established.

To prove my case in the following chapters, I need to describe the tools that I have made and used to manipulate the MRI images and other sources of data in order to reach the objective.

The thesis continues in the next chapter by introducing the two domains of data used throughout the experiment, the atlas model and the MRI scan.

Chapter 3

General experimental procedures and generation of data volumes

3.1 Introduction

The purpose of this chapter is to present the different tongue modelling approaches found in the background literature, and the procedure that I followed in order to create the tongue atlas that is used as reference in further steps of the experiment. Additionally, I outline the MRI sequences and experimental procedures used to acquire the MRI volume that depicts the tongue of the subject. The chapter, therefore, has two objectives:

- To find a method and construct an atlas model of the tongue, in the form of a volume, with the muscles identified by their type: extrinsic or not extrinsic.
- To extract the volume of the tongue in an MRI scan of a subject: unlike the previous procedure, with which I construct a fixed atlas, this one needs to be repeated for every different subject or examination.

The description of the MRI sequences and procedures is straightforward, as I employ standard techniques of imaging in order to acquire the MRI volume. The construction of the atlas model, however, is more complicated, as a mechanism must exist to identify the muscle groups and correctly tag the cross-sectional images at voxel level. The identification of tongue muscularity in the MRI volume is the purpose of chapters 5,6 and 7.

3.1.1 Organization of sections

In this chapter, the different concepts presented are arranged into sections that follow a logical order, from the general background theory of tongue modelling approaches to the approach that I followed in this project. In more detail:

- Section 2 presents a historic survey of tongue models developed for academic or clinical purposes, from the early simple 2-dimensional models to the more recent, complex 3-dimensional models.

- Section 3 summarizes the demands for the atlas model that was needed in the current project. A decision was made to use cross-sectional photographs of the human tongue, derived from the Visible Human Project and post-processed according to the demands outlined. Based on these images, a software tool was then built to identify manually, with the help of an anatomist, the regions that depict the different muscle structures of the tongue.

- After describing the methods that I used to generate the atlas model, section 4 outlines the procedure of retrieving the tongue volume of the subject through MRI scanning. The section summarizes the MRI sequences employed, as well as methods used to enhance the quality of the data retrieved, in order to assist image processing of the volume in question at a later stage.

- The chapter ends with an evaluation of the results so far, and how they fit in the wider scope of this project.

3.2 Survey of tongue modelling approaches

3.2.1 Introduction

In the past several researchers have attempted to develop 2D and 3D models of the tongue. When modelled, the tongue is often represented as a simple parallelepiped [J.74]. The "model" in this context is a mathematical or computer representation of the geometry of the structures in question, in static and dynamic arrangements. Depending on the arrangement, the modelling approaches can be roughly split into the following two categories [Eva04]:

3.2.2 2D articulatory modelling

The scope of articulatory modelling is to attempt to describe the dynamics of the tongue in speech production, as viewed mainly from the sagittal plane. The prediction of the behavior of the system is addressed using a set of parameters and initial conditions derived from linguistic and physiological measurements. X-ray cineradiography, being the only modality that can offer high frame-rate recording, is used during speech in order to trace the position, shape and trajectories of the tongue body, surface, tip and other components pertaining to the tongue. The acquired information may then be used as a basis for defining a mathematical model that exhibits very similar physical properties (inertia relationships, displacement limits, body stress, physical impenetrability) to the features under investigation. The scope of the abstract model to try to emulate reality and the laws of physics that support it.

Several different physical-based models have been developed: for example, Honda et al. and Sanguineti et al. ([HKJ94] and [SVY97]) use finite element analysis to emulate tongue deformation, while Perrier et al.([PPE88])

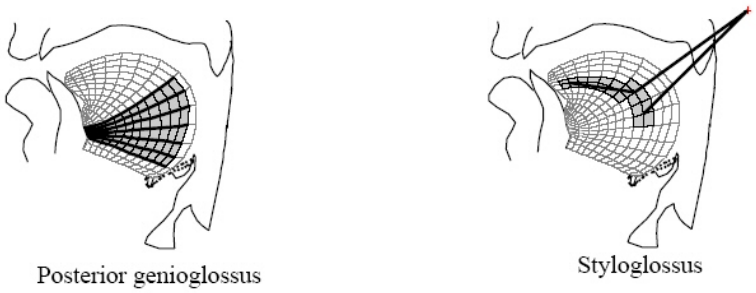


Figure 3.1: Posterior Genioglossus and Styloglossus muscles as depicted using the model by Sanguinet [SVY97].

use springs with time-varying stiffness to model muscle synergies and stress. The geometrical modelling is usually based on a mesh of triangulated elements ([HKJ94], [J.74]). A visualization of the model based on triangulated elements can be seen in figure 3.1. Perkell [J.74] was the first to propose a physiological model of the tongue that takes into account the presence of intrinsic and extrinsic muscles, but the arrangement and labeling was based on anatomical references from dissection work. This model can be seen in figure 3.2. However, the tongue's overall shape and behaviour was defined by X-ray cineradiographic tracings of a speaker [J.69].

Normally the model is applied in linguistic research in order to recreate articulatory movement and generation of acoustic output under various circumstances. As such, the emphasis of the abstract model is not on recreating a realistic representation of the tongue musculature but rather on giving an insight into the function of the tongue over time in speech reproduction. The singular plane of projection is inadequate for modelling a structure as complex as the tongue, with its overlapping muscles, in 3 dimensions. Additionally, potentially harmful exposure of the subjects to X-rays may be restrictive for certain types of clinical research experiments.
It is impractical to use single plane acquisition to map a realistic atlas of the

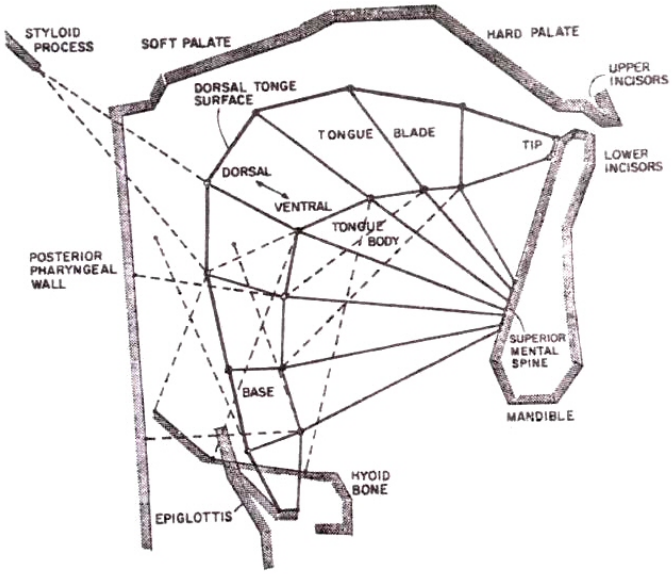


Figure 3.2: The early 2D model suggested by Perkell [J.74].

tongue musculature and hence it is outside the scope of this work to analyze further the attempts in 2D bio-mechanical modelling of the tongue.

3.2.3 3D tongue modelling

In this category, the modelling methods attempt to represent the muscles comprising the tongue in a 3D configuration.

The first 3D model of the tongue was suggested by Kiritani ([ea76], depicted in 3.3), in which the tongue was considered as an elastic object. The model was utilized using the finite element method in which volume incompressibility of the structure (which simulates the muscular hydrostat) and assumption of uniform tension between each element were some conditions of the simulation. The model is comprised of 28 elements, divided equally between intrinsic and extrinsic muscles, which are represented as a configuration of prisms. Kakita ([Y.77]) and Hashimoto ([KS86]) used similar finite element methods, but increased the number of elements to 86 tetrahedrons and 496 tetrahedral nodes respectively. Their work was not only an improve-

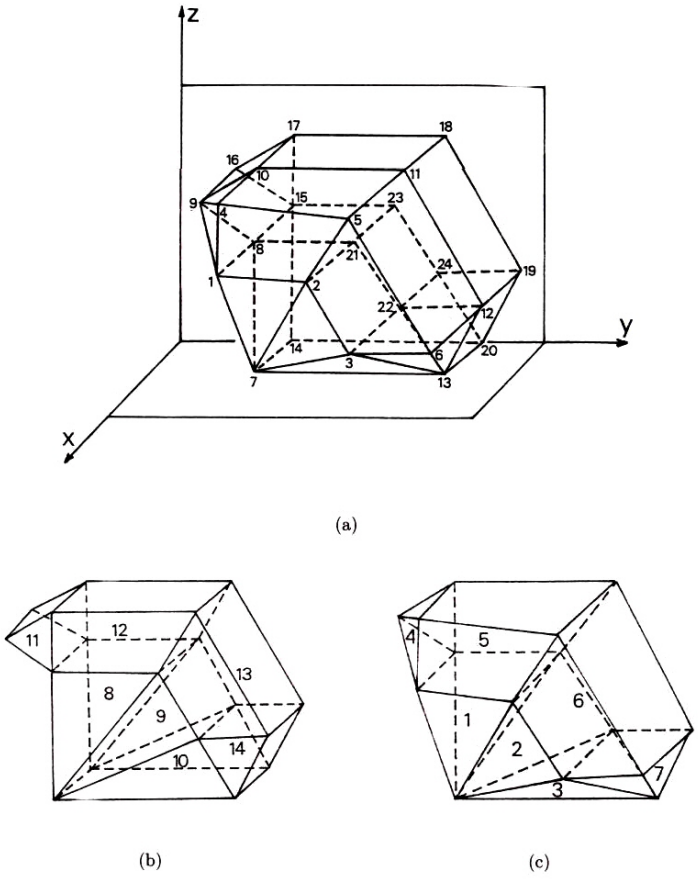


Figure 3.3: The 3D finite element model by Kiritani. [ea76]

ment upon the model suggested by Kirinati in terms of accuracy, but also in terms of speed.

In their models, a mid-sagittal plane symmetry was assumed. The nodes were grouped into functional units that could simulate muscle contraction for a given stress and strain condition. Similarly to the 2D methods, the physical parameters of the emulation were acquired from tracings of X-ray radiographs. Stone ([M.90]) combined imaging techniques and feature tracking to build a model of tongue movement. In this work, a number of metallic pellets is introduced on the tongue surface and tracked using X-rays, while

the coronal plane cross-sections of the tongue and mid-sagittal plane curve are traced using a developmental real-time 3D ultrasound device.

Other imaging modalities that have been used as a basis for constructing geometric models of the tongue are MRI and photography. The two different approaches are presented in the following two subsections.

3.2.4 Geometrical modelling of the tongue based on the MRI modality

The MRI modality has not been used extensively in the literature of 2D or 3D tongue modelling approaches, when the target of the modelling is to construct a realistic atlas of the tongue musculature. This is mainly due to rather limited spatial resolution for volume acquisition (matrix size, and most importantly slice thickness), compared to the alternative, which is photography of histological sections. Another issue with MR imaging is the inability to delineate clearly between all the muscle groups of the tongue, as they are not necessarily separated by a medium with different fat or water density, which would show up on the MR images as a region of different intensity. The ability of MRI to display the tongue is discussed in more detail in section 2.5.5. However, unlike histological or cryo-specimen photography, MR images are captured in-vivo. Therefore, while MRI is not the best choice for constructing a detailed atlas of the tongue musculature, it can be used in phonetics research (in animated sequences, again limited by generally low frame-rates compared to X-ray cineradiography) and attempts at quasi 3D geometrical representations of the tongue for use in custom applications.
Dang and Honda ([DH96]) conducted an experiment in phonetics in which the objective was to produce an improved vocal tract model of vowel production. An important prerequisite was the calculation of the cross-sectional area of air as a function of the distance from the glottis to the tip of the tongue as the subject was reproducing certain vowels.

To achieve this, they used both physical measurements and MRI to ac-

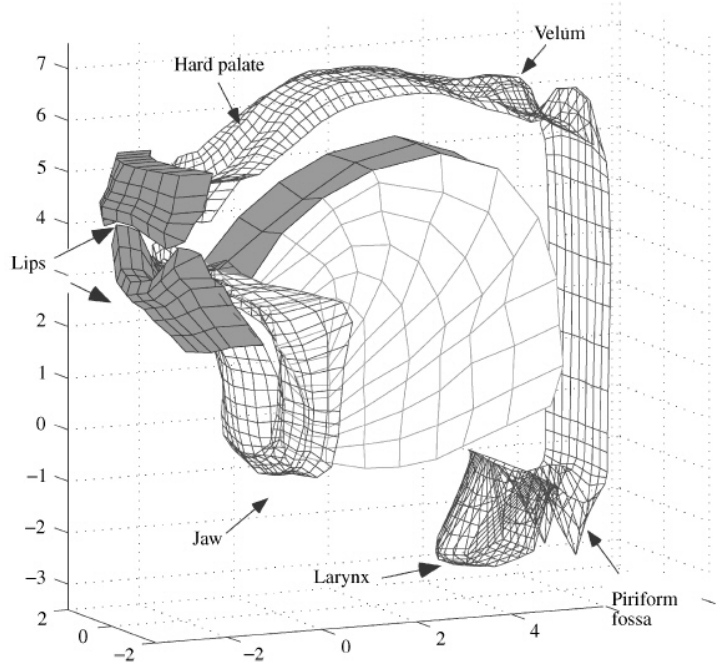


Figure 3.4: The geometrical model of the tongue suggest by Honda, as an aid in phonetics experiments [DH02].

quire morphological data of the vocal tract. Volumetric MRI data were obtained in the transverse and coronal planes using a 256 x 256 matrix for each slice and 0.4 cm slice thickness. Over these images they traced manually the tissue-air boundaries on the mid-sagittal slice, forming the vocal tract mid-line, and calculated the area measurements. Since their objective was not detailed modelling of the tongue anatomy, but rather a model that would assess the temporal variations of the tongue orientation with articulatory movements, their geometrical model of the tongue was an abstraction and of little use in experiments of different nature.

Further use of manually post-processed MRI data was made in an experiment of similar scope by the same authors [DH02]. Instead of using a full 3D model of the tongue, they constructed a partial 3D model based on a 2 cm thick mid-sagittal layer (Figure 3.4). The orientation of tongue muscles was examined with reference to existing literature ([K.74]) and visualized over

the mid-sagittal plane. This model is an extension of the previous one in the sense that it is more detailed and takes into account the difference between the muscles. However, it is not fully 3D but rather 2.5D: a term that refers to a surface which is a projection of a 2D construction into the 3rd dimension.

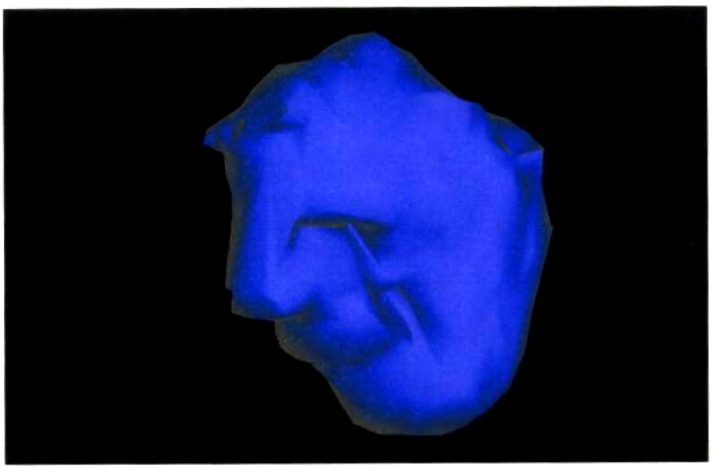


Figure 3.5: A shaded 3D render of a tongue model calculated by Evangelou using MRI images [Eva04].

Finally, Evangelou ([Eva04]) was concerned with modelling the detailed structure of the human tongue musculature in-vivo using MRI. The motivation behind this work was to build a database that contains geometrical models of different subjects and then to generate a statistical model based on these multiple observations. The approach followed by Evangelou was to use image processing feature extraction, with some manual support, to retrieve the tongue structure as contours in consecutive axial planes, and triangulate them into a 3D surface mesh. His model is illustrated in figure 3.5. The work succeeds in providing a relatively straightforward framework to build upon but the quality of meshes produced is moderate: while the tongue as a whole is reproduced in adequate detail, the internal structure is not sufficiently designated, mainly due to the inability of the medium (MRI) to delineate the muscle groups in question.

3.2.5 Geometrical modelling of the tongue based on photography

Cross-sectional photographs of the tongue (for example those of the "Visible Human Project") provide much higher spatial resolution than medical imaging, and the use of colour makes it easier to distinguish between different muscle groups. The "Visible Human Project" is a digital atlas of the human body, a database containing colour images of physical cross sections of a male and a female cadaver. The Visible Human Project is reviewed in more detail in section 3.3 of this chapter.

In a model developed by Wilhelms et al. ([WTC91]), the tongue was constructed as a network of 3D geometric primitives based on histological sections of specimens. Their system also took into account the muscle fibre directions that dictate the stress and elongation parameters during muscle movement and deformation. Building upon the same histological sections, Wu ([C.M92]) developed a custom graphics environment, loosely akin to a CAD [1] program, that provided an interface to sketch, tag and define fibre direction on the various muscles. In similar fashion, Tricarico ([R.95]) attempted a computational simulation of both bio-mechanical and physiological features of the structures of the vocal tract with a model derived from anatomy books and cryo-section photographs (of the Female Visible Human Project) of the tongue.

Tricarico's work mainly deals with the modelling of the dynamic behavior of the tongue for quantitative studies of tongue gestures. To facilitate the construction of a realistic bio-mechanical model, it is important that the fibre directions and densities are specified on an individual muscle basis. For this, the muscles are represented as blocks of prisms and hexahedra and their fibre direction is directly computed through the normals of the curvi-linear coordinate lines of the geometric models. Figure 3.6 shows a 3D rendering

[1] Computer Aided Design

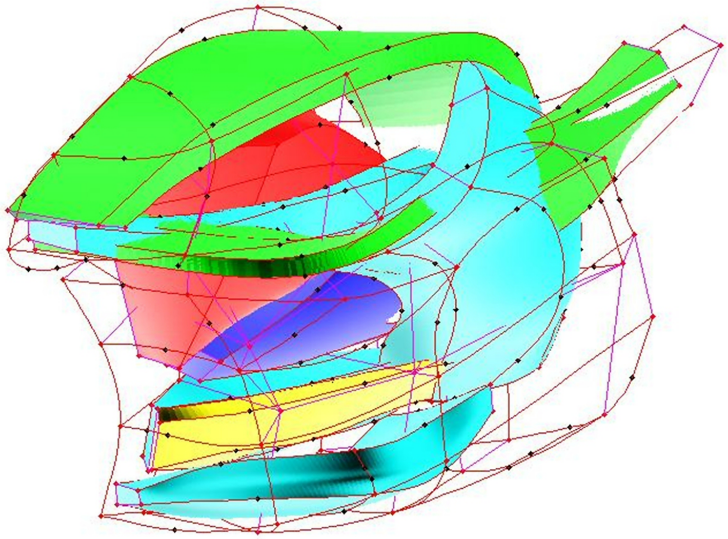


Figure 3.6: A representation of the model developed by Tricarico, with the different muscles depicted in different colour [WT06b].

of Tricarico's tongue model.
Tricarico's work is an ongoing project that over the past 10 years (latest update [WT06b]) has significantly explored the domain of tongue modelling and has arrived at the following design patterns and conclusions about future research strategies:

- The anatomy of the tongue is complicated enough to require a high degree of refinement: a high number of finite elements (of the order of 200-250) is needed for adequate cross sectional resolution.
- A tongue is built up of tissue with varying characteristics; muscles are surrounded by membranes and other passive tissues (fat), that have distinct elastic properties. Their definition is important in articulatory gesture modelling, and thus must be taken into consideration by improving spatial refinement - using more elements.
- The directions of the fibres within the muscles vary with respect to

their boundaries. It is a simplification to assume that within one finite element the fibres of the same muscle types are parallel since, in reality it is possible to have two or more muscle fibre directions within the same element. As an example, the transversus and verticalis muscles are usually entangled. The solution is to specify an explicit function that relates the fibre orientation vector of a given node to its position within the muscle, or a network of muscles.

An extension to the original model proposed by Tricarico was attempted by Wu ([CM.96]). His geometrical model was based on anatomical landmarks placed over the Visible Human (male) Project and constructed using 2D and 3D thin-plate splines. The landmarks were dictated by drawings of tongue sections made by Miyawaki ([K.74]). The process of reconstruction is interesting in that it shows that it is possible to use real-life data and hand-drawn sketches and fuse them into a single model. Miyawaki's approach is illustrated as follows:

- For a given tongue section, a hand-drawn sketch from Miyawaki's work is used (Figure A).
- For the same section, an image from the Visible Human Project is retrieved and the outline is manually traced (Figure B).
- There is now a need to map the sketch into the Visible Human Project outline. To do so, a number of corresponding landmarks is placed manually.
- Figures A and B are mapped into each other using thin-plate spline and grid-based image warping.

A more recent model of the human tongue musculature based on images from the Visible Human Project was developed by Sanders et al. [SIA02] (figures 3.7 and 3.8). The purpose of the work was to produce a comprehensive study of the 3D anatomy of the human tongue, identifying the position

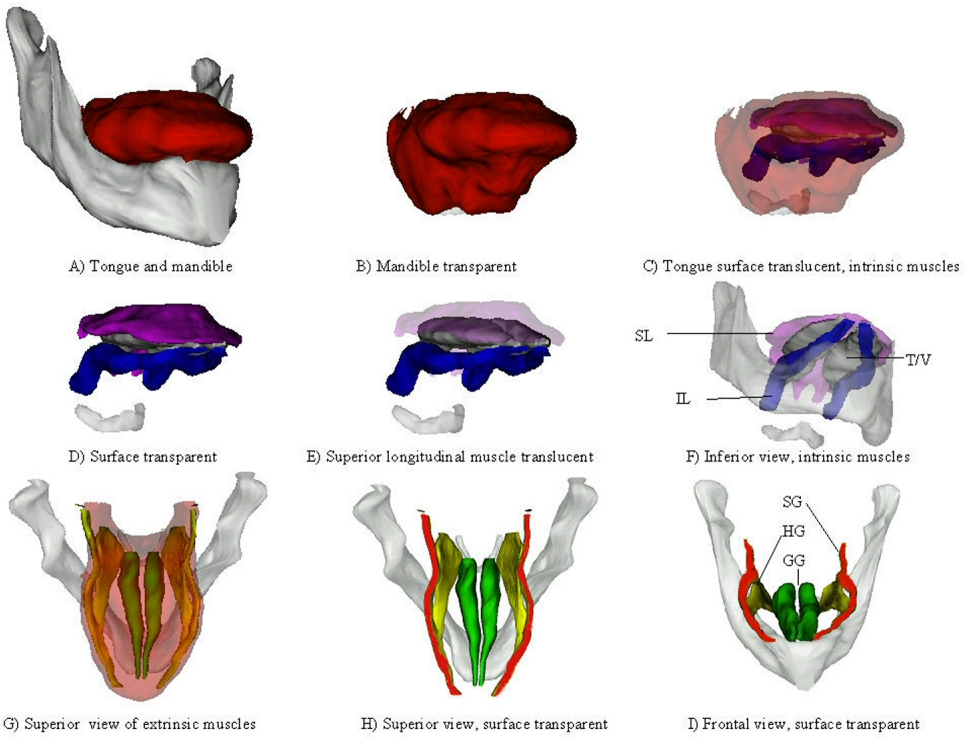


Figure 3.7: 3D shaded models of the tongue muscles, as depicted in Upper airway project website [SIA02].

and shape of each muscle and to represent them geometrically as 3D meshes. This 3D atlas of human tongue musculature would have been a valuable source of information for my project. At the time of writing, however, the data was not available for public or research use.

3.2.6 Summary

Tongue modelling is an elaborate problem because of the complexity of the tongue musculature. Several researchers have attempted to construct geometrical and/or bio-mechanical models but most of them are mainly concerned with tongue movement in phonetics. As such, they are based on arbitrary hypotheses on the shape, the number and the layout of the individual muscles of the tongue. In contrast, the required model for my project needs

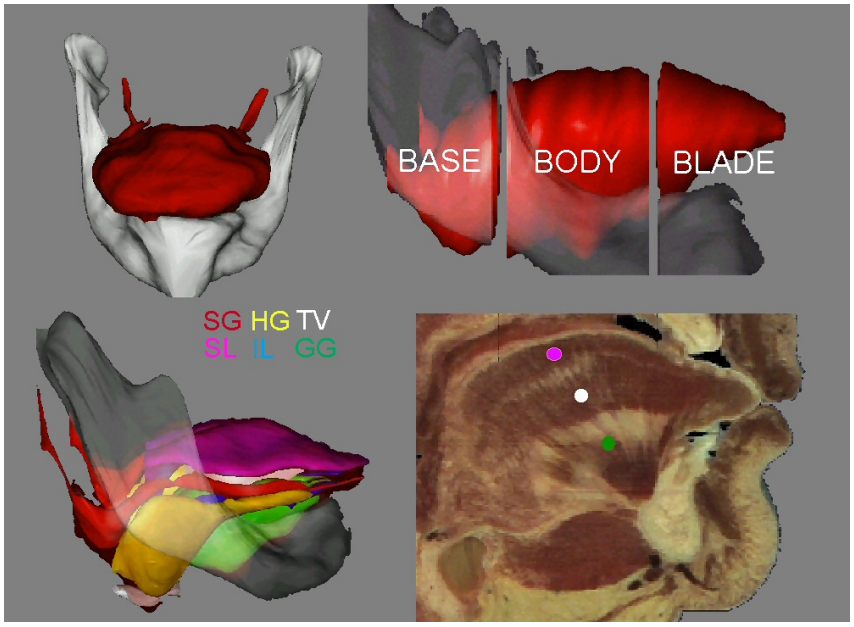


Figure 3.8: Different views of Upper airway project models (Upper airway website) [SIA02].

to be an accurate geometrical abstraction of reality, without the extra computational complications that a full-blown finite-element based bio-mechanical model would introduce.

3.3 Geometric modelling of the tongue and extrinsic muscles

Most of the models referred to in the previous section are either publicly unavailable or not suitable for my project, which has the following demands:

- A static geometrical model of the tongue, from the apex to the root, and of the extrinsic muscles of the tongue, either as bulk object (Genioglossus, Styloglossus, Palatoglossus and Hyoglossus muscles combined as a single object) or with the muscles as separate objects. There is no concern for the intrinsic muscles of the tongue.

- The model must either be in the form of a 3D surface mesh or in voxel space - a 3D space where the presence of the model is indicated on a per volumetric pixel basis.

- The model must have adequate detail: for example, a representation of a muscle with a single polyhedron is simply not sufficient. However, there is a margin of error allowance of ¡3 mm to accommodate for various misalignments and miscalculations that may occur.

Based on these targets and published experience, a decision was made to use photographic material as a basis for the model's morphology, rather than MRI. The reasons are that, based on evidence from the literature review, I believe that MRI images are not adequately delineated (at least on the 1.5 Tesla scanners that I have access to), and hence an attempt to designate the tongue musculature is impaired. MRI has lower spatial resolution than the alternative high-resolution cryo-tomography images, which are also in colour and give a very good indication of muscle fibre orientation[2]. Furthermore, the Visible Human Project database was readily available, with a plethora of reference material (most importantly [R.95]) to assist my reconstruction. Since my tongue model is adapted to that of the Visible Human Project specimen, it is necessary to expand on the database itself and the methods that I followed to construct the model.

3.3.1 The Visible Human Project

The Visible Human Project is a collection of highly detailed cross-sectional photographs of human anatomy used in visualization and research experiments. The project's aim was to create detailed 3D illustrations of the male and female bodies. The project begun in 1989 by the U.S. National Library of Medicine (NLM), and the dataset was available to the public in late

[2]Muscle fibre orientation is only approximated. Due to small, finite resolution, the fibre direction here refers only to macroscopic observations, which might be different to a microscopic study of the same tissue specimen.

1994 [AYJ96]. A male and a female cadaver were cut axially into thin cryo-sections, which were digitally photographed in high resolution. The bodies were also CT scanned, and further axial section of the head and neck were obtained by MR imaging. Before they were cut, the bodies of the cadavers had to be embedded in blue gelatin, frozen to $-70°C$ and remain at very low temperature for days. The process produced some artefacts on the male images, such as a number of collapsed blood vessels and a lost inner ear ossicle. Nevertheless, the problems with the datasets are now documented and can be compensated for. Also, in order to aid the alignment of the slides, 4 plastic rods were placed in the corners of the volume that could act later as reference guides.

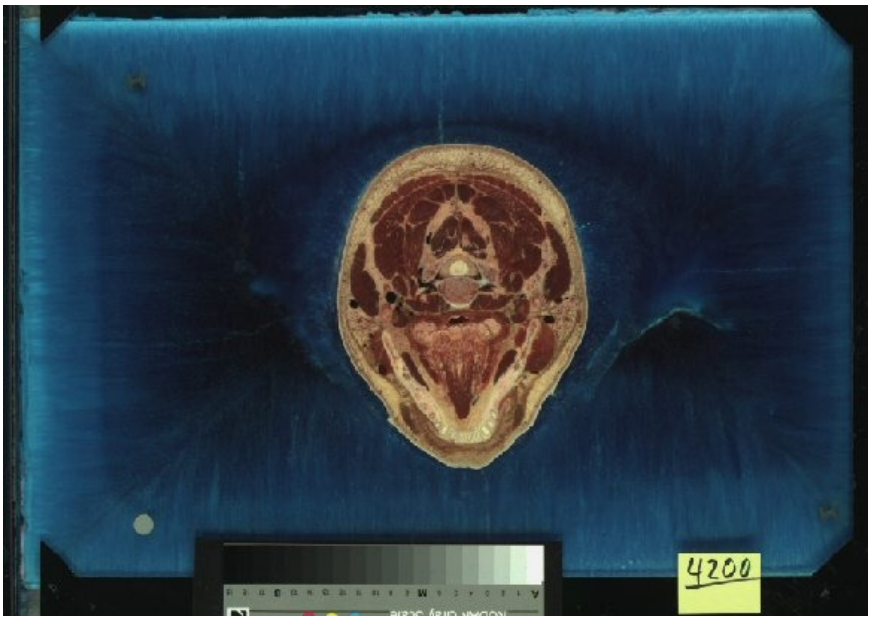


Figure 3.9: The Visible Human Project in original axial form.

My experiment uses the female dataset for its superior quality, as it was cut into slices at 0.33 mm intervals, compared to 1 millimetre intervals for the male. The female cadaver dataset was also chosen because of a major "glitch" in the male dataset: the tongue protrudes unilaterally through the empty space where the molar teeth are missing, as a consequence resulting

in atypical post-mortem deformation.
Figure 3.9 shows an image of the female Visible Human Project axial set.

3.3.2 The Female Visible Human Project dataset

The female dataset is for a 59 year old woman who died of natural causes. The colour images are of very high definition: uncropped, they each have a resolution of 2048 $\times$ 1216 pixels at a 0.33 mm pixel size and make up around 7.5 Mbytes of data. At 0.33 mm interval there are around 5000 images covering the body from the top of the head to the bottom of the feet, resulting in some 40 Gbytes of uncompressed raw data.

3.3.3 Preparation of the Visible Human Project data

Since my experiment is focusing on a subset of this dataset, the following three steps had to be taken so that a smaller dataset focused around the tongue was produced:

Reading the images - conversion

The images of the Visible Human Project were in .RAW format, with no header and a 24-bit RGB non-interleaved colour structure. A non-interleaved bitmap has all the bytes assigned to the red component of the colour of a given pixel appended first, followed by the green and the blue, rather than in triplets of RGB. In the absence of a header, I found the correct X-Y resolution of the images by studying the accompanying documents that arrived with the Visible Human dataset DVD.

Selection of axial slices containing the oral cavity

This process involved going through the slices and selecting the ones that contain the oral cavity. A viewer of the full-body Visible Human axial images was written in order to assist the selection of the critical slices. As such, 330 Images were tagged, yielding a volume 10 cm tall, between the plane passing

through the center of the eyes to the one passing through the body of the mandible.

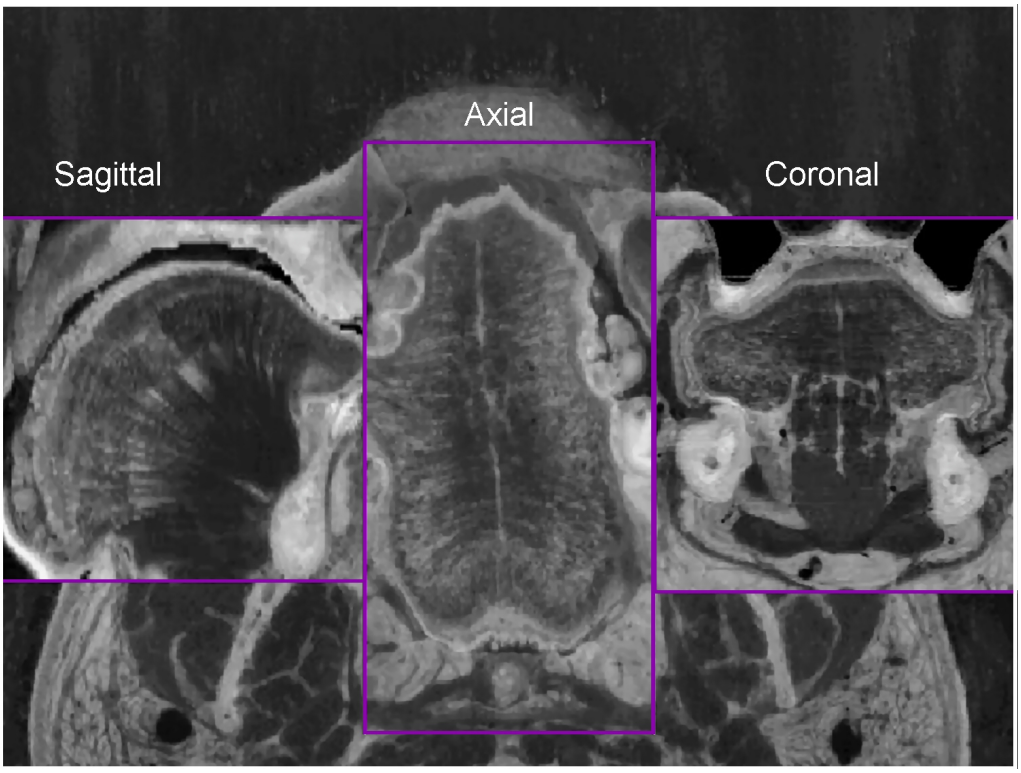


Figure 3.10: The Visible Human Project: different views of the tongue in grayscale, generated through Multiplanar reconstruction (MPR).

Cropping of the selected axial slices

The axial images were then cropped from 40 cm to 10 cm in the coronal axis and 8 cm in the sagittal. The objectives were to reduce the amount of data (a reduction by a factor of 6), hence decreasing the memory requirements and increasing the speed of subsequent operations, and to discard surrounding structures which are irrelevant to my experiment. I did not find any use for the fiducial rods that were placed originally at the corners of the axial images, and these were cropped as well.

After performing these three preparation steps, the next objective was to an-

notate the voxels within the newly constructed volume with the musculature information.

3.3.4 Geometric modelling

I wrote a software design tool to visualize the Visible Human Project cross-sections of the extracted block that contains the tongue in any orthogonal plane[3], as illustrated in figure 3.10. The tool also provided the methods to manually annotate a region of voxels as belonging to either an extrinsic muscle, the rest of tongue, or something else (i.e. air, bone and other structures). The operation is similar to a painting program: the muscle groups are "tagged" with a semi-transparent colour. There are "helper" functions for ease of use, such as an adjustable line of symmetry to take advantage of the bilateral symmetry of the tongue: muscle groups tagged in any lateral sections are automatically tagged in the other. Also there is a provision to tag the hard palette with curves, in case this is needed in the future. I used the reconstructed coronal plane (frontal plane) for most of the annotation work. The reason for this is that, as expected, almost all of the tongue muscles were oriented lengthwise and could be viewed as elongated structures that can be well described by a surface of revolution - a surface generated by rotating a two-dimensional curve about an axis, the axis being in this case the coronal. It was noticed, however, that the tongue as originally photographed was not perfectly symmetrical on the coronal plane. The reason for this was an apparent slight rotation of the head (and presumably the rest of the body) around the axial plane. To compensate for this I added a "skew" factor of $\sin(\frac{\pi}{36})^4$ into the plane reconstruction after trial-and-error experimentation with small values.

[3]The process is also known as MPR - Multiplanar reconstruction. It is a simple method of reconstruction of different planes orthogonal to the original plane, in this case the original plane being axial and the reconstructed sagittal and coronal. If the original volume data is isotropic (and this is the case here), the problem is trivial.

[4]The skew factor implies that the normal of the reconstruction plane is now $\left|\overrightarrow{[0, \sin(\frac{\pi}{36}), 1]}\right|$.

I recruited the assistance of a specialist[5] in oral anatomy who used my software to do the annotation work and prepare the first draft of the model. Once it was finished, the volume of identified voxels was saved for validation and future reference. Figure 3.11 shows a screen-shot of my tool in action.

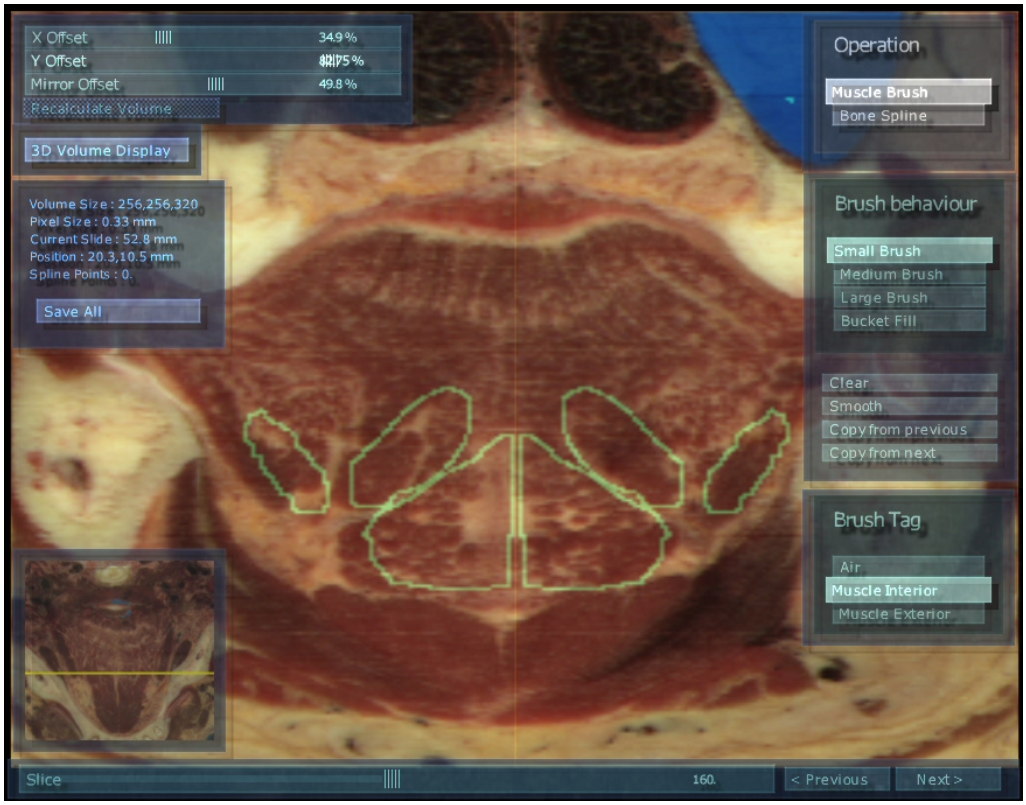


Figure 3.11: Muscle tagging using the annotation tool.

3.3.5 Validation through data visualization

At this stage, the model is in the form of a voxel model: for each voxel in the volume that corresponds to the original set of data, an musculature "I.D." scalar value can be retrieved. The model was designed using the custom software tool that I wrote, based on pre-processed images taken from the

[5]Dr Lucy Butler, Department of Maxillofacial surgery, John Radcliffe hospital, Oxford.

Female Visible Human Project. The following list presents the additional steps that I took to validate and update that original model:



Figure 3.12: MIP rendering with shading of the Visible Human Project based tongue and extrinsic muscles atlas.

- I compared and revised my model with the publicly available model from the website of Tricarico ([WT06b]), which, however, is provided as a dense point cloud of finite element control points, and also with drawn models from anatomy books. To do so, I first had to visualize the volume data. There are two main methods for doing so, and I followed both: a voxel volume can be visualized either by direct volume rendering or through the rendering of its polygon iso-surface, which follows the contours of the objects within. A direct volume rendering approach is the Maximum Intensity Projection (MIP) method, in which parallel rays are traced from the viewpoint to the plane of projection, where the object(s) lie, and the maximum value of voxels along the path of the ray is displayed. This is a computationally fast technique that provides a relatively adequate sense of depth of the original data, and was used to find the relative positions of the tongue and extrinsic muscle components in the dataset. I wrote an MIP renderer and integrated it into the annotation tool, as a real-time feedback/visualization helper function to assist the trial-and-error process of building the model.

Figure 3.12 shows a rendered screen-shot, with added depth shading to improve depth cues. To create and display the iso-surface polygon mesh from the voxel-space volume, I used the marching cubes technique, which is referred to in chapter 7. The mesh (Figure 3.13, visualized on coronal and sagittal plane) was compared to the 2.5k point cloud model from Tricarico, which was rendered as an overlay using thick pixel billboards. As such, gross disparities between the two models could potentially be observed and corrected by referring to a third model, this time from anatomy books ([JB05] and [Net06]) and the 3D renderings provided by Sanders et al. [SIA02].

- Further to visualization, the extraction of the polygonal mesh of the tongue was useful in that it was then possible to adjust the 3D model using one of the sophisticated 3D software modelling suites available, such as 3DStudio or Wings3D. These tools support per-vertex manipulation, a vast library of possible geometrical transformations, measurement tools (for metrics such as distances between vertices, volume of object etc.), and an elaborate WYSIWYG[6] graphical environment, which simplified the process of updating the model. Once the editing of the objects was finished, it was a straightforward task to render the 3D model into a more useful voxel-space object again, by using a custom ray-casting application and scanning through the objects for each slice of the volume. I developed both paths, from voxel-space into 3D mesh and vice-versa, so that I could update the model in a 3D modeller. However, the latter path was not much used, as the original model I got using my own design tool was sufficiently precise, and hence needed only minor corrections.

A direct visual comparison of my final model, as presented in figure 3.12 and 3.13, with the background literature model as presented in figure 2.3,

[6]Acronym for What You See Is What You Get. It is used in modern editing software to describe a system in which displayed content is almost identical to the end product.

would confirm the similarities between the two models. In my model, the Genioglossus and Styloglossus/Hyoglossus muscles can be seen clearly, the Genioglossus forming the bulk of the central portion of the model at the front of the tongue, while the Hyoglossus and Styloglossus, one almost indistinguishable from another[7], travel over the sides and along the length of the tongue.

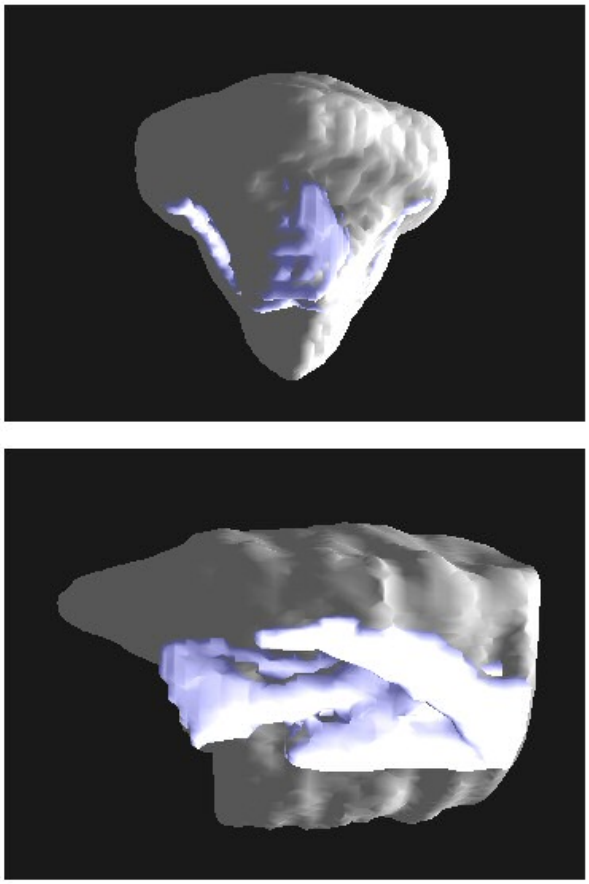


Figure 3.13: The 3D rendered models of tongue and extrinsic muscles superimposed. Top image in coronal plane, bottom in sagittal.

[7]The two muscles should separate near their point of insertion. This is barely visible at the very last images of the atlas.

3.3.6 Summary

This section presented the method that I followed in order to create the tongue musculature atlas that is used further in this project. The method is completely manually based: the annotation of muscle type was made with the assistance of the anatomist, who relied on visual information provided by reconstructed images from the female Visible Human Project data. The model was then validated and updated by comparing it to existing models from literature in various forms (point cloud, 3D renders of triangular meshes and parallelepieds, drawings). Figure 3.14 illustrates a selection of coronal images of the tongue atlas (colour-coded gray) and the extrinsic muscles (colour-coded pink) superimposed.

The next step is to provide all the necessary information concerning the second volume that is used in this experiment. Unlike the atlas, which is a fixed, gold standard and is unique throughout the experiment, this volume is retrieved once for every subject using MR imaging.

3.4 Retrieval of tongue from MR scans

3.4.1 Description of the MR sequences

As it has been already mentioned in sections 2.4.4 and 2.6.3, clinicians use both T_1 and T_2 weighted images to gain insight into anatomy and pathology respectively. For tongue cancer it is common to use T_2 to define the boundaries of the tumour within the tongue, which in general is well delineated because of the signal difference between the tissue.

For T_1 and T_2 weighted imaging of the tongue it is common to use the coronal plane, a matrix size of the order of 256×256 to 512×512, with an FOV of 18 to 24 cm and a slice thickness of the order of 2 to 6 mm with a 0 to 1 mm interslice gap. The sequences usually employ Fast-Spin Echo (FSE) and Fat Saturation (FAT). I used these settings, which are described in more detail below, for all scanning of volunteers. Figure 3.15 shows a set of T_2 MR

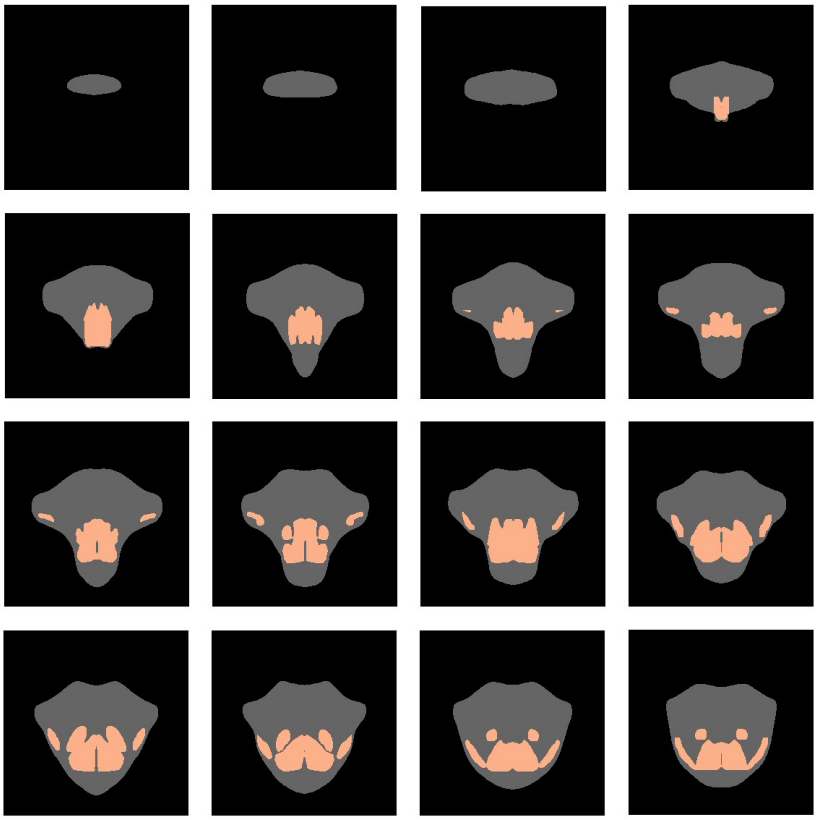


Figure 3.14: 16 colour-coded images of the Visible Human Project based tongue atlas in coronal plane.

images acquired using these settings. The same settings are also employed in standard clinical examinations, as will be seen in chapter 8.

3.4.2 Shortcomings of the MR images

However, one problem common to T_1 and T_2 weighted images of the tongue is that is usually difficult to delineate the boundaries of tongue and palate. This is important for automatic or semi-automatic segmentation of the tongue. In order to overcome this problem I have developed and used a new technique to enhance the tongue imaging using cavitatory enhancement with Gadolinium contrast agent suspended in a pectin solution to give adherence to the mus-

cular surface. The following section describes the experimental procedures of this technique.

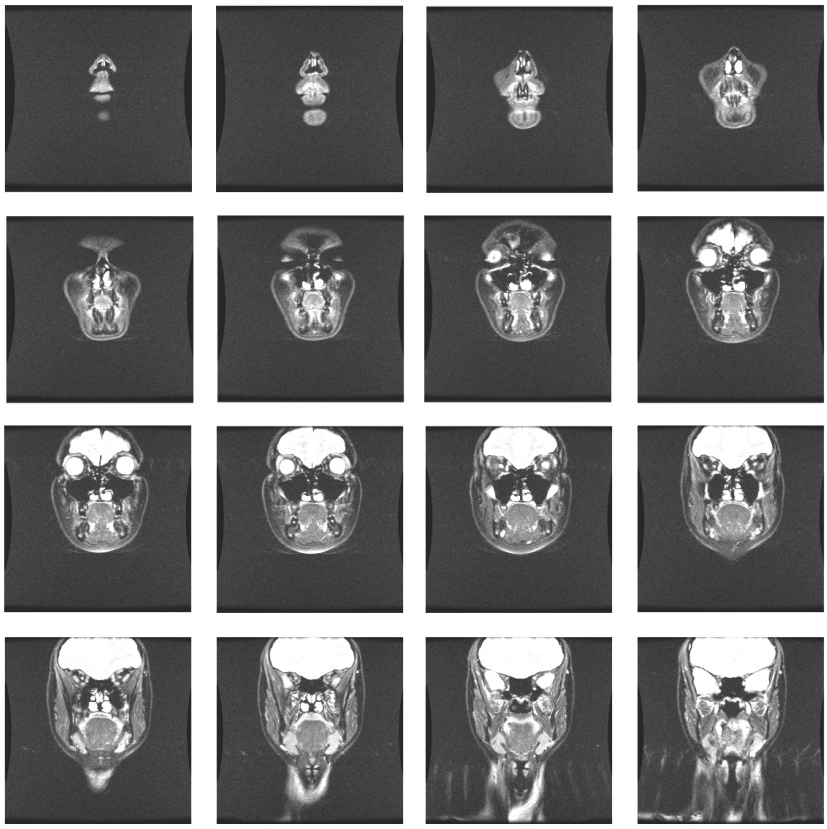


Figure 3.15: 16 Coronal MRI images of the head of a subject (Lucy), scanned with T_2 weighting.

3.4.3 Experiment of image enhancement using Gadolinium

The objective of image enhancement in the context of image acquisition is to improve the clarity of the scanned image, by increasing the spatial resolution or Signal to Noise and Contrast to Noise ratios, thus making the displayed structures more straightforward to locate and segment. However, it is not possible to change one of these factors in the MR sequence without affecting

the others, if the scanning equipment is fixed. The influence of the different sequence characteristics on the image quality is referred to in more detail in section 2.5.3.
The sequences employed in this experiment and described in the previous subsection are two standard sequences that have been used extensively in clinical cases of tongue cancer. The sequences balance these factors in order to provide images of adequate quality, without relying on unattainable image acquisition times and unrealistic slice thickness.
An alternative is to rely on hardware with different characteristics: a 5T scanner, for example, has the potential to improve all these factors simultaneously. This concept, however, is beyond the scope of this experiment, because of limitations of resources in hardware, and also because I considered that it would be better if the project could work without relying on rare equipment.

With the sequence and hardware locked, the only other method that could be employed is to use a contrast agent during the scan. This method is commonly applied in radiography, with the aim of improving the contrast of the areas where the agent resides. In this experiment, I explored the value of application of a contrast agent during the scan in order to improve the delineation potential of the surface of the tongue on the MRI images.

The use of Gadolinium in pectin

Before the scan, the tongue of the subjects was covered in a solution of contrast agent. A general description of contrast agents and their uses has already been given in section 2.4.3. The agent that I used was Gadolinium and the solution was based on pectin, which is an heterogeneous grouping of acidic structural polysaccharides that occurs in ripe fruits and vegetables, and can be found in bakery jams. The solution was then externally administered over the dorsum and anterior tongue in order to aid the contrast between the boundaries of soft tissue, the hard palate and surrounding air

pockets.

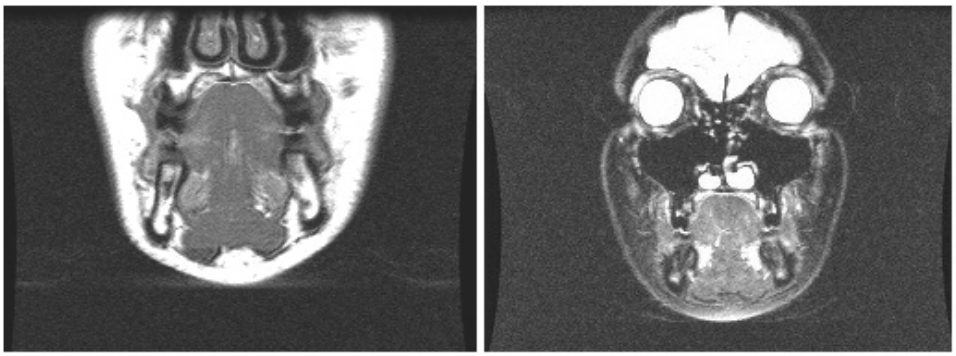


Figure 3.16: Effect of contrast agent on contrast of dorsum. Left image is T_1 weighted (Matthew), right image is T_2 weighted (Lucy).

In my experiment, I decided to prepare the solution by following the instructions given by the manufacturer of the Gd[8] which recommended mixing 0.2 mmol of Gd per kg of subject. The optimal density of the Gd in the solution, which was based on apple jam, was found to be 0.1ml in 20ml of jam, after trying different values starting with 0.05ml in 20ml and going up to 0.2ml in 20ml of jam. The optimal density was found by scanning test tubes filled with different solutions, while keeping the sequence locked, and observing the intensity patterns on the images.
The solution of Gd in jam was a viscous fluid that was easy to apply as a thin film (1-2 mm) and remained in place during the long acquisition times of the sequence. The only problem was that it had to be kept in relatively warm temperatures (over $30°C$) otherwise the pectin set. This meant that the fluid had to stay in the mouth for a period of time (5-10 minutes) before the scan.

[8]Gadoteridol (Prohance, Bracco, Spain).

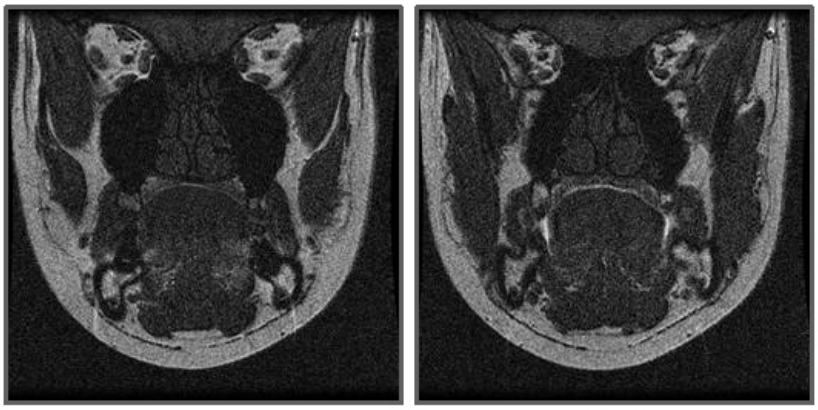


Figure 3.17: T_1 weighted scan of Lucy before and after contrast enhancement.

Experimental procedures: scanning sequences and subject

This experiment was performed twice with two different subjects, all volunteers. The first was a Female (Lucy) in her mid-20's, and the second a Male (Matthew), also in his mid-20's. Both had no contra-indication to MRI.
The scans were done on a 4 channel 1.5 Tesla Superconductive G.E. scanner using a neurovascular phase array coil. Tables 3.1 and 3.2 describe the scanning sequence used for both subjects. The scanning sequence characteristics are typical of the ones used for T_1 and T_2 weighted clinical scans. The projection plane used was sagittal in both cases.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_1 SE ET: 0 TR: 700.0 ms TE: 9.0 ms
Matrix Size	512 × 512
FOV	18.0 × 18.0 cm
Slice thickness	2 mm, 0 mm spacing

Table 3.1: Scanning sequence characteristics for scan of Matthew.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_2 FSE FAT ET: 12 TR: 4500.0 ms TE: 69.5 ms
Matrix Size	512 × 512
FOV	24.0 × 24.0 cm
Slice thickness	2 mm, 0 mm spacing

Table 3.2: Scanning sequence characteristics for scan of Lucy.

3.4.4 Findings and evaluation of the experiment

Figure 3.16 shows two images from the Matthew and Lucy scans. For both T_1 and T_2 images the contrast between the dorsum and the palette is visibly improved over similarly weighted images[9] that do not employ contrast enhancement. Realizing the potential of cavitatory contrast enhancement especially on T_1 weighted images, I repeated the T_1 sequence with Lucy (using a smaller FOV of 14.0 × 14.0) and achieved the best result in terms of delineation, which can be seen in Figure 3.17 (before and after the application of the solution).

By using this method it is now possible to devise an automated and semi-automated image processing algorithm to attempt to find the position and size of the tongue, based on the high contrast pattern that is observed on the images. This algorithm is described and used in more detail in chapter 5. On the other hand, the disadvantage is that the scan becomes more time consuming, since the solution must be prepared and applied with care. Nevertheless, the benefits of using this method are significant to the extent of making it almost impossible for many of the segmentation algorithms described in Chapter 5 to function as expected without the contrast agent. For

[9]Common coronal T_2 images of the head that do not employ this concept can be seen in chapter 8.

this reason, all of the non-clinical experiments[10] performed throughout this text employ this method.

3.5 Significance of findings

This chapter signifies the starting point of the experiment as a whole: the methods that I describe produce the raw data that will be processed further in the next chapters. While the format of the two datasets produced is the same (a 3D volume of scalar values), the domains are different:

- The atlas based on cryo-photography, using the Visible Human Project data, is fixed. It was formed using a wealth of cross-reference material from past and recent studies. While all precautions were taken to ensure that the atlas is as realistic as possible, there is still the possibility it may be proved to be erroneous in parts. That could happen either because of faults in the original design or because information about the tongue anatomy will become available that contradicts the current models. Regardless, I assume that the current atlas is the gold standard for the rest of the experiment, and even if a possibility exists that more accurate models can be constructed now or in the future, that does not interfere with the objectives of the project as a whole. In fact, a more accurate model can be easily interchanged with the one constructed in this chapter, even if it exists in a different format (for example, as a 3D mesh, or point cloud).

- The scanning procedure involves MR imaging and has to be repeated for every different subject. I tried to follow the sequences that can be produced with a standard 1.5T scanner, without resorting to exotic solutions that require, for example, specialized coils, or high field strengths. The only non-standard method I use is the application of

[10]The clinical-only scans that were used and whose sequence and procedure was out of my control are the exception. These experiments are referenced in Chapter 8.

the contrast agent, which is also not critical for the experiment as a whole.

The methods I followed to acquire the data are illustrated in the flow diagram in figure 3.18.

Having acquired the data, the next step is to extract the anatomical features relevant to the project that are depicted on the images within the volumes. This step is explored in section 5. The next chapter, however, sidesteps from that path and investigates a potential post-processing solution to the problem of relative anisotropy of the data. The problem is explained in detail in the introduction to that section.

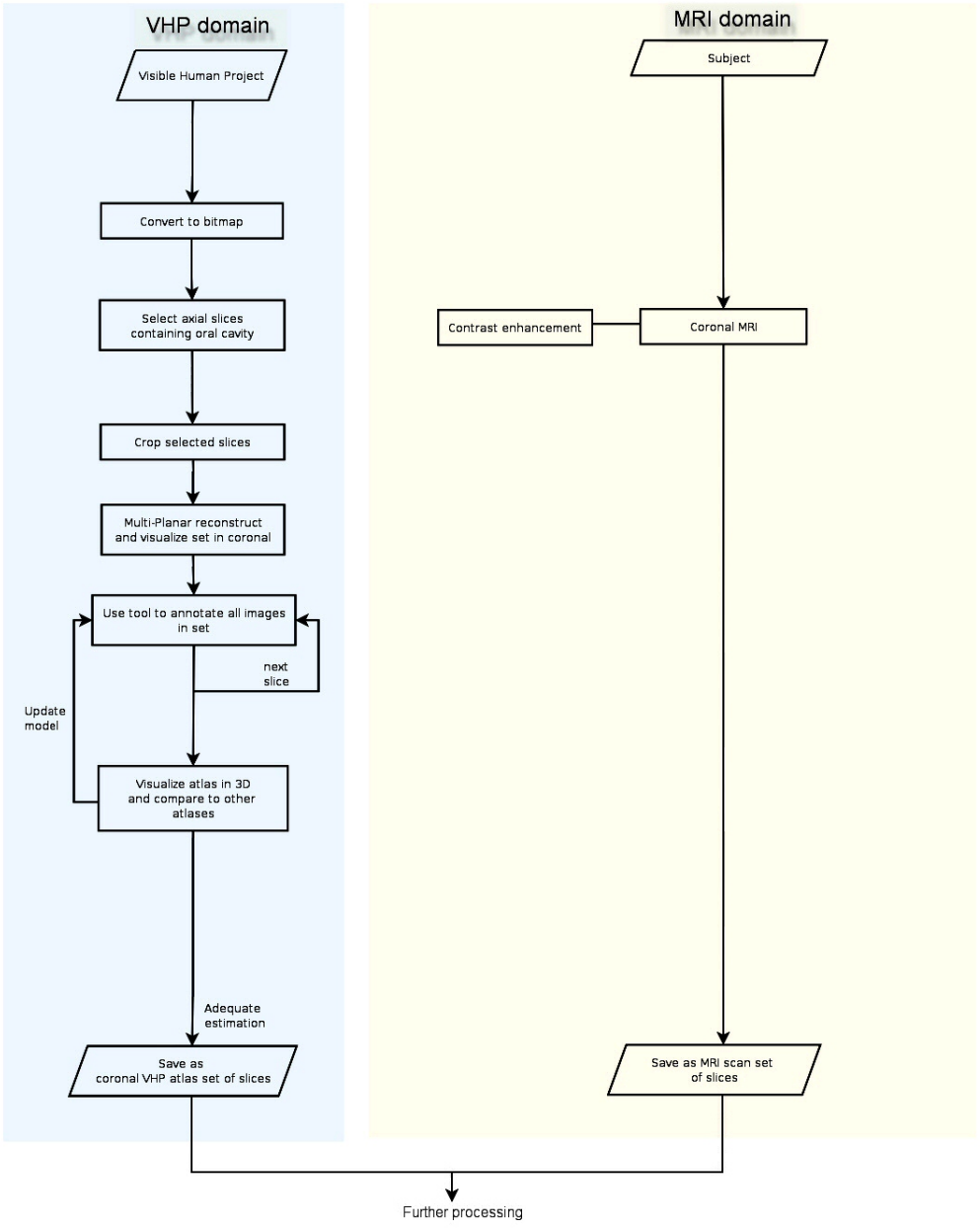


Figure 3.18: Overview of the work-flow of chapter 3.

Chapter 4

Coplanar anisotropic volumes fusion

4.1 Introduction

This chapter investigates a potential solution to the problem of working with volumes of anisotropically sampled voxels in medical imaging. While several solutions are possible, the chapter describes one that was tried initially but was not pursued further, for the reasons stated below. This can be regarded as a separate small project. As will be shown, the outcome of this chapter does not significantly influence the operation of the project as a whole, but it represents one possible method of handling the MRI data derived from the previous chapter (chapter 3) and processing them for use in the next chapter. This approach was not followed in this thesis for the reasons stated but does provide one viable alternative approach.

4.1.1 Requirement for isotropy

The ability to extract an accurate, dense, 3-dimensional structure from a given MRI volume is mainly dictated by the signal and contrast to noise ratios, and the combination of in-plane resolution, slice thickness and matrix size. The main criterion for image quality is the signal intensity and Signal

to Noise ratio (SNR), which increase in proportion to voxel volume, the number of phase encodings and number of acquisitions. However, as the voxel volume increases, the amount of information that can be extracted per cubic unit of space is proportionally reduced. Additionally, larger voxel sizes lead to sampling artefacts such as aliasing, as the sampling rate violates the Nyquist criterion[1].Reducing slice distances, on the other hand, introduces cross talk between slices that affects contrast. An interleaving sequence can compensate somewhat for these artefacts but produces significant intensity differences between adjacent slices.

4.1.2 Fusion of coronal and sagittal volumes

A solution to this problem would be to construct an isotropic MRI volume from the combination of two anisotropic volumes, in such way so that both **textural** and **structural** information is preserved. Figures 4.1 and 4.2 present two sets of high resolution slices, with significant slice thickness, as viewed on different planes, sagittal and coronal.

Structure, in this context, implies the collection of objects within the volume that are made out of connected voxels, and can be extracted through a shape-based feature extraction mechanism. These objects are delineated from the surrounding environment, and describe anatomical features such as, for instance, the mandible. Texture, on the other hand, is the distribution of intensity values within the structures that can be evaluated to characterize the type of tissue. The tongue, for example, as viewed in the coronal plane, consists mostly of pixels that portray texture - there is no indication of structure within the tongue, only a collection of pixels at an almost fixed level of intensity. However, the contour pixels of the tongue, if delineated

[1]Also known as the Shannon's sampling theorem, which states that for lossless digitization and reconstruction of the original signal, the sampling rate should be at least twice the signal bandwidth. If the criterion is not satisfied, the frequencies will overlap, causing a distortion called aliasing. Aliasing is prevented by increasing sampling rate or introducing anti-aliasing filters.

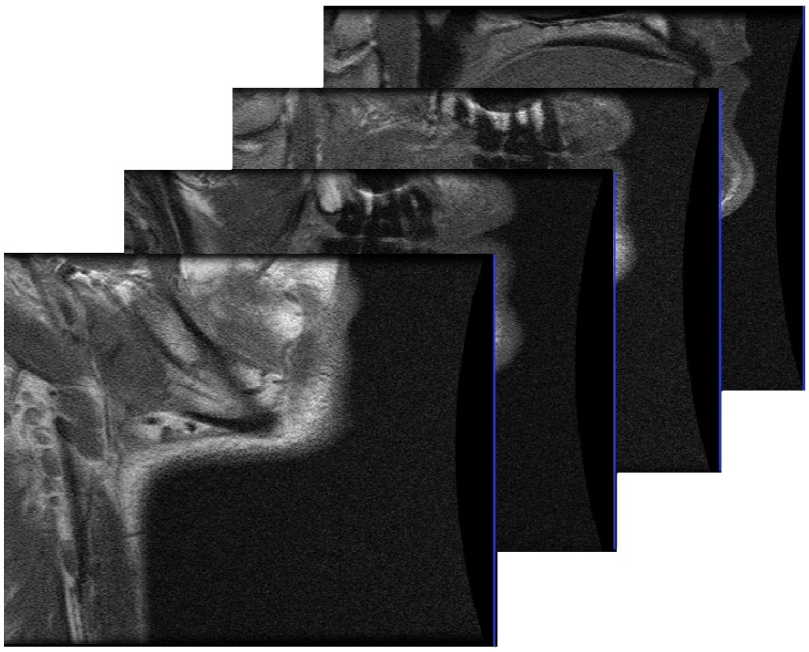


Figure 4.1: High resolution slices in sagittal domain (Matthew dataset, T_1 weighted).

sufficiently, can be assigned as carrying "structural" information.

The distinction between texture and structure can be very roughly approximated through a different path: assume that the MRI volume is viewed in slices on the coronal plane. In this mode, a solid object (such as the mandible) is viewed in cross-section. As the 3D object changes shape along the coronal direction, so do the 2D projections of the object on the cross-section plane change appearance, shape or position. If the coronal direction was time, it would appear as if the shapes were moving on the plane. On the other hand, the same cannot be said for a large textured area, which, for the most part of it appears to be rather static, with only the texture varying slightly over adjacent slices.

If this "motion" information is retrieved for every voxel in the volume,

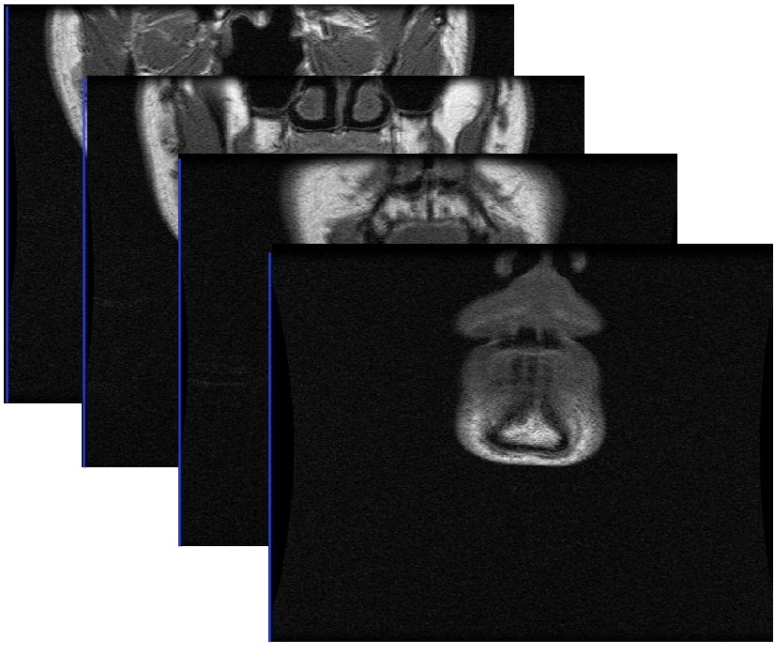


Figure 4.2: High resolution slices in coronal domain.

according to its displacement over two consecutive frames in a set projection, then it can be used as a weighting factor in the following scenario: there are two anisotropically sampled volumes of the same subject. The first one has high plane resolution and a substantial slice thickness. The second one has low-resolution images but the slice thickness is much smaller than in the first volume, and hence the resolution in the direction of projection is higher. The high-plane resolution volume is associated with textural information - high resolution texture has to be preserved. The volume with the high projection resolution is associated with structures : the "motion"of the projected shapes of the 3D objects within the volume is sampled at a higher frequency. The drawback is that, due to reduced in-plane resolution, the projected shapes appear more pixelated.

Combining these two produces a third volume, in which the emphasis is given to one volume or the other depending on whether the voxels depict an

element of structure or texture. The generation of this volume is pursued in this chapter.

4.1.3 Organization of sections

The chapter is divided into sections that describe the theory necessary for the implementation of the fusion mechanism. In more detail:

- Section 2 describes the concept of an optical flow algorithm and what information it extracts from the scene it is applied to.
- Section 3 expands on the specific optical flow solution I implemented, which uses the Lucas-Kanade Flow.
- Sections 4 and 5 are the core of the chapter, where the mechanism of fusion, in spatial rather than temporal sense, is further explained. At this stage the description emphasises the connection of the optical flow with the objective of the anisotropic fussion of MRI images set at an earlier stage.
- Section 6 describes the single modality registration schemes implemented in order to fuse the two different scans correctly. The section does not expand on rigid registration; registration is a major topic that is covered in detail in chapter 6.
- Finally chapter 7 refers to implementation details, results and significance of the experiment to this project.

4.2 Image processing of optical flow

Optical flow is the dense velocity field that describes the motion - the direction and speed - of intensity patterns in an animated image, assuming that intensity is conserved during any displacement. It was first mentioned in a

seminal paper in 1981 by Horn and Schunck [BH81], and since then a plentitude of algorithms have been produced that attempt to calculate optical flow. Given a series of images that represent moving objects in a scene, an optical flow algorithm should be able to estimate:

- Which pixels belong to objects in the foreground (that are in motion) and which belong to background (static).
- The direction of motion.
- How fast the objects in question are moving.
- Whether the motion is linear or rotational

An **optical flow 2D vector map can be defined that has the same spatial dimensions X/Y as every image in the motion sequence. For every pixel in a sequence of two consecutive frames, the velocity vector describes the direction and speed of displacement. For static objects, the corresponding displacement vectors should be the zeroth vector.**

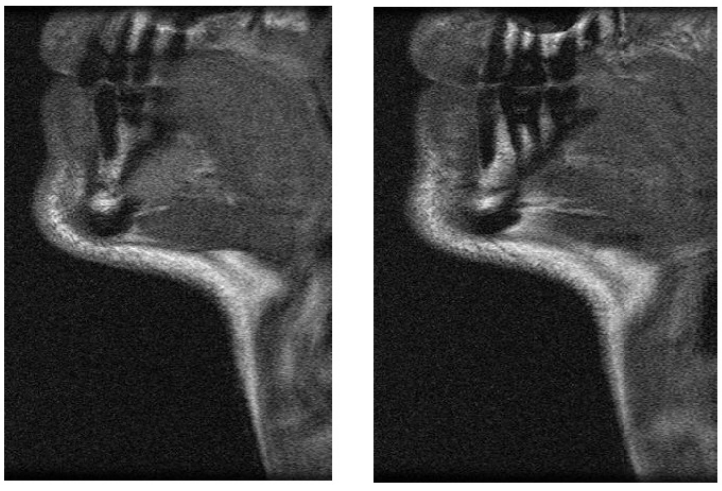


Figure 4.3: Sequential images in a sagittaly scanned volume.

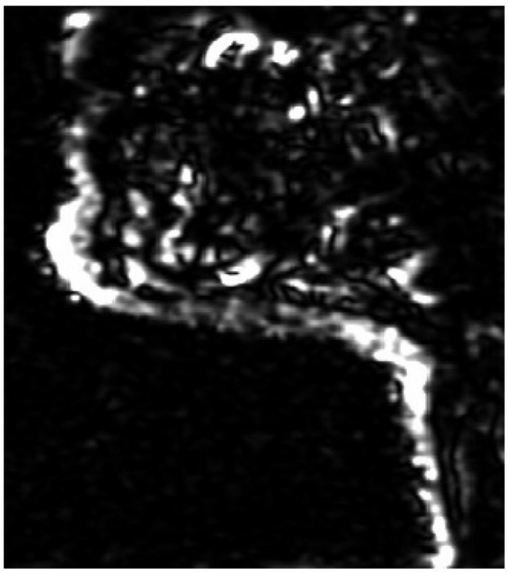


Figure 4.4: Optical flow: normalized displacement magnitude.

4.2.1 Optical flow constraint equations

In order to calculate the motion between two image frames taken at times t and $t + \delta t$ the assumption should hold that a pixel at location (x, y, z, t) with intensity given by $I(x, y, z, t)$ has moved by $\delta x, \delta t, \delta z, \delta t$. Hence

$$I(x, y, z, t) = I(x + \delta x, y + \delta y, z + \delta z, t + \delta t)$$

Since the movement is small enough, the image constraint at $I(x, y, z, t)$ can be expressed using Taylor series as:

$$I(x + \delta x, y + \delta y, z + \delta z, t + \delta t) = I(x, y, z, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial z}\delta z + \frac{\partial I}{\partial t}\delta t$$

The higher order terms of the Taylor expansion are negligible and ignored. From here, it follows that:

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial z}\delta z + \frac{\partial I}{\partial t}\delta t = 0$$

which is

$$\frac{\partial I}{\partial x}\frac{\delta x}{\delta t} + \frac{\partial I}{\partial y}\frac{\delta y}{\delta t} + \frac{\partial I}{\partial z}\frac{\delta z}{\delta t} + \frac{\partial I}{\partial t}\frac{\delta t}{\delta t} = 0$$

and

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial z}V_z + \frac{\partial I}{\partial t} = 0$$

V_x, V_y, V_z are the x, y, z components of the velocity vector. $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial z}, \frac{\partial I}{\partial t}$ are the corresponding derivatives in the x, y, z, t direction. If they are expressed as I_x, I_y, I_z, then the equation can be rewritten as

$$I_x V_x + I_y V_y + I_z V_z = -I_t$$

which, using the del operator, is

$$\nabla I \cdot \overrightarrow{V} = -I_t$$

This equation cannot be solved as it stands, because it has three unknown quantities. Another set of equations is needed with additional constraints.

4.3 Lucas-Kanade Flow

The Lucas and Kanade [LK81] solution imposes additional constraints by assuming that the flow field is smooth locally, hence the pixel's neighbours have the same motion vector. If the flow is constant in a small cubic window, the size of $n \times n \times n$ centered around a pixel at x, y, z then the set of equations expands to:

$$\begin{aligned}
\nabla I_{x_1} \cdot \overrightarrow{V} &= -I_{t_1} \\
\nabla I_{x_2} \cdot \overrightarrow{V} &= -I_{t_2} \\
&\vdots \\
\nabla I_{x_n} \cdot \overrightarrow{V} &= -I_{t_n}
\end{aligned}$$

or

$$\begin{bmatrix} I_{x_1} & I_{y_1} & I_{z_1} \\ I_{x_2} & I_{z_2} & I_{z_2} \\ \vdots & \vdots & \vdots \\ I_{xn} & I_{yn} & I_{z_n} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} = \begin{bmatrix} -I_{t_1} \\ -I_{t_2} \\ \vdots \\ -I_{t_n} \end{bmatrix}$$

or

$$A\overrightarrow{V} = -I$$

The problem now is that there are more equations than unknowns. The system of equations is over-determined. The solution is to solve by minimum least square[2] for:

$$\overrightarrow{V} = -(A^T A)^{-1} A^T I$$

being the compact form of

$$\begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} = \begin{bmatrix} I_x^2 & I_y I_x & I_z I_x \\ I_y I_x & I_y^2 & I_y I_z \\ I_z I_x & I_z I_y & I_z^2 \end{bmatrix}^{-1} \cdot (-A^T I)$$

In order to solve the system, $(A^T A)$ should be invertible and not be too small due to noise. A matrix inversion operation is needed and a method such as the LU decomposition [NCV85] or the Newton-Raphson approximation can be used. Also the derivatives of the original image in four dimensions X, Y, Z, t must be calculated. This is a trivial problem however, as they are estimated through the first-order differences between adjacent pixels.

Errors can be caused by the violation of the constant brightness and smoothness of velocity assumptions made at the beginning. This is a case for highly textured regions, a window size n too large or regions with additive noise. The solution usually given is

[2]Solutions for optimization problems in mathematics are referred to, in more detail, in chapter 6.

derived from local regions: the image is divided into smaller regions where the assumption of brightness consistency holds. However, for small regions with slowly changing gradients, the optical flow estimate becomes unreliable due to the lack of motion information.

4.4 Morphing along optical flow

Optical flow is a descriptor of motion in animated images. It is been used in studies of a variety of motions that include both moving and static objects (and observer), as a tool for collision prediction (in robotics) and as a mechanism to estimate the field of view and 3D Cartesian coordinates of moving shapes in a scene. The common theme in these implementations of the algorithm is the use of the time as the 3rd dimension. This is not a restriction: indeed, the 3rd dimension could just as well be spatial, so that the information consists of a 3D volume rather than a sequence of 2D images in time. The optical flow mechanism in this case would automatically attempt to determine the locations of feature points across each projected plane of the volume and measure how each pixel has to move from one plane into the next, an operation akin to 'wrapping' one image into another [EP97].

As an example, assume that the volume describes an MRI scan of a subject. The volume is quantized in all 3 axes, $X \times Y \times Z$. We can apply the optical flow between pairs of projected images $Im[n]$ and $Im[n+1]$, Im being a 2D image in the X/Y plane and $n \in [0, Z-1]$. We also assume that the Nyquist-Shannon sampling theorem is satisfied between the planes (that is, the sampling rate along Z is sufficient so as not to cause motion aliasing artefacts) and that the features change smoothly from one plane to the next. Ideally, the optical flow calculation would yield a per-pixel dis-

placement vector map for each of the $Z-1$ pairs of images. A tri-linear/cubic interpolation scheme can then be employed to calculate any number of intermediate slices (the same process can be repeated for the X/Z and Y/Z plane as well).
In practice, the optical flow vector maps are not very accurate for displacement and wrapping. The 'motion' smoothness assumption does not always hold true as the spatial gap between slices increases. Also, the wrapping mechanism itself, even if based on a well-posed displacement map, does not guarantee a realistic transformation. While the direction of the displacement vectors is of little practical use, however, the magnitude can indicate the rate of change of the perceived morphing of one structure into another. The per-pixel scalar value provides a measure of "motion activity" that can be used as a blending weight in a different scenario:

4.5 Fusion of multi-planar anisotropic volumes

There are two volumes V_1 and V_2 of MRI data of the same scanned subject. Both are anisotropic in perpendicular directions. The dimensions of the volumes at $X/Y/Z$ axis are $n \times n \times n/m$ and $n/m \times n \times n$, with $m > 1$. The question is how is it possible to mix information from both volumes so that a third volume V', with a resolution of $n \times n \times n$ is generated, with each pixel assigned according to the following principles:

- If the structure to which it belongs to is altering significantly over the next projected slices (without loss of generality, assume that being the Z axis), then retrieve value with a bias towards the volume that provides with the best spatial resolution at X/Y in order to retain texture and fine structures.
- If there is significant motion, then the value of the pixel is

affected more by the volume with the high resolution in the Z axis.

The normalized weighting factor $\kappa \in [0, 1]$, calculated per pixel of the volume, is used in:

$$V'[x, y, z] = V_1[x, y, z] \cdot \kappa + V_2[x, y, z] \cdot (1 - \kappa)$$

with volumes V_1 and V_2 re-sampled into isotropic volumes $n \times n \times n$ using an 1-D interpolation scheme (linear or cubic) along the direction of the axis with the least resolution. Factor κ is dependent upon the magnitude of optical flow vector $\overrightarrow{F}$ at $[x, y, z]$, scaled and offset:

$$\kappa = \min(1, \max(0, \alpha \cdot \left|\overrightarrow{F}\right| + \beta))$$

α and β are defined globally. There is no mechanism to estimate α and β, which have to be set manually, according to image types - a process akin to brightness/contrast windowing employed in images in radiology. The 'windowing' of α and β can be in real-time, as the optical flow map - a time consuming process - is only calculated once and kept in a 3D buffer of scalar values. Figure 4.4 shows a map of the per pixel magnitude of the optical flow, normalized to within 0 (black) and 1 (white). The high intensity components track the jaw-line, while the low intensity components show the regions that are either empty or contain slow-varying structure. Again, "slow", in this context, implies displacement in a spatial dimension rather than time.

4.5.1 1D Magnification filtering

As I re-sample the volumes from a spatial resolution of $n \times n \times n/m$ into $n \times n \times n$, I employ a "Magnification filter" in order to stretch

the volume in one dimension by an order of m. The "nearest neighbour interpolation" scheme is the most basic method. It works by assigning each new pixel value as equal to its nearest neighbour. If the adjacent pixels are P_1 and P_2, then the new in-between pixel value P' spaced between P_1 and P_2 at normalized distance $\theta \in (0, 1)$ is P_1 if $\theta < 0.5$ and P_2 if $\theta > 0.5$. Nearest neighbour interpolation is computationally efficient but not as accurate as linear or cubic interpolation.

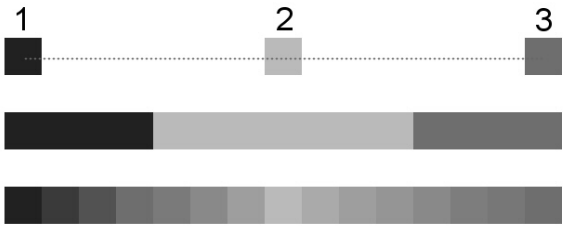


Figure 4.5: Magnification in 1 dimension. Top row: the 3 original pixels. Middle row: after nearest neighbour interpolation. Bottom row: after bilinear interpolation.

In linear interpolation, P' is set to construct a smooth gradient between P_1 and P_2, and defined as:

$P' = P_1 + \theta \cdot (P_2 - P_1)$ **A visualization of the two operations can be seen in figure 4.5.**

Higher order piecewise polynomial based interpolation (like the third order Hermite [ea98a]) or non-linear low-pass filter schemes with an intensity dependent filter kernel [ea02a] would often yield sharper images, and a slight visual improvement over linear interpolation. However, this comes at a cost, since the complexity of the algorithm (and hence the time spent for each calculation) increases dramatically. In reality, m is rather small, of the order of 4, and therefore only 4 extra pixels have to be generated for every

pair of pixels in the magnification process. The difference between linear and cubic becomes too small to be noticed visually, and so the linear filtering is used for its better balance of image quality and speed.

4.6 Single modality rigid registration

The acquisition of the two volumes, while it is of the same subject and of the same modality, takes place at different times. Hence the datasets are not necessarily rigidly coordinated with each other. In order for any comparison and fusion to be valid, the data sets must be aligned so that the same structures under both acquisitions are overlapping. One dataset has to be fixed (target) while the other (source) is transformed into the first. This calls for a rigid registration scheme. Registration criteria fall mainly into three categories:

- Segmentation-based: binary structures are rigidly aligned
- Landmark-based: markers (such as, but not exclusively, control points) are manually or automatically selected and matched
- Intensity-based: registration reconstructs source so as to minimize intensity differences with target.

Woods [Woo00] provides a detailed explanation on rigid transformations involving translation, rotation and scaling. Thirion ([J.P96]) suggests a landmark-based method based on the extrema of the crest lines. However, these algorithms deal with the more complicated situations of multi-modality registration, and/or registration involving the full spectrum of affine transformations - translation, rotation, scale and shear. For the given single-modality problem, I applied a solution based on similarity-based methods.

Such methods use an image similarity metric, which quantifies the amount of correspondence between source and target images, after a given transformation is applied. An optimization algorithm then tries to minimize or maximize this value by choosing different transformation parameters. The procedure repeats until the best fit parameters are found.

In the datasets that I used the variance between the volumes was observed as being minimal, with no significant translation, scaling/rotation or elastic deformations. The reason is that the two acquisitions take place within a very short period of time - over the order of 2-5 minutes - during which period the subject is stationary and the FOV, slice thickness and matrix size fixed. Two different registration schemes were tested, and both produced satisfactory results for the given problem, as follows:

4.6.1 Rigid registration using cross-correlation

In signal processing, cross-correlation is a metric of the similarity of two signals. It is used as a tool for identifying a pattern within an image [L.G92]. The cross-correlation of two given images Im_1,Im_2 with resolution $X \times Y$, would produce a $X \times Y$ map C defined by:

$$C[i,j] = \sum_{y=-Y/2}^{Y/2} \sum_{x=-X/2}^{X/2} (\mathrm{Im}_1[x + \frac{X}{2}, y + \frac{Y}{2}] - \overline{\mathrm{Im}_1}) \cdot (\mathrm{Im}_2[i+x, j+y] - \overline{\mathrm{Im}_2})$$

with $\overline{Im_1}$ and $\overline{Im_2}$ being the mean value of the intensity of all pixels in Im_1 and Im_2. The peaks in C are the positions of the best matches, and dictate the amount of shifting that needs to be applied for a best match. Cross correlation of every image is performed with a number of images before and after in the sequence, so as to find a potential shift in the 3d dimension. A similar technique

can be used to check for slight global minification/magnification mis-alignments: the template image is rendered with its size varying in steps between 90% and 110% and cross correlated as shown. The magnification process (which required sub-pixel level accuracy) uses bilinear rendering, which is essentially a 2-dimensional version of the 1D magnification filter already described.

4.6.2 Rigid registration using phase-correlation

Phase correlation is another approach to the registration problem that works in the frequency domain in order to determine the relative translation between two images ([EDC87] and [WSH03]). Correlation is based on phase information, and is not sensitive to changes in image intensity and independent noise (unlike cross correlation in spatial domain). Phase correlation works in the following sequence:

- Calculate the discrete 2D Fourier transform of Im_1 and Im_2.
- Store conjugate of Im_2.
- Multiply together and normalize all elements of the Fourier transforms of Im_1 and Im_2, producing a normalized cross power spectrum map.
- Inverse transform the cross power spectrum and find peak value.

Likewise, the peak value determines the location of best fit. Figure 4.6 shows an example of using phase-correlation to find the relative translation between two images that have been corrupted by independent Gaussian noise. The peak (white pixel) on the phase-correlation spectrum indicates a relative translation of (20,20) pixels.

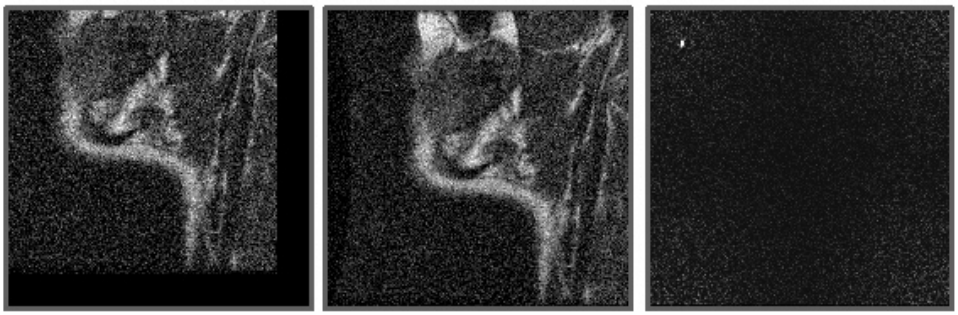


Figure 4.6: Use of phase-correlation to determine relative translation. Rightmost image shows the phase-correlation spectrum with peak.

4.7 Implementation and results

4.7.1 Software implementation

Optical flow was implemented on CPU with C and OpenCV, and also on a dynamic branching capable GPU (Graphic Processing Unit) in Cg and Assembly, by following the method suggested by Strzodka and Garbe ([SG04]). For the purposes of linear and bilinear interpolation, OpenGL and hardware texture mapping was employed. Re-sampling is much faster in hardware than in software, as it can be parallelized (the CPU is free to work on the next set of operation while the GPU is rendering) and also because it employs dedicated circuits that were designed for this purpose.

The registration of the volumes was executed on CPU only. For the phase-correlation, a forward and inverse discrete Fourier transform using the Fast Fourier Transform (FFT) algorithm was implemented, based mainly on [FJ05] and [FJ98].

The rendering of the slices of the volume was finally treated wholly on a pixel shader (A GPU program dedicated to operate on pixel level), that had access to the two volumes of data, the volume of pre-calculated displacement intensities and the α and β

windowing scalars.

4.7.2 Results

Scanning procedure

The fusion mechanism was evaluated with head scans of two different non-clinical subjects (Lucy and Matthew).
The first scan was a test that helped to calibrate the sequences, as the radiographer attempted to achieve the best image quality for the smallest possible FOV, noise and slice thickness. What was noticed in the first test was that motion artefacts not only degrade the image quality but are also a source for misalignment problems between the two volumes. As a result, extra care was taken to reduce the motion artefacts in the second scan, by asking the subject to keep as still as possible. However, because this procedure requires twice the time of the routine procedure, this was not very comfortable for the subject, and additionally increased the risk of artifacts from involuntary movement.
After acquiring the two sets of images, they were then fused using the mechanism described in this chapter, and displayed or tested with a segmentation algorithm used in chapter 5.

Fusion of the two volumes

For the structures surrounding the tongue, the optical flow map seems to correctly identify the presence of components that alter significantly over consecutive slices. As viewed from the coronal plane, the inside of the tongue has a low flow intensity while that of the surrounding tissue is higher. The rigid registration of the two volumes is also a very straightforward operation, and assuming that the subject has tried not to move much during the scan, it gives satisfactory results, as the peak value in the cross-correlation

and phase-correlation spectrum map is clearly evident, and lies very close to point $(0, 0)$. The latter signifies that the displacement is minimal, and even in the absence of any registration scheme, the procedure would work within an acceptable margin of error.

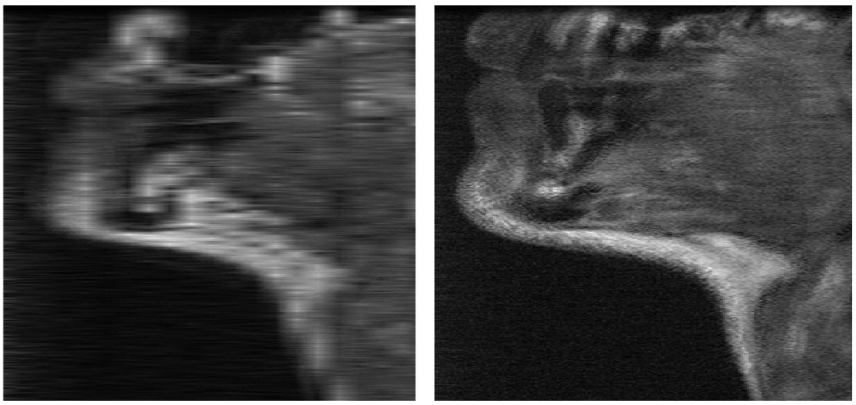


Figure 4.7: A slice of an anisotropically sampled volume. Low and high spatial resolution domains.

It must be noted that this procedure relies on the radiographer to define the correct parameters for the FOV and slice thickness with high precision so that the multi-planar images are matched. Failure to do so would result in volumes that cannot be co-registered by any of the mechanisms described in section 4.6. Figure 4.7 represents two sagittal images from two scans of the same subject sampled anisotropically in perpendicular directions. The left image is originally scanned in the coronal plane and is reconstructed through MPR. As such, it has visible aliasing artefacts. The right one was captured in the sagittal plane and has a resolution that is sufficient to capture texture and finer details, but represents a slice of significant thickness. The merging of the two (Figure 4.8) produces an image that retains the most useful information with a minimal virtual slice thickness (which is not possible to visualize on paper). As expected, there are some aliasing artefacts also present

in this image. However, these artefacts are much less prominent than in the low resolution image of figure 4.7, or in an image that is a blend of the two images using a fixed, global factor for κ.

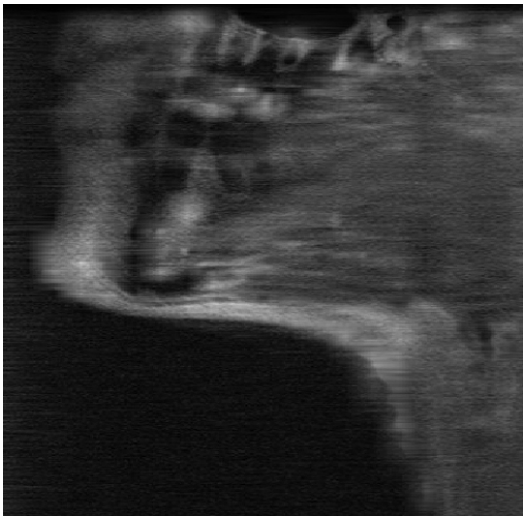


Figure 4.8: Fusion of the two images based on optical flow calculations.

Efficiency of software

In terms of speed, both tested optical flow implementations, on the CPU and GPU, performed adequately for the needs of the experiment, with no significant difference in speed, yielding an acceptable average of 15 slices per second for a slice resolution of 256×256.

4.8 Discussion of Results

In the two cases I tested, the fusion of the two volumes makes for a much clearer visual representation of the anatomy, something which can potentially assist the clinician/anatomist in finding the contour of the structures and segmenting the tongue. This is due

to the extra slices inserted between the thick consecutive coronal slices that can be perceived as smoothing out the geometrical gap between the contours of the structures they describe. As such, the tongue is visually perceived as a surface of revolution that is continuous in the coronal axis, rather than heavily discretized.

However, when tested with a fully automated segmentation scheme, such as an Active Contour initialized using a template shape[3], there is no significant improvement or deterioration over anisotropic volumes acquired using routine procedures. This is mainly due to the existence of some aliasing artefacts inherited from the low resolution images that introduce "steps" in image intensity that the active contours cannot always cope with, as their operation is sensitive to the presence of continuous intensity gradients. Additionally, the operation of active contours depends on the delineation potential of the contours of the objects they attempt to segment. Unfortunately, the aliasing artefacts produced by the fusion of the two volumes are more apparent around the edges of displaced structures, where there is naturally more displacement activity. By altering the variables α and β it is possible to improve the segmentation potential, but there is no automated mechanism to find the best fit. The anisotropic fusion mechanism and an automated segmentation scheme such as the active contours can work together only if there is a higher level interaction with the user, who visualizes the results, evaluates and manually controls the operation based principally on the augmented visual information provided by the mechanism.

There are still two important disadvantages with using this non-routine procedure. The first one is related to the management of the scan. The scan time has to increase by a factor of 2, which

[3]For a complete description of the scheme, implementation and relevance to this experiment, the user is referred to the next chapter, chapter 5.

results in extra effort from both patient and radiographer. As a consequence, additional motion artefacts may be present near the end of the scan, as it is usually difficult for the subject to keep still during that extended period of time. Any motion artefacts would undermine significantly the quality of images and the co-registration step.
Furthermore, if the experiment as a whole relied on this procedure, then the experiment would not be potentially applicable to previous examinations that were taken using routine scans.

4.9 Conclusions

This chapter describes a concept for improving the delineation/segmentation potential of medical images using a combination of software and hardware techniques. The methods I followed are illustrated in the flow diagram in figure 4.9.

It has been argued that, while it is not suitable for pairing with automated segmentation schemes, it can provide useful assistance to an anatomist. However, one needs to consider the fact that technology progresses and the advent of more powerful MRI scanners, with potential for higher resolution scanning and new sequences, would soon render the concept of fusion as a means of extracting more information out-dated, or irrelevant. For these reasons, I chose to follow the alternative path, which is to use standard scanning procedures for the bulk of the experiment, and try out the concept of anisotropic fusion with non-clinical subjects only. It is unknown whether there is scope for more experimentation, with different procedures (for example a fusion of 3 anisotropic volumes) or different parts of the anatomy. However, it is common for hardware technology to catch up with research and render such

schemes obsolete[4]**.**

At this point I have established that standard clinical procedures should be used for the creation of the volume in the MRI domain. What follows in chapter 5 is the description of the methods I used to extract the model of the tongue from MRI volumes that were acquired using these procedures.

[4]An example from the world of computer graphics is the Talisman concept by Microsoft. Introduced in 1996 as a wholly new graphics pipeline, it attempted to overcome certain limitations of the hardware of the era by using very unusual (and rather counterproductive) tricks to compose images. Within two years, the advent of powerful dedicated graphics hardware rendered the project irrelevant, as hardware could now compose real-time graphics effortlessly using the old techniques.

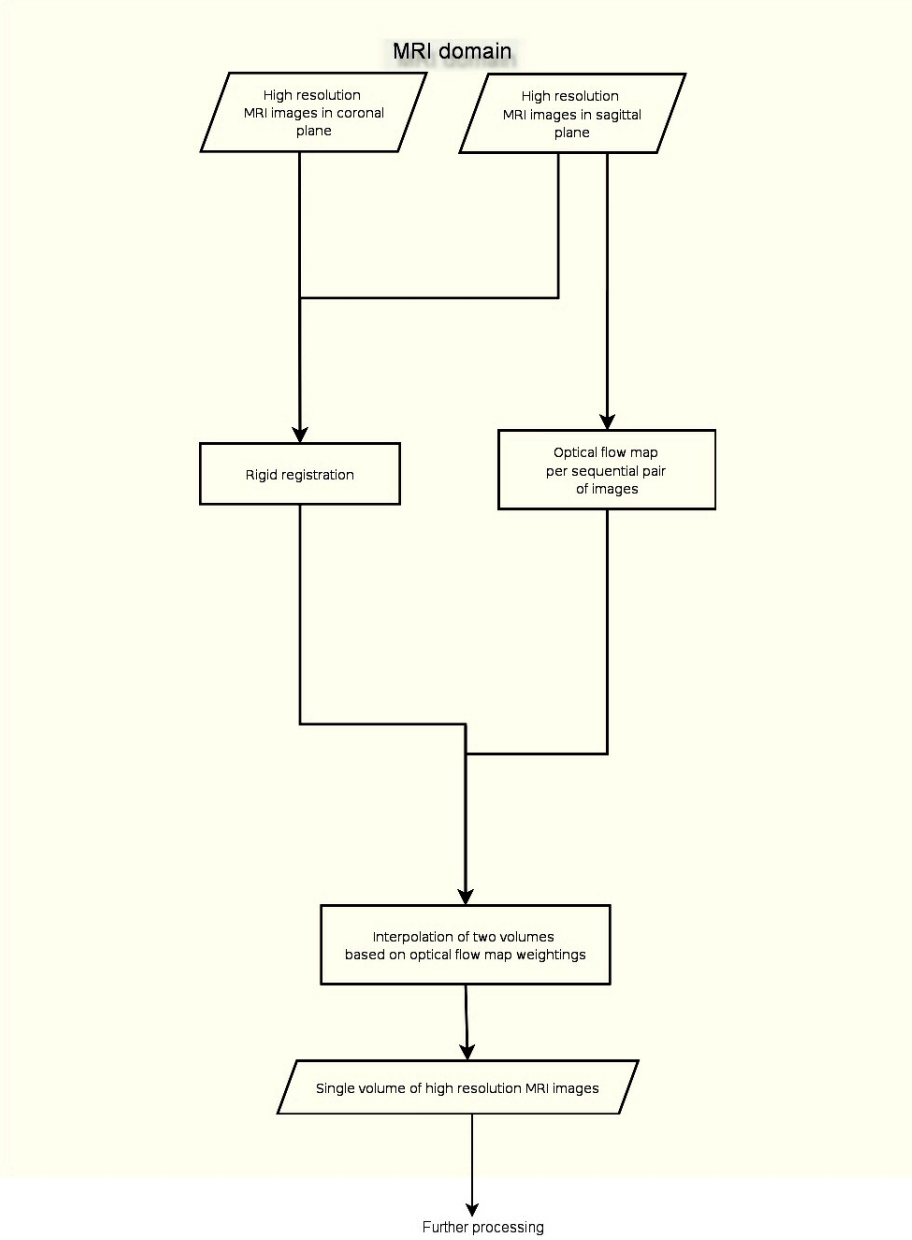


Figure 4.9: Overview of the work-flow of chapter 4.

Chapter 5

Automated extraction of tongue model

5.1 Introduction

At this stage in the project, there are two different sets of data available for experimentation: the Visible Human Project Atlas (referred to as VHP) and the MRI image acquisition, as described in chapter 3. These are both volumes of scalar values: for the MRI volume, they represent intensity values on consecutive coronal slices of the head, while for the VHP the values represent an ID number which identifies each voxel as "Air", "Extrinsic muscles" and "Tongue excluding Extrinsic muscles". What is now needed is a mechanism to identify and extract (segment) the image features that represent the physical objects necessary for further calculations (as will be described below in chapter 6).
These objects are:

- **For the VHP**: the contours of the tongue and extrinsic muscles expressed in the more simplified and manageable form of a chain code.
- For the **MRI**: the contours of the tongue as a chain code.

This establishes a common frame of reference between the two different domains and assists the correlation of the objects from the two different sets of data.
The purpose of this chapter is to explain the image processing concepts and implementation details of the tools that I have developed in order to retrieve the two sets of contours. In more detail, sections 5.2 and 5.3 refer to the background theory of feature extraction and image preprocessing. Section 5.2 analyzes the different mechanisms that are implemented, including the Canny edge detector, the Freeman chain encoding and, most importantly, Active Contours. Section 5.3 refers to the preprocessing that is applied to the images in order to assist the feature extraction mechanisms.

The chapter ends with sections 5.4, 5.5 and 5.6, where these mechanisms are combined and evaluated with the two volumes VHP and MRI of a volunteer. As a result of analyzing the options in the background, I decided to base my methods on the approaches described in the next sections.

5.2 Feature extraction mechanisms

One important mechanism used extensively in the field of image processing and robot vision is "feature extraction" [PBB93], which is a form of dimensionality reduction[1]. Feature extraction is normally used to detect the defined various portions or shapes of an image that have certain characteristics. Depending on the nature of characteristics, the following classification can be made [Mur99]:

[1] In statistics, dimensionality reduction is a common tool to either find and use a subset of the original features or map a multi-dimensional space into a space with less dimensions that usually capture the most variance. The goal is to categorize, visualize and simplify large sets of data.

- **Spectral features** such as colour histogram, tone, spectral index etc. [L.01]. Spectral feature extraction is used in many operations, for example to compute vegetation density based on multi-spectral scanner data from satellites. There is also a broad use in medical imaging, and many solutions incorporate examining spectral quantities. Threshold-based segmentation is one such example.

- **Textural features** that look into patterns of texture, spatial frequencies, distribution of intensity values (through the use of co-occur

- **Geometric features** that are associated with topological information, mainly edge based [MH80]. Other features used are corners [HL77], points of interest [H.P77] and curvature [HJ92]. A more extensive list is presented at [Den99a].

Feature extraction based on topological information is a very common approach for medical images like the ones used in this project. Spectral and textural features are normally not very prominent, or reliable, mainly because of the low Signal to Noise ratio, intensity inhomogeneities and limited spatial resolution that leads to aliasing problems. On the other hand, geometric features such as edges are more consistent and easy to track, as images have adequate Contrast to Noise ratio and features are sufficiently delineated from background. The next section introduces the concept of segmentation based on geometric features and expands on the methods used, the Canny edge detector, the Freeman chain encoding, and the Active Contours.

5.2.1 Segmentation based on geometric features

Feature extraction methods are commonly based on the geometric feature of edges. Edges are encountered in areas of the image where the levels of intensity fluctuate and are indications of object

boundaries. We can use these boundaries to separate important objects (the foreground) from the rest of the scene (background). The techniques used to find the objects of interest are known as *segmentation* techniques. There is no universally applicable segmentation technique that works for all situations and all images, and no segmentation technique is perfect. Therefore, the usual path is to combine different algorithms that are streamlined to cope with the needs of a given experiment. This approach is also followed in this experiment.

5.2.2 Edge detection: the Canny edge detector

Edge detectors

Edges in an image can be found in places with strong intensity contrast and usually occur at object boundaries. The detection of these edges is important in image segmentation, which is the process of partitioning the image into different structures. The assumption is that the structures have a predominant intensity value that is different to each other, therefore an "edge" can be detected at the crossing from one to another. Edge areas are described by the presence of high frequency components in their spectrum, so, in theory, a high-pass frequency filter in the Fourier domain can act as an edge detection mechanism. However, edge detection is usually performed in the spatial domain, mainly because it is less computationally expensive. In the spatial domain, gradient operators and Laplacian/Zero-crossing operators are used for edge detection. In gradient operators, the detection is based on the assumption that edge are the pixels with a high gradient and therefore the magnitude of the gradient indicates the strength of the edge. Even if a gradient cannot be defined for a discrete function, it can be estimated using special operators that use convolution with a mask, such as those described by Roberts, Sobel and Prewitt ([Pre70] and [Rob63]). The Laplacian based filters work by detecting the zero crossings of the second derivative of the image intensity, an operation, however, that is very susceptible to

noise. This can be explained by the fact that a derivative calculation in the spatial domain corresponds to intensifying high frequencies in the Fourier domain, hence accentuating noise (which exists in a broad spectrum, including high frequencies) as well. Because of this counter-effect, the image is first convoluted with a smoothing operator (such as Gaussian), an operator that reduces noise, noise points (but also small structures and finer texture details), before any second derivative operation takes place. The last step of all edge detectors is to identify those pixels corresponding to an edge out of all the candidates that were produced by the operator in question. One solution is to apply a fixed threshold to the gradient image and assume that a gradient with values above that threshold qualifies as an edge. The selection of the correct threshold can be either manual or automated (through statistical measurements) process. It is important to get this value right, as a lower threshold would yield more detected edges but would introduce more susceptibility to noise. A higher threshold would miss faint edges, or sections of them.

Another technique would search for the local maxima in groups of N $\times$ N pixels. This solution avoids a global constant threshold value and produces pixel-wide edges, essential for a later stage contour tracing. However, the size of the N $\times$ N kernel must still be defined. A more advanced method, that attempts to avoid the pitfalls of these techniques, is the Canny edge detector [Can86] which I use in this work.

Canny edge detector

The edge detector that is also used in the step prior to the Hough transformation and Active Contour stage is the Canny filter, which was designed to be an optimal edge detector. A large number of edge detector operators have been published, but, for general situations none has exhibited any notable advance over the Canny operator. It takes as input a gray-scaled image, and outputs a binary image that shows the marked intensity discontinuities. The function tries to achieve the following optimization constraints:

- **Noise reduction:** maximize Signal to Noise ratio.
- **Determination of intensity gradient:** mark edges by achieving good localization - the positions of edge points marked are as close as possible to the original edge.
- **Tracing of edges through the image:** identify true edges and reject non-edges by minimizing the number of responses to a single edge.

The Canny filter operates as a multi-stage process:

Noise reduction

The unprocessed image is convolved with a Gaussian mask, which produces a slightly blurred, anti-aliased version of the original. This step aims to reduce the noise and other high-frequency components that might be present in the original image and could affect the intensity gradient.

Determination of intensity gradient

Vertical, horizontal and diagonal convolution masks are used to detect potential edges in a variety of direction. For every operation on each pixel, the maximum result and the direction of mask is stored in a new map of intensity gradients. Canny's operator essentially looks for the maxima of:

$$\frac{\partial^2}{\partial k^2}(G \otimes I)$$

with

$$k = \frac{\nabla G \otimes I}{|\nabla G \otimes I|}$$

where I is the image, σ variance a scalar and G the Gaussian mask:

$$G_\sigma(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{x^2 + y^2}{2\sigma^2}\right]$$

At this stage an additional surround suppression mechanism can be used that will eliminate texture edges [GPW04].

Tracing of edges through the image

The gradient magnitude map is filtered through a non-maxima suppression mechanism that eliminates all pixels that are not on the ridge top of edges. Any gradient value that is not a local peak is set to zero. The purpose of this function is to restrict thick edges into linear segments that have a single pixel width, hence are more easily traceable by a "trace contour" mechanism. The last step required is thresholding. However, there is not a global, fixed value at which an intensity gradient shifts from being part of an object into being an edge. For this reason Canny introduced the concept of thresholding hysteresis. This involves using not a single but two different thresholds, a low τ_1 and a high τ_2 (the high having a value of the order of 3 times the value of the low threshold). All pixels in an edge that represent gradients with an absolute value above than that of the the higher threshold are classified as edge points. Any other pixels connected to these pixels, that also have a gradient value above the lower threshold are also classified as edge points. The implication is that an edge can start off with a point represented by a strong gradient value and continue until the gradient value has dropped considerably. Hence, faint sections of lines can be traced as long as they are connected to a genuine starting point.

The result is a binary image, where each pixel is marked as being part of an edge (1) or of the background/part of an object (0).

Figure 5.1 illustrates the application of different edge detectors. From left to right and top to bottom: original image, Laplace, Sobel and Canny. Laplace and Sobel zero point is offseted in order to make it possible to represent negative values as darker shades of gray.

The Canny filter, while being superior to other gradient-based edge detection schemes, is still not perfect. A pair of threshold values and the width of masks have still to be manually chosen, a problem common to all gradient based methods. However, one of the benefits of threshold hysteresis is that these values are flexible and can be more liberally selected. Edge detection

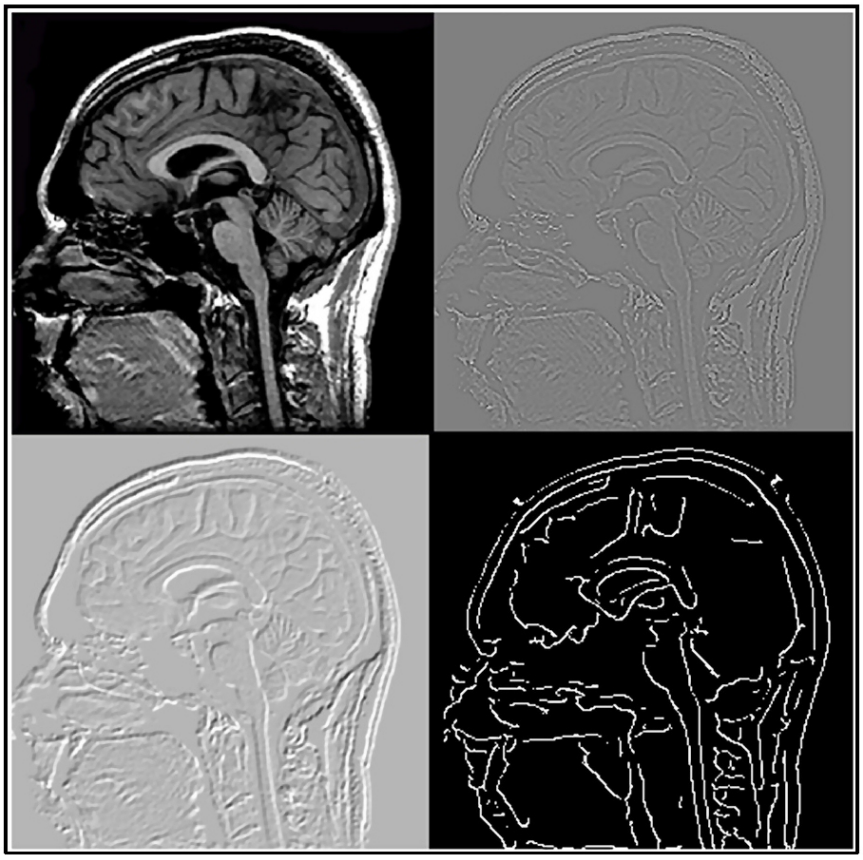


Figure 5.1: The effect of using edge detectors: original image, Laplace, Sobel, Canny.

in images depicting objects of the same nature would require very similar low and high thresholds, that can be found once (even by trial and error) and used as constants throughout the whole dataset. This is certainly the case with medical images of the same modality, where the objective is to find the contours between well delineated structures. Another problem of Canny is that it may miss corners, as they exhibit a very small 1-dimensional gradient. That can be an issue with line labeling schemes that depend on corners and junctions being tagged accurately. Medical images, however, are devoid of such features, as they consist mainly of organic shapes whose contours can

be better described by high-order splines[2] rather than polygons.

5.2.3 Extraction of contour

According to the previous section, the Canny edge detector produces a binary image, where the edges/contours of the objects have a value or 1, and everything else is 0. A method is needed to collect the pixels that belong to the contour of the objects, by identifying them and storing them in an ordered sequence. This method is usually called contour tracing or boundary following [Pav82], and is extensively used as a post-processing step in feature extracting algorithms.
In the context of contour tracing algorithms, the contours are paths that visit consecutive pixels that belong to the boundary of a closed region. A region is defined as the set of pixels such that any pixels in the set can be reached from any other pixel in the set via a 4- or 8- neighbours path. Once a contour is established, it is stored into a list of contours, as the image might contain more than one contour. Therefore, the product of contour tracing algorithm is a list of contours, which are a list of sequential points on a plane.

Several algorithms have been developed to cope with the problem. The four most common contour tracing algorithms are Square Tracing algorithm, Moore-Neighbour tracing, Radial Sweep and Theo Pavlidis' algorithm. The first two algorithms have very straightforward operation and are commonly used in image processing applications. However, they have weaknesses due to connectivity issues in certain binary patterns. For this reason, I used the Radial Sweep algorithm, which is also rather simple to implement and very fast to execute.

[2]Splines are piecewise parametric polynomial functions used extensively in computer graphics because of their simplicity and capacity to approximate smooth curves. A spline, in the context of this chapter, is an expression for a smooth curve regardless of implementation. Splines are further investigated and used in chapters 6 and 7.

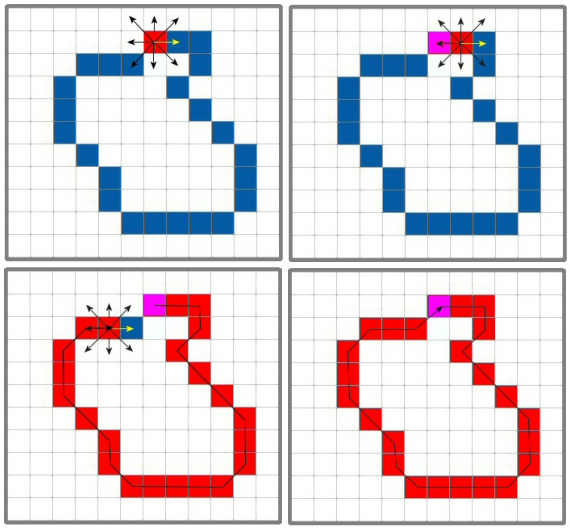


Figure 5.2: Contour tracing: the algorithm traces iteratively the contour and terminates when it visits the starting point again.

Radial Sweep algorithm

All contour tracing algorithms start by tracing the image until they come across a pixel with value 1 (contour pixel), that has not been visited before. This is the starting point of the contour. Then all neighbours of that pixel that belong to the contour are traced in an iterative manner, until the iteration runs out of valid neighbours to trace or visits the starting point again. Each time a contour pixel is visited, is tagged as such, so that in the next iteration it is not regarded as a valid visiting/starting point.
The search for the next pixel is always restricted to the 4- or 8- neighbour, in a certain direction. In this implementation, without loss of generality, the contour is traced in a clockwise direction. Figure 5.2 demonstrates the general concept of tracing algorithms. The operation of Radial Sweep algorithm is identical. The only difference is the stopping criterion: Square Tracing and Moore-Neighbour tracing terminate when they simply visit the starting pixel for a second time. That can lead to problems with more complicated patterns with splits and merges, as demonstrated in Figure 5.3 where the algorithm stops and ignores the upper half of the contour. The Radial Sweep

attempts to rectify this problem by proposing the following stopping criteria:

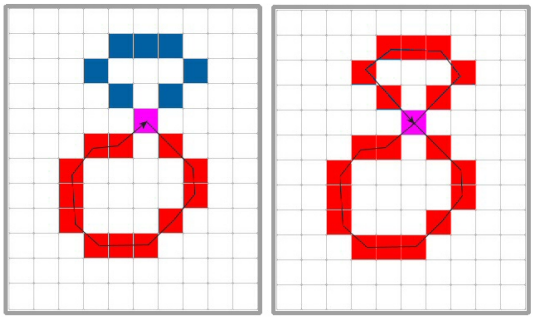


Figure 5.3: Termination criterion problem in other algorithms (left) and the solution provided by the Radial Sweep.

Each time a boundary pixel P_i is visited, is declared as the current boundary pixel and inserted in a stack of pixels $P_1, P_2, P_3..., P_i$. The algorithm terminates when P_j exists, with $j < i$ such that $P_i = P_j$ (which means that the current boundary pixel has appeared previously) and $P_{i-1} = P_{j-1}$. Hence, the algorithm terminates when it visits the starting point for a second time only in the same direction as the first time around. That solves a vast number of ambiguous patterns that would otherwise confuse a more simplistic contour tracing mechanism. Figure 5.3 presents a situation where the Radial Sweep solves a pattern with a split and merge.

5.2.4 Encoding of contour using Freeman chain code

As the contours of the objects are retrieved from the scene, an extra mechanism must exist to allow for systematic packing and storage of this information. Freeman et al. [Fre74] proposed several methods to represent line or curve segments efficiently, by quantizing and encoding the 2D figure into a "Chain" code.

A chain code is a compact way to represent a curve or a contour of an object. It is an ordered sequence of n links of C_i with $i = 1, 2, ..., n$, C_i being

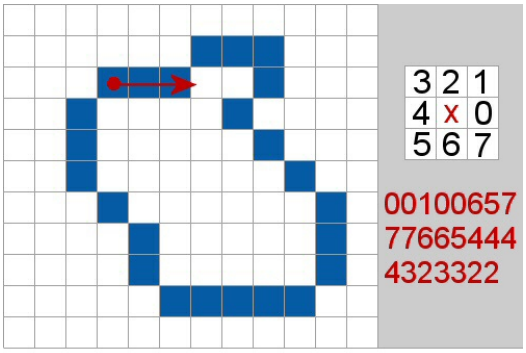


Figure 5.4: Freeman chain encoding of a curve.

a vector connecting neighbouring pixels. C_i codes the direction of movement from current pixel to the next, by using integer values $k = 0, 1, ..., N - 1$ in a counterclockwise fashion, $k = 0$ representing the direction of the positive x-axis, or any other fixed, predetermined direction [FS78]. N in this case is at most 8 (the "Queen's case" move set), as for any curve point on a quantized grid, the next one in the sequence can only be at most eight directions away. It follows that the Freeman chain code is translation invariant, as it is independent of the starting position of the chain: two objects with exactly the same contour but at different positions within an image will yield the same code. Figure 5.4 presents the Freeman chain encoding of a random closed curve.

The reverse sequence, the compilation of a curve from a given chain code happens in a similar fashion. The control points that will drive a polygon or a set of splines are determined each from the previous neighbour and their relationship, as described by the sequence of C_i codes. If the points of the original curve lie on a well-ordered, quantized grid, and N is 8, then the reconstruction can provide an exact replica of the encoded curve. Otherwise, the encoding is lossy and the accuracy depends on the quantization factors (the step of the direction) of the chain code itself [Bos00]. When using chain codes to describe a contour, one needs to clarify whether the boundary encoded pixels are perceived as belonging to the structure inside the region, or as being part of the background. Choosing which is the case is not critical

if the same rules are applied consistently, as measurements derived from the encoding will differ proportionally, according to which approach is followed [Mor04].
The benefit from encoding, apart from the compactness that is derived from not having to store a long string of coordinate pairs, is that it is a basis for a number of techniques used in estimating the metrics of a given contour, such as area and perimeter [Kul77], which, while have not been utilized in this project, could be used in statistical comparisons of different tongue models in the future.

5.2.5 Active Contours - Snakes

Another feature extraction algorithm used as a segmentation mechanism in my research is "Snakes" - a subset of the Active Shape models family. The Active Shape model is a mechanism for contour determination used extensively in image segmentation and analysis. It differs from low-level feature extraction mechanisms such as the Canny edge detector already outlined, in that it is a local, dynamic, iterative and more complicated procedure which moves one step further: contours are used to segment regions that are not necessarily bounded by continuous edges, a phenomenon very common in MRI images. Active Shape models are also different to classical feature extraction techniques as they are model driven: they are given an approximate *a priori* shape and a starting position somewhere near the desired contour of the object they should look for.
The comparative advantages of Active Contours over other feature extraction mechanisms are represented in the following list [Pin97]:

- Active Contour models can be interactively controlled with the aid of springs and volcanoes, placed in appropriate places by the user.
- They are more robust than non-model based methods that do not take the structure of the image into consideration.

- They are less sensitive to ambiguities, discontinuities and noise in the image.
- They can be used in more than 2 dimensions, either spatial or temporal. For example, Snakes have been used to track dynamic objects in animations.
- There are many different models fine-tuned to cope with various problems in computer vision. Also, because of the popularity of the Active Shape models, there are numerous software implementation details available in literature to reference to.

The various Active Shape modelling mechanisms are normally well tested and documented and it is beyond the scope of this text to describe all of them. I only present the algorithms that I tried and used in this project (in this chapter and in chapter 7). For further information the reader is referred to [Den99b].

Snakes

The development of Active Contour models results mainly from the work of Kass, Witkin, and Terzopoulos [MK87]. Active shape models encompass a variety of forms, most notably dynamic contours, deformable templates and Snakes. A snake model is an energy function, a combination of external and internal forces. The scope of the model is to match a deformable spline (the snake), to an image contour by means of minimization of the energy function. The snake is given a starting location and shape and deforms to conform with the nearest prominent contour (Figure 5.5). Snakes are a contour matching mechanism and not a contour detector by themselves like, for example, the Canny edge detector.

Specifically, a snake is represented, through a parametrized curve $v(s) = (x(s), y(s), z(s))$ with s ranging from [0,1] as:

$$E_{snake} = \int_v E_{snake} ds = \int_v (E_{internal} + E_{image} + E_{constraint}) ds$$

where:

- E_{snake}: the energy of the snake, that must be minimized.
- $E_{internal}$: internal energy used to control deformation of the curve; Alterations to internal energy are caused by bending and/or stretching.
- E_{image}: the attraction effect of image features, such as contours, on the snake.
- $E_{constraint}$: external constraints from a higher level or energy applied externally by user.

For $E_{internal}$, Kass et al. proposed a definition based on a measure of elasticity $\alpha(s)$ and stiffness $\beta(s)$.

$$E_{internal} = \alpha(s) \left| \frac{dv}{ds} \right|^2 + \beta(s) \left| \frac{dv^2}{ds^2} \right|^2$$

with constant $\alpha(s)$ controlling the tension along the spline, and $\beta(s)$ the rigidity of the spline. In the case of $\beta(s)$=0, $E_{internal}$ has first order discontinuity - hence, while positional continuity is preserved, the same cannot be guaranteed for its tangents, and so sharp corners can be developed at some points. In the case of both $\beta(s)$ and $\alpha(s)$ =0, then the discontinuity can be also positional.

The snake has to be attracted and lock on to salient image features, and it is the image energy E_{image} that is used to provide the forces needed. The snake may be attracted to lines, edges and terminations:

$$E_{image} = E_{line} \cdot w_{line} + E_{edge} \cdot w_{edge} + E_{termination} \cdot w_{termination}$$

with w_i defined as scalar weightings.

The line functional E_{line} is defined by the image function as :

$$E_{line} = Im(x, y)$$

so that (depending on the sign of w_{line}) the spline is attracted to bright areas on the image. The edge functional factor looks into the magnitude of the image gradient, with:

$$E_{edge} = -\left|\nabla Im(x,y)\right|^2$$

As a result, the spline is attracted to parts of the image with large image gradients - parts that exhibit strong edges. The definition of E_{edge} can be extended to:

$$E_{edge} = -Grad(x,y)$$

with $Grad(x,y)$ a gradient map derived from original image $Im(x,y)$ through a more sophisticated operator that is weighted, for example, according to direction of gradient.

Also, the factor $E_{termination}$ is included so that the snake is attracted by free ends of lines (terminations) or by corners. One can consider the perception of illusory contours to visualize the operation better.
The termination functional is defined as:

$$E_{termination} = \frac{\partial\theta(x,y)}{\partial n_r}$$

with $\theta(x,y)$ being the gradient direction along the contour and

$$n_r(x,y) = (-\sin\theta(x,y), \cos\theta(x,y))$$

being the unit vector perpendicular to the gradient direction.
Finally, the external constraint energy $E_{constraint}$ is applied by the user, and can be set in the form of a number of springs attached to an external entity so that the snake is inflated or deflated by positive or negative expansion along the snake normals. Other possible additions are a "volcano and spring" system to drive a snake or steer away from certain features in the image.

Snakes are frequently initialized manually, but for effective computer vision, it is necessary to have an automatic way of starting the snakes. There are various possibilities: given enough computing power, it is possible to start off many snakes at random, and see which ones find structures they like. The others can be killed off. Conventional techniques can be used to find structures to start the snakes. For example, in image sequences, differencing might provide regions of interest round which snakes can be put [You94]. In other cases, a Hough transform (that will be described in more detail later in this chapter) or other recognition technique can provide a starting point. Groups of line segments or unusual texture, prior knowledge of the geometrical shape, colour or intensity patches might be used.

The Greedy Algorithm for Snakes

The Kass model described above and its implementation details, as described by [MK87], have certain limitations. The model requires the use of many points to represent adequately the contour, something which causes a heavy computational burden - the algorithm runs at $O(nm^3)$, with n the number of control points and m the size of the search neighbourhood $\aleph$, which is usually $3x3$ or $5x5$.

The Greedy[3] algorithm [DM92] which I use in this project is a newer and more efficient (running at $O(nm)$) implementation of snakes, in which the energy minimization is contacted as a purely discrete algorithm based on dynamic programming [AAW88].
In the Greedy model mode, the equation to minimize is:

$$E_{snake(s)} = \int (\alpha(s)E_{cont} + \beta(s)E_{curve} + \gamma(s)E_{image})ds$$

with $s \in [0, n-1]$. $\alpha(s)$ and $\beta(s)$ are user defined scalars that control

[3]In general, a Greedy algorithm is an algorithm that attempts to solve a problem by making the locally optimum choice at each iteration with the target of finding a globally optimal solution algorithm. Greedy algorithms mostly (but not always) fail to find an optimal solution, but instead give a quick and good approximation to the optimum.

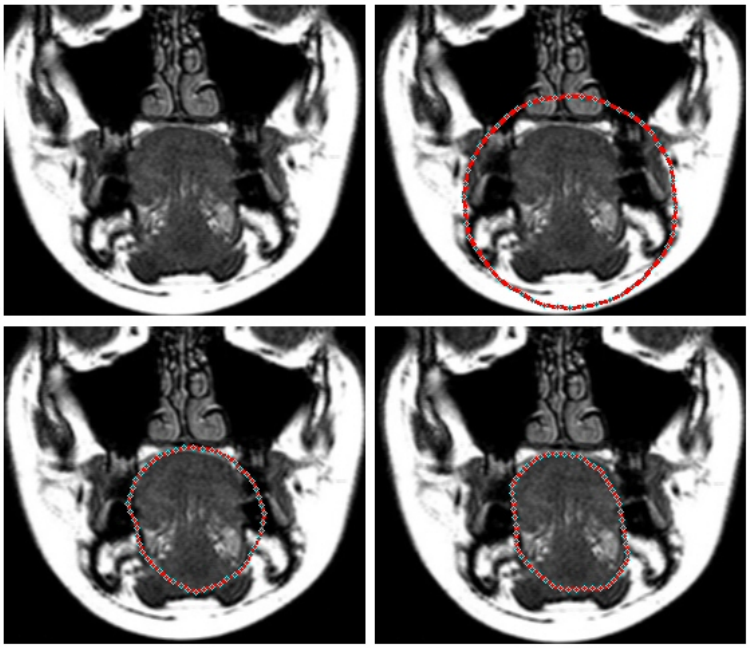


Figure 5.5: A Snake in action: convergence is achieved after many iterations.

the first and second order continuity. The constant $\gamma(s)$ scales the strength of edges.

Like the Kass model, the Greedy model starts with an initial contour whose points are manually or automatically predefined. The Greedy algorithm works then in iterations and performs a neighbourhood search sequentially around all contour points to find the new positions that have lower snake energy.

v_i is defined as the set of points (x_i, y_i) of the Snake. The minimal energy values are evaluated at the first point v_0. To do this, the next point v_1 is evaluated over a predefined neighbourhood value in order to find the point that minimizes the Energy function. The point v_1 is then changed to that of the minimizing value. The process is repeated for all points all points v_i and v_{i+1}, with index i computed module n - the number of snake points on the closed contour.

The three terms of the equation are relatively straightforward to implement in discrete form. E_{cont} is given by the equation:

$$E_{cont,j} = \overline{d} - |v_j - v_{i-1}|$$

with $j \in \aleph$.
$\overline{d}$ is the average distance between two points:

$$\overline{d} = \frac{\sum |v_i - v_{i-1}|}{n}$$

The E_{cont} term regulates the distance between the points, so that it does not get extremely small or large at the edges. The next term is given by:

$$E_{curve,j} = |v_{i-1} - 2v_j + v_{i+1}|^2$$

which is essentially an implementation of the second derivative. The last term, E_{image} is given by the gradient equation:

$$E_{image,j} = -|\nabla I(x,y)|^2$$

which, if G_{max} and G_{min} express the maximum and minimum gradient in each neighbourhood:

$$G_{\max} = \max_{j \in \aleph}(G_j)$$
$$G_{\min} = \min_{j \in \aleph}(G_j)$$

The equation becomes:

$$E_{image,j} = \frac{G_{\min} - G_j}{G_{\max} - G_{\min}}$$

At the end of each iteration, the curvature of the snake is determined and compared to a predefined $threshold1$. The curvature is defined as $d\theta/ds$, θ being the angle between the positive X-axis and the tangent vector, and s the length of the arc. In discrete approximation, curvature c_i for each of the n points is:

$$c_i = \left[\frac{\Delta x_i}{\Delta s_i} - \frac{\Delta x_{i+1}}{\Delta s_{i+1}}\right]^2 + \left[\frac{\Delta y_i}{\Delta s_i} - \frac{\Delta y_{i+1}}{\Delta s_{i+1}}\right]^2$$

with $\Delta x = x_i - x_{i-1}$, $\Delta y = y_i - y_{i-1}$ and $\Delta s = \sqrt{\Delta x_i^2 + \Delta y_i^2}$.

The edge magnitude under the point is also compared to another $threshold2$. If conditions are satisfied, β_i is set to 0 - an indication of a corner. Williams and Shah ([DM92]) propose initial values for $threshold1 = 0.25$ and $threshold2 = 100$. Larger values will result in a rather circular contour, while smaller will have more corners and look less "organic". The threshold values are often determined through a trial and error process, where many slightly different images of the same structure that needs to be segmented are tested.

There is an third threshold value, $threshold3$, which is the least amount of points that need to be displaced in an iteration, before a convergence has occurred. As such, it determines the iteration in which the Greedy algorithm terminates. $threshold3$ is rarely zero, and small values work quite well for certain images. $threshold3$ has to be balanced: very small values will give rise to oscillatory behavior. On the other hand, large values result in inferior, early estimations, since the snake is not allowed to progress much before stalling.
The Greedy algorithm implementation, which I used in this chapter, has the following form in C:

```
// Snake control points: u[i] , i index modulo n
// Snake constants: a[i],b[i],c[i]


// Set a,b,c to 1 for all i
for ( i=0 ; i< n ; i++) {
    a[i]=b[i]=c[i]=1;
}
// iterate through snake
do {

// Loop through points
    for ( i = 0 ; i< n ; i++) {
        E_min=LARGE;

// m is the size of neighbourhood
        for ( j = 0 ; j< m-1 ; j++) {
            E[j] =
            a[i]*E_cont (j) + b[i]*E_curve (j) + c[i]*E_image (j);
            if ( E[j] < E_min ) {
                E_min = E[j];
                jmin = j;
         }

// Call function to move point V[i] to new position jmin
         Move_Point (V[i] , jmin);
         if ( jmin != current_location ) {
             points_moved += 1;
         }
      }
```

```
// Find corners in the next iteration
      for ( i=0 ; i< n-1 ; i++ ) {
          curvature[i] = Compute_Curvature (u[i],u[i+1]);
      }

// Check curvature; if larger than neighbours
// and edge strength is above threshold2 then
// relax curvature through b[i]
      for ( i=0 ; i< n-1 ; i++ ) {
          if ( (curvature[i] > curvature [i-1]) &&
               (curvature[i] > curvature [i+1]) &&
               (magnitude(edge[i] > threshold2)  ) {

               b[i] = 0;
          }
       }
   }
}
while (ptsmoved>=threshold3);
```

The Greedy algorithm presented is fast, flexible and simple to implement. Williams and Shah support that the method gives very similar final results compared to the original variational calculus method of Kass et al. [MK87] and the dynamic programming method developed by Amini et al. [AAW88]. Additionally, it is faster than both approaches and more flexible and stable with the inclusion of hard constraints.

Active Contours are very popular in research, and therefore, in addition to the Greedy algorithm, there are other methods to approximate a solution for a snake. Recent methods have introduced a further external force, called Gradient Vector Flow (GVF, [CJ98]), that replaces the commonly used image

edge magnitude. However, the GVF based solutions are more computational expensive than the Greedy model. Additionally, their principal improvement, the ability to move better into concavities, is not appropriate in the context of this work, as the sought-for shape of the tongue is almost convex. For these reasons they have not been explored further in my research.

Summary

In this subsection I present the notion of Active Contour models and the implementation method I followed - the Greedy algorithm. I have used Active Contours in order to aid the segmentation of tongue in coronal MRI slices, by finding a contour that best approximates the perimeter of the object. I have argued, however, that Snakes cannot solve the entire problem of contour estimation as they depend on user interaction or interaction with a computer vision mechanism that has a higher level understanding of the problem. Active Contours are "naive" in that they do not, necessarily, have an *a priori* knowledge of the position and shape of the object they need to approximate. A way to solve this problem is to initialize the Snakes by placing them near the image contour of interest, and then let them iterate. The placement of the Snake's starting point can either be manual or automatic.The rest of the Snake parameters are set usually after trial-end-error experimentation with a plenitude of similar images, and are fixed for all examinations.

5.2.6 Straight line detection: Hough transformation

The Hough transformation ([DH72] and [Hou59]), originally used in automated analysis of bubble chamber photographs, is a versatile tool for performing feature extraction. It is used extensively in image processing, to identify arbitrary shapes that can be expressed by a set of parameters, such as curves, polynomials, ellipses and straight lines. In the latter case, the technique aims to determine the line(s) that pass through most features of the image, as demonstrated in figure 5.6. One set of such features that can provide with a meaningful set of lines through the Hough Transform are

binary edges. The binary edge detected image highlights (by assigning a value of 1) the pixels where the intensity changes sharply, while keeping the rest of the image at 0. As already mentioned, there are many techniques for edge detection (Roberts Cross, Prewitt, Sobel, Laplacian, Canny) and choosing a suitable one depends on the nature of problem in question, with the Canny edge detector being the favorite for problems involving medical images. As such I used the Canny edge detector to produce the binary edge images required as an input to the Hough transformation.

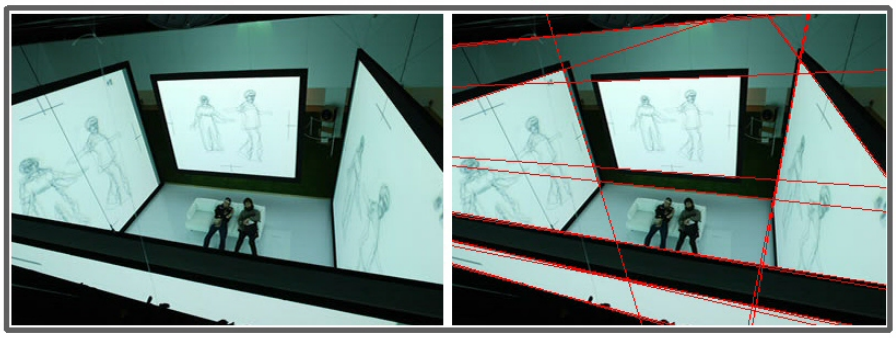


Figure 5.6: Hough transformation: original image, and image with superimposed lines.

In order to detect the straight lines in an image, the Hough transformation assumes lines parameterized in the form:

$$\lambda = x \cos\theta + y \sin\theta$$

where λ the distance from origin and θ the orientation of λ with respect to the horizontal axis. Points of interest:

$$(x_i, y_i)$$

laying on the same line are transformed into K sinusoidal curves in the polar Hough parameter space:

$$\lambda = x_i \cos\theta + y_i \sin\theta$$

in the (λ,θ) plane, which intersects at (λ,θ). The sinusoidal points calculated are mapped and summed into an accumulator, initialized with zeros. As the algorithm scans the original image, the accumulator map cells are incremented. The last step is to determine the local maxima in the map: its parameters (λ,θ) represent the most probable candidates for lines in the original image. The candidate lines are then stored as infinitely long segments and represented with two floating-point number (ρ, θ), where ρ is the distance of the line to $(0, 0)$ point and θ the angle between the X-axis of the image plane and the normal vector of the line.
There are plenty of methods that can be employed to pinpoint the local maxima in a given image. The most simple is to search and record for the highest value in neighbours of N $\times$ N pixel sized non-overlapping rectangles over the image. Another solution would be to apply a fixed threshold, or calculate one through the histogram of the map, and then apply a thinning morphological operator[4] to the isolated clusters of highlighted pixels [Mat04] .
The Generalized Hough algorithm for lines is straightforward to implement in discrete arrangement. In C, it has the following form:

```
// Input image  : Source [Xmax,Ymax] - boolean
// Hough transform map : Hough [Distance_Resolution,Angle_Resolution] - floats
// MSize : Local maxima search size
// Lines : Storage for lines

// Zero all Hough components
for ( x=0 ; x< Distance_Resolution ; x++) {
    for ( y=0 ; y< Angle_Resolution ; y++) {
        Hough[x][y]=0;
    }
```

[4]Thinning is a morphological operator, similar to erosion or opening, used to remove certain foreground pixels in binary images. It is used in skeletonization of the output of edge detectors, where it reduces all lines to single pixel thickness.

```
}

// Hough transformation
for ( x=0 ; x< Xmax ; x++){
    for ( y=0 ; y< Ymax ; y++){
         if ( Source[x][y]){
             for ( s=0 ; s< N ; s++){
                 r=(x-XMax*0.5)*cos(s)+(y-YMax*0.5)*sin(s);
                 if (r> 0) Hough [s][r]++;
             }
         }
     }
}
```

One of the advantages of the Hough transform is that it is relatively tolerant to image noise and gaps in features. As such, the Hough transform is used to define lines derived from an image where there is only a suggestion of a linear configuration, in the form of discrete dots or disconnected components. I used the Hough transformation to calculate the midsagittal brain fissure line on the coronal MRI images (on scans of sufficient FOV that include a portion of the brain), in order to determine the locality and global rotation of the oral cavity, as explained in section 6.5.3.

5.3 Image Restoration (Preprocessing)

The previous section referred to the background theory of the different feature extraction mechanisms that I used in the project. Before applying these algorithms to the images, an extra step is taken to ensure that noise inherent in the data is eliminated. This process is called filtering. Filtering is a fundamental operation in image processing. It can be understood as an algorithm applied to the pixels of a given image, so that the value of the pixels at a given location within the image becomes a function of the values of the

pixels in a neighbourhood around that location. Most filtering algorithms compute the equivalent of a mean intensity value on a local neighbourhood for each pixel of the image. As a result, boundaries of objects (edges of high contrast) are affected - blurred out - and structures are displaced. This side effect is critical for segmentation algorithms (i.e. Active Contour) that depend on accurate detection of boundaries. The filtering algorithm used in this work is anisotropic diffusion.

5.3.1 Anisotropic diffusion

Anisotropic diffusion [PJ90] is a solution to this problem, as it smooths the images in order to cut-out noise and other unwanted small fluctuations, while preserving edges. This is accomplished through looking at the geometric closeness as well as the intensity similarity between pixels and choosing near values in both range and domain.

Similar edge-preserving filtering (including anisotropic) have been used very frequently in an extensive range of applications in image processing and robot vision. For an early survey and comparisons with other methods the reader should consult [MT79] and [RC83] respectively.

In medical imaging, anisotropic diffusion has been successfully applied in the filtering of MRI data [ea92a], and was selected to be used in this project as well.

Definition

Anisotropic diffusion is an iterative process. We define:

$$\frac{\partial I(t)}{\partial t} = \nabla \bullet (c(t) \cdot \nabla I(t))$$

with $I(t)$ the gray-scale image at iteration step t. The function $c(t)$ is the diffusion function, a monotonically decreasing function of the magnitude of the image gradient:

$$c(t) = f(|\nabla I(t)|)$$

which allows for adaptive diffusion strength based on the local image gradient magnitude. Any monotonically decreasing continuous function is sufficient, but [PJ90] suggests, amongst others, the following:

$$c(t) = \exp\left(-\left(\frac{|\nabla I(t)|}{K}\right)^2\right)$$

with K referred to as diffusion constant, that affects the behavior of the filter. If we define a flow function:

$$\Phi(t) = c(t) \cdot \nabla I(t)$$

and integrate that into the original equation:

$$\frac{\partial \mathrm{I(t)}}{\partial \mathrm{t}} = \nabla \bullet (\Phi(t))$$

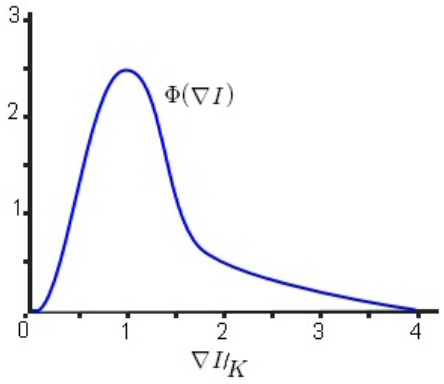


Figure 5.7: Flow function plotted as a function of image gradient.

A flow function Φ corresponding to image gradient (figure 5.7) can be constructed and plotted. Observe the local maximum at the point where:

$$|\nabla I| \approx K$$

which implies that the flow is small in regions with $|\nabla I| << K$ and $|\nabla I| >> K$, and so the diffusion process preserves the edges at these regions. When the gradient magnitude is around K then the flow is maximum and the regions are smoothed out. K is usually selected to correspond to the gradient magnitudes produced by noise. Selecting the K value to be slightly less than the gradient magnitude of the edges of objects in the image can result in the enhancement of the edges, which is the objective in many situations.

Discrete implementation in 2D

The discrete implementation of the filter in any dimension is very straightforward, since, in the discrete domain, gradients and derivatives can be approximated as the intensity difference between adjacent image elements. According to Blair [Mac95], for an image element at $I(x, y, t)$, the 2D case is formulated as:

$$\frac{\partial I(x,y,t)}{\partial t} = \frac{\partial}{\partial x}\left[c(x,y,t)\cdot\frac{\partial I(x,y,t)}{\partial x}\right] + \frac{\partial}{\partial y}\left[c(x,y,t)\cdot\frac{\partial I(x,y,t)}{\partial y}\right]$$

$$\approx \frac{1}{(\Delta x^2)}[c(x+\frac{\Delta x}{2},y,t)\cdot(I(x+\Delta x,y,t)-I(x,y,t))-$$

$$c(x-\frac{\Delta x}{2},y,t)\cdot(I(x,y,t)-I(x-\Delta x,y,t))]+$$

$$\frac{1}{(\Delta y^2)}[c(x,y+\frac{\Delta y}{2},t)\cdot(I(x,y+\Delta y,t)-I(x,y,t))-$$

$$(x,y-\frac{\Delta y}{2},t)\cdot(I(x,y,t)-I(x,y-\Delta y,t))]$$

which if $\Delta x = \Delta y = 1$ becomes:

$$\frac{\partial}{\partial t}I(x,y,t) = \Phi_{east} + \Phi_{west} + \Phi_{north} + \Phi_{south}$$

and

$$I(x, y, t + \Delta t) \approx I(x, y, t) + \Delta t \cdot (\Phi_{east} + \Phi_{west} + \Phi_{north} + \Phi_{south})$$

Therefore, in every iteration t, each pixel is updated by the sum of the flow contributed by its four neighbours. The flow functions are calculated independently and once for each of the neighbouring elements, and are cached for the duration of the iteration.

Figures 5.8 and 5.9 shows the potential of the anisotropic diffusion filtering. The original image, a standard Gaussian low-pass filtered and the anisotropically filtered versions are presented.



Figure 5.8: Anisotropic diffusion filtering: original image.

In the anisotropic diffusion version, the picture is simplified as the texture and fine details are removed. There is no loss of overall shape features, and the overall shading is preserved, although some intensity banding artefacts are present. On the other hand, the Gaussian filtered image looks blurred, with intensity "auras" around the boundaries of objects which potentially complicate a segmentation attempt on the image. For these reasons, the anisotropic diffusion filter was used on the MRI images to eliminate noise while retaining the shape and boundaries of objects.

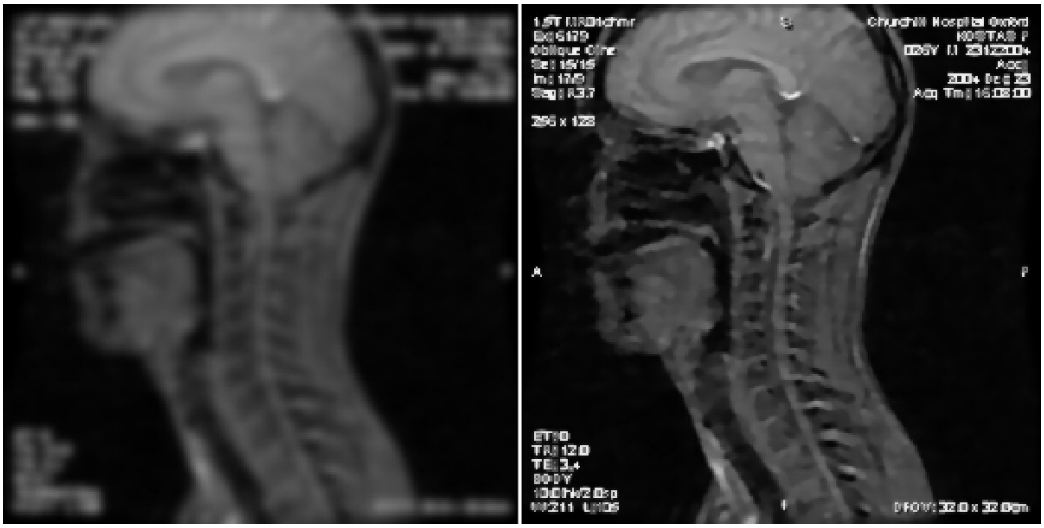


Figure 5.9: Anisotropic diffusion filtering: Gaussian and anisotropically filtered images.

5.4 Implementation

The following two sections describe the steps I have taken in order to extract, on a slice by slice basis, the contours of the tongue from the Visible Human Project (VHP) atlas volume and the MRI scan volume, based on the common image processing techniques described so far in this chapter. The extraction of the information in the VHP domain is straightforward, since it was designed to be an atlas of the human tongue and only that, meaning that the VHP volume does not carry any irrelevant (or "background") information. Nevertheless, the same cannot be said of the MRI volume content, which holds a substantial amount of background data and noise that must be discarded.

5.5 Extraction of contours in VHP volume

For the extraction of contours out of the VHP volume, I simply applied the Canny edge detector without preprocessing[5], and Freeman chain encoding to save the contours of each coronal slice of the volume into a more manageable structure. To cope with the Freeman chain translation invariance, the 2D position of the first pixel of the contour was also saved.

5.6 Extraction of contours in MRI volume

On the MRI coronal slices, it is only required to extract the contour that describes the tongue, without any tagging of muscle groups - if that was possible, this experiment would not be necessary altogether.
On every slice, the contour must define the outer limits of the tongue structure. There is only one contour to be determined per coronal slice, and this is due to the anatomy of the tongue which, in relaxed form, does not produce any cavities along the coronal direction.
The basis of the tongue segmentation that I have used is an Active Contour that is initialized somewhere close to the place where the tongue contour is, and iterates until an estimation for the shape of the object in question is reached. This estimation is a polygon, which then can either be saved or manually updated and saved.
This work-flow introduces a major problem: without any manual intervention, how is it possible to know where to place the Active Contour on the image? The solution I propose combines *a priori* knowledge about the structures on the images (that there exists a midsagittal line - the brain fissure line - that its imaginary extension crosses the tongue in the middle) and further localization information derived from the open contour that is delineated on the surface of the tongue. This contour is easily extracted through simple edge detection, as its margins are marked by cavitatory contrast enhance-

[5]The VHP volume contains ID values expressed as intensity scalars. As such, there is no "noise" and hence no need for image restoration or other enhancements

ment through the employment of the Gadolinium solution during the scan, as described in section 3.4.3.

The following subsections describe the steps followed to derive the contour from the images, in sequential order.

5.6.1 Preprocessing

The application of medium strength anisotropic filtering on each image is needed in order to smooth out textural noise while retaining the delineation between high-contrast structures. This eventually helps the movement of the snakes, as they avoided getting "stuck" in heavily textured areas (as, for example, the inside area of the tongue). The effect of preprocessing is mild but nevertheless important. The preprocessing step is not costly, as it is only performed once per slice, and my implementation can filter tens of images a second.

5.6.2 Midsagittal line calculations

The Hough transformation was used to find the midsagittal fissure line on the lower portion of the brain, as viewed on the coronal planes. I assume that the FOV of the sequence is sufficiently large to make an adequate amount of the brain visible (figure 5.10).
I used this information for localization of the tongue and also to determine the global rotation on the coronal axis.

A similar method to the one I used was first proposed by [Bru91], although completely different methods have been proposed in the past (most notably [RGC95] that uses linear Snakes). In my method, the sequence of processing for all coronal images in the MRI volume is as follows:

1. Pre-process using anisotropic diffusion filtering.

2. Apply the Canny Edge detector to detect edge points.

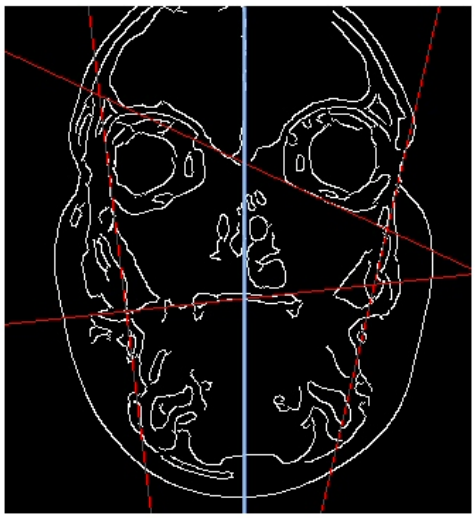


Figure 5.10: Use of Hough transformation to detect midsagittal fissure line. Example uses Lucy's dataset.

3. Hough transform for line detection. Straight lines are detected and stored in a single array.

4. Discard all lines that are not vertical (that is, lying over the axial axis) with a 10% variation over 2π either side. These lines cannot be suitable candidates, as the fissure line is almost (but not necessarily exactly) vertical as viewed on the coronal plane.

5. For each of the lines calculate a weighting factor based on: 1) Distance of line from the center point of the image - for 256×256 images, that is the point at (128,128). 2) Intensity of the accumulator cell during the Hough transformation step that describes the line. At the end there are t lines $\overrightarrow{n}$[6], each with a weight w. The weighting factor is introduced here so that vertical segments that are away from the fissure line (which should be near the middle of the image) are taken less into account than those lying near the middle.

[6]I use the vector notation for (ρ, θ).

6. The weighted mean of lines is then calculated through:

$$\overrightarrow{\overline{n}} = \frac{\sum_{i=1}^{t} w_i \overrightarrow{n}_i}{\sum_{i=1}^{t} w_i}$$

The anisotropic diffusion and Canny filter values were selected mainly through trial and error, after repeating the experiment many times with different images and correlating the results with what could be observed. The values used for anisotropic diffusion filtering were: number of iterations=5, K=0.5. The values used for the Canny filter were σ=2.5, τ_1=0.3 and τ_2=0.7.

5.6.3 Use of midsagittal line $\overline{n}$

I used the estimated midsagittal fissure line $\overline{n}$ in two places:

1. **To determine the locality (position) of the oral cavity** . The extended midsagittal line should pass through the contour that describes the dorsum of the tongue. I use ray-casting to find whether the line intersects a given contour or not:

   The midsagittal line is represented as a ray. Any point $\overrightarrow{R}$ that lies on the ray must satisfy:

   $$\overrightarrow{R}_1(\lambda) = \overrightarrow{P} + \lambda \cdot \overrightarrow{D}$$

   with $\lambda > 0$ a scalar, $\overrightarrow{P}$ and $\overrightarrow{D}$ the origin and unit direction of the ray respectively. As the Hough transformation represents the line in (ρ, θ), $\overrightarrow{P}$ can be expressed as:

   $$\overrightarrow{R}_1(\lambda) = \begin{bmatrix} \rho \cdot \cos(\theta) \\ \rho \cdot \sin(\theta) \end{bmatrix} + \lambda \cdot \begin{bmatrix} -\rho \cdot \sin(\theta) \\ \rho \cdot \cos(\theta) \end{bmatrix}$$

   Similarly, each retrieved contour can be expressed as a series of lines. A line with end-points at $\overrightarrow{P_0}$ and $\overrightarrow{P_0}$ is given by:

   $$\overrightarrow{R}_2(\kappa) = \overrightarrow{P_0} + \kappa \cdot (\overrightarrow{P_1} - \overrightarrow{P_0})$$

with $\kappa \in [0,1]$. The two simultaneous equations $\overrightarrow{R}_1(\lambda)$ and $\overrightarrow{R}_2(\kappa)$ are solved, and checked so that λ and κ are within the specified ranges. If a contour has at least one line segment that satisfies the above criteria, then that contour is a strong candidate for further evaluation.

2. **To find the rotation transformation necessary to compensate for any possible coronal rotation of the head** .

   The coronal rotation with regards to the X-axis is found through θ of $\overline{n}$. All contours in all slices, expressed through control points $\overrightarrow{V}_n$ are then rotated by $-\theta$:

$$\overrightarrow{V}'_n = \begin{bmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{bmatrix} \cdot \overrightarrow{V}_n$$

5.6.4 Use of a snake to segment the tongue

In the final step in the sequence I initialize and iterate a snake so that it finds the organ boundaries on every slice. Throughout this procedure, a closed contour, expressed as a finite sequence of 2D points is extracted for every slice. The points are then stored as a Freeman chain code, in exactly the same fashion as the ones extracted from the VHP volume.
As with all Active Contours, the procedure is broken down into 3 parts. In summary:

- **Initialization**: it is important to appropriately initialize the position and size of the Active Contour in order for it to converge to the sought-after shape on the image. I propose an automated and manual method that adequately estimates these initialization parameters.

- **Iteration**: the Active Contour iterates through each contour point and determines the energies (curvature, contour and image) until the number of contour points moved is below a threshold. The process is fully automated (as described in the theory section), but the assignment

of parameter values intrinsic to the operation of the Active Contour is not: it is a product of trial and error experimentation with different values in several test cases.

- **Validation**: once the iterations terminate, a higher level mechanism must validate whether the estimated contour is roughly accurate or not. The higher level mechanism is the user, who can update any points of the contour that mismatch with the actual contour.

Each of these implementation points is analyzed in the following subsections:

Initialization: problem of initial conditions - requirements for a bounding ellipse

The initial conditions of an Active Contour, or rather the position of the points that prescribe the shape of the Active Contour at the first iteration are usually set by a pre-defined contour that vaguely follows the shape of the object under segmentation [CT98]. In this context, the object is a cross-section of a tongue, which can be roughly approximated by an ellipse. Hence I can use an ellipse to derive the initial conditions of the Active Contour that will attempt to segment the tongue from the rest of the image. The parameters of the ellipse, such as its locality and major and minor axes are calculated through the points that describe the dorsum contour: the concept is that the dorsum contour is an open curve that can be part of an ellipse that is localized to the region of interest, and scaled to bound the bulk of the tongue.
In order to calculate the equation of this bounding ellipse, I use a direct least squares fitting method of ellipse to points, with the points derived directly from sampling the dorsum contour.
The problem of ellipse fitting is well documented, and different approaches exist in the literature. In my work, the approach of Pilu et al. ([MP96]) is followed, in which:

The ellipse is presented as an implicit second order polynomial:

$$F(A, X) = A \cdot X = a \cdot x^2 + b \cdot x \cdot y + c \cdot y^2 + d \cdot x + e \cdot y + f$$

where $A = \begin{bmatrix} a & b & c & d & e & f \end{bmatrix}$ and $X = \begin{bmatrix} x^2 & x \cdot y & y^2 & x & y & 1 \end{bmatrix}^T$. $F(A, X_i) = d$ is the distance of a point x_i to $F(A, X) = 0$.
The goal is to minimize the algebraic distance over the set of N data points in the least squares sense:

$$\arg\min_A \left\{ \sum_{i=1}^{N} F(A, X_i)^2 \right\}$$

Bookstein ([Boo79]) suggests that, having set a quadratic constraint on the parameters, the minimization is solved through a generalized eigenvalue system:

$$D^T \cdot D \cdot A = \lambda \cdot C \cdot A$$

where:

$$D = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T$$

and C is the constraint matrix. This is a well posed optimization problem, that can be solved through matrix decomposition, for which task there is a variety of tools available in OpenCV, the mathematical and image processing platform that was used throughout this project.
A very similar method, but for head rather than tongue images has been presented before as a collaborative work in our department [KPG05].

Problem of initial conditions - alternative methods

Another method, albeit less automated, that I devised for the initialization of the Active Contours is based on a pre-defined template. The template consists of a series of contours in the coronal plane, from the tip of the tongue to the base, that are a rough approximation of the shape of the organ. This

template exists already in the form of the VHP contours which were extracted earlier. In order to use the VHP contours as a starting point for the snake, a mechanism must exist to take care of the localization problem and the 2D scaling of the contours with respect to the tongue shapes as depicted on the MRI images.

For the **localization** problem, an automated solution is to use the tongue surface contour as an offset. In order to do so, the template contour is clamped by the tongue contour itself. The clamping mechanism works in two steps for every slice:

1. Let template and tongue surface contours be $\overrightarrow{Temp}[i]$ and $\overrightarrow{Surf}[j]$. Scan through $\overrightarrow{Temp}[]$ and $\overrightarrow{Surf}[]$ and find contour points $\overrightarrow{Temp'}$ and $\overrightarrow{Surf'}$ so that $\forall i \in [0, n]$ and $\forall j \in [0, k]$ control points, $\overrightarrow{Temp'}_y \leq \overrightarrow{Temp}[i]_y$ and $\overrightarrow{Surf'}_y \leq \overrightarrow{Surf}[j]_y$. These are the topmost points of each contour and should lie on the same position. To do so:

2. Set new offset for every point in contour $\overrightarrow{Temp}[i] = (\overrightarrow{Temp}[i] - \overrightarrow{Temp'}) + \overrightarrow{Surf}$.

The next issue is to adjust the relative **scaling** . An automated approach would involve a similar technique, where the X-axis boundaries of $\overrightarrow{Surf}[]$ and $\overrightarrow{Temp}[]$ are found, and from their Euclidean distance the X-axis relative scaling is acquired. The Y-axis scaling is then set equal to that value:

1. Scan through $\overrightarrow{Temp}[]$ and $\overrightarrow{Surf}[]$ to find scalars $Temp_{minx} \leq \overrightarrow{Temp}[]_x \leq Temp_{maxx}$ and $Surf_{minx} \leq \overrightarrow{Surf}[]_x \leq Surf_{maxx}$. These numbers represent the minimum and maximum values of the x component of all control points in each contour.

2. Relative scale of $\overrightarrow{Temp}[]$ to $\overrightarrow{Surf}[]$ is

$$\theta = \frac{(Surf_{maxx} - Surf_{minx})}{(Temp_{maxx} - Temp_{minx})}$$

3. Set scale for all points in contour $\overrightarrow{Temp}[] = \overrightarrow{Temp}[] \cdot \theta$.

The operation is similar to matching the bounding boxes[7] of the two contours. It can be noticed that the re-scaling of $\overrightarrow{Temp}$ distorts the offset matching set earlier. A method to cope with this problem is to perform the sequence in reverse, and hence set the relative scale first and then adjust for offset.

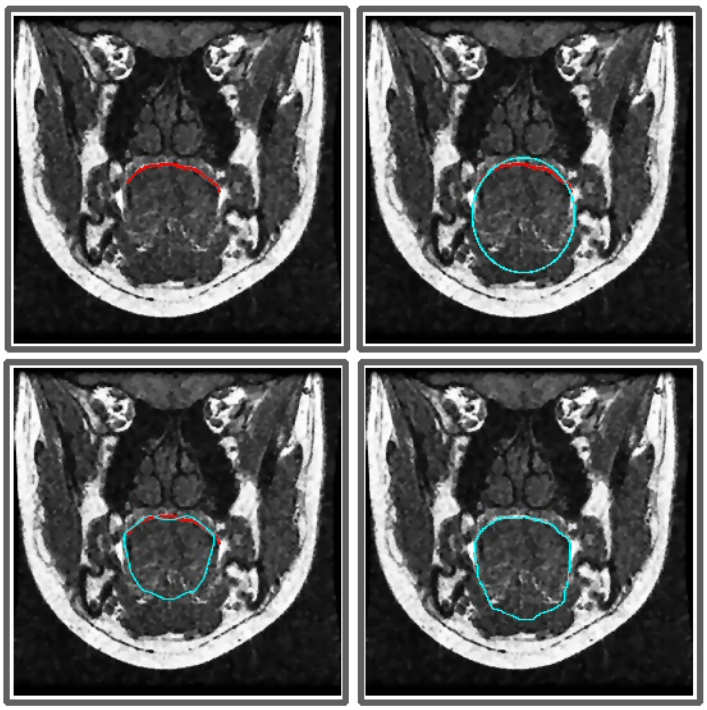


Figure 5.11: Initialization step of Active Contour: different approaches.

The assumption that I made in this alternative method is that the tongue surface contour gives a very good indication of the maximum width of the tongue for any given slice, which may not always be the case. Because of this, a manual scaling 'override' mode is more applicable to this problem. In contrast, the automated offset calculations are found to be quite precise and hence there is no need for a manual mode.

[7]Bounding box alignment is also investigated in chapters 6 and 7, in 2 and 3 dimensions.

Summary of method

The automated initialization of the Active Contour poses a reasonable problem that is tackled through the use of the highly delineated open curve that is described at the tongue surface, as a result of the application of the contrast medium during the scan. An overview of the two methods that I propose is illustrated in figure 5.11: the top left image shows the marked tongue surface. The next step is to use either the bounding ellipse (top right) or the template contour, scaled and translated to fit upon the marked spline, as described in the previous subsection. The last step shows the tongue contour extracted after the cycle of iterating and manually evaluating the generated snake. This last step is described in more detail in this subsection.
However, one can assume that this curve is missing, either due to poor Contrast to Noise ratio, resolution or because the application of the contrast medium was not feasible in the first place. The alternative in this case is either to manually place the contour points on every slice (a process which could take less than one minute per slice) or use the VHP atlas contours and manually adjust their relative offset and scaling parameters. It can be seen that even in the "worst case scenario", the process of initializing (and, as it will be demonstrated, the process of iterating and evaluating) the snake contour takes a few tens of seconds per slice, which is deemed adequate even for a purely clinical environment.
A more elegant solution for this "worst case" scenario is to manually place the contour points on a slice around the middle of the volume, for example about 2 cm deep, reading the slices from the tip of the tongue to the back of the head. After the initialization, the iteration of the snake at this point will produce a contour C^0. The next two slices (going now forwards and backwards over the coronal slices) can now use C^0 in the initialization step of their snake. The result (C^{-1} and C^1) will also propagate to the next slice (to give C^{-2} and C^2), and so on, until the end of volume is found. This semi-automated method requires the manual initialization of only one contour, but on the other hand is very sensitive if the gap between the slices is too

large, and hence the shape of the contour in one slice is not a even a rough estimation of the shape of the contour of the next one. The fusion of sagittal and coronal slices (as described in chapter 4) seems to improve results, as it populates the volume with in-between slices that assist the smooth transition between contours of adjacent slices.

With the initialization of the contour set, the next step is to iterate the snake to converge to the tongue cross-section:

Iteration of snake

I found the arguments for the Snake iteration process after trial and error tests with different images. The following arguments are fixed for every point and every test case:
Number of points= 100
α= 3, β = 5, γ=2
Window size = 7×7
Termination criteria: At least 5 points must be moved during any iteration to keep the iteration process running, otherwise the function terminates.

The snake was implemented using a combination of C++ programming and OpenCV tools. For every slice it needs a few tens of iterations to converge, something which happens in almost real time - with an average PC (as of 2006) capable of calculating thousands of iterations per second.

Evaluation of snake

At the end of the cycle it is important to evaluate the estimated contour in terms of how accurately it envelops the textured region in question. Pixel-perfect accuracy at this stage is not a very important factor, and an error tolerance[8] of the order of 10 % is acceptable. The reason is that there is a

[8]A 10 % error tolerance would mean that an answer could deviate up to 10 % from the correct answer in either direction and still considered a correct answer. As an example,

level of uncertainty anyway, due to the limitations of the scanning technology (mainly noise, contrast and spatial resolution) and of the pose of the subject in the scanner (rotational artefacts).
The evaluation cycle that I propose is essentially a feedback mechanism: at the end of each iteration, the user can either save the contour or update the position of the points that are not positioned correctly and start the process again, using that contour as the initial conditions for another snake iteration. The points are moved with a drag and drop ("snap") mechanism, an operation similar to the one employed by desktop publishing tools for working with vector graphics. Again, in "worst case" scenarios, the speed of the whole operation is high enough to ensure sub-minute evaluations per slice. This means that an entire volume can be processed in less than 10 minutes, which is a sensible and practical value.

5.7 Results

5.7.1 VHP domain contours

It has been established that in the VHP volume, the extraction of contours is a one-off process that is reasonably straightforward and fully automated. The validity of the output of this procedure depends solely on the soundness of the data fed, which was produced manually with the help of the anatomist, as explained in chapter 3. The result of processing the VHP domain images can be seen in figure 5.12, where an example set of 3 VHP contours is presented. Similar to the three slices presented in this example, the rest of the images of the set produce a single contour per slice.

assume that the width of the tongue is in reality 5 cm, and our measurement comes up as 5.3 cm. We can accept the later as correct.

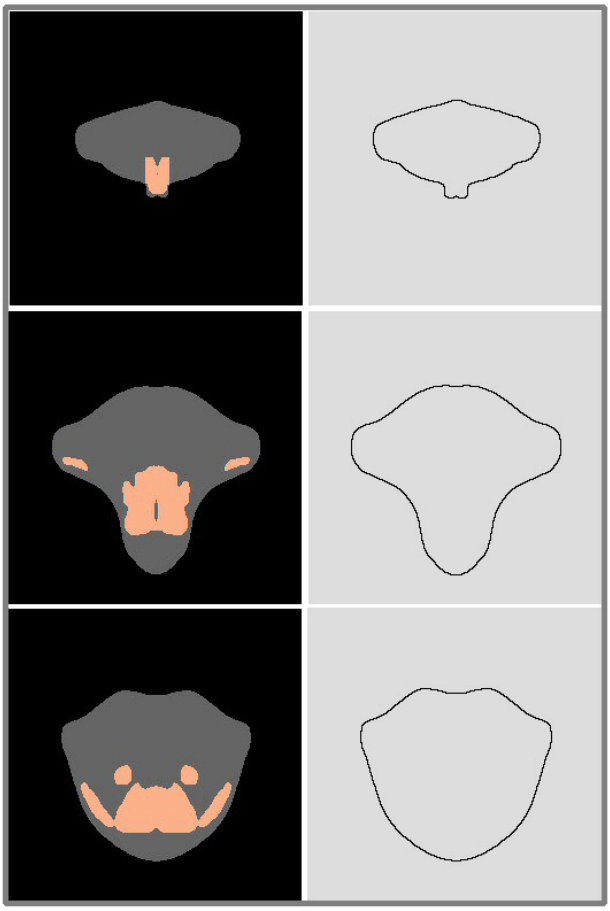


Figure 5.12: VHP domain contour calculation: 3 cases, visualized before and after the operation.

5.7.2 MRI domain contours

The MRI images were processed using the algorithms described in this chapter, on volunteers and clinical cases, yielding always one contour per slice. The use of pectin on the volunteers assisted the automated extraction of the tongue using the Active Contours, with minimal user interaction. The clinical cases, which are presented in chapter 8, were outlined semi-automatically with the help of a clinician.

An example set of 3 MRI images and their corresponding contours is visu-

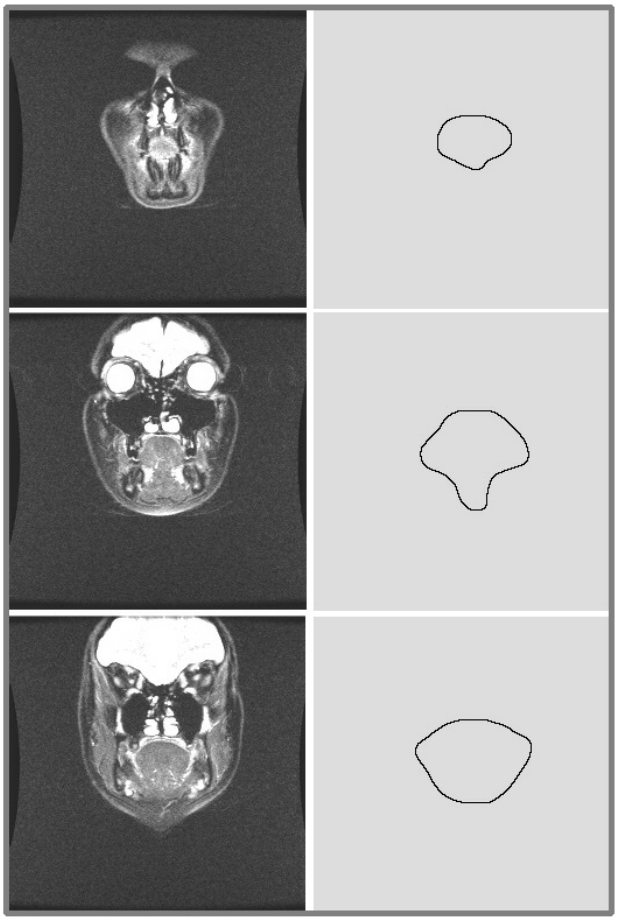


Figure 5.13: MRI domain contour calculation: the same 3 cases, visualized before and after the operation.

alized in figure 5.13. The images shown are the ones corresponding to the VHP atlas images shown in figure 5.12. The scan presented is of a volunteer (Lucy, T_2 weighted coronal) using cavitatory enhancement with Gadolinium contrast agent suspended in pectin solution, as described in chapter 3. The rest of the images also result in a single contour per slice, which conforms to the rule of convexity[9] for the shape of the tongue, as described in section

[9]A concavity would indicate a "split and merge" and hence would require more than one contours per slice to fully describe the shape of the object in question. This potential problem is investigated in chapter 7.

5.2.3.

Generation of in-between contours by interpolation

The contours shown in figure 5.13 are derived from a volume that has a slice thickness of 2 mm with 0 mm spacing. This means that any voxel (and therefore a contour) has a thickness of 2 mm total, which is sufficiently small to produce an accurate end 3D model of the tongue.
However, it is possible that the coronal resolution is not sufficient: for example, a volume scanned with slice thickness of 6 mm would yield 10 or less coronal images and contours, which would introduce aliasing artefacts to the end 3D model. Assuming that the slice thickness is locked to a value of over 3 mm then an interpolation method must be employed in order to generate the "in-between" contours. The interpolated contours are an approximation which is required in order to produce models that are smooth. Essentially this problem can be traced back to chapter 4 which also attempts to solve the anisotropy between coronal and axial/sagittal plane.
The method I used constructs any number k of contours from given C^n and C^{n+k} so that the shape of the newly generated contours $C^{n+1}, C^{n+2}...C^{n+k-1}$ becomes progressively more similar to C^{n+k}, starting with a shape close to C^n for C^{n+1}. This is an interpolation operation, essentially the same as the operation of the magnification filtering described in chapter 4.5.1. The smooth transition is generated between the corresponding control points of the cubic spline representation of the contours. These concepts (correspondence and spline representation) are all analyzed in detail in the next chapter.
Figure 5.14 is an illustration of the concept of interpolating between two contours in order to generate a third - the one in the middle. In reality the differences in shape between the contours derived from consecutive slices are not that visible. I used interpolation between contours for all cases where the slice thickness was over 3 mm, such as the clinical cases presented in chapter 8.

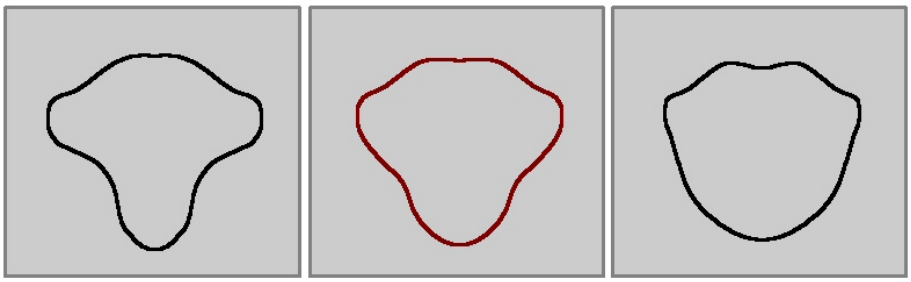


Figure 5.14: Interpolation of contours. Red contour (middle) is the result of interpolating the contours on left and right.

5.8 Discussion

This chapter has achieved its objective, which is to generate two sets of contours that describe the outline of the tongue structure in each of the VHP and MRI domains, as viewed in coronal plane. As a result, a direct comparison of the derived contours presented in figures 5.12 and 5.13, yields obvious similarities in shape[10].

In the VHP domain, the contour extraction is trivial and there is no scope for errors or mis-interpretation, as long as the VHP images generated in the previous step of the experiment are anatomically correct.

In the MRI domain, the problem, if an automated-only solution is required, is much more convoluted as the validity of the contours extracted also depends heavily on the quality of the images processed, of which very little can be assumed. However, if it is assumed that the MRI sequence is fixed, the most important factors that contribute to imprecise estimations are low contrast to noise ratio, absence of motion artefacts and tongue surface concavity. The first two lead to blurring and ghosting of structures, which restricts the delineation potential of the Canny algorithm and of the Active Contours. The latter may introduce problems with the ellipse/rectangle bounding mecha-

[10]The scale and offset differences are irrelevant in this comparison, since they can be trivially corrected. This will be demonstrated in the following chapter.

nism, which would force the procedure to follow the fall back path to manual setting of contour points. This, however, can be easily compensated for by asking the subject to push their tongue so that the dorsum covers the surface of the hard palate, which is concave. Then, without any air-pockets in-between the two structures, the dorsum will appear to have convex shape. Another problem area with the automated retrieval of contours in the MRI domain is that the cross sectional shape of the bundle of Genioglossus, Geniohyoid and Mylohyoid muscles (which form the root of the tongue) is very difficult to precisely delineate from the lower jaw and the anterior belly of the digastric muscle, which is a small muscle located under the jaw. There are instances where the distinction is very difficult even for the trained eye. A solution to this problem would be to perform a sagittal volume aquisition as well and to use this information to assist either the Active Contour or the end user who is manually placing the contour points. The sagittal view (especially a high resolution midsagittal plane) can illustrate the difference between the structures, as the tongue, dorsum, teeth and lower jaw are each viewed as a unified structure, as demonstrated in an example image in figure 5.15.

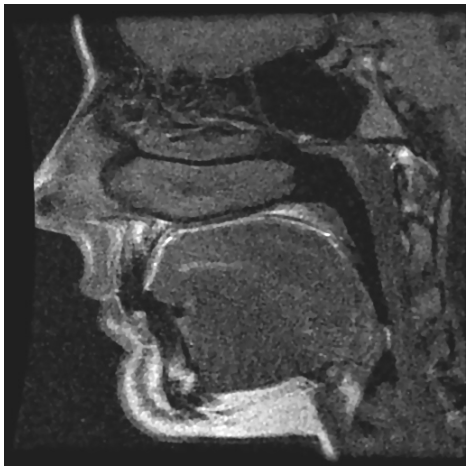


Figure 5.15: A sagittal view of the dataset. This view can potentially assist the segmentation performed by the user.

To summarize, the problem of detection of the contours in the MRI domain is convoluted, mainly because of the poor delineation between the different structures adjacent to the root of the tongue. In this chapter, some different approaches that deal with this problem from different perspectives have been presented: other than the obvious manual tagging (which, while giving the best results, is costly in terms of time required for a full volume evaluation[11]), an automated and semi-automated method was suggested.
My automated method, while being almost always successful in **localizing** the tongue contour in the presence of the tongue surface guide, it can have problems in the initialization phase of the Active Contour. Because of that, the Active Contour is more likely to converge to an erroneous outline. Thus, a manual evaluation cycle is required in many cases. As it stands, one could argue that the fully automated solution is actually incomplete, since it cannot guarantee an acceptable solution all the time, but rather most of the time. Additionally, my algorithm cannot classify which case is which, so it cannot say, for example, that in these slices the features in question have been successfully extracted, while in these they need a manual evaluation.
Regardless of this, with better MRI acquisition techniques (higher spatial resolution, faster acquisition - less motion artefacts - and better signal and contrast to noise ratios) the fully automated path can perform much more solidly. However, in the absence of a highly delineated tongue surface guide (i.e. when cavitatory contrast enhancement is not feasible), the localization is possible only through manual interaction. But once this step is performed, the Active Contours should approach the true outline of the tongue with less higher-level assistance.

[11] Of the order of 30-50 minutes per evaluation. Naturally, this number will vary according to experience and clarity of images.

5.9 Conclusions

At this stage of the experiment, both volumes in the VHP and MRI domain have been successfully processed, and that the contour of the tongue and the image of extrinsic muscles, for the VHP, and the tongue for the MRI are stored for later processing. A visualization of the work-flow, as achieved in this chapter, is presented in figure 5.16.
What follows in the next chapter is a description of the registration mechanism that I developed in order to fuse the two domains together and label the structures on the MRI volume accordingly.

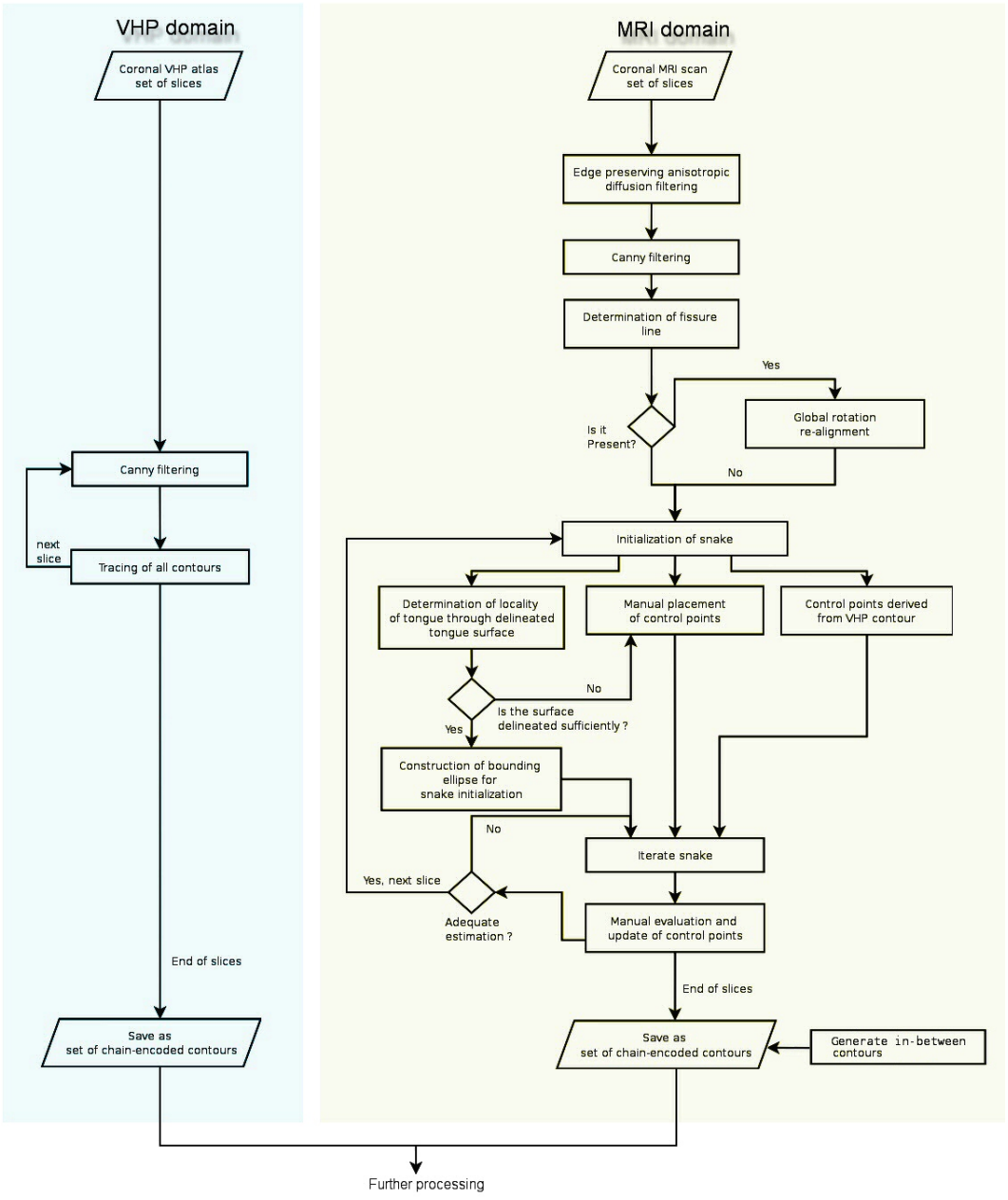


Figure 5.16: Overview of the work-flow of chapter 5.

Chapter 6

Soft body dynamics in registration

6.1 Introduction

In this chapter the mechanics of the fusion (or registration) of the Visible Human Project based atlas of the tongue and a given MRI volume are explained in detail. All the steps taken up to this point dealt with pre-processing of the VHP and MRI domains and extraction of the contour information that is the basis of my registration procedure.
This registration can be thought of as a black box with input and output:

- The input is a series of contours that signify structures in the two different domains, and the original domains in voxelspace.
- The output is a one-to-one mapping of the textural contents of the VHP contours, that carry the ID of the structures in question, over the MRI images. The implication of this is that it is possible to display the tongue as seen on MRI images, with the muscle groups denoted by, for example, different colour and superimposed over the MR images.

The registration procedure is responsible for finding this correlation, based solely on information provided by the association between the corresponding

contours between the two domains.

6.1.1 Organization of sections

For clarity, the chapter is divided into sections according to the sequence of tasks in the work-flow:

- Section 2 is introductory. The background theory of medical imaging registration in general and of my work specifically is established. It is noted that the problem demands a non-rigid registration scheme, such as the elastic registration that has been used in other, similar work.
- In section 3 I describe my concept supporting elastic registration and boundary matching. This is the core section of the chapter, and it touches many concepts and sequences of algorithms. All algorithms are described also in terms of their input and output, and how they fit in the general work-flow. The concepts described are based on published material and original ideas.
- Section 4 evaluates the implementation of the system and the results acquired using procedural and clinical data, and their significance to the project as a whole.

6.2 The problem of registering the two domains

Registration, in computer vision, is the process of approximating the point-by-point correspondence between two images [Gos05]. It is a mandatory process in comparing data obtained from separate measurements or, in the case of medical data, different modalities.

In more detail, registration is fundamental in dealing with the following applications [ea98b]:

- Scenes that are acquired for the same body in different modalities, for example CT with PET and MRI with functional MRI. The purpose is to link the functional information with the associated anatomic structures and vice-versa and combine them in a single presentation.
- Scenes of the same anatomical feature, same modality, but acquired at different times. This process helps to identify displacement in time, and hence is used for studying growth or motion (e.g. the movement of tongue during swallowing).
- In surgery, the registration of the body region and the devices used in the procedure is a vital navigational aid that establishes a frame of reference and motion between the body and the surgical apparatus.
- Scenes of a body region that needs to be correlated to a digital atlas that is artificially or otherwise constructed and describes features of the same region.

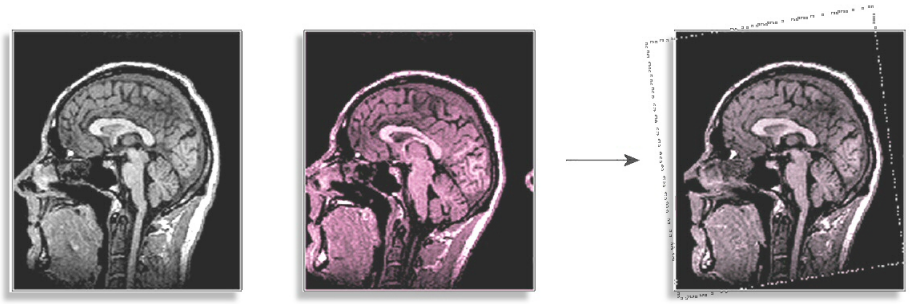


Figure 6.1: A registration example between two MRI images.

Registration, which can also be referred to as image fusion, is based on the definition of the set of transformations that transform an image of one modality (source image) to the image of the another modality (target image) [KK00]. Image registration attempts to correlate each point in the source image onto the corresponding point in the target image. Unlike calibration

procedures, where the two coordinate systems are in the same spatial and temporal domain [ea98b], in registration the coordinate systems can be totally independent. A visualization of a registration operation is presented in figure 6.1.

While it is impossible to develop a general algorithm that can cater for all uses, a general framework for registration is described in detail by Udupa [ea98b] and is based on the three steps described in the following subsections.

6.2.1 Definition of a relation between coordinate systems

Modalities A and B are associated with coordinate systems Ref_A and Ref_B. There exists a transform T that links the coordinates of A and B. T can be selected from different types of rigid or non-rigid (elastic) transforms:

- Rigid body transforms parameterized by Euler angles, rotation vectors or scaling [ea96b].
- Affine transforms (the combination of translation, scale, rotation and shear) on global or local scale [MVH94].
- Polynomial or spline based transforms [CSH89].
- Displacement vectors on regular grids [RS89].
- Local transforms with thin plates [Boo84].

6.2.2 Segmentation of reference features and definition of a disparity or similarity function between extracted features

The next step is to define the features of correspondence between Ref_A and Ref_B, and extract them by defining a similarity (or disparity) function be-

tween them. The definition of a "feature" can vary according to the nature of the problem. A "feature" is a geometrical shape, a point, a strong corner, a boundary curve or an intensity value. Depending on the nature of the features, the registration methods can be broadly classified into two groups:

Segmented feature-based techniques

These registration methods first segment structures in both modalities and correlate using point, curve, surface or volume matching [ea92b]. Some of the correlation metrics used are the matching of Fourier properties, the minimization of distances between matched points or surface to surface distances, or the fitting of interia axes of volumes and local moments attributes [LA92].

Image-based techniques

Image-based techniques directly match the intensity levels of the images without a need for a segmentation step. The correlation methods include cross correlation of the two images [ea86b], matching using the difference of the intensity values, or the difference between the filtered version of the images [JJ93], and matching using mutual information metrics such as the joint entropy [D.H94]. The disadvantage of image-based techniques is that they are inherently sensitive to noise and intensity differences between the images, and also slow (the cross correlation function between two images, for example, is an extremely time consuming process - a N^2 operation).

6.2.3 Optimization of the similarity or disparity function

For the registration between Ref_A and Ref_B to be valid, the similarity (or disparity) function must be maximal or minimal. The following non exhaustive list describes the most common methods, which are selected based on the choice of features. However, the association between features and optimization is not always straightforward.

- Pairing 2D and 3D points by rigid transform: a "least-squares" system (such as the singular value decomposition) can be applied to solve the optimization. In general, a rigid registration approach is inflexible and can only cater for a limited number of applications. It can be demonstrated that the affine-based methods cannot always provide a solution to a given registration problem: as a simple example, applying an affine transformation to an image that depicts a pair of parallel lines would result in an image with those lines remaining parallel. It is impossible to provide of a match between the parallel lines and an image where those lines are slightly bent (and hence non-parallel), since that is inherited in the nature of linear transformations (in the case described, a perspective projection could possibly cope). Affine transformation based fusion methods are usually used in matching multiple modality images of the same subject, derived from SPECT, CT, PET and MRI. When there is variability in the structure between the images, for example a known geometric "wrapping artefact" distortion that is common in MRI, then some extra non-linear (but still global) calibration transformations may be applied to counter-balance the curvature.

- Using the iterative closest point algorithm (ICP). The ICP [PJB92] works iteratively in three steps: 1) find correspondence between pairs of features in the two images, 2) estimate the rigid transformation needed to map the first member of the pair into the second, 3) transform all features in the first structures. The steps have to be repeated until a satisfying convergence level, based on the minimization of the euclidean distances between the pairs of features, is achieved. Given a good initial estimate, the algorithm is very efficient.

- Non-linear minimization, which can be derived through the analytic expression of gradients of the disparity function [ea92c], calculated using, for example, the iterative method of conjugate gradients (for machine implementation details consult [JD91]), or the generalized Kalman fil-

ter [And01]). The non-linear minimization solution is straightforward for rigid body registration, as there are only six parameters to be calculated. In the non-rigid scenario the displacement parameters are increased to several thousands and the evaluation becomes very slow and impractical. To compensate, a speed-up of the iterative search that is based on "octree-spline" distance representation would normally be used, as explained in [RS96].

- Physical modelling: an approach equivalent to least-squared minimization of distances is to use a framework based on elastic registration, which can be modeled using physical entities such as mass-points, spring and dampers between points and contours/surfaces. The principal idea behind elastic registration is to consider images as continuous entities and model the geometric differences between them such that they have been caused by elastic deformation [WFJ01].
  Another case in which elastic registration has been used is that of boundary based image registration, where there is no intensity level correspondence between the two images but rather a set of non-rigidly corresponding parametric curves or chains. [CDB96] proposed a solution where an iterative Gauss-Seidel method is used to drive the contour points (step 1) and shift the remaining image pixels (step 2), by using a Spring-Mass-Damper network to avoid pixel position folding and maintain smoothness and continuity. This technique is called Elastic Deformation Transformation (EDT). With EDT the scope is not only to attain correspondence between the pixels in the two pictures, but also to make the underlying deformation realistic, eliminating large or sharp deformations. In EDT, images are modeled as elastic sheets that are distorted by external forces applied to those points that are part of the boundaries mapped in step 1.

6.3 Elastic registration based on Boundary mapping

I have proposed, implemented and evaluated an approach to the registration of the VHP atlas into the MRI-derived contours, by first aligning the contours and then performing elastic image warping to map the textural content of the atlas over the MRI images. The textural contents of the atlas are then used to identify the musculature of any given voxel that lies within the tongue in the MRI domain.

The elastic, non-rigid registration path was selected as it is impossible for any rigid-body type deformation to manage with the potential structural variations that exist between different MRI scans. My procedure aims to remove this structural variation by matching the VHP atlas to each individual study in order to have a common coordinate system for further reference.

6.3.1 Overview

The procedure is mainly based on the work of Davatzikos et al. ([CDB96]), Bajcsy and Kovacic ([RS89]) and Peckar et al. ([PSRS96]). Below, I identify the main steps; I will then expand on the theory and implementation for each one:

1. Align the two contour datasets (VHP and MRI) in the Z-axis, as the registration scheme proposed in this chapter is applied on a slice-by-slice basis.

2. Associate the coordinate systems of the datasets, by setting identical bounding conditions for the VHP and MRI contours.

3. Map homothetically the VHP contour onto the MRI contour.

4. Apply an Elastic deformation to map the pixel contents of VHP contour over the MRI image.

6.3.2 Z-axis contour alignment

I assume that the alignment of the datasets in Z-axis is global, that is, each element in all coronal slices on X/Y is scaled in the Z dimension by an equal quantity.

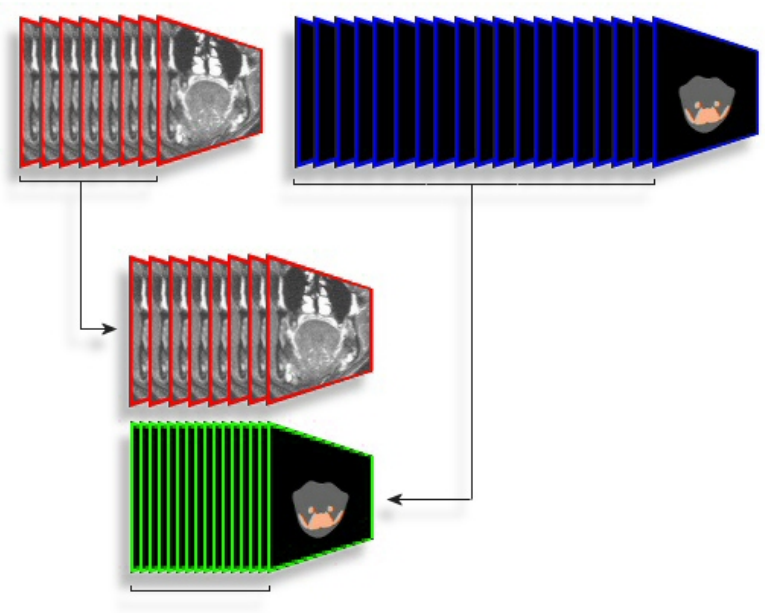


Figure 6.2: Z-axis alignment of the datasets. Blue set aligns with red by relative scaling and offset in the Z axis.

The Z-axis alignment is readily practicable: on the MRI volume the slices are identified that correspond to the two extremes of the tongue (tip, base) in the sagittal direction. In VHP, the extremes are fixed and already known. The 4 known quantities are then sufficient to solve the equations for the relative scaling and offset between the two sets: if the indeces of the two extremes for VHP and MRI slices are $i_{VHP1} < i_{VHP2}$ and $i_{MRI1} < i_{MRI2}$, then the slice correspondence is given by:

$$MRI_{slice}[n] = VHP_{slice}[(n - i_{MRI1}) \cdot \left[(i_{VHP2} - i_{VHP1}) \cdot \frac{(n - i_{MRI1})}{(i_{MRI2} - i_{MRI1})} \right]]$$

and since always $(i_{MRI2} - i_{MRI2}) < i_{VHP2} - i_{VHP2})$, that is, there are more slices in the VHP volume than in the MRI, the VHP is under-sampled in the Z dimension. Figure 6.2 shows the operation: the blue database (VHP), which has more slices is indexed (green) so that its slices correspond to the slices of the red (MRI).

6.3.3 Coordinate system transformation

A coordinate transformation is a conversion from one system to another in order to describe the same space. The two datasets, while expressed in the rectangular coordinate system, are still not aligned in terms of scaling and distance from the origin of the coordinate system.

The VHP contours are localized by translation and scaling so that the bounding boxes[1] of the two contours in VHP and MRI are identical. There is no need for rotation, as the VHP and MRI contours (after the coronal rotation compensation applied after extraction) are axis aligned.

The first step is to find the bounding boxes of the VHP and MRI contours. This is done by finding the minima and maxima for each axis X and Y, and thus determining the leftmost, rightmost, uppermost and lowermost points of the contours. If the 4 vertices of the bounding box for VHP and MRI, starting with the left topmost and winding clockwise, are:

$\overrightarrow{P}_{VHP1}$, $\overrightarrow{P}_{VHP2}$, $\overrightarrow{P}_{VHP3}$, $\overrightarrow{P}_{VHP4}$

and

$\overrightarrow{P}_{MRI1}$, $\overrightarrow{P}_{MRI2}$, $\overrightarrow{P}_{MRI3}$, $\overrightarrow{P}_{MRI4}$

then the vertical and horizontal bounding box spans for VHP and MRI are

[1] A bounding box is an axis aligned 2-D rectangle containing an object, so that no parts of the object can lie outside the rectangle. While there are potentially infinite bounding boxes for any given single object, I consider here the unique minimum bounding box that has width and height equal to the width and height of the object.

given by:
$VHP_{hor} = (\overrightarrow{P}_{VHP2} - \overrightarrow{P}_{VHP1}).x$, $VHP_{vert} = (\overrightarrow{P}_{VHP3} - \overrightarrow{P}_{VHP2}).y$
and
$MRI_{hor} = (\overrightarrow{P}_{MRI2} - \overrightarrow{P}_{MRI1}).x$, $MRI_{vert} = (\overrightarrow{P}_{MRI3} - \overrightarrow{P}_{MRI2}).y$

All $\overrightarrow{VHP}[n]$ contour points are mapped to lie inside the MRI bounding box by:

$$\overrightarrow{VHP}[n] = \begin{bmatrix} \frac{MRI_{hor}}{VHP_{hor}} & 0 \\ 0 & \frac{MRI_{vert}}{VHP_{vert}} \end{bmatrix} \cdot (\overrightarrow{VHP}[n] - \overrightarrow{P}_{VHP1}) + \overrightarrow{P}_{MRI1}$$

with n the index of any given point on the contour.
The operation is essentially identical to the Z-axis contour alignment, but in 2 dimensions. A visualization of the operation can be seen in figure 6.3 where two images with different bounding boxes (red and blue) are scaled and translated so that their bounding boxes align (green).

6.3.4 Homothetic mapping of the contours

At this point a one-to-one mapping between the VHP and MRI contours is created by finding the corresponding points in the two contours. I make the assumption that the contour of one domain can be transformed into the contour of the other homothetically, i.e., by a uniform scaling of its length and an arbitrary offset. This assumption normally yields very good correspondence points between homologous boundaries of anatomical features in similar studies [CDB96] and, in theory, can be extended to allow for non-uniform scaling.
The three-step homothetic mapping procedure is:

1. Select the sum of κ total corresponding points, based on the average length of the contours.

2. Create a Catmull-Rom cubic spline representation of each of the two

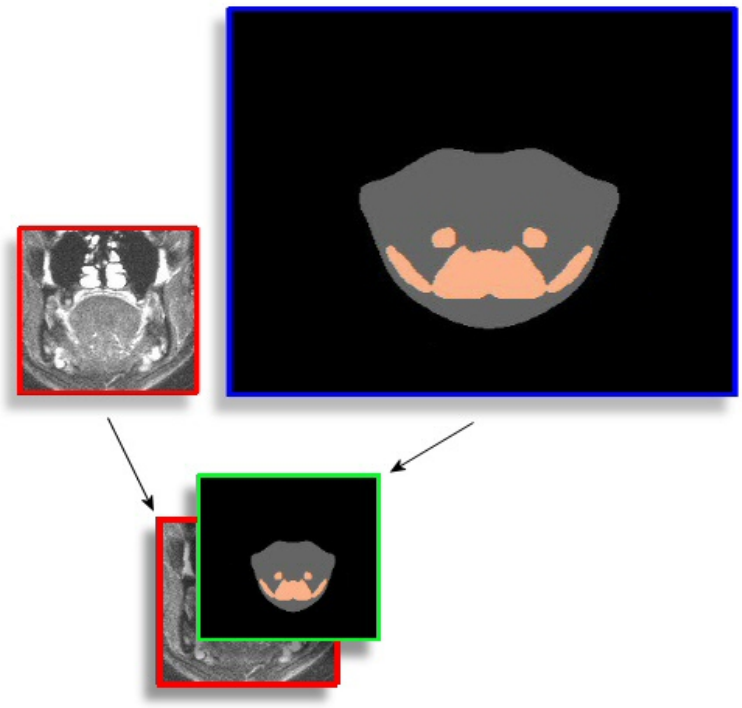


Figure 6.3: Alignment of the bounding boxes of two objects. Blue bounding box aligns with red by scaling and translation.

contours, and re-parametrize by a constant speed parametrization so that the κ points are evenly spaced along the contour.

3. Apply a Least Squares method to find the optimal circular shifting fit between the two sets of points, based on the curvature of the spline on each point.

The following subsection explains the theory of cubic splines and how it is possible to re-parametrize and create a constant speed spline.

Spline representation

For each chain encoded contour a parametrized curve $\overrightarrow{\gamma}(t)$ is produced, with $t \in [0, 1]$. $\overrightarrow{\gamma}(t)$ is a smooth, continuous approximation of the original

contour. The reason for doing so is that the original representation of the contour, which, in chain form is essentially a piecewise linear function, has only zero order parametric continuity[2]. However, first order continuity is at least needed, to ensure that the curvature of the spline for any t is possible to calculate.

Splines represent a curve by specifying a series of points (called "control points") at intervals along the curve, and defining a function that calculates the points within the intervals [Boo01]. Splines are defined piecewise by low degree polynomials: the most common form of spline is the natural cubic spline that is based on 3rd degree polynomials with at least first order continuity. I focus on the simple case of Catmull-Rom spline, which guarantees that the the curve $\overrightarrow{\gamma}(t)$ will pass through all of the control points - something which is not true for all types of spline[3].

Given the sequential control points $\overrightarrow{P_0}$, $\overrightarrow{P_1}$, $\overrightarrow{P_2}$, $\overrightarrow{P_3}$, then the interval points between $\overrightarrow{P_1}$ and $\overrightarrow{P_2}$ are given through:

$$\overrightarrow{\gamma(t)} = 0.5 \cdot [1, t, t^2, t^3] \cdot \begin{bmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{bmatrix} \cdot \begin{bmatrix} \overrightarrow{P_0} \\ \overrightarrow{P_1} \\ \overrightarrow{P_2} \\ \overrightarrow{P_3} \end{bmatrix}$$

A spline with κ control points can be constructed and any interval point is calculated as a function of the 4 neighbouring control points. For a close spline, the control points are calculated modulo-κ. For example, if $\kappa = 5$,

[2]Zero order parametric continuity implies that the first derivative function of the curve is not necessarily a continuous function. That is guaranteed in first order continuity, meaning that there are no discontinuities in the tangent magnitude and direction.

[3]Cubic splines, however, do not necessarily lie inside the convex hull created by their control points (convex hull is the smallest convex polygon containing the points). This often leads to "overshooting" which appears at intermediate points between tightly sampled control points with radically different gradient. Spline overshooting does not affect our experiment; nevertheless, the problem can be easily constrained by a hybrid linear/cubic solution [Kru].

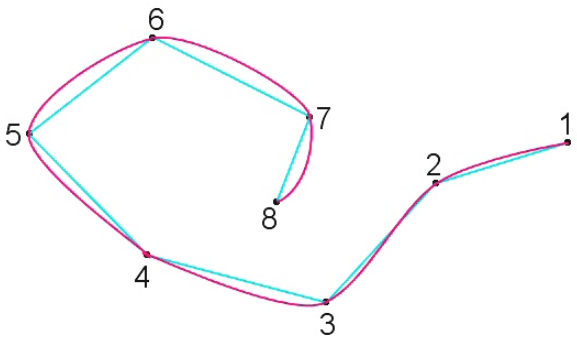


Figure 6.4: Polygon (cyan) and Spline (red), defined by 6 control points.

then the groups of 4 control points affecting each span are:

$$(0,1,2,3),(1,2,3,4),(2,3,4,0),(3,4,0,1),(4,0,1,2)$$

Selection of number of control points

I under-sample the tightly sampled contour points stored in the chain, in order to generate a spline that has a lower number of control points. The benefits are two-fold: a gain in terms of calculation speed (all the calculation-intensive steps from this point on are dependent upon the density of control points), and also an indirectly performed naive "contour smoothing" that is helpful in eliminating the contour wiggling that appears due to background noise.

I arbitrarily select the number of control points to be equal to 1/4 of the average of chain points of both VHP and MRI contours, based on trial-and-error experimentation.

Creating a constant speed spline

I defined spline $\overrightarrow{\gamma}(t)$ with κ control points, that, while sampled at fixed intervals for t, do not necessarily have equal path distances between them.

The path distance in this context refers to the length of the spline between the control points, rather than the Euclidean distance, which is the straight line connecting the two points. As such, the spline arbitrarily speeds up and down, since t and path distance covered in t are not related by a constant value throughout the spline.

The problem with using non-constant speed splines is that homothetic mapping can only be applied to points evenly spaced along the boundaries if a global scale and offset parameter is to be used (a function of spline density is otherwise required, which complicates the problem considerably). Additionally, the curvature (gradient) calculation has to change in order to accommodate for the non-linearities in terms of t.

There is no straightforward formula to calculate accurately the length of a given segment of a spline. I propose a simple, approximate method to create a constant speed spline from any given spline. The "black box" function receives a spline $\overrightarrow{\gamma}(t)$ with κ control points and an integer scalar λ, and returns spline $\overrightarrow{\gamma}'(t)$ with λ control points so that the spline path distance between each consecutive control point is constant. The steps in the procedure are:

1. Calculate overall length of spline by subdividing and approximating each segment as a series of linear segments. 8 Lines for each segment between 2 control points is sufficient. Divide overall length by λ to get desired distance between new control points, $Dist$.

2. Cache the distance from the origin for each of the $8 \cdot (\kappa - 1)$ start points of the linear segments (for a closed spline, first and last control points are the same).

3. Loop i through from 0 to λ to find the position of new control point $\overrightarrow{\rho}[i]$, that must be $i \cdot Dist$ units away from the origin.

4. Find, through the distance cache, the line segment with end points $\overrightarrow{P}_{start} \rightarrow \overrightarrow{P}_{end}$ with distances from origin d_0 and d_1 so that $d_0 <$

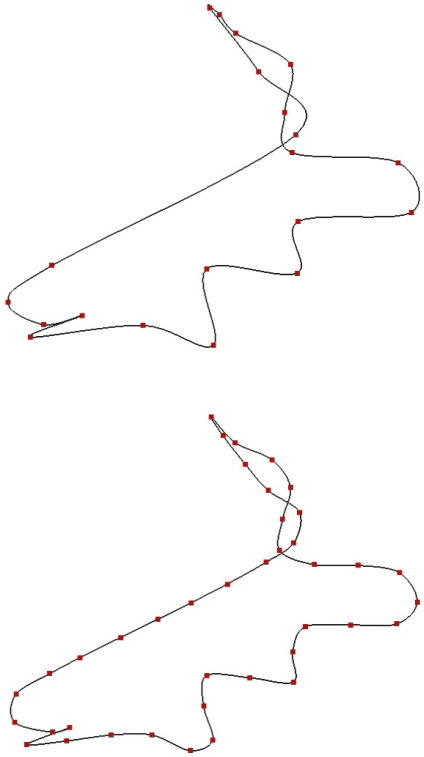


Figure 6.5: Example of a spline with control points at unequal distances (top). Bottom version is the constant speed spline.

$i \cdot Dist < d_1$. Calculate interpolation factor:

$$\phi = \frac{i \cdot Dist - d_0}{d_1 - d_0}$$

To search through the cache use either brute force iteration or binary search.

5. Linearly interpolate to find $\overrightarrow{\rho}[i] = \overrightarrow{P}_{start} + (\overrightarrow{P}_{end} - \overrightarrow{P}_{start}) \cdot \phi$.

The procedure, while approximate, yields acceptable results and is fast enough to iterate through tens of thousands of splines, with over 100 control points each, in a second. An example is given in figure 6.5.

Homothetic mapping of the splines

Homothetic mapping in mathematics is an affine geometrical transformation which dilates distances with respect to an origin point [Wei99]. If the origin is O and real number c (which can be also negative) the dilation factor, then the transformation of any point M to a point M' is given by:

$$M' = c \cdot M + O$$

At this stage of the problem, the two contours, VHP and MRI, are both represented as constant speed splines $\overrightarrow{\gamma}_{VHP}(t)$ and $\overrightarrow{\gamma}_{MRI}(t)$, with the total length of the splines being different. If a fixed number of corresponding points λ is to be matched in the two splines, then dilation factor c is simply the uniform scaling required so that the overall length of the VHP spline matches the overall length of the MRI spline:

$$c = \frac{Length(\gamma_{VHP})}{Length(\gamma_{MRI})}$$

Finding the origin parameter (or, "Circular shift", since this process is dealing with periodic 1-D functions) O is slightly more complicated. First I need to define below what "correspondence" means in this context.

Correspondence of control point: use of curvature as a similarity metric

I already referred to the "curvature" of the spline as a metric of the perceptually meaningful similarity between regions of different contours. I define a metric of similarity of curvature[4] as the dot product of the normal vectors

[4]While this method is sufficient for the purpose of this task, there is a plethora of other similarity methods for curve comparison in literature, such as Fourier descriptors and Similarity measurement. In their paper, [DL03] perform a more extensive analysis on the topic.

$\overrightarrow{N}_1(t_1)$ and $\overrightarrow{N}_2(t_2)$ of the two splines [SS04].

The normal vector $\overrightarrow{N}(t)$ of $\overrightarrow{\gamma}(t)$ is approximated explicitly using the central finite difference operator to find the tangent vector and cross product with $\overrightarrow{(0,0,1)}$. In discrete form, the tangent vector can be evaluated through convolution with the mask $[-1, 0, +1]$. As such:

$$\overrightarrow{N}(t) = \left| \lim_{h \to 0} \frac{\overrightarrow{\gamma}(t+h) - \overrightarrow{\gamma}(t-h)}{2} \right| \times \overrightarrow{(0,0,1)}$$

Determination of circular shift O

The problem now becomes: the circular shift O must be found so that the sum of residuals of $N_{VHP}(M) \bullet N_{MRI}(c \cdot M + O))$, with c known, for all λ different values of M in $[0, 1]$ is minimal. This is a common problem in mathematics, and it is solved through Least Mean Squares (LMS), an optimization technique which, given a series of measured data, attempts to find a "model" function that closely approximates the data [ea92c]. In order to do this, it minimizes the sum of the squares of the residuals (the ordinate differences) between the model generated points and the corresponding points in the data.
The formulation of the least squares problem for the two contours becomes: Find O that minimizes:

$$\sum_{i=1}^{\lambda} [N_{VHP}(i) \bullet N_{MRI}(c \cdot i + O)]^2$$

The problem can be solved either by using Brute force or a more elaborate method such as the iterative Newton's method for optimization:

- A naive "brute force" finds the best solution for O by solving the sum for all possible solutions for $O \in [0, 1)$. Since O is a real number, the search is quantized and restricted to, for example, 2 decimal digits, resulting in 99 total operations per pair of splines. This is a very straightforward approach to the problem, and although it is much slower than

the Newton's method, this operation - in the grand scheme of things - is one of the least computationally expensive. The resolution of the approximation, if set to 2 decimal digits, is also sufficient.

- The Newton's method in optimization:
  Newton's method is an algorithm for finding roots of equations in one or many dimension, and can be used for functions at least twice differentiable. The method can also be used to find local maxima and minima points, where the first derivative of the function is a root.
  It attempts to find O_* that minimizes

$$f(O) = \sum_{i=1}^{\lambda} [N_{VHP}(i) \bullet N_{MRI}(c \cdot i + O)]^2$$

  with the sequence (O_n) defined by:

$$O_{n+1} = O_n - \frac{\left[\frac{df(O_n)}{dO_n}\right]}{\left[\frac{df^2(O_n)}{dO_n^2}\right]}$$

  with $n \in [0, \infty)$. The sequence converges toward O_* after a number of repetitions. The bail-out condition is triggered when $|O_n - O_{n-1}| < threshold$, a very small number. The first and second discrete derivatives of $Curv$ are estimated by convolving $Curv$ with $[-1, 0, 1]$ and $[1, -2, 1]$ respectively [RFW00]. Initial guess for O is $O = 0$.

This method is much faster than the brute force method, but it only searches for local, rather than global, minima. However, I still have not found a situation where the local minima around $O = 0$ is not the global minima. This is due to the fact that the chain encoding for all the contours commonly starts at the top-left corner. This establishes a common frame of reference ($O = 0$) which can dramatically deviate only due to major shape dissimilarities and differences in global rotation. However, the shape dissimilarities are not that extensive and

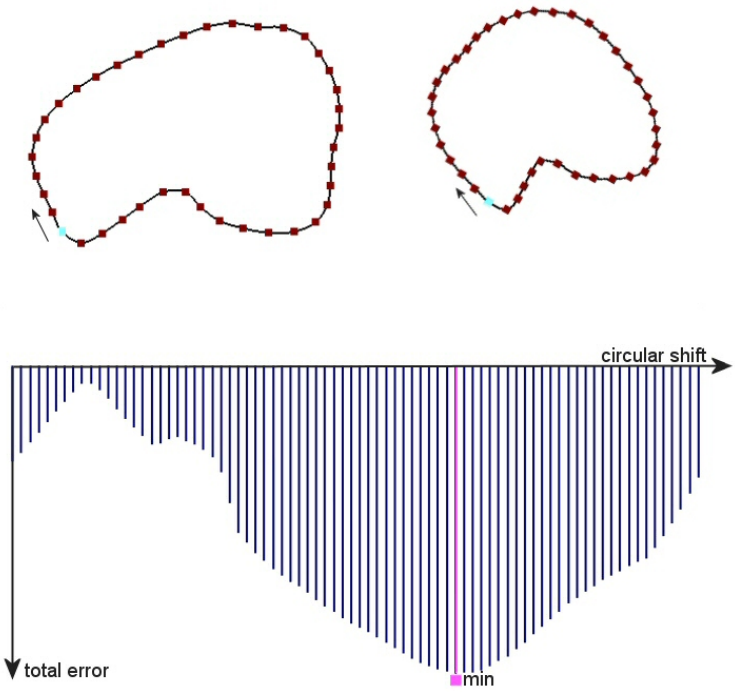


Figure 6.6: Homothetic mapping. Top: the two contours. Bottom: a graph showing the global error as a function of circular shift.

global rotation on the MRI curves is also compensated for at earlier stage.

In figure 6.6, two contours are compared. The first one is a slightly distorted version of the other, and is also rotated and scaled by a random value. The target is to find the correspondence between the control points. The graph underneath shows the sum of least squares as a function of relative shift. Figure 6.7 shows the mapping of one set of control points into the other, as a result of a circular shift applied after finding the minimum value.

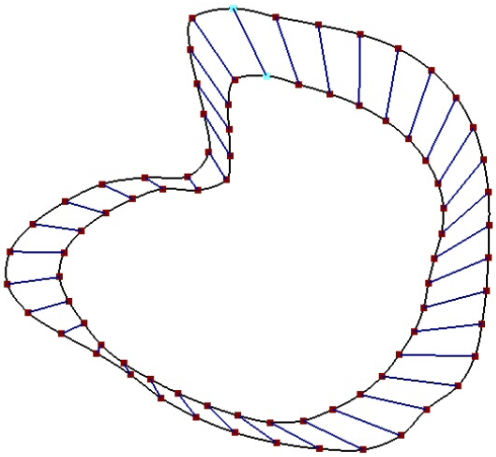


Figure 6.7: Homothetic mapping. The mapping of control points between the two contours.

6.3.5 Elastic deformation: Spring-Mass-Damper systems

This section introduces the concept of two-dimensional elastic deformation that I use in order to find the correspondence of features between the two sets of images. This approach attempts to emulate the change of relative positions of particles in a continuous, isotropic, homogeneous body when external forces are applied to it and deform it, until a state of balance is reached between the external forces and internal forces within the body resisting the deformation [RS89]. In this abstraction, the continuous body is the atlas tongue that is registered into the MRI derived tongue model and the external forces are the mechanism that deforms the contour of the atlas model into the contour of the MRI tongue model. As already described in chapter 6.2.3, the modelling of this system can be based on Hookean/Newtonean mechanics, with each particle interacting with each other through a network of Spring-Mass-Damper entities. This approach was followed in this work.
The description of this section follows a bottom-up approach, where the individual components of the system are first specified in detail and then linked

into more complex structures. The description of the methods starts with the Spring-Mass-Damper (SMD) unit, which links into a Spring-Mass-Damper network, which in turn is linked to the specific problem of this work.

Spring-Mass-Damper unit

In the simplest model, the configuration of the elastic unit would consist of a point-mass connected to a linear spring. If the mass is displaced, there is a restoring force (according to Hooke's law) that tends to bring the system back to equilibrium. However, as the mass passes the equilibrium point, it has acquired momentum which keeps it traveling beyond that position. The spring restoring force is now in the opposition direction. The Spring-Mass system's motion is an oscillation described by the simple harmonic motion equation. To prevent the oscillation a damping factor is introduced parallel to the spring (Voigt model of Viscoelasticity). Figure 6.8 is an abstract illustration of the model.

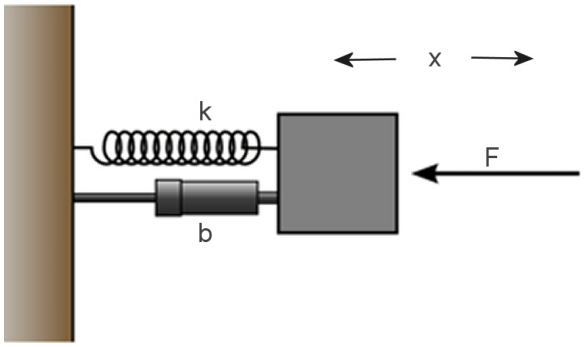


Figure 6.8: Illustration of a Spring-Mass-Damper unit.

Spring-Mass-Damper network

Moving into a higher level, in a Spring-Mass-Damper network model (a technique first introduced by [HB00]) the units are connected in a grid configuration: this system consists of a number of point masses connected through a network of springs and dampers (Figure 6.9). The solution to the posi-

tion and state of energies (kinetic and potential) for every point-mass in the system is given by iterative application of Hooke's Law:

$$\overrightarrow{F} = -k \cdot \overrightarrow{x}$$

where x is the displacement between the two mass-points at the ends of the spring, and k a constant describing the stiffness of the spring. If there is more than one spring connected to a given mass-point, then the forces are calculated for each one and added.

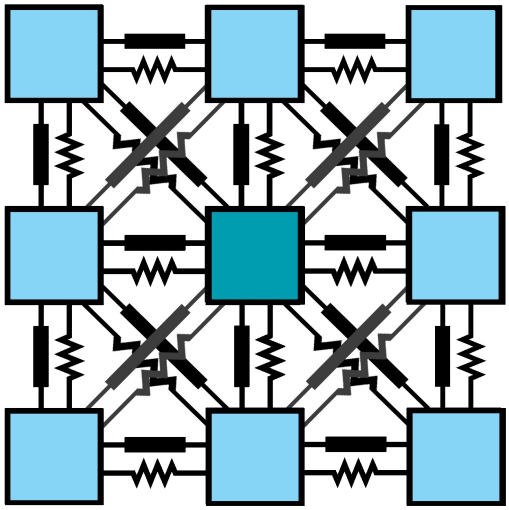


Figure 6.9: The Spring-Mass-Damper network configuration: interconnections of mass units through springs and dampers.

Introducing a damping factor into the equation results in:

$$\overrightarrow{F} = -k \cdot \overrightarrow{x} - b \cdot \overrightarrow{v}$$

where b is the coefficient of damping and v the relative velocity between the points connected by the spring. According to Newton's second law:

$$\overrightarrow{F} = m \cdot \overrightarrow{a} = m \cdot \frac{d\overrightarrow{u}}{dt}$$

The force exerted on the mass over a specified time step dt produces a resulting acceleration. The acceleration gives rise to a velocity, which causes the position of the mass-point to change. As a result, the length of each connected spring-damper also changes. The displacement of the point-mass is obtained through the equations of motion, for non-constant velocity and acceleration:

$$x = x_0 + \int u(t) \cdot dt$$

with velocity u given by:

$$u = u_0 + \int a(t) \cdot dt$$

A method is needed to integrate the system in order to find a solution at any time t. There is a number of different explicit and implicit integration schemes used in other studies [ea02b]. As examples I refer to the Explicit and Implicit Euler method, the Semi-Explicit Euler, the Midpoint, and the Fourth order Runge-Kutta method, which I use and explain in more detail in this document. For further information and implementation details, the reader is referred to [ea92c].

Explicit Integration - Fourth order Runge-Kutta

Euler integration is the most basic type of numerical integration technique. The method is usually used for solving first-order differential equations with a given initial value. When applied to the motion equations, the velocity and position are given by:

$$u(t_0 + \Delta t) = u(t_0) + \Delta t a(t_0)$$

and:

$$x(t_0 + \Delta t) = x(t_0) + \Delta t u(t_0)$$

It can be shown that the Euler integration is accurate in integrating velocity over time, only if the acceleration is constant [AS72]. The magnitude of the errors can be demonstrated by comparison to the Taylor expansion of $x(t_0 + \Delta t)$. If $x(t)$, $a(t)$ and $v(t)$ are known for $t = t_0$, then Euler integration gives:

$$x(t_0 + \Delta t) = x(t_0) + \Delta t u(t_0) + \Delta t^2 a(t_0)$$

while the Taylor expansion gives:

$$x(t_0 + \Delta t) = x(t_0) + \Delta t u(t_0) + 0.5 \cdot \Delta t^2 a(t_0) + O(\Delta t^3)$$

The difference between the two equations is the Euler integration error:

$$-0.5 \cdot \Delta t^2 a(t_0) + O(\Delta t^3)$$

which, apart from the Δt^2 term also has third order terms in Δt. A better alternative is needed in terms of residual error, one that can take into account the curvature in the derivatives when integrating a value.
One of the best numerical integration schemes that is commonly used in research and commercial software (i.e. physical modelling in computer games and virtual reality) is the Runge-Kutta order 4 integrator (RK4). RK4 has error $O(5)$ in the Taylor's Series expansion of the solution to the differential equation, and while it is not completely accurate (as the $O(5)$ error indicates) , it is reasonably simple and robust, and for these reasons I used the RK4 integrator in this experiment.
RK4 uses a trial step at the midpoint of the interval $t_n + h$ and calculates an approximation for y_{n+1}. It then uses a weighted average of approximated values of $f(t, y)$ several times within the interval $(t_n,\ t_{n+h})$. If the initial values are specified as:

$$y' = f(t, y)$$

and:

$$y(t_0) = y_0$$

then the RK4 provides a solution:

$$y_{n+1} = y_n + \frac{h}{6} \cdot (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4)$$

where:

$$k_1 = f(t_n, y_n)$$

$$k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2} \cdot k_1)$$

$$k_3 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2} \cdot k_2)$$

$$k_4 = f(t_n + h, y_n + h \cdot k_3)$$

With multiple variables, the RK4 algorithm remains essentially the same, and it is only the variables that become vectors.

Dynamic modelling of the SMD network

The modelling of the SMD network is an iterative process, in which the intention is to reach a potential energy minimization state, much like it is with Active Contours. There are further similarities to be drawn between the two schemes, as the Active Contours can essentially be regarded as exhibiting a range of elastic behavior, albeit in one dimension.
The basic steps that I follow in the SMD modelling iteration are:

1. Establish the density of SMD units per surface unit. In general, higher mesh densities increase the accuracy potential, but require more iterations, and hence processing time, to reach an equilibrium state.

2. Assign to all SMD units a starting position, values for the spring, mass, damper and rest length constants and an energy (kinetic and potential) state of rest.

3. Assign the inter-connections (thus creating a manifold) based on the locality of the SMD units. For each SMD unit, a list is drawn that contains all the other SMD units connected to it.

4. Introduce energy into the system via SMD units that are attached to a higher level entity that exerts a displacement force.

5. For each unit, iterate through all connected units and sum all forces exerted due to displacement from the point of rest. Update acceleration, velocity and finally position by means of explicit integration.

6. Update Δt by a relatively small amount. Large amounts for Δt improve the speed of operation, but may result in chaotic, uncontrolled oscillations that break continuum of the network.

7. Iterate through step 4 until the maximum displacement amongst all SMD units has dropped below a certain threshold.

The model has then reached a state of equilibrium between the units.

6.3.6 Construction of SMD mesh in 2D

I have already defined the SMD block as a unit that consists of a mass point which is connected to a spring with a parallel damper. A computer model of a discrete 2-dimensional network (mesh) of interconnected SMD blocks must be constructed, in which each SMD block is connected to a number of neighbour SMD blocks via the spring/damper connection. These rules are followed:

- All SMD blocks should lie within the contour(s). Each SMD is attached to a neighbor SMD, unless it is at a boundary position. In this case it is attached to the boundary contour.

- The SMD blocks should be isotropically placed. This is not a very strict requirement as they will move into a more isotropic equilibrium state in the first few iterations anyway, but if the blocks are stochastically sampled that may require more than a few iterations for the system to settle.

- Each SMD block carries a UV coordinate[5] over the VHP atlas image. The UV coordinates are simply found by linear 1-1 mapping of the XY position of the block on the plane (the transformation matrix is the inverse of the bounding box transformation applied to the contours so far). A texture look-up using the UV coordinates should map to the VHP atlas texel that lies under the SMD block at any given position, as the SMD mesh is projected on the VHP image. Hence, each SMD block is associated, through the texture lookup, with an ID scalar that defines the muscularity type of the underlying pixel.

- The operation should be irrespective of the SMD mesh density. The SMD mesh should evaluate for any given number of SMD blocks.

- The connectivity[6] is set to 8. This yields better approximations than 4- ([Mor98]) but at the cost of increased processing time.

- The mesh construction process is off-line, so there is no need to take speed into consideration.

I came up with four different solutions for building the SMD mesh. I tried all of the following, which yielded almost identical results, in terms

[5]UV mapping is a coordinate system for textures. U and V coordinates are the equivalent of X and Y for 2D space. UV coordinates are used in a 2D texture Tex [][] look-ups to fetch the texel at position Tex[U][V]. The concept can be extended to 3D textures (UVW mapping) or more.

[6]Connectivity defines how many elements can be connected to other elements. In standard 2D operations the connectivities can be either 4-way (elements are connected if their edge touch, along the horizontal or vertical direction) or 8-way (like 4-way and additionally if their corners touch along the diagonal direction).

of spacing, interconnection of units, and how long it took to generate the mesh. It is beyond the scope of this text to analyze extensively each method. Instead, I only outline the basic ideas and expand on the one I used in the final version of the program:

- Probably the most straightforward solution is to apply Delaunay [Del34] Triangulation[7] to the vertices over the contour, (described as a polygon), and to a number of stochastically or isotropically sampled points within the contour. The Delaunay method is often used in a similar fashion to build meshes for a finite element method, or to refine a mesh for modelling objects used in computer graphics. The difference of this method from the others described below is that the connectivity is not guaranteed to be fixed (it can, potentially, get any value from 2 upwards).

- Another option is to find an SMD block with the contour, and apply a seed fill [FD93] algorithm: a (usually) recursive algorithm that determines the space connected to a given node on an array. The seed fill (also known as "flood fill") is normally used in painting programs (bucket fill tool) in order to apply colour to an enclosed area. In my implementation, the initial block generates 4 block neighbours at east, west, north and south, all connected to the first one. Then each of these blocks spreads 4 more and so on recursively, until a boundary has been found. In this case, the connection is made to the boundary and no block is generated. The algorithm stops when no more blocks are generated.

- Use of span-lines: the contour is represented as a polygon on a discrete 2D map. A loop processes each horizontal line of the map and

[7] In this context, a triangulation is the subdivision of an area into triangles. In *general position*, the Delaunay triangulation is a unique solution, in which the circumcircle of every triangle generated does not contain any points of the triangulation. Delaunay triangulations maximize the minimum angle of all angles of triangles, thus avoiding sliver triangles.

constructs a simple span buffer, using ray-casting: a horizontal ray intersects the sides of the polygon to find the range of pixels that lie within. The SMD blocks are assigned to the position of these pixels, with connections made by looking at the span buffer of the previous and next line.

- Finally, the method I used and refer to in more detail in the subsection that follows, samples isotropically the 2D map, decides which pixels are inside the contour(s) and interconnects them according to their relative position. While I tried all of the methods presented in this section, I used the following method because of its implementation simplicity, and speed of operation.

Mesh generation in detail

To start with, the contour is represented as a finely sampled regular polygon on the X/Y plane. The density of points per contour is decided and then the contour is built as a sequence of small consecutive linear segments, each having the same length. It is sufficient to simply assign each segment of the original Freeman code to a single line, without the need to oversample, either through sampling more control points in between or by using, for example, splines.
Then a 2D map is constructed, that lies on the X/Y plane, and has a resolution that is a direct function of the desired SMD mesh density. A map of 128 $\times$ 128 is sufficient. All pixels in the map are iterated sequentially through , and checked whether they are within the contour or not. To do so, for each pixel a ray is cast from its origin, with direction towards $\overrightarrow{(1,0,0)}$ (or any other fixed direction on the X/Y plane). Then, each edge of the polygon is tested against the ray, using the ray/line intersection mechanism described in section 5.6.3. A counter is increased each time the ray crosses a boundary. If the counter is an odd number, then the pixel is surrounded by the polygon; otherwise it is outside the polygon. This method works for both convex and concave polygons.

Once it is established which pixels are within the contour, the next step is to define the interconnections. For each pixel/SMD block, a search in all 8 directions around it is performed in order to find the closest element that is tagged as part of the inside of the contour. A ray is also cast towards the same direction and tested against the contour polygon. If the central element already has a connection in this direction then the search is abandoned. If not, then there can be two cases:

- An element is closer than an edge of the polygon. A connection is then made between the two elements, and both elements are tagged as connected in the appropriate directions.
- An edge of the polygon is closer. The connection is established to the closest control point of the spline/polygon, in a clockwise and anti-clockwise fashion.

The three steps are represented in Figure 6.10: The original contour is in the top left image. Then the potential SMD blocks are sampled over the plane (top right). The ones inside the contour are kept (red colour - bottom left). The SMD network is built with these points (bottom right): red connections are between SMD units, and green connections between units and control points of the contour.

6.3.7 Mapping of UV coordinates

While the Spring-Mass-Damper mesh is spatially dense, there is no guarantee that there will be perfect coverage of point-masses over the texels in source and target images. A texel cannot be mapped to a point-mass and provide at least a one to one coverage (as demonstrated in figure 6.11) over a continuous texture without a mechanism that ensures that intermediate values are approximated regardless of the mesh density and resolution of the images.
For this purpose, I use a bilinear interpolation scheme between the UV coordinates carried by each of the 4 point-mass nodes that define a quad within

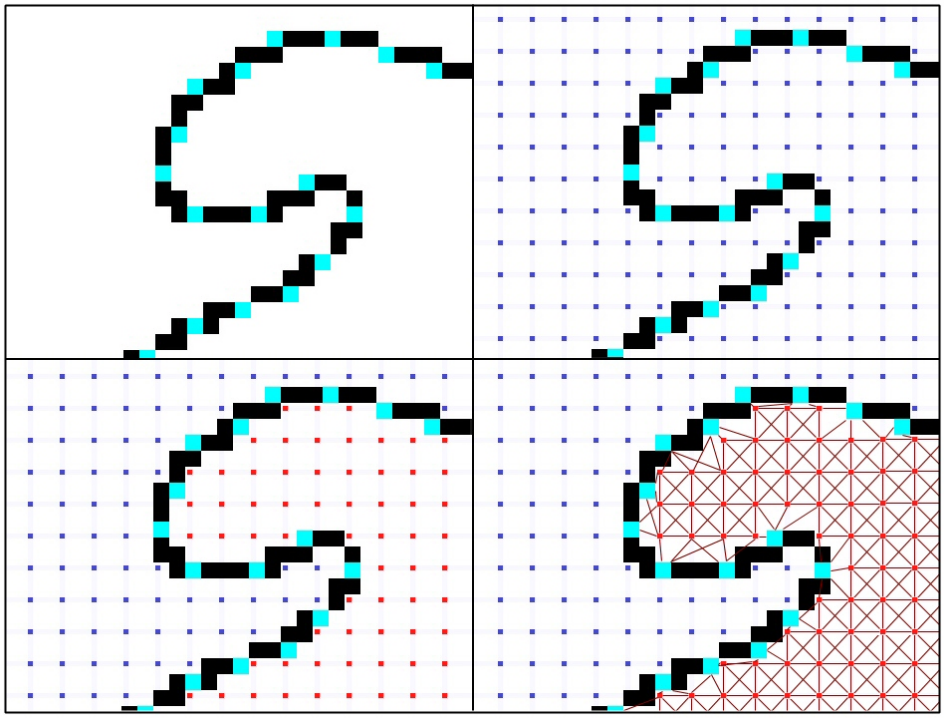


Figure 6.10: Construction of SMD network.

the mesh. The process of converting a vertex representation into a pixel representation is called rasterization and is the basis of computer graphics [JF95]:

- For every 4 nodes (vertices), split the quad into 2 triangles that are created from vertices 1,2,3 and 2,3,4.
- For every triangle (with vertices 1,2,3), create 3 edge lists expressed as sequence of points (X, Y) in 2D, with endpoints the pairs of vertices 1/2 , 2/3 and 3/4. The edge lists contain the interpolated intensity between the endpoint vertices.
- Generate in-between texels by interpolating the value of intensity of the points on the edge list that have the same scanline Y. For each scanline at Y, a new line is created with endpoints the points of the opposing edges.

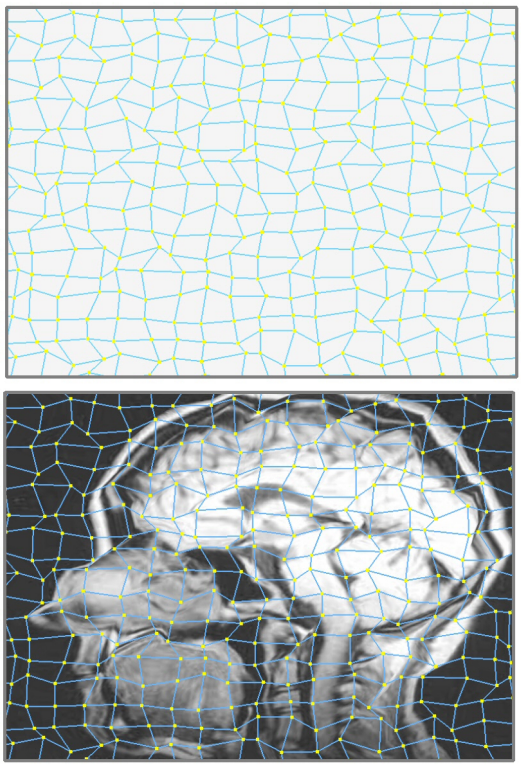


Figure 6.11: Rasterization of texture mapped quads. Top: point-masses constructing a sparse quad grid. Bottom: texture mapping applied for each quad.

Rendering of thousands of quads, depending on the implementation, can be a slow process. In the past, rasterizers used to run solely in software on the CPU and, depending upon the implementation, were susceptible to rounding errors: in order to increase the speed of operation engineers would use floating point arithmetic with insufficient resolution to cope with sub-pixel level accuracy.

Nowadays, with the advent of dedicated graphics processing units (GPU) and specialized protocols to communicate with the graphics cards, the rendering of computer graphics primitives (such as quads) is a straightforward

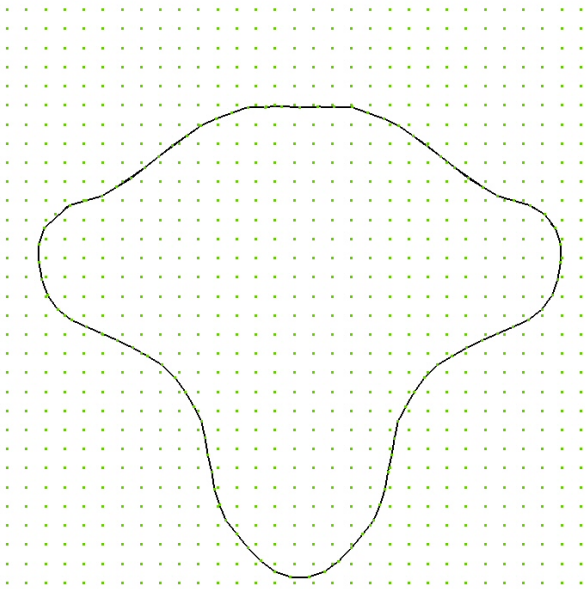


Figure 6.12: A VHP contour over a grid of potential SMD units.

affair taken care of by a specialized system.
I have used OpenGL, a common language for writing 2D and 3D hardware accelerated applications that is used from within our code-base as a "black box": it receives a list of quads defined each by 4 vertices carrying positional and intensity information and renders them into an image.

6.4 Result images

The 2D non-rigid registration scheme presented is serialized into "black-box" units, each of which was analyzed in detail in section 6.3. At high level, the target is to map a 1-to-1 correspondence between the voxels within the VHP tongue contours that carry the extrinsic muscles' ID information, and the voxels within the MRI contours that are yet to be identified. To do so, first a shape correspondence between contours (which express the outlines of the tongue in coronal domain, in VHP and MRI) is established (sections 6.3-6.3.4). Based on that, a textural correspondence for the information that is held within these contours is approximated (sections 6.3.5-6.3.9).

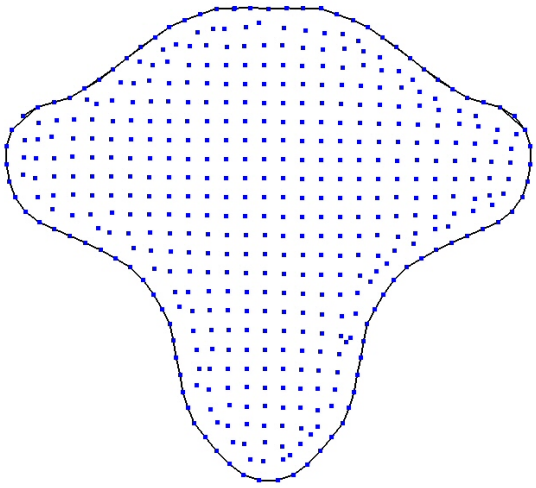


Figure 6.13: Highlighted SMD units that will form an SMD network.

The procedure can be seen in a series of figures, starting with figure 6.12, which shows a VHP contour over the grid of potential SMD units. Then in figures 6.13 and 6.14, the SMD units that lie within the contour are selected and form an SMD network.

After bounding box alignment, the VHP contour is homothetically mapped to the equivalent MRI contour. The SMD network iterates and reaches a state of equilibrium 6.14.

The mapping of the musculature ID information through texture mapping is shown in figure 6.15. The red colour denotes extrinsic muscles. The musculature type can now be projected over the original MRI image, as shown in figure 6.16.

It is clear from the two images that in the second contour, the side lobes of the tongue, as viewed in the coronal plane, appear to be wider than in the first contour. As a result, the Styloglossus muscle, which appears on the images as a dot on each side, is slightly scaled up on the second image. The Genioglossus muscle also appears to be scaled up and shifted upwards with respect to the Styloglossus. The side lobes that contain Styloglossus muscles

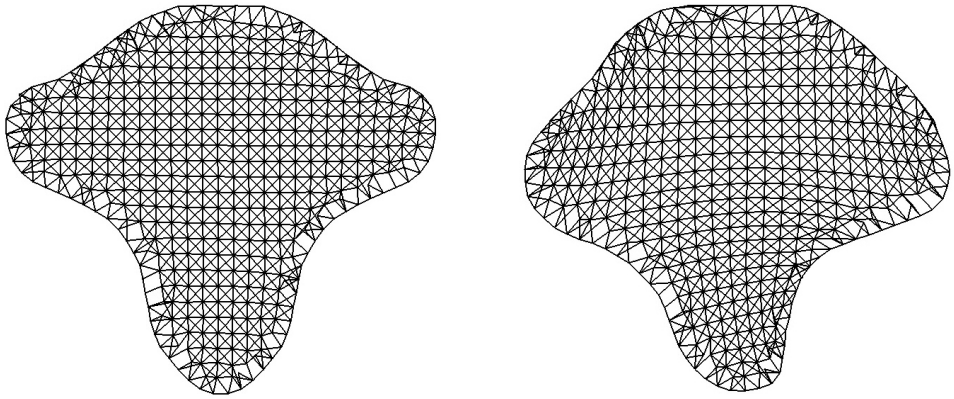


Figure 6.14: The SMD network, with highlighted interconnections between the SMD units. The left image shows the initial state, and the right the final state after mapping.

are bent downwards, pushing the Styloglossus further down with respect to the Genioglossus which is in the centre and is not affected by the lateral deformation.

6.5 Evaluation of methods and discussion

In order to evaluate the procedure as a whole, it would be good practice to split it down to the units that it is made of, and comment on their functionality and efficiency separately. This is practicable since the units are serialized, and hence the operation of one does not depend on the synchronous operation of another, but rather on its inputs alone. The "units" to consider in this case are the bounding boxes alignment of the volumes, the contour matching, the elastic deformation and eventually the UV texture mapping.

6.5.1 Bounding box alignment

The alignment of the two sets of contours is a step taken so that their bounding box describe the same volume in space. This is achieved through the Z-



Figure 6.15: Extrinsic muscles denoted with red colour. Mapping from VHP onto MRI contour.

axis alignment, which matches the two sets in the coronal direction so that the VHP and MRI blade tips and bases of the tongues are correctly aligned, as are the coordinate system transformation between the contours on each coronal XY plane. The Z-axis alignment is much more important than the bounding box alignment in the XY plane, as it dictates the correspondence between the contours of VHP and MRI that lie on the same coronal plane. If this is not correct, then the tongue models are mismatched and no calibration at a later stage of the registration is possible to rectify that error. Nevertheless, the process is very straightforward, with a very narrow margin for error. Both ends of the tongue (tip of the blade and base) are very easily identifiable even to the untrained eye. The only manual operation required is to identify the MRI contour that corresponds to a predefined position on the base, which is also the last contour on the VHP set. Of course, a similar requirement exists, in theory, for the tip of the blade. In reality, the two tips are reserved for contour set number 1: in VHP, it is part of the design of the model, and in MRI is the starting point, the first image to be segmented. This image is identified by the user as image number 1 before feeding the



Figure 6.16: Estimation of extrinsic muscles blended with the original MRI image.

images into the segmentation unit.

6.5.2 Contour matching

The contour of an object described in a image and saved as a chain code, has a starting point that depends on the position of the point of the contour that the scan-line search comes across first. The contour matching is a step required to find the relative offset between two contours that describe a very similar object and have an arbitrary starting position. Without contour matching, the result would be an apparently random and erroneous circular shift of the mapping of all UV coordinates between the two contours.
The metric of similarity used is curvature, which is suggested by other authors in similar work [CDB96], expressed through the tangent of the cubic spline representation. The casting of the polygon into cubic spline is necessary in order to acquire a good approximation of the original contour as an iso-speed closed curve (a requirement of homothetic mapping), and to enable calculation of the smooth gradient at any point along the curve. The

homothetic mapping works best with circular shapes, as every point has an almost unique normal. The potential and speed of the homothetic mapping was tested with different curves that were altered by a non-linear deformation and an arbitrary offset. The next subsection describes in detail the test I devised in order to evaluate the performance of the homothetic mapping:

Evaluation of Homothetic mapping

This aim of this evaluation is to show whether the homothetic mapping can be used between different splines that have an overall similarity in shape. This is the case between the contours of the VHP and the MRI sets.
The evaluation is performed in the following steps:

1. Start off with a closed spline, for example one of the contours of the VHP set. This spline is $\overrightarrow{C_0}()$.

2. Filter the spline by calculating a distorted version by means of linear and non-linear scaling, shearing, deterministic random noise etc[8]. This is spline $\overrightarrow{C_n}()$.

3. Interpolate between $\overrightarrow{C_0}()$ and $\overrightarrow{C_n}()$ thus creating n splines that represent the "morphing" of $\overrightarrow{C_0}()$ into $\overrightarrow{C_n}()$. n, therefore, represents the filter "strength".

4. Map homothetically $\overrightarrow{C_0}()$ and $\overrightarrow{C_k}()$, $k \in [1, n]$. Calculate the residuals for all circular shifts, by the method described in section 6.3.4. Find the median of the residuals and the optimal circular shift, which, in all cases, is a known quantity (it is zero). The distance of the optimal circular shift value (which should appear on the circular shift function as a spike) and of the median of the function is then a metric of the

[8]There is no point in testing translation as the mapping is translation invariant, as indicated in chapter 5.2.5

strength of the spike[9]. In this context, the metric will be referred to as "Homothetic mapping confidence" level, a scalar value. A high value indicates that the determination of circular shift is straightforward as the peak value is clear; hence the margin for error is very small. A value under 0 means that the mapping has completely failed, and the offset result is erroneous. Therefore the test can indicate the limits of the algorithm - the point after which it is not reliable.

5. Plot the confidence level scalars for all k to visualize the operation. As k increases, it is expected that the confidence level will drop, from a maximum value for $k = 1$ to a minimum value for $k = n$ and beyond.

I tested all VHP contours with different levels and types of distortion. The test case presented in this subsection uses 6 chosen contours from the VHP set (figure 6.17).

The distortion function used is represented in figure 6.18, where the original spline is on the left and the distorted version on the right. The distortion is a combination of displacement, global scale, skew and Perlin[10] noise. It can be noticed that the similarity is retained between the two splines, as both are loosely T-shaped.

The next figure (figure 6.19) visualizes the different splines $\overrightarrow{C_k}()$ as a surface of evolution, from the original (bottom) interpolated all the way to the final distorted version (top).

Finally, figure 6.20 shows the graph of the calculated confidence level over the filter strength for the 6 splines displayed in figure 6.17, with $n = 75$.

[9]Or, alternatively, can be calculated through the first derivative of the residual function around this point. There are potentially many different metrics of "signal strength" that can be used in this context. The one I used is sufficient for the scope of this test.

[10]Perlin noise [Per84] is a smoothly varying deterministic pseudo-random function over space and/or time that is constructed using spectral synthesis. Perlin noise can be defined in any dimension and is most commonly used for 2D and 3D applications. In this context, it is used to distort the contour points in 2D. The selection to use Perlin noise was based on the fact that it is simple to implement and very easy to control [Per02].

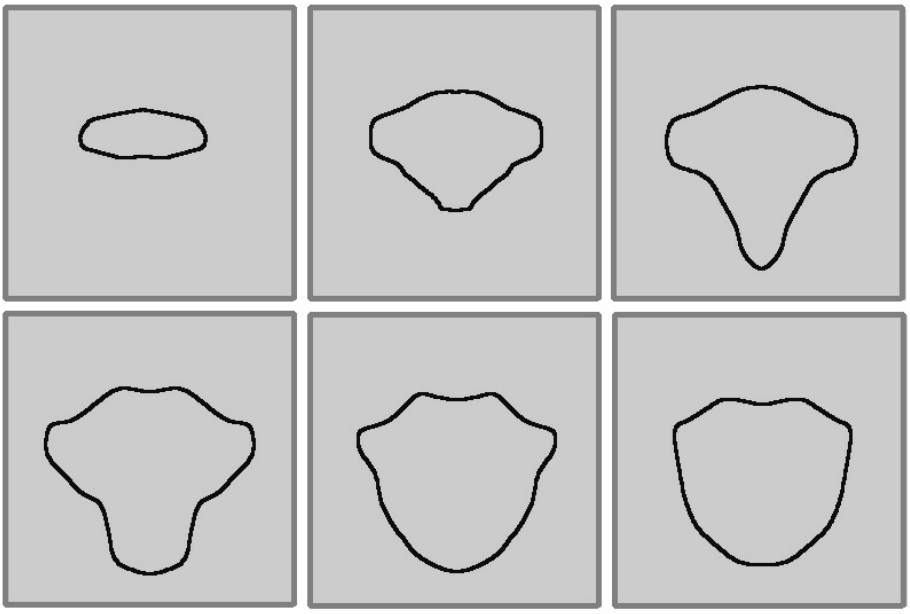


Figure 6.17: The 6 splines derived from the VHP contour set. Left to right, top to bottom: slice no.220, 200, 186, 142, 114, 100.

As expected, the confidence level[11] drops as n increases, albeit not always linearly. Also, the maximum confidence level is reserved for the contours with very simple shape, like Slice 220 which is an ellipsoid. In the more complicated case of Slice 142 and 186, the level even drops below zero, which signifies a complete fail.

Conclusions on Homothetic mapping evaluation

This test has proved that there is minimal scope for error. Even when the two shapes are only marginally similar, the method rarely fails. The only exception is when the structure of the curve is very complicated, which is not the case with the contours of either VHP or MRI.
On a side note, the speed of operation is also very adequate, well within the

[11]The confidence level is projected unnormalized on purpose, in order to show the relative levels between the different contours.

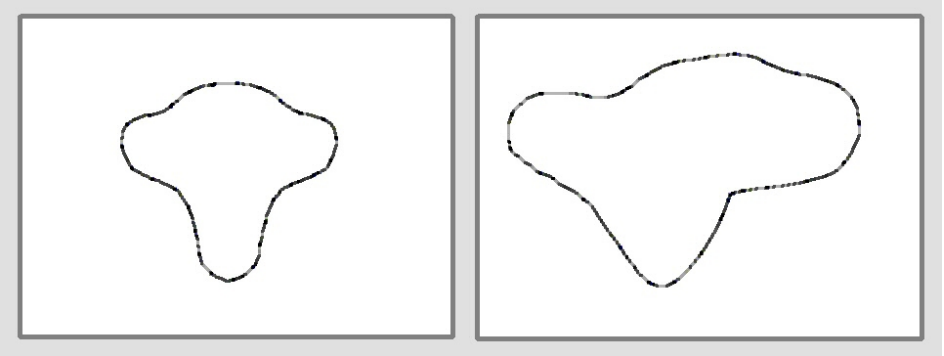


Figure 6.18: The original spline and its distorted version.

limits of real-time.

6.5.3 Elastic deformation based on SMD network

The image warping of the textural contents of the VHP tongue contour into the MRI tongue contour is implemented using a simulation based on a grid network of SMD units, as described in section 6.3.1. The iterative operation can be thought of as an abstraction of a mechanical or thermodynamic system, where force/energy exchange between units increases the entropy of the system until a balance is reached (or is perceived to have been reached, at a macroscopic level).

Technically, the simulation of the elastic operation of the units and their interaction with each other does not have many complexities. This is due to the careful selection of the integrator (the Fourth order Runge-Kutta), the fine discretization level of the variables used internally (64-bit floats), and small δt step, all of which minimize the errors due to "leak" energies/forces. These residual quantities are carried from one iteration to another and may throw the system off balance as they can introduce positive feedback paths, which result in uncontrollable oscillations between the units. Of course, the drawback of using high-precision integrators and, especially, small δt step is that the progression of the network becomes rather slow, and convergence may be reached in tens of iterations or more. However, for a network of a

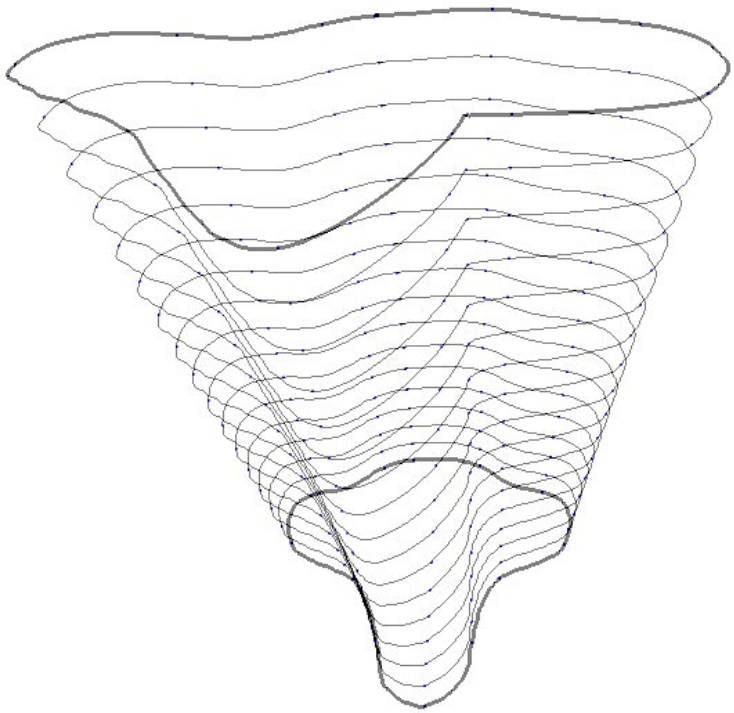


Figure 6.19: The different interpolated versions visualized as a surface of revolution.

few hundred SMD units, the operation is still within the limits of real-time, allowing for an almost instant mapping of VHP into MRI.

Evaluation of the Elastic deformation step

The method that I used to evaluate the operation of the Elastic deformation is very similar to the one used in association with the homothetic mapping described in the previous subsection.
Assume that the contour $\overrightarrow{C_0}()$ is the basis for the construction of an SMD network, the solution of which, after UV texture mapping, yields a 2D map $Map[x, y]$ of $Max_x \cdot Max_y$ unique scalar values. The contour is deformed by an external mechanism, like a linear transformation through a 2×2 matrix M, so that new contour $\overrightarrow{C_0'}() = M \cdot \overrightarrow{C_0}()$. The objective is to show that the new 2D map $Map'[x, y]$ produced based on $\overrightarrow{C_0'}()$ is very similar to

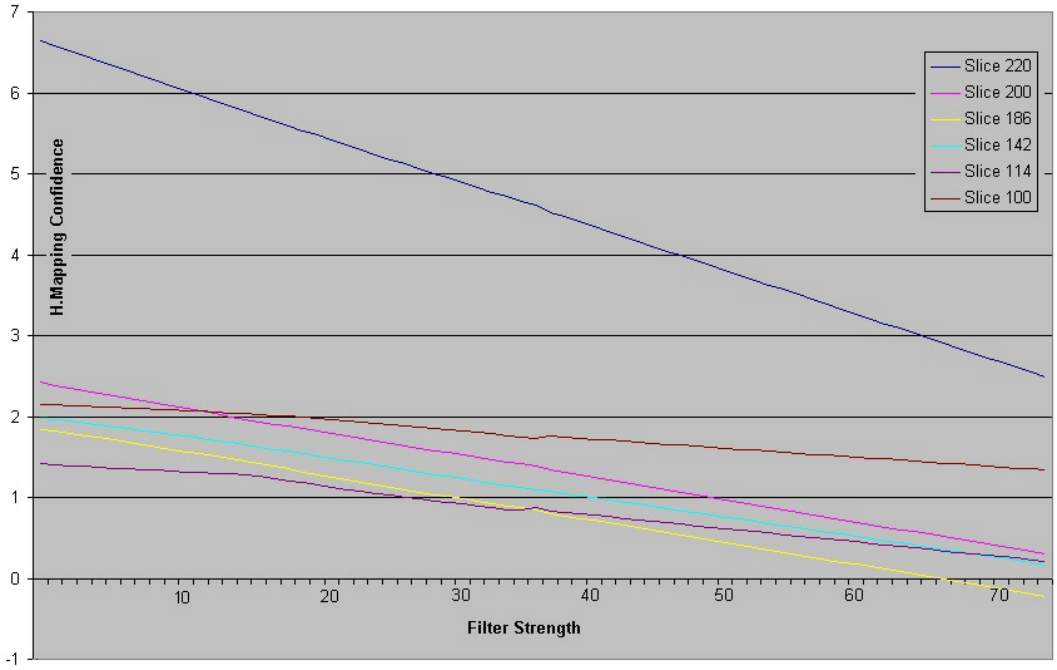


Figure 6.20: A plot of Homothetic mapping confidence level over filter strength, for 6 selected VHP contours.

$Map[M^{-1} \cdot (x, y)]$. The similarity metric in this context (or, rather, dissimilarity) is regarded as the average displacement of all the elements that have the same value in $Map[M^{-1}[x, y]]$ and $Map'[x, y]$, which implies that a perfect match would yield a similarity value of 0[12].
The point of this exercise is to see if the *approximation* achieved using the SMD network iterative procedure, which is essentially a fitting of data points (the map) to a sparse set of data points (the contour)[13], is satisfactory.

The operation is performed in the following steps:

1. Use one of the VHP contours to construct a SMD network. Assign a texture with all pixels having a unique ID.

[12]This metric is essentially the sum of the residuals of the approximants.

[13]This is the abstract concept; the difference of domain between the map and the contour is irrelevant here.

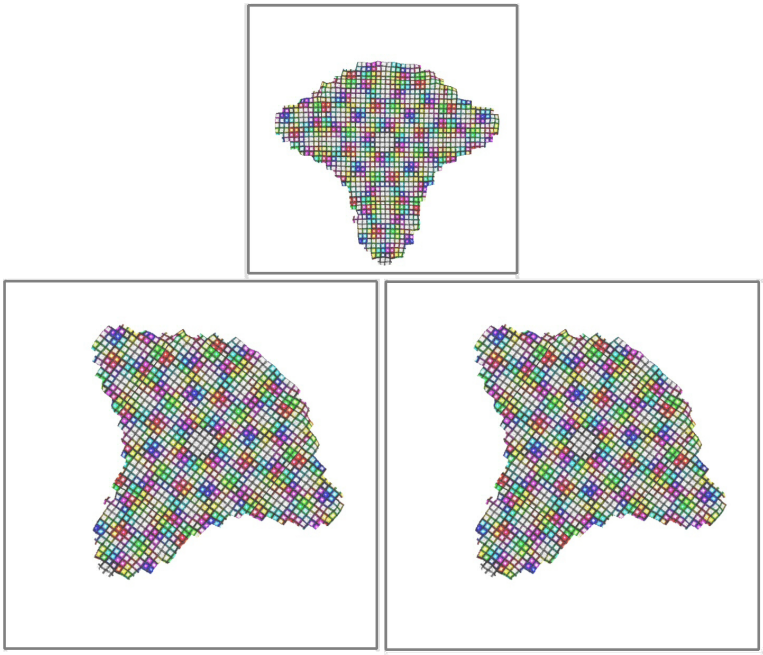


Figure 6.21: Elastic deformation evaluation. Original model (top), and solutions after rotation: elastic deformation (left), transformation of image (right).

2. Iterate and reach a state of equilibrium. UV texture map throughout the SMD network. Render and capture image in $Map[x][y]$.

3. Transform the contour by multiplying every vertex of the spline with a 2 $\times$ 2 linear transformation matrix M. Iterate until a state of equilibrium is reached.

4. UV texture map and capture image. This is $Map'[x][y]$.

5. Compare $Map[M^{-1} \cdot (x, y)]$ and $Map'[x][y]$ by finding the average displacement of all pixels that are the same in both maps. This can be done with a "brute force" algorithm, or through a modified optical flow map, as explained in a similar context in chapter 4.

The operation is visualized in figure 6.21. The top image shows the

model[14] prior to transformation (step 1+2). The image on the left shows the approximation achieved using the elastic model (step 3+4), after rotating the contour points. The image on the right shows the result of rotating the top image by the same amount of rotation. The two images are almost identical (step 5).

I automatized the test, so that all different VHP contours were tested with different types of linear transformation: translation, rotation, scaling, shearing and their combinations. The similarity was always found to be almost 0 - a perfect 0 was not reached due to trivial non-fixed point arithmetic inaccuracies found in most experiments dealing with floating point numbers in computing. This shows that the elastic deformation solution can, at least, be used to provide a solution to a linear fitting problem. However, is it possible to test it with the more common (to this experiment) *non*-linear deformations? The answer to this problem is yes: the sequence would be the same, as described in this subsection. The non-linearity could be introduced by means of a 2D displacement map based on, for example, Perlin noise. However, the similarity metric results would be nonsensical due to aliasing: the elastic model would only "sample" the deformation at the contour points, as there is no mechanism to provide any information regarding the state of the map anywhere else. Hence, the two solutions provided by the second rendering of the elastic model and the straightforward transformation of the original image of model would yield different results, as expected. Nevertheless, a validation of the mechanism with a non-linear deformation is presented in chapter 8 by following a rather different procedure (elastic deformation of a physical object - an ox tongue) that is not based on purely synthetic data.

[14]The colour texture used in this illustration is an abstraction, for visualization purposes only. The original texture is not RGB, but a 32-bit intensity that has a "white-noise" appearance.

6.5.4 UV texture mapping

The final step in the registration process is the mapping of texture information at pixel level, inside the sparse SMD network. This is done by rendering texture quads over the SMD grid, with all location, connectivity and UV information readily available. The rendering of the quads is done in OpenGL using the Graphics Processing Unit and is an extremely fast and well documented operation, common in computer graphics. As such, there is no scope for error or mis-interpretation of results at this step. The only thing to consider is the density of the textured quads: the bi-linearly mapped quads are essentially a linear approximation to a problem that is originally not linear. The level of approximation is determined by the number of quads per unit surface. This number is balanced by the required speed of overall operation: the amount of quads for rendering is a direct function of the number of SMD units in the network. Better approximation is achieved with more units, which, however, require more time to compute. After trial and error experimentation based on the validation method described in section 6.5.3, it was noticed that SMD networks with units sampled more densely than 3×3 pixels and then rendered with bilinear texture mapping gave almost identical results to the network that uses 3×3 pixel size quads. Therefore, the 3×3 sampling distance is used through-out the project. At this density, the number of SMD units in the network are in the order of a few hundred.

6.5.5 Conclusion

The registration method proposed in this chapter is based on a sequential operation, which is presented as a flow diagram in figure 6.22.
Each step of the process has been tested with procedural patterns or real-life scenario data in order to assess the validity and efficiency of the algorithms involved. This chapter uses as inputs the output of the data produced in the previous chapter, which must be valid, otherwise the registration will produce erroneous/nonsensical results. The same is true for the inputs of the next

chapter, which deals with the extension of the 2D mechanism described in this chapter into full 3D.

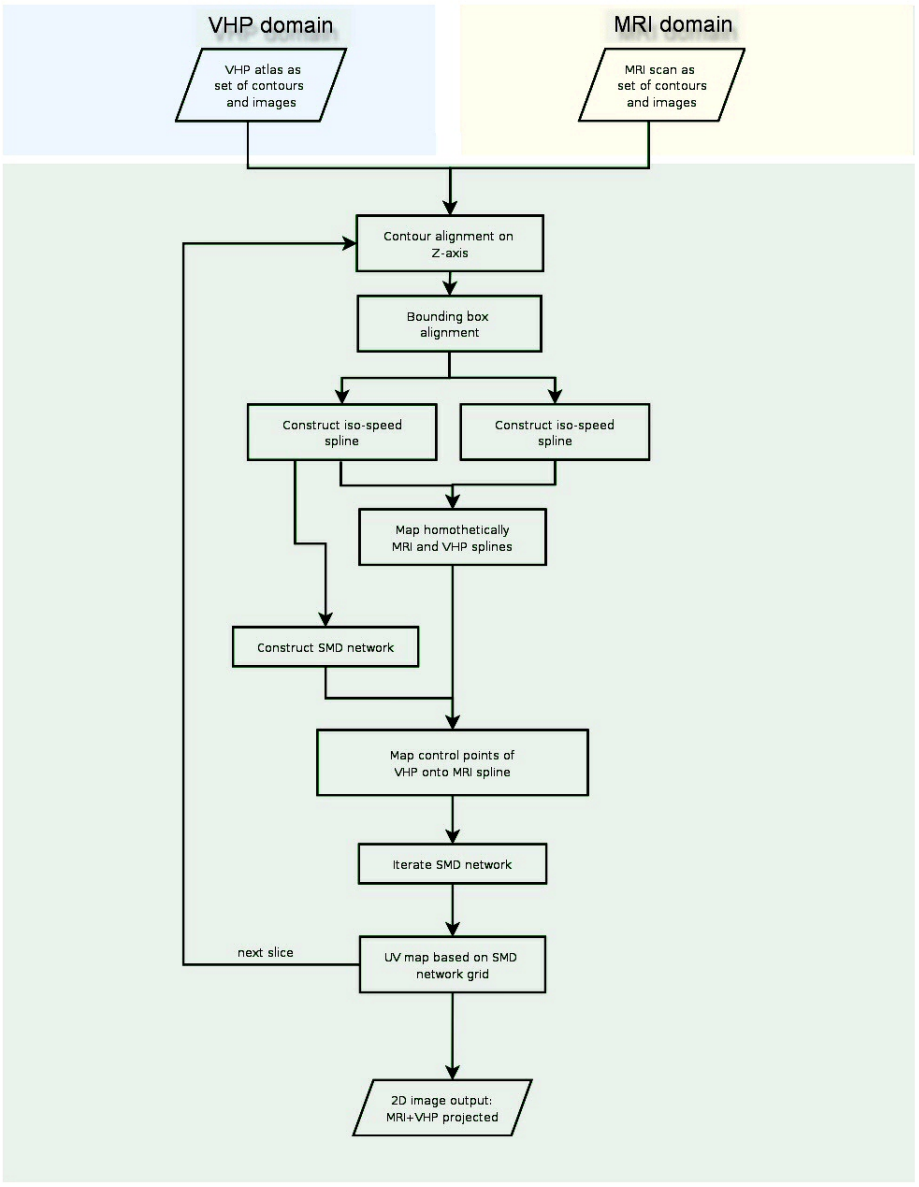


Figure 6.22: Overview of the work-flow of chapter 6.

Chapter 7

Extensions in 3D

7.1 Introduction

This chapter is conceptually a continuation of the previous chapter, which evaluates the underlying hypothesis and the solution given to the registration problem addressed in this work. The solution presented in the previous chapter essentially dictates that the two volumetric data are split into planes and sets of 2D contours, and then fused into each other, together with their textural content. The reason for this 2D approach is the fact that the tongue, if dissected in the coronal plane, does not separate into two or more components, as there are no air, bone or other tissue concavities to be found between tongue-only tissue (and this assumptions holds true for the VHP atlas and MRI scans tested in this work).

As such, the tongue can be modeled as a surface of revolution extending into the coronal direction, with a single closed curve needed to describe the projection of the object at any position along that direction. The benefit of this approach is that, having aligned the two objects in the 3D dimension (coronal) using a straightforward rigid alignment scheme (section 6.3.2), the rest of the process becomes more simplified. This simplification results in a much faster operation that requires limited computing resources and is also more flexible and intuitive if a manual override mechanism is required for

fine-tuning.

While the registration procedure presented is adequate for this experiment, it may be instructive to view the problem from a different perspective: what if the above assumption is not valid, and the object appears to have concavities even if projected on *any* plane? Would it be possible to extend the concepts already described to cope with this situation? An example is demonstrated in figure 7.1: the coronal projection of the sagittaly viewed tongue-like object in the upper left image is a single component. However, the dissection of a more complicated object (bottom left) yields two separated components. The algorithm presented in chapter 6 cannot deal with this situation.

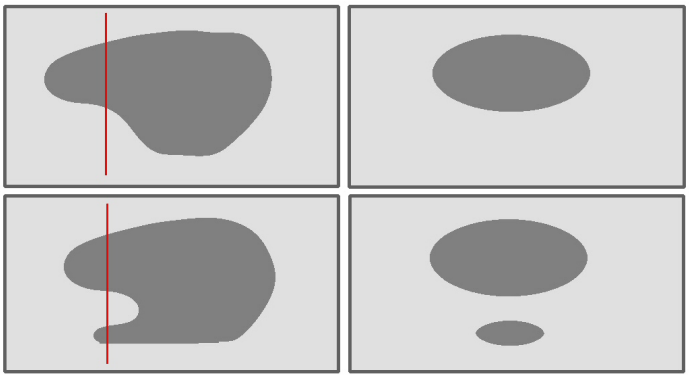


Figure 7.1: Coronal dissection of two objects. The bottom object produces two separated components.

This chapter investigates the potential of a full 3D non-rigid registration scheme, and investigates whether it would be more beneficial for the objectives of this experiment. The structure of the chapter is similar to that of chapter 6, which serves as a detailed reference for most of the background concepts utilized.

7.1.1 3D boundary mapping

The purpose of this version of the elastic registration scheme is the same as it was in the previous chapter, to register the atlas VHP texture contents into a given MRI scan, so as to identify whether a given MRI voxel within the tongue is part of the extrinsic musculature or not. The data used for this objective are the two sets of closed contours (one set for the VHP and one for the MRI) which describe the outlines of the objects, as viewed through a coronal projection.

7.1.2 Overview

The procedure followed in the 3D variant is very similar to the one proposed in chapter 6. The main steps are:

1. For each set of contours (in VHP and MRI domain) to produce a planar B-spline curve to approximate the closed surface that the contours describe. To align the two splines so that their bounding boxes are matched.

2. To map homothetically the VHP B-spline curve into the MRI curve using a similarity metric of curvature.

3. To apply the Elastic deformation so as to map the voxel contents surrounded by the VHP surface over the MRI images.

7.2 Generation of planar B-Spline curve

The problem can be formulated as follows. Given a set of stacked contours, how is it possible to generate a closed planar curve that:

1. Has a shape very close to the 3D surface that the stack of contours describe.

2. Has topological continuity (at least for Genus-0 cases) and a two-manifold. This cannot be guaranteed with a surface reconstruction mechanism such as the marching cubes ([SWC04] and [TLT03]) or with lofting ([Fer01]) or by using other surface reconstruction schemes such as the one proposed by Keppel ([Kep75]), in which the formulation is based on graph theory, and that by Fuchs ([HFU77]). For this reason, these algorithms are not considered for use in the context of the problem in hand.

3. Has continuous surface gradient, so that the curvature at any given point can be readily calculated.

These rules are identical to those set in chapter 6, where cubic splines were utilized to describe and compare contours. In 3D, clauses 2 and 3 can be satisfied using Bi-cubic uniform B-spline surfaces. The definition of the Bi-cubic spline is provided first, followed by a solution for clause 1.

7.2.1 Bicubic uniform B-spline surface representation

The bicubic spline is the equivalent of a cubic spline in 2 dimensions. It is an interpolation scheme that is used in this work to provide a smooth interpolant (rather than approximant[1]) over 3D vertices arranged on a surface. The result is the parameterization of the surface by variables u and v [Joy00]:
Given the 4×4 array of control points:

$$P = \begin{bmatrix} \overrightarrow{P_{0,0}} & \overrightarrow{P_{0,1}} & \overrightarrow{P_{0,2}} & \overrightarrow{P_{0,3}} \\ \overrightarrow{P_{1,0}} & \overrightarrow{P_{1,1}} & \overrightarrow{P_{1,2}} & \overrightarrow{P_{1,3}} \\ \overrightarrow{P_{2,0}} & \overrightarrow{P_{2,1}} & \overrightarrow{P_{2,2}} & \overrightarrow{P_{2,3}} \\ \overrightarrow{P_{3,0}} & \overrightarrow{P_{3,1}} & \overrightarrow{P_{3,2}} & \overrightarrow{P_{3,3}} \end{bmatrix}$$

[1]The difference is that an interpolant passes exactly through the data points. This is the case with Hermite interpolation at any dimension. The drawback is that the spline does not conform to the convex hull created by the original data, which can lead to "overshooting". This is, however, a minor issue with evenly spaced control points.

then

$$\overrightarrow{\gamma}(u,v) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \cdot M \cdot P \cdot M^T \cdot \begin{bmatrix} 1 \\ v \\ v^2 \\ v^3 \end{bmatrix}$$

where $u, v \in [0,1]$ and M is a 4×4 matrix of constants:

$$M = \frac{1}{6} \cdot \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

A surface inside a 2×2 patch is therefore controlled by 4×4 adjacent control points over the surface patch. The function can be constructed as a piecewise, so that $u, v \in \Re$, and $\overrightarrow{\gamma}(u,v)$ can compute any interpolated point over any patch of the surface.

7.2.2 Approximation of general shape of object with a B-Spline surface

There is much literature on surface fitting techniques in computer vision. In the context of this experiment I only consider the techniques that use dense data in the form of point clouds or a triangular mesh of arbitrary topology and produce a smooth surface approximation. They are broadly divided into two categories [KL96]:

- Manual and semi-automated techniques, including methods for digitizing a physical model directly through user interaction (using, for example, a touch probe) [Ost86]. Since these methods require user intervention they are time-consuming, especially when the models are

complex. Other manual techniques use conventional modelling systems linked with the scanned data that serve as a template to assist the construction of the smooth surface approximation [Ost86]. More automated methods still have a user-guided process to assist the projection of a base plane or surface to a specified region of the original data. Then the approximation for this region is done automatically, using a least squares fit method. A research system based on this principle is presented by [MJ95]. While these techniques normally produce the best results in terms of how closely the surface approximates the original data, they require a complex underlying modelling system and the assistance of a specialist in 3D modelling. For these reasons I decided to base my modelling on a more automated technique, which does not require high level intervention but, on the other hand, sacrifices a level of quality.

- There are a number of automated techniques that deal with fitting problems that have different constraints. One method ([EH96]) fits irregular meshes with a number of bi-cubic spline patches to produce surfaces of arbitrary topological type. In reviewing this work, Krishnamurthy and Levoy ([KL96]) remark that the technique produces high quality surfaces but includes a number of expensive optimization steps that make it very slow for an interactive system. Other techniques ([RH91], [DN89]) use hierarchical algorithms for fitting the parametric surfaces, but only work with parametrized, regular polygonal meshes. Other techniques proceed further to use displacement maps ([SP91]) that also work only on regular data sets.

The method I use is based on the work of Terzopoulos et al. ([DTK88]), Westin et al. ([ea96a]) and Osher ([SF03]), adapted to address the problem in hand. The overall concept is based on 3D Active Contours: a closed surface that employs energy minimization is guided by external constraint forces and is influenced by image forces, starts from a properly specified seed

shape and converges[2] towards the target shape through an iterative process. The image forces in this context are the VHP and MRI stack of tongue outline contours, rendered as filled contours and represented as voxels in the volumes. The procedure has to be repeated twice and from the start for each of the two volumes and there is no interaction between the two processes. The operation satisfies all clauses set in the previous subsection:

1. The closed surface is expected to deform and align itself with the salient image features of the volume so that its final shape approaches the shape of the object in the volume.

2. The two-manifold of the closed surface is defined at the initialization step and remains unchanged throughout the operation.

3. The gradient of the surface can be calculated through the bicubic spline representation that has continuous surface gradient and therefore the curvature of the surface at any given point can be readily calculated.

The following list shows the work-flow sequence for this task:

1. Render the VHP and MRI tongue as a separate volumetric object in a 3D volume of scalars, using one set value for the voxels that belong to the tongue and another for the empty space surrounding the object.

2. Initialize the shape of the 3D active contour as a two-manifold sphere-like polyhedron.

3. Iterate the snake by using the Greedy algorithm until convergence is achieved. The two bicubic splines are then generated based on the geometrical and topological information of the control points of their corresponding 3D Active Contours.

4. Finally, post-process the two splines so that they are bounding box aligned.

[2]The operation can be mentally visualized by thinking of a balloon with a smaller object inside it. The balloon then slowly loses air and snaps around the object within.

The next four subsections describe each of the tasks in more detail.

Construction of volumetric object

A volume of the dimensions of the original VHP or MRI volume is initialized with all voxels set to 0. Each contour is then rendered to its corresponding slice of the volume as a filled polygon, by first rendering the outline as a line strip and then employing a "seed fill" algorithm. The seed fill algorithm is fast and simple to implement and can cope with the convex and concave shapes of the outlines. The intensity of the fill is fixed for the whole volume, and is set to maximum ($2^{16} - 1$ for the 16 bit volumes used in this project).

Initialization of 3D active contour

The requirements for the initialization shape of the 3D active contour, as outlined in section 5.2.6, are that its shape should be coarsely similar to the shape of the target object, and that the density of control points of the curve should not deviate significantly (that is, their mutual distances must be regular). The shape that I use is a spherified cube that has the same bounding box as the volumetric object. In this case, the shape similarity clause is satisfied because a spheroid is an adequate approximation of any tongue model[3]. The density clause is also satisfied[4] because, unlike a real sphere whose point density changes as a function of the azimuth, a spherified cube is an almost fixed density of control points.

The spherified cube is produced by first making a two-manifold cube out of 6 planes of interconnected control points and then normalizing the vector position of each control point in the object. The result can be seen in figure 7.2. Another method that has been suggested in the literature and gives similar results is to construct a quasi-regular geodesic sphere by

[3]And also of most internal anatomical features and organic patterns found in nature.

[4]It must be mentioned that the shape created is not perfectly regular. There is no regular polyhedron with more than twenty vertices, as proved by Plato. The shape proposed, however, is a good approximation.

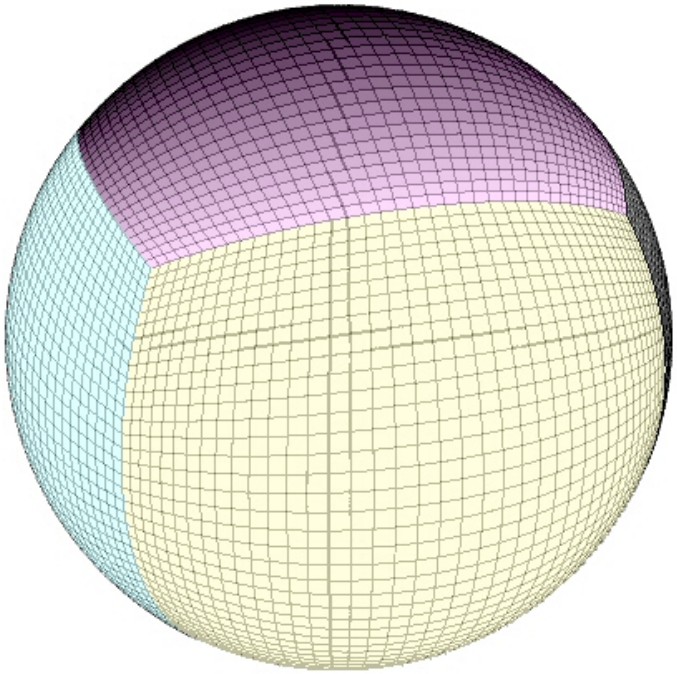


Figure 7.2: A spherified cube, with the different faces of the original cube colour-coded.

repeating a mesh smooth operation that employs polygon face subdivision on an icosahedron. However, my explicit solution is more simple and faster as it does not require the additional step of recursion.

Iteration of the 3D active contour

With the snake initialized the next step is to iterate until convergence is achieved. The snake mechanics are essentially as defined in chapter 5, with some difference in the way the energy of the contour is modeled and in the representation of the attractor image, which is now a volume. This subsection describes only the changes made to the original model, which is referred to in detail in chapter 5. For the implementation of the 3D active contour

I consulted the work of Ahlberg ([Ahl96]) which describes a Matlab based system in detail.
The original equation of the snake, described by a function $v : [0,1] \times [0,1] \rightarrow \Re^3$ retains its structure in 3D:

$$E_{snake} = \int_s r E_{snake} ds = \int_v (E_{internal} + E_{image} + E_{constraint}) ds dr$$

with Kass definition changed to:

$$E_{internal} = \alpha_s(s) \left| \frac{dv_s}{ds} \right|^2 + \alpha_r \left| \frac{dv_r}{dr} \right|^2 + \beta_s \left| \frac{dv_s^2}{ds^2} \right|^2 + \beta_s \left| \frac{dv_r^2}{dr^2} \right|^2 + \beta_{sr} \left| \frac{dv_r dv_s}{dr ds} \right|^2$$

In this equation α_s and α_r are the elasticity along the s and r axis of the surface. β_s and β_r are the corresponding rigidities, and $\beta_s r$ the resistance to twist. Having defined the internal energy function, the next step is to define the image forces that balance the internal forces at the point of convergence. The image considered here is a 3D image $Im(x, y, z)$.
In the original equation

$$E_{image} = E_{line} \cdot w_{line} + E_{edge} \cdot w_{edge} + E_{termination} \cdot w_{termination}$$

the last term is abandoned as there are no free ends (terminations) in the model. The term E_{image} is the image function $Im(x, y, z)$. The E_{edge} term, which expresses the magnitude of the image gradient term, similarly becomes:

$$E_{edge} = - \left| \nabla Im(x, y, z) \right|^2$$

The active contour iterates through a Greedy algorithm implementation and deforms to conform with the contour of the object in the volume. When the number of snake control points that moved in each iteration has dropped

below a fixed threshold specified before the operation, the snake stops. The shape of the contour now approximates the shape of the volumetric tongue. The procedure is repeated for both VHP and MRI tongues, with exactly the same parameters for the snake (weighting factors, density of control points and termination criteria). Figure 7.3 shows the resulting mesh for the VHP domain.

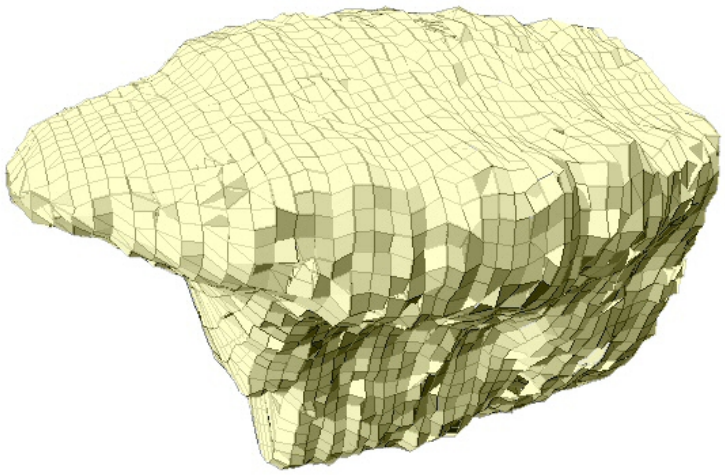


Figure 7.3: The resulting mesh after the iteration of a coarse tessellated 3D snake on the VHP set of contours.

Post-processing: bounding box alignment

The two bicubic splines produced in the previous step, one describing the VHP atlas and the other the tongue derived from the MRI scan, are not aligned in terms of scaling, which means that they do not necessarily occupy the same 3D space. A coordinate system transformation is needed to ensure that the new bounding boxes of the VHP and MRI splines are identical. The bounding box alignment mechanism has already been provided in section 6.3.3 for 2 dimensions, and can be simply extended for 3D.

Similarly to the procedure of chapter 6, the next step now is to map homothetically the two derived splines.

7.3 Homothetic mapping of the bounding surface

This task is based on an extension of the theory of homothetic mapping, already referred to in section 6.3.4, into 3D. The objective of this task is to create a one-to-one mapping between the corresponding control points of the VHP and MRI derived splines. "Correspondence", in this context, denotes the matching of points of homologous anatomical features between the two domains. Also, the assumption made in section 6.3.4 is still valid, that the spline of VHP can be transformed by uniform scaling and a 2D offset (as opposed to the 1D that was used with the 2D contours). The operation can be thought of scaling and shifting one closed surface to match with another that describes a similar feature.
The two-step procedure is:

1. To generate a bicubic spline representation of each mesh, so that is possible to:

2. Transform each mesh to have a total of κ evenly spaced vertices, a value based on the average surface between the two meshes.

3. Perform an optimization method to find the optimal homothetic mapping factors based on a curvature metric of the spline on every vertex.

7.3.1 Selection of number of control points

The number of overall control points for each mesh is selected based on two criteria: calculation speed (the fewer control points, the faster the overall operation), and whether a refined (more polygons) or a more coarse (less polygons) representation is needed for the experiment. The overall tongue

structure, being not that complicated and hence requiring less detail to describe adequately, calls for an overall elimination of control points. A factor of 3/4 reduction (for every 4 control points, 1 is saved) is sufficient.

Creating a constant speed B-Spline

The control points of the splines at this step are already placed reasonably regularly and therefore do not require an additional step to make them iso-speed. This is provided without any computational penalty by the active contour process in which the control points have a tendency to align themselves so that the distance to their neighbouring points does not deviate from the average distance of neighbouring points in the whole contour. As such, the average distance is determined mainly by the overall length of the contour and the number of control points.

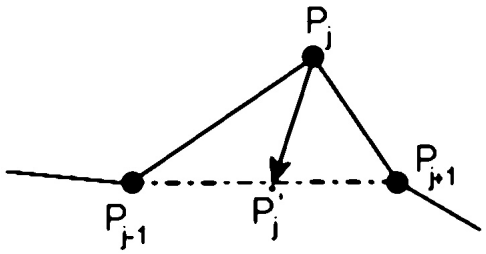


Figure 7.4: Displacement of control point to the average of its neighbouring points due to elasticity forces.

This phenomenon can be explained by considering the effect of elasticity (or tension, which is the first order continuity factor) in the active contour. Elasticity can be thought of as the dragging force that moves a control point to the average of the neighbouring control points, acting like a membrane. As the point is forced to its new position, the distance between itself and the previous and next point becomes equal. The operations can be seen in figure 7.4: the new position of point P_J is P'_J and is set at the average of

P_{J-1} and P_{J+1}, because of the elasticity force. The effect of this force can be generalized to higher dimensionality.

Curvature as a similarity metric

At this point, the two splines (MRI and VHP) have equal numbers of control points which are spaced equally over the surface they describe. The topology of the manifold is also known.

The next step in the process is to map the two surfaces by means of 2D dilation and 2D shift, so that a maximum possible similarity summed for all the pairs of points between the two surfaces is achieved. First, it is important to define the similarity metric.

As with the 2D contours, the metric of curvature is used. First, the two tangent vectors $\overrightarrow{T_u}(u,v)$ and $\overrightarrow{T_v}(u,v)$ are approximated explicitly using the central finite difference operator:

$$\overrightarrow{T_u}(u,v) = \left| \lim_{h \to 0} \frac{\overrightarrow{\gamma}(u+h,v) - \overrightarrow{\gamma}(u-h,v)}{2} \right|$$

and

$$\overrightarrow{T_v}(u,v) = \left| \lim_{h \to 0} \frac{\overrightarrow{\gamma}(u,v+h) - \overrightarrow{\gamma}(u,v-h)}{2} \right|$$

It can be seen that the tangent at (u,v) is defined by two vectors (that define a plane) rather than one. The two vectors $\overrightarrow{T_u}$ and $\overrightarrow{T_v}$ are linearly independent, and can be used to determine the normal of the surface at (u,v). The normal is found through the cross product of the two tangents:

$$\overrightarrow{N}(u,v) = \overrightarrow{T_u}(u,v) \times \overrightarrow{T_v}(u,v)$$

which can be used as a measure of "curvature", based on the direction of the face rather than the gradient of the curve. The similarity metric between two normal vectors $\overrightarrow{N}$ and $\overrightarrow{K}$ is then set as their dot product, which is maximum (1) for the case $\overrightarrow{N} = \overrightarrow{K}$, and minimum (-1) for $\overrightarrow{N} = -\overrightarrow{K}$ (all vectors being unit vectors).

Homothetic mapping - determination of circular shift

Having defined a similarity metric between the normals of the two surfaces, the problem follows the solution given in section 6.3.4: an optimization problem, for which the following function of O_u and O_v (circular shift scalars for u and v) is minimized:

$$f(O_u, O_v) = \sum_{u=1}^{\kappa} \sum_{v=1}^{\lambda} \left[\overrightarrow{N_{VHP}}(u, v) \bullet \overrightarrow{N_{MRI}}(c_u \cdot u + O_u, c_v \cdot v + O_v) \right]^2$$

with c_u and c_v as known scaling factors.

The solution can be solved again through either Brute force or the iterative Newton's method for optimization. The iterative scheme is extended to 2 dimensions, so that the original Newton's method sequence:

$$x_{n+1} = x_n - \frac{\left[\frac{df(x_n)}{dx_n} \right]}{\left[\frac{d^2 f(x_n)}{x_n^2} \right]}$$

becomes, by replacing the derivative with the gradient $\nabla f(\mathbf{x})$:

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma \cdot [H(f)(\mathbf{x}_n)]^{-1} \cdot \nabla f(\mathbf{x}_n)$$

with small step size $\gamma > 0$. The $H(f)$ is the Hessian matrix of f. Given the function $f(x1, x2)$, the Hessian[5] matrix is then:

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \cdot \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \cdot \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}$$

Once a solution is found, the spline representation is not needed further. For each of the two splines (VHP and MRI) I construct a two-manifold mesh of quads by calculating the position of the vertices through the B-spline equations on the known control points. The number of vertices can be

[5]The inverse of the Hessian is a somewhat expensive operation, which is usually solved approximately with the help of the conjugate gradient method. In other versions of the Newton method, an approximation for the Hessian matrix is used instead.

arbitrarily selected, depending on the requirements for speed and accuracy in the step of the algorithm that follows, but must be the same for the two objects.

7.4 Spring-mass-damper system

The purpose of the previous step was to find the corresponding vertices between the two surfaces, so that one surface can be morphed into the other. What is still required is a mechanism to map the contents of the object described by one surface (the atlas VHP) into the other (MRI), so that a 1-1 correspondence between VHP and MRI voxels is achieved. This calls for an elastic transformation similar to the one defined in section 6.3.5, where the background theory of the elastic unit and network were also presented. This section builds upon that and extends the network to 3 dimensions. Apart from this detail the implementation and objectives are identical.

7.4.1 Construction of 3D Spring-mass-damper network

In the 2D version of the problem, the Spring-Mass-Damper (SMD) block (chapter 6.3.5 and 6.3.6) is connected to 8 other SMD blocks, in the 8 directions on the same plane around the central block. In 3D, the connections are increased to $3 * 8 = 24$. The mechanism of constructing the network is identical:

1. Quantize the space into cubic units. A resolution of $256 \times 256 \times 256$ is sufficient. This units represent the SMD blocks.

2. For each block, find whether it lies inside or outside the surface by performing a Ray to Polygon intersection. Use a Ray with a starting point at the centroid of the cube, and fixed direction $\overrightarrow{(1,0,0)}$. Intersect with all polygons of the surface mesh. If the sum of intersections is an odd number, then the cube is within the surface, otherwise it is outside.

3. Find interconnections by searching, for each block, the 24 possible neighbours. If a neighbour is found, then an SMD-SMD connection is set, otherwise an SMD-Surface connection is set.

The network is built for one of the two surfaces, depending on which object is considered to register into the other. In this experiment I build it around the VHP surface, which is a known, fixed quantity.
This process is not real-time, as it is only performed once for the VHP atlas, so there are no speed or memory use considerations.

7.4.2 The Spring-mass-damper system in action

Once the network is established, the vertices of one surface are matched with the surface of the other. As a result of this displacement the springs and dampers that connect a large number of vertices with SMD units are stretched, or squeezed. That introduces forces into the network of interconnected SMD units, which are converted into acceleration, velocity and finally displacement. The system is given a time step and enough iteration cycles to maximize the entropy within, and reach a state of equilibrium. The only difference between this operation and the operation described in chapter 6 is that the units within the network described in this chapter can move in 3 dimensions rather than in 2.
As expected, with the increase in network complexity comes a very significant increase in computational time and need for resources. The operation can take several minutes to complete, depending on time step and resolution of network.

7.4.3 Mapping of UV coordinates

The progression from the sparse 3D SMD network to a 1-1 voxel matching requires an extra step in which the intermediate texel values are computed. Section 6.3.9 describes a bilinear mechanism that uses the UV coordinate values at the 4 nodes on the surface, and calculates the in-between values

through a textured quad, rendered using OpenGL.

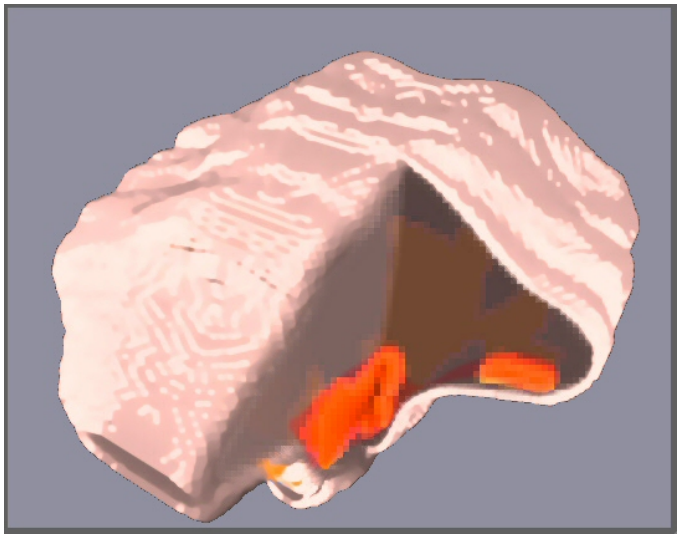


Figure 7.5: Cross-section of the VHP atlas in voxel-space. The extrinsic muscle voxels are colour-coded with red.

The 3D case is slightly more complicated: rather than 4 nodes, there are 8 nodes forming a prism rather than a quad. A mechanism is needed to find the texel value at any point within the prism, assuming that UVW (3D texture) coordinates are assigned for all 8 nodes. This calls for a tri-linear interpolation of texture coordinates. In this case, the solution can be in either software or hardware. The software solution would be slow, so I opted for the hardware solution. I made use of 3D textures and programmable pixel shaders[6]. Figure 7.5 is a rendering of the cross-section of the 3D texture inside the surrounding mesh. The Genioglossus and Styloglossus muscles can be clearly seen.

As a result of using the dedicated hardware path, the operation of rendering

[6]A programmable pixel shader allows for very fast execution of arbitrary commands that operate on pixel level, for rendering primitives. Before the advent of modern GPUs and programmable pixel shaders (circa 2002), the operation described in this section would be impossible in hardware. Pixel shaders have been used again in this work, in section 4.7.1.

the UV coordinate values in-between the SMD nodes is not time-consuming enough to be considered as a bottleneck. The serious bottleneck of the procedure remains the progression of the SMD network to the state of rest, which involves tens of iterations.

7.4.4 Result images

I used the algorithm described in this chapter to register the VHP atlas on the MRI scan of Lucy. I then visualized the results with my own volume rendering engine that uses OpenGL. I used this 3D visualization scheme to show how the structures have registered in 3D. A set of 2D images, like the ones shown in the result section of the previous chapter, would not provide the necessary visual depth cues to appreciate the registration operation as a whole.

Figure 7.6 shows two different views of the VHP atlas mesh as it was estimated through the 3D active contour. While it is not a perfect match for the original data, it is a very close estimation. The internal colour-coded structure represents the sum of voxels identified as extrinsic muscles. Figure 7.7 shows the different views of the tongue derived from the Lucy MRI scan after the registration. The extrinsic muscles have changed shape as they follow the change in overall shape of the tongue. This image is analogous to the image in figure 6.16, where I show the results rendered in 2D and overlayed over the original MRI image.
For this test I started with a 3D active contour that is based on 6144 (32 * 6) quads. On my test machine, the 3D active contour step required 60-80 iterations and around 40 seconds to converge. The SMD network was built on 30000 SMD units interconnected in a 24- neighbour configuration in 3D. The operation of the network required 100 iterations before settling and each operation cost around 5.5 seconds, for a total of around 550 seconds, or 9.1 minutes of calculations.

Comparison test to model generated using the 2D registration scheme

The 3D registration scheme was tested with the model of Lucy and gave similar results in terms of position and distribution of extrinsic muscles within the tongue to the 2D scheme tested in chapter 6 using the same model. This was evaluated by comparing the images generated per slice and taking their saturated absolute difference. A binary volumetric map was then created (relative error map), in which voxel value of "1" indicated a relative anomaly. The comparison of the two estimations produced a map with a very low presence of relative error voxels - less than 2 %. A similar method is also used and illustrated in chapter 8.

7.4.5 Discussion and conclusion

The 3D registration scheme presented in this chapter is essentially an extension of chapter 6. The work-flow, which is presented in figure 7.8, is conceptually identical:

- The geometrical models of the VHP and MRI derived exterior of the tongue are first constructed and represented as two independent sets of isospeed closed cubic splines in 2D or as two independent closed bi-cubic surfaces.
- The models are scaled so that their bounding boxes are identical.
- The correspondence of spline control points between the VHP and MRI models is found by minimizing the overall sum of residuals of their dot products.
- An iterative physical based elastic registration is then used in which the image is treated as an elastic material that deforms to match the constraints set by the models.
- Visualization of results uses overlays or volume rendering.

The benefit of using the 3D registration scheme is not apparent in the test case I present. This is because none of the volunteers or clinical subjects[7] presented a concavity in the sagittal plane (between the Genioglossus and Styloglossus muscles) that would require a fully 3D registration scheme as opposed to the 2D scheme presented in chapter 6. While this was the reason for exploring the fully 3D solution in the first place, as I described in the introduction of this chapter, I did not find the suitable data to prove my case. However, it is clear from the design of the algorithm that it does not depend on special restrictions, as the 2D scheme does, and it should function as long as the basic rules are followed: that the topology of the closed surface is fixed to genus - 0 and that the source and target objects (VHP and MRI derived exterior of the tongue) carry a geometrical resemblance.

The main drawback of the 3D registration scheme is that it is very slow, as it is based on a complex operation that requires a large number of calculations. This is not the case with the 2D scheme which can be updated almost in real-time. The ability to register in real-time is desirable (but not crucial) in this application in order to allow the clinician to fine-tune the boundaries of the tongue on the MRI images and update the model on the fly in cases where the registered model has obvious errors. The real-time registration scheme is also necessary for future applications of the system that involve 4D data and virtual reality, as will be discussed in chapter 9. Consequently, it is preferable to use the original 2D registration scheme at this stage, which is magnitudes faster than the 3D version. All the result data and images presented in the next chapter are therefore based on the system that was built upon the algorithms described in chapter 6.

[7] The experimental procedures of the scans are described in chapters 3 and 8 accordingly.

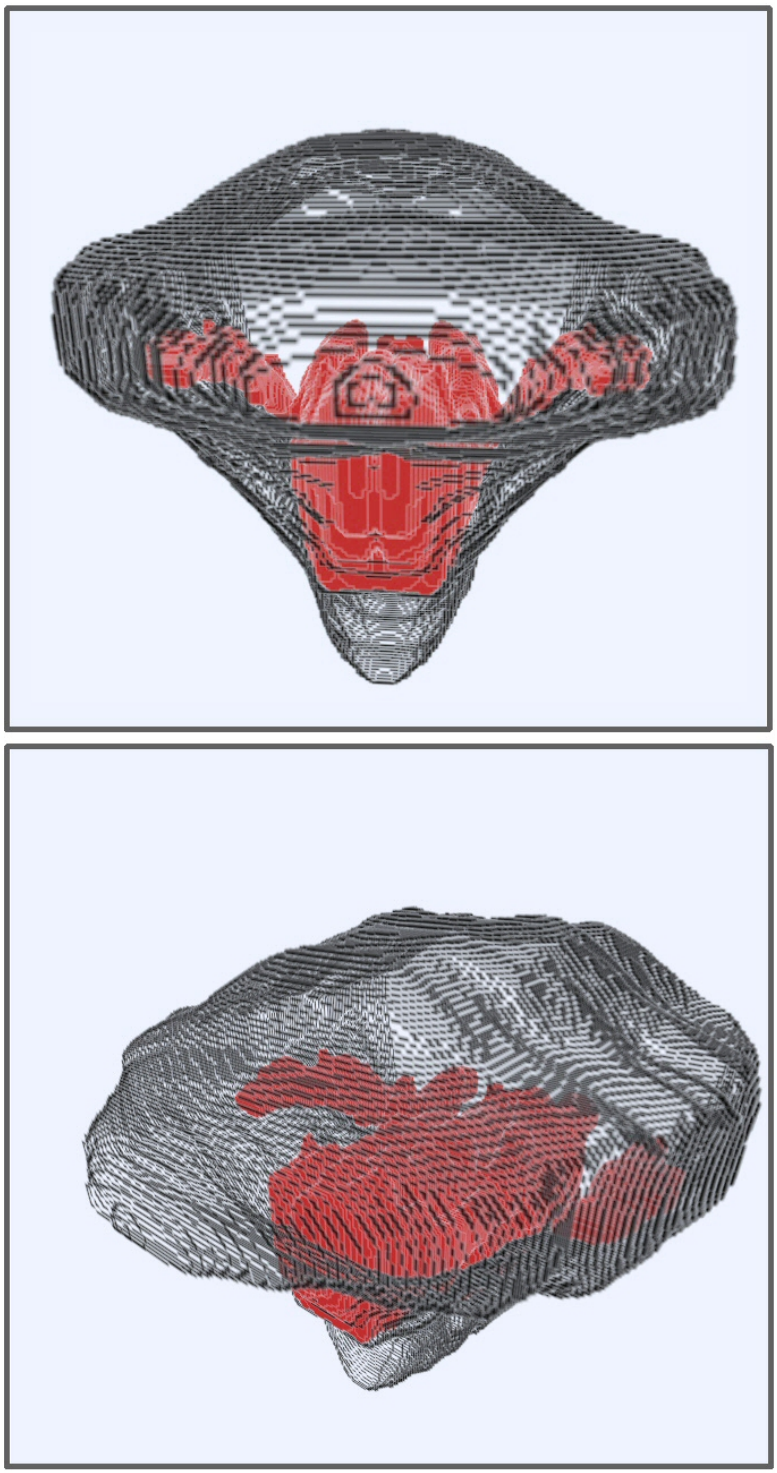


Figure 7.6: Two different views of the VHP atlas rendered in voxel-space.

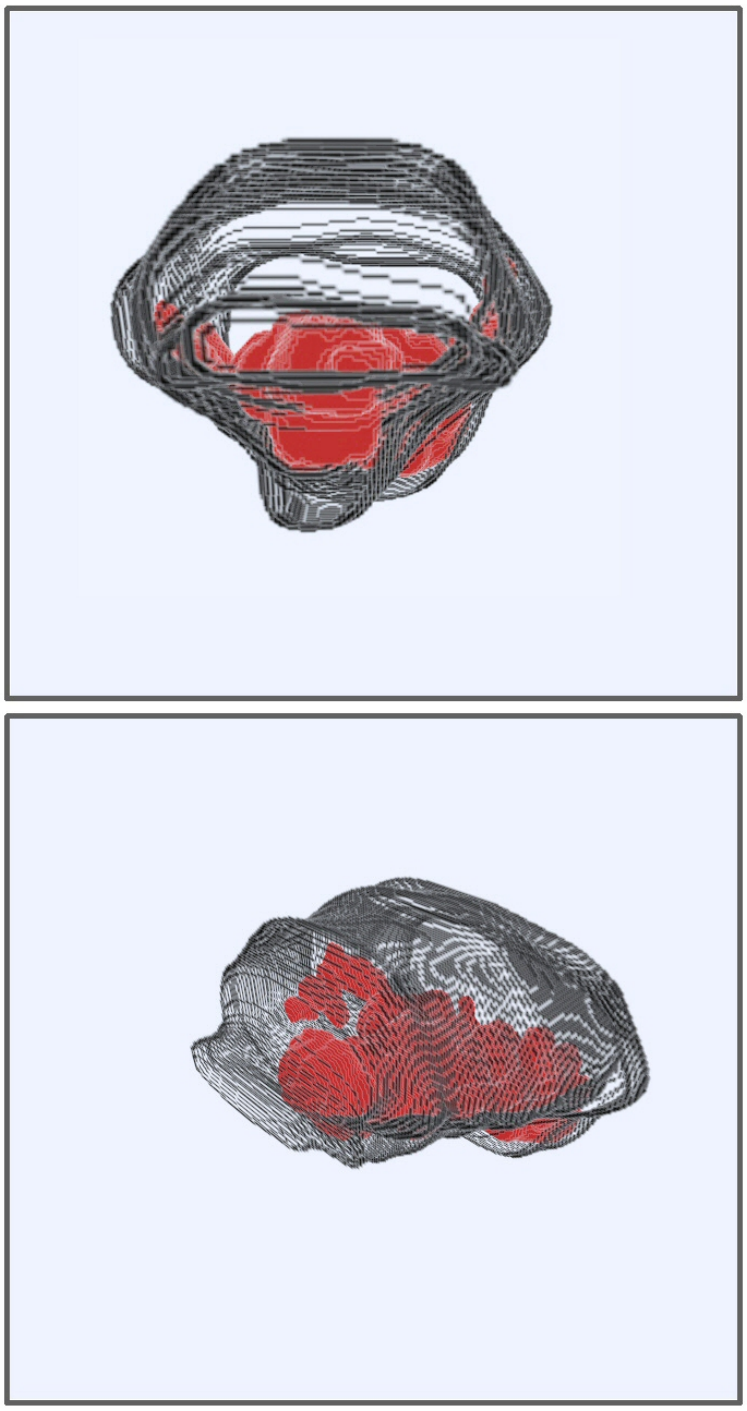


Figure 7.7: Two different views after registration of the VHP atlas into the MRI scan of Lucy, rendered in voxel-space.

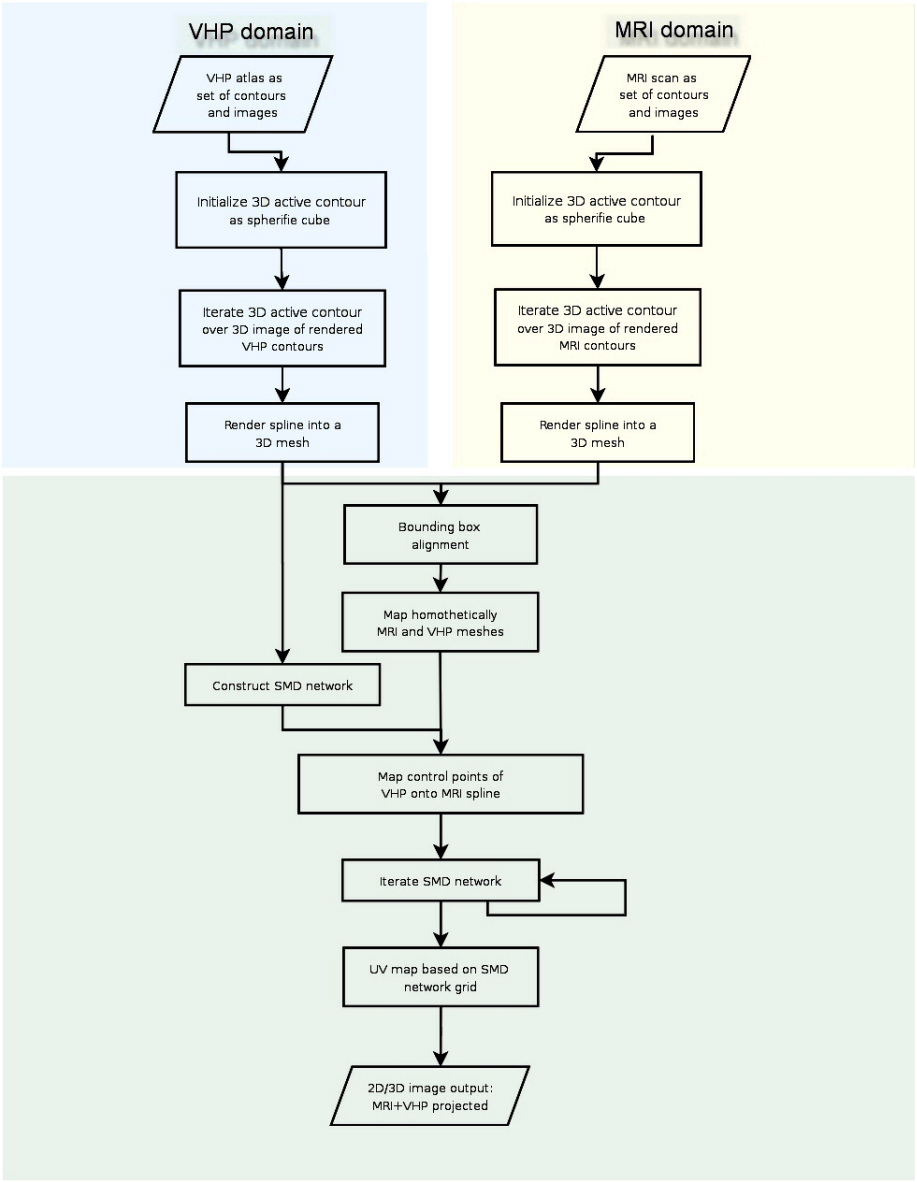


Figure 7.8: Overview of the work-flow of chapter 7.

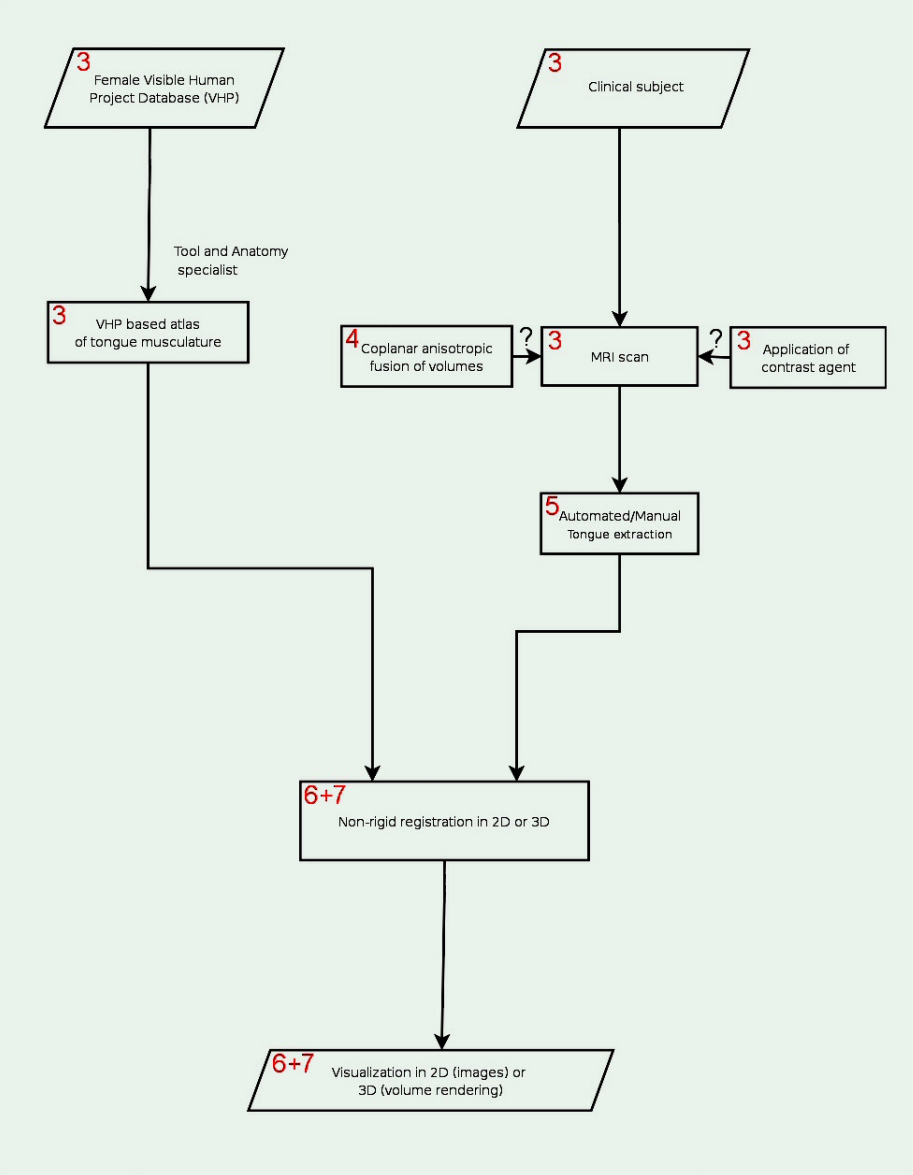


Figure 7.9: The work-flow of the system described in this thesis. The red numbers indicate the number of the relevant chapter. Question-marks denote non-essential units.

Chapter 8

Validation and Results

8.1 Introduction

The previous chapters have presented the intrinsic details of the conception and operation of my system. In order to describe the system in a tangible manner, I have organized it into separate chapters that, while conceptually connected to each other, describe each phase independently. The results presented in every chapter were part of the process to validate the software and concepts developed in that particular chapter and not the system as a whole. The source material consisted also of scans of volunteers that did not have tongue cancer or other tongue abnormalities, as I attempted to prove first that the concepts work on normal subjects and then proceed to test them on more complex clinical cases.

This chapter addresses these issues by presenting results obtained after using my system in a clinical environment. There are two experiments in this chapter:

- A validation experiment using a phantom (ox tongue), in which I test the efficiency of the integrated system, the main steps of which are atlas generation, image acquisition, contour extraction and, most importantly, non-rigid registration.

- A validation experiment using clinical subjects with tongue cancer.

8.2 Validation using a phantom

This section describes a test I performed to validate the registration method I used in this thesis. The test was carried out using an MRI phantom. A phantom is usually an anthropomorphic article that is imaged in order to test the performance of an imaging system [Hor06]. In general, one of the reasons for using a phantom in place of an in-vivo human part is that the latter is not easy to acquire. The experiment I describe in this section used an ox tongue that was dissected, painted and scanned. It was obviously impossible to carry out the same experiment on a living human subject as it was not practicable to acquire a human tongue ex-vivo.

8.2.1 Objective

My registration algorithm, as presented in this text, maps the reference atlas data into the measured data by aligning the contours of the two structures and then texture mapping the space they inscribe by using an Spring-Mass-Damper model that mimics particle behaviour under elastic deformation. This is an abstraction from reality; there is no perfect solution, as indicated by the fact that there are potentially infinite solutions for a problem like this where there are not enough known quantities required to reach a unique solution. Therefore, the registration scheme is an approximation.
The model used has been separated into stand-alone units that are already described and tested in chapters 5,6,7. The experiment I describe in this section attempts to bring all these units together and evaluate if the approximation achieved is sufficient for the scope of this project.

8.2.2 Materials

The materials used in this experiment were a fresh, unsalted ox tongue. There was no special preparation for the tongue and it was not treated in contrast enhancement solution[1].
For the photography and scanning I used a Canon S50 5 MPixels digital camera and a 1.5T G.E. scanner in clinical use.

8.2.3 Experimental procedure

The experiment emulates the atlas/MRI registration process in a controllable fashion. The procedure I followed is:

1. I dissected the ox tongue coronally into 2 cm wide slices. I selected the two slices with the largest cross-section.

2. I carefully marked the large Genioglossus muscle and Styloglossus on each slice using a black marker pen. I tried to follow the anatomy of the ox tongue by consulting all the relevant anatomy information for humans, as I could not find more information on ox tongue anatomy[2].

3. The two slices were then placed horizontally and photographed from a distance of 40 cm using a digital camera. The photography technique (distance, exposure time, colour balance etc.) was identical for each slice. The acquired images represent the atlas images.

4. The two slices were then placed horizontally and submitted to MRI. Table 8.1 lists the scanning sequence used examination. One practical

[1]There was no need to. The cavitatory application of contrast agent that was described in chapter 3 was needed in order to assist the segmentation of the tongue in the oral cavity. In this case, the ox tongue is easily segmented by simple image processing (thresholding) as it is surrounded by air.

[2]The fact that the atlas 'model' of the ox tongue might be wrong is irrelevant in this context, as this is a comparative experiment. If the model was a set of random lines, the comparison would still be valid, as it is with respect to the same model.

difficulty with the scan was that the signal from the thin slices was not sufficient to trigger the scanner. The solution to this was to place two bags holding 1 litre of normal saline beside each of the tongue slices. This allowed satisfactory signal detection and the scan commenced as normal.

5. The next step was to apply an elastic deformation to the slices: I moved (translation) the slices and changed their overall shape by manually applying compressive (pushing) forces, tensile (pulling) forces, shear and bending. The objects stayed in their new position and shape.

6. I photographed and scanned the slices again using exactly the same photography and scanning techniques. The combination of the two gives the correct musculature map of each slice.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_1 FSE FAT SAT ET: 0 TR: 700.0 ms TE: 9.0 ms
Matrix Size	384 × 384
FOV	26.0 × 20.0 cm
Slice thickness	2 mm, 0 mm spacing

Table 8.1: Scanning sequence characteristics for experimental procedure in 8.2.3.

The first set of photographs were used as the atlas set: it was a simple matter to extract the outline of the tongues on the photographs. The pixels with very dark colour denoted the presence of an "extrinsic" muscle. That produced two maps (one for each slice) of contours and textural content similar to the VHP atlas map.
The second set of MR images were used as the target MRI scans. I then used

my registration algorithm to register the atlas map into this scan, twice for the two different slices and produce a registration image. This was compared to the second digital photograph with the naturally deformed images of the two slices. The comparison method that I used was a displacement map which was produced through the optical flow of the two images[3]. My hypothesis was that the comparison should yield very similar results (minimal overall displacement), meaning that the synthetic deformation follows the naturally occur VHP and MRI contours, the synthetic as the procedural attempt to assess and approximate this difference.

8.2.4 Results

Figure 8.1 shows the two photos, before and after the apparent deformation (all figures can be found at the end of this chapter). The next two pages show the combined results (MRI, photography and processing) acquired for both slices. In figures 8.2 and 8.3 the top images show the atlas produced by manipulating the first photograph, superimposed on the first MRI image. The extrinsic muscles are colour coded cyan. The middle two images show the result of the registration (cyan) and natural deformation (red).The last image shows the difference between the two, visualized as the pixels with non-zero displacement vector. It can be seen that there is a minimal error which is of the order of 1.5-2 mm at worst, and that occurs for the Styloglossus muscles that lie on the side lobes and hence are deformed comparatively[4] more than the Genioglossus which is in the middle.

8.2.5 Conclusion

The purpose of this experiment was to test the elastic registration beyond the conventional methods that I used in chapter 6. The experiment essentially simulated the process of atlas registration on MRI images of the tongue,

[3]The same concept is presented in chapter 4 and in validation methods of chapter 6.

[4]Mainly due to coronal rotation which affects the extremities of the circular object more than the center.

as was described in previous chapters. The MR sequences are virtually the same, the atlas generation is analogous to the Visible Human Project based one described in section 3, and the registration scheme is identical. The important addition, however, was that the true position of the structures tagged as 'extrinsic muscles' was also known, and hence a comparison could be made between the placement and shape of the structures according to reality and estimation. This is clearly not possible with a clinical MR scan of in-vivo subjects, where the true musculature cannot be acquired by imaging. The next section investigates the use of my VHP atlas registration scheme on clinical MRI scans, where, as mentioned, the validation of the results is possible. A metric of accuracy or inaccuracy therefore does not exist. Instead, I base my validation on comments made by a clinician, as indicated below.

8.3 Validation using clinical data

This section presents the results acquired by using the system developed in this thesis on three clinical cases of tongue cancer. The cases were selected from the MRI database of the John Radcliffe hospital in Oxford as representative of malignancy in early to advanced stage. The format I use to present the experiment is first to provide the scanning sequence in a table, followed by a short clinical evaluation of the stage of the disease. I then present an annotated MRI image selected from the set as characteristic of the state of the disease and a set of volume renderings of the results.
The outlining of the tongue was done using the semi-manual method (outlining and snake "snapping" of contour) described in chapter 5. The cancer area was visually identified from the original images with the help of a clinician who identified it on each of the original images.

8.3.1 Case 1

This 47 year old man has an early stage of tongue cancer (T1).

The tongue outline is relatively easy to trace on the images. The carcinoma is also very well delineated and does not distort the tongue outline. It can be seen on figure 8.4 that the carcinoma is small and may affect the Styloglossus muscle. Figure 8.5 is a 3D volume rendering of the tongue outline with the extrinsic muscles colour-coded as red and the carcinoma as light blue. It can be seen that the carcinoma is not extensive and does not affect the large Genioglossus muscle.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_2 FSE FAT SAT ET: 12 TR: 5200.0 ms TE: 73.3 ms
Matrix Size	512 × 512
FOV	20.0 × 20.0 cm
Slice thickness	5 mm, 1 mm spacing

Table 8.2: Scanning sequence characteristics for case number 1.

8.3.2 Case 2

This is a case of a 52 year old with extensive tongue carcinoma, involving the base of the tongue.

The carcinoma has distorted the outline of the right side of the tongue, which is apparent in figure 8.7. Figure 8.6 shows that the carcinoma (blue) has invaded the right side of Genioglossus and it fully surrounds the left Hyoglossus and Styloglossus muscles.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_2 FSE FAT SAT ET: 12 TR: 6100.0 ms TE: 69.2 ms
Matrix Size	512 × 512
FOV	24.0 × 24.0 cm
Slice thickness	4 mm, 1 mm spacing

Table 8.3: Scanning sequence characteristics for case number 2.

8.3.3 Case 3

This is a case of a 36 year old man with carcinoma of the tongue, involving the floor of the mouth, relating to the Mylohyoid and Hyoglossus.

Because of the extensive size of the carcinoma that distorts the tongue and the large slice thickness, the overall tongue shape is not easy to determine. As a result, the approximation is not satisfactory and the registration model fails in a few slices. This can be seen in figure 8.8 where the Styloglossus is fractured in places. Regardless of that the overall shape of the extrinsic muscles and their spatial relation to the carcinoma is retained. It can be seen in figure 8.9 that the cancer has spread onto the Hyoglossus and Styloglossus muscles, but not so much onto the Genioglossus.

Scanner	1.5T GE Twinspeed super conductive
Weighting	T_2 FSE FAT SAT ET: 12 TR: 46400.0 ms TE: 73.3 ms
Matrix Size	512 × 512
FOV	20.0 × 20.0 cm
Slice thickness	6 mm, 1 mm spacing

Table 8.4: Scanning sequence characteristics for case number 3.

8.4 Significance of findings

In this chapter I presented the models constructed by using my system on images of clinical subjects. From visualizing the models in 3D it was then possible to gain insight into the the spatial distribution and relation of the extrinsic muscles of the tongue and the carcinoma in order to assist surgical staging.
The findings agree with the original evaluation of the clinician:

- The visualization of case 1 indicated an early stage tongue cancer in which the tumour partially surrounded the Styloglossus muscle. The clinician agreed that this could be the case but it that would be not possible to prove on the original, unprocessed MR images.

- Case 2 indicated a T4. The clinician suggested that the tumour had definitely affected the extrinsic muscles and that the spatial correlation of muscles and tumour, as presented in the rendered images, is anatomically correct.

- Case 3 shows also an extensive carcinoma. My system illustrated that the tumour had invaded Hyoglossus and Styloglossus but left Genioglossus relatively unaffected. The clinician indicated that this was very probable because of the elongated shape and lateral position of the tumour.

While there is currently no other method to evaluate the end result (other than refer to an evaluation made by the clinician, based on the original MR images), it is possible to analyze the quality of data and methods used in the sequence from raw data (MRI) to the 3D model.
The two primary sources of data are the VHP constructed model and the set of contours based on the MRI images. With the VHP model and registration method fixed and already validated, any gross deficiencies in the end 3D model could be related directly to the accuracy of the contours which are derived from the MR images.

The 3 sets of MR images have two important shortcomings: the tongue outline is not adequately delineated and therefore the segmentation of the tongue itself is challenging (as it was observed in case 3). This can be improved by employing the cavitatory contrast enhancement with Gadolinium contrast agent on T_2 or T_1 weighted images (where the effect is more prominent0, as described in chapter 3. The other problem with the images is that the slice thickness of the clinical sequences available is large (up to 6 mm with 1 mm gap) compared to the slice thickness of the scans of the volunteers (Lucy and Matthew both had 2 mm sections and produced a much smoother 3D model), which affects the accuracy of the end 3D model in the third (coronal) direction. The solution is to change the clinical sequence characteristics to minimize the slice thickness which, as a side effect, will increase scan time and decrease the overall SNR.

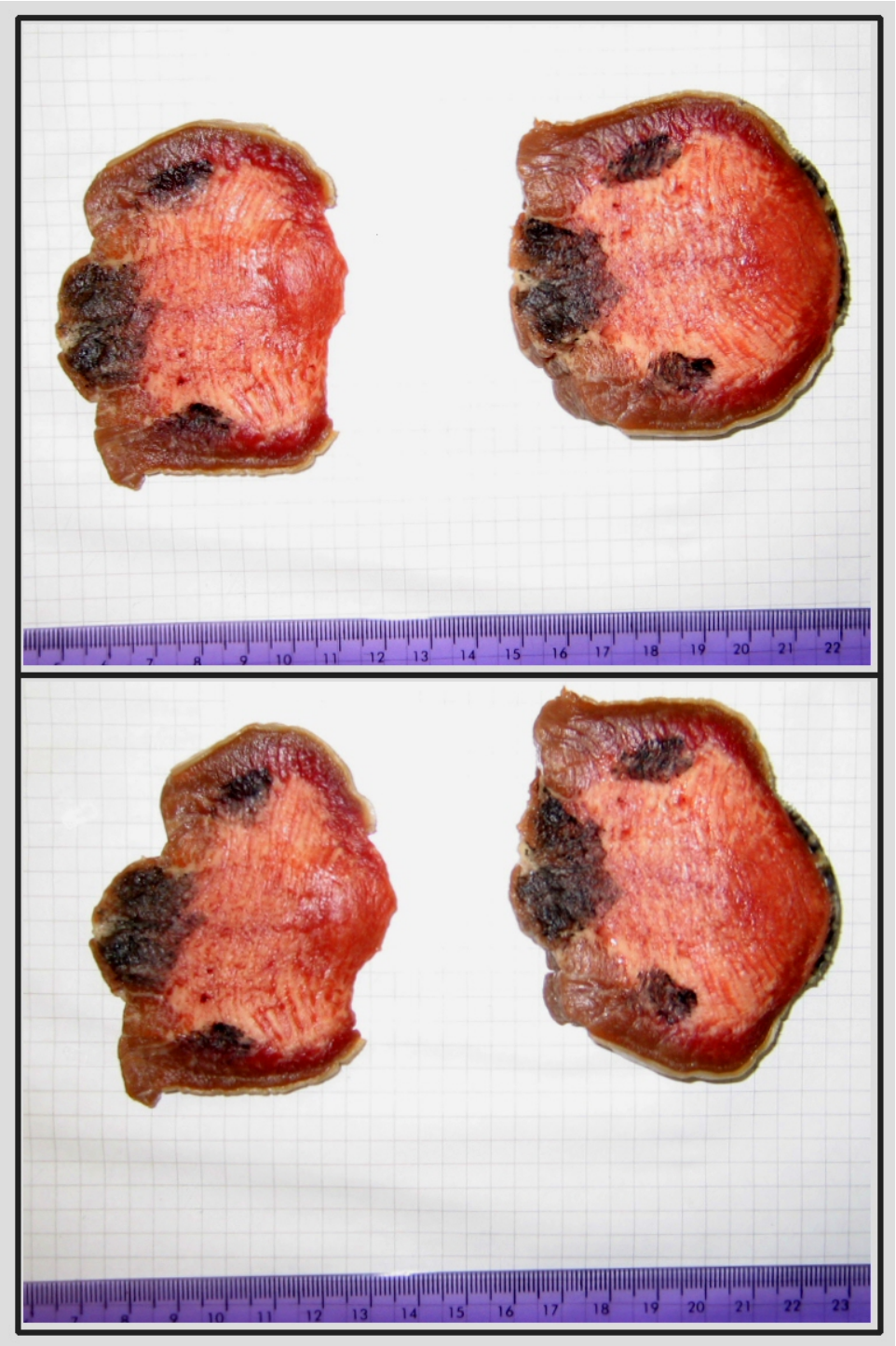


Figure 8.1: Photographs of the ox tongue slices before and after deformation.

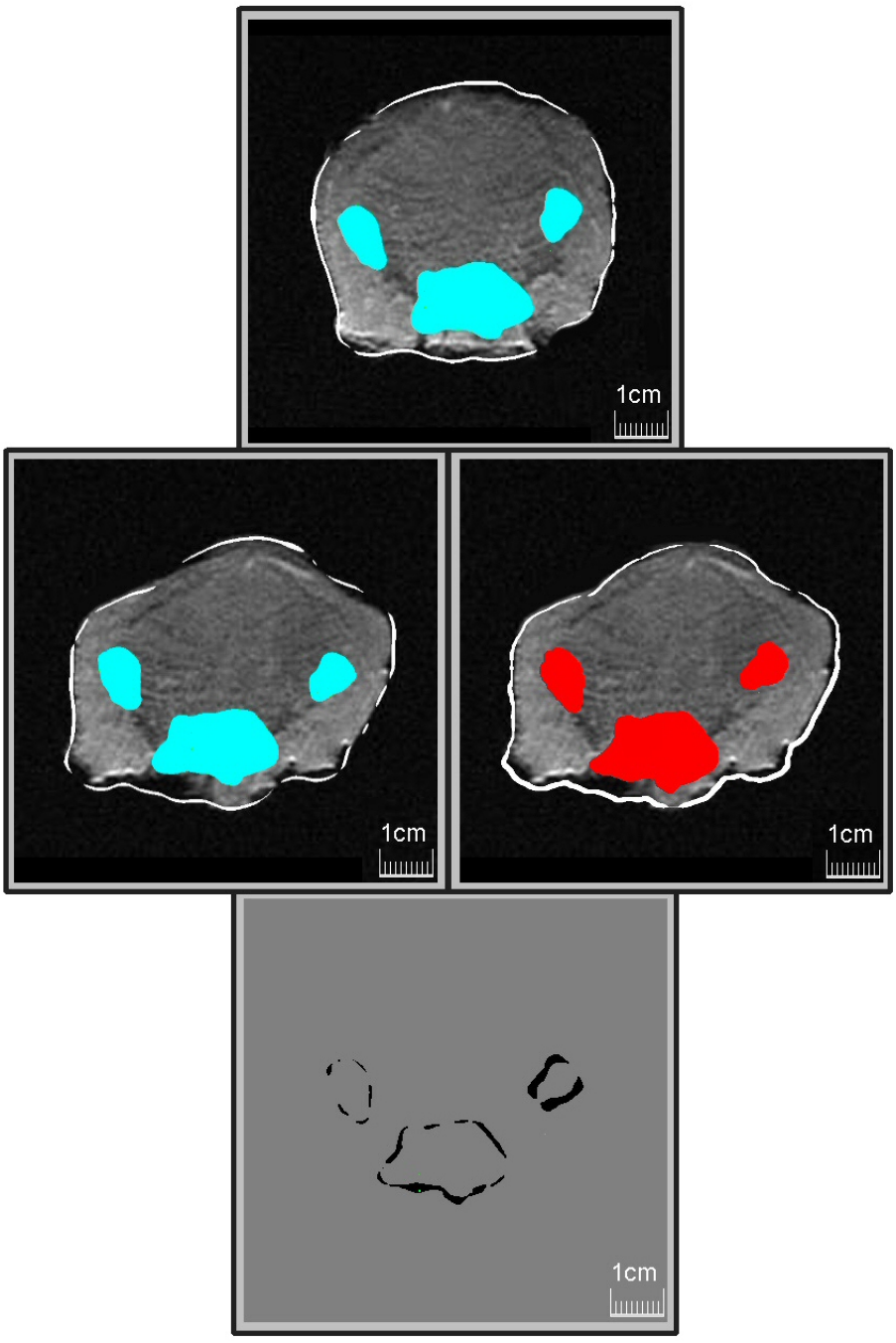


Figure 8.2: Results for case 1: image before and after deformation - middle left shows estimated deformation. Bottom image shows error map between estimation and reality.

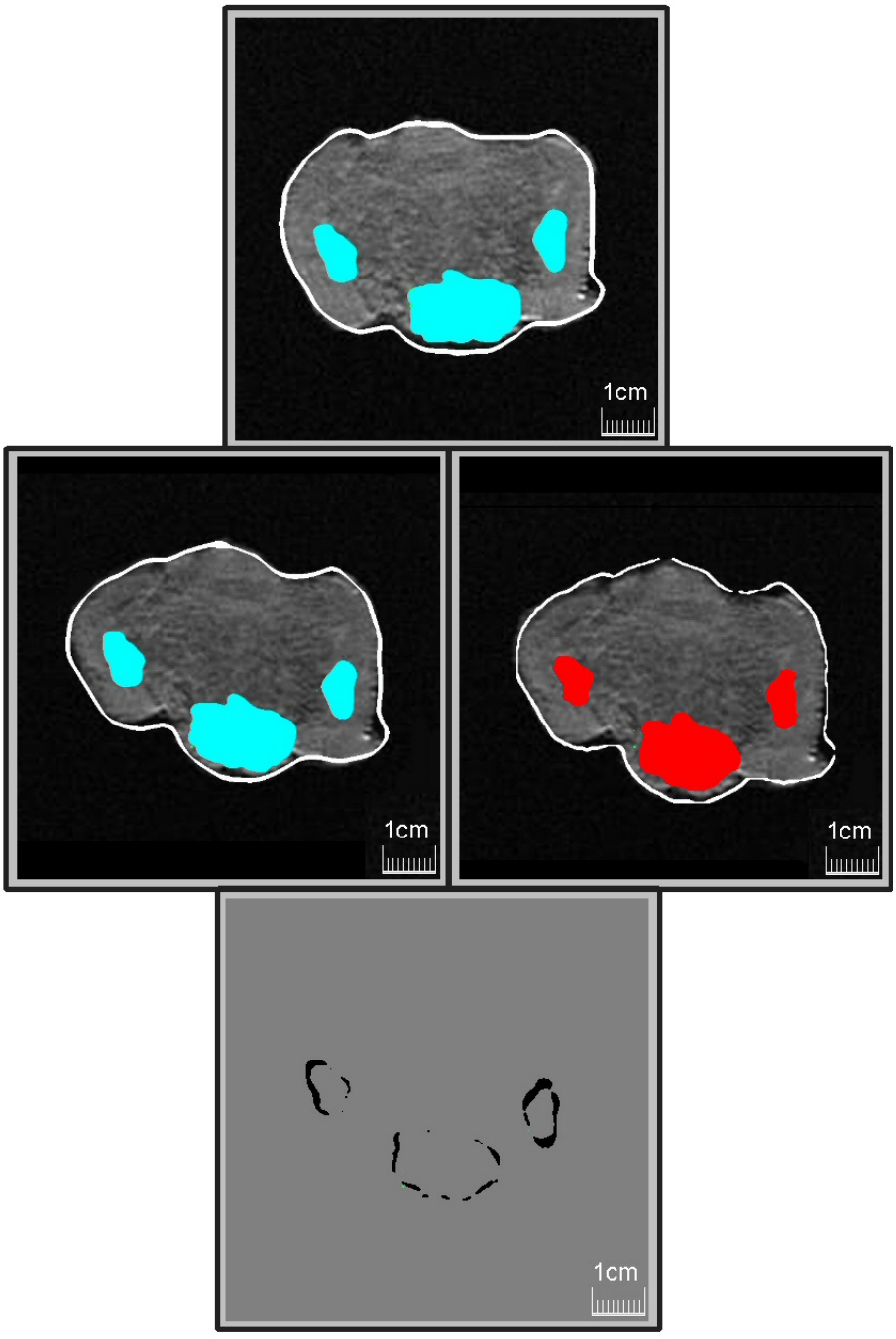


Figure 8.3: Results for case 2: image before and after deformation - middle left shows estimated deformation. Bottom image shows error map between estimation and reality.

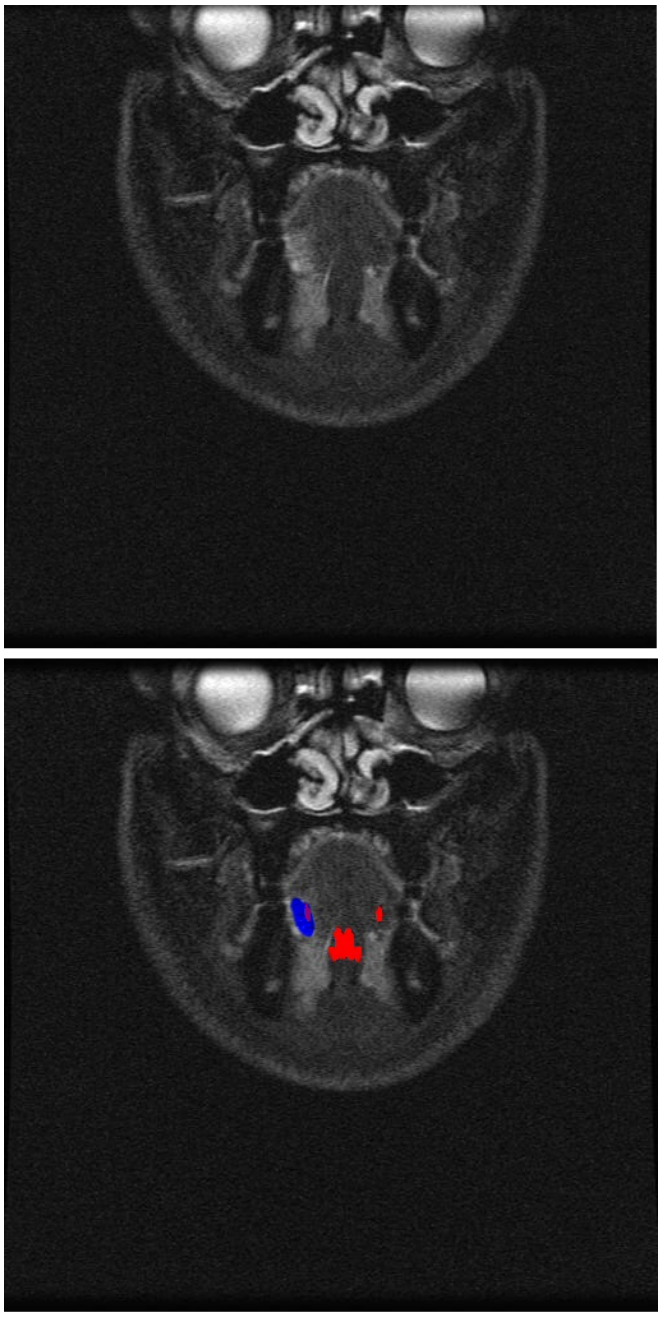


Figure 8.4: Case 1: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom.

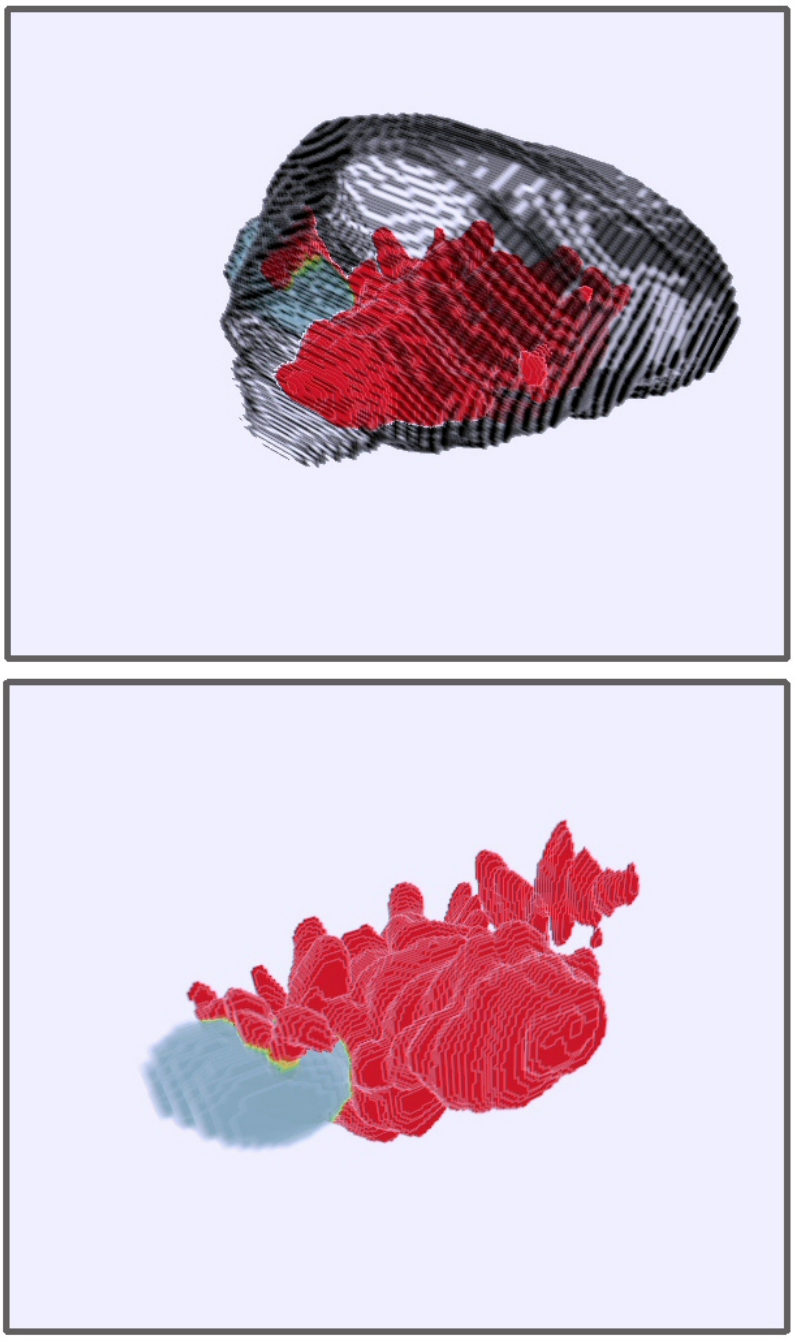


Figure 8.5: Case 1: volume rendering of cancer (blue) and extrinsic muscles (red).

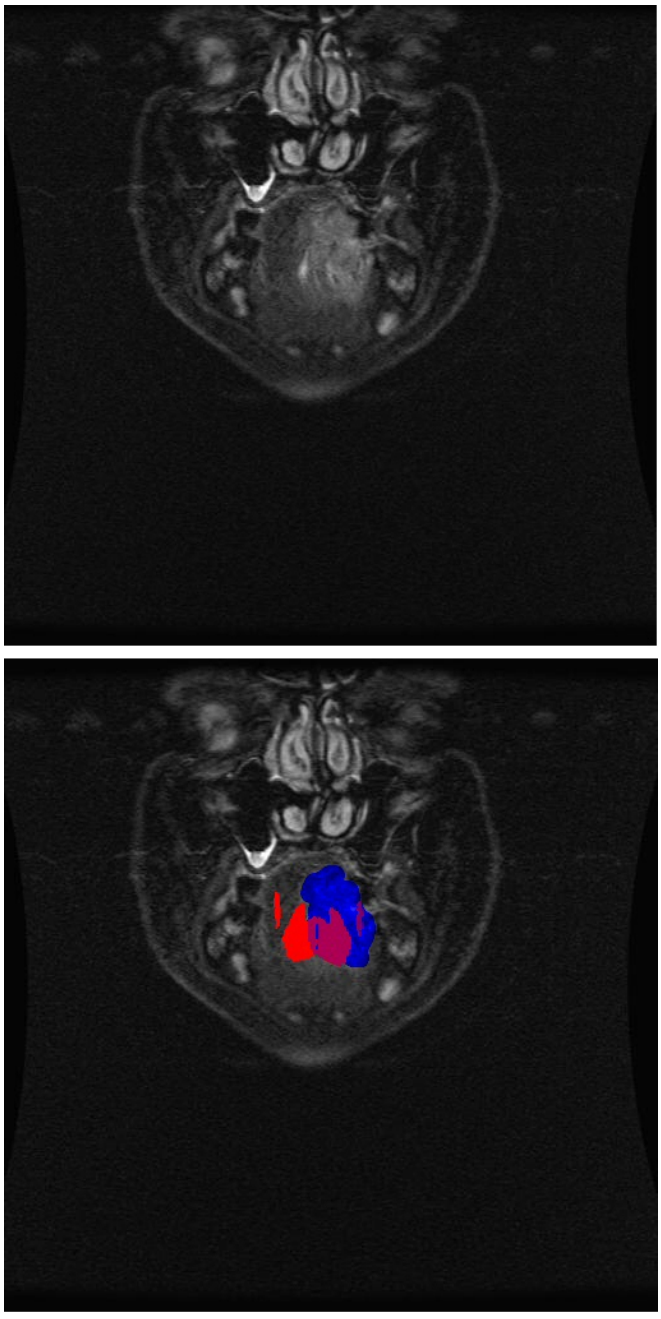


Figure 8.6: Case 2: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom.

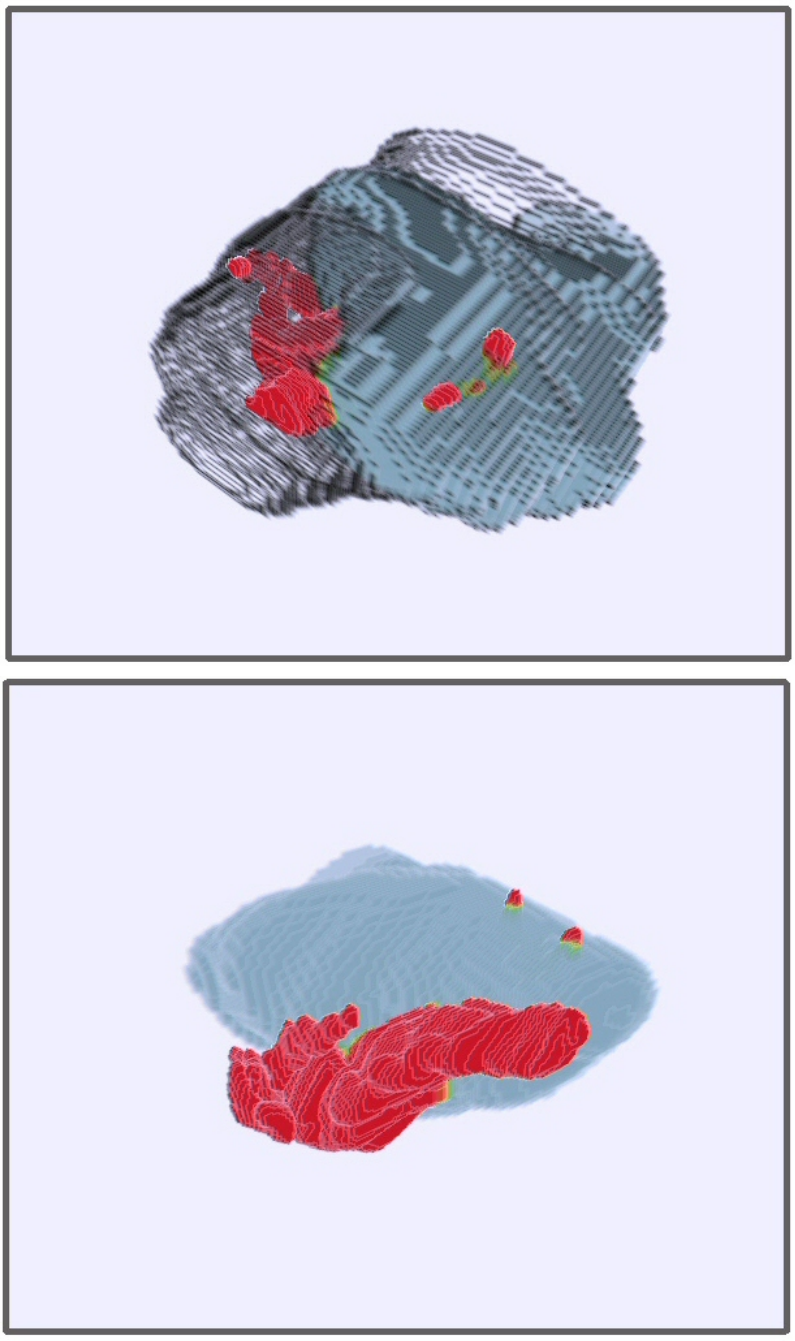


Figure 8.7: Case 2: volume rendering of cancer (blue) and extrinsic muscles (red).

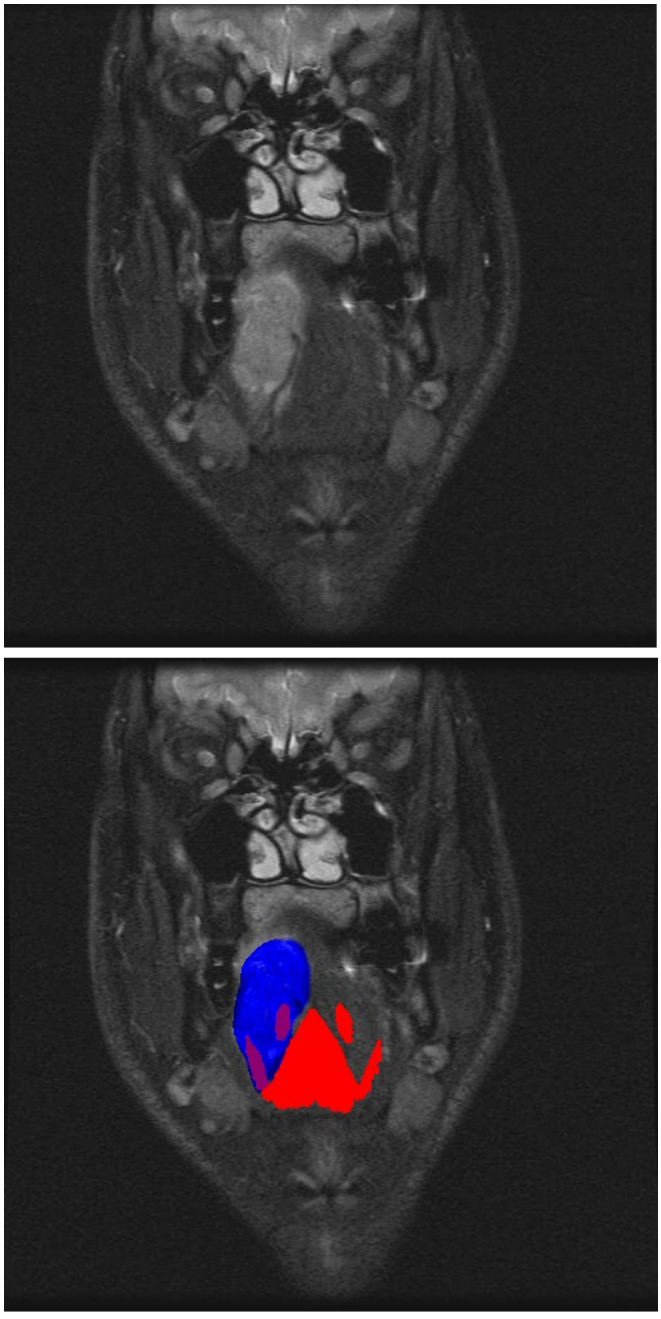


Figure 8.8: Case 3: original MRI (top) and MRI with superimposed cancer area (blue) and extrinsic muscles (red) in bottom.

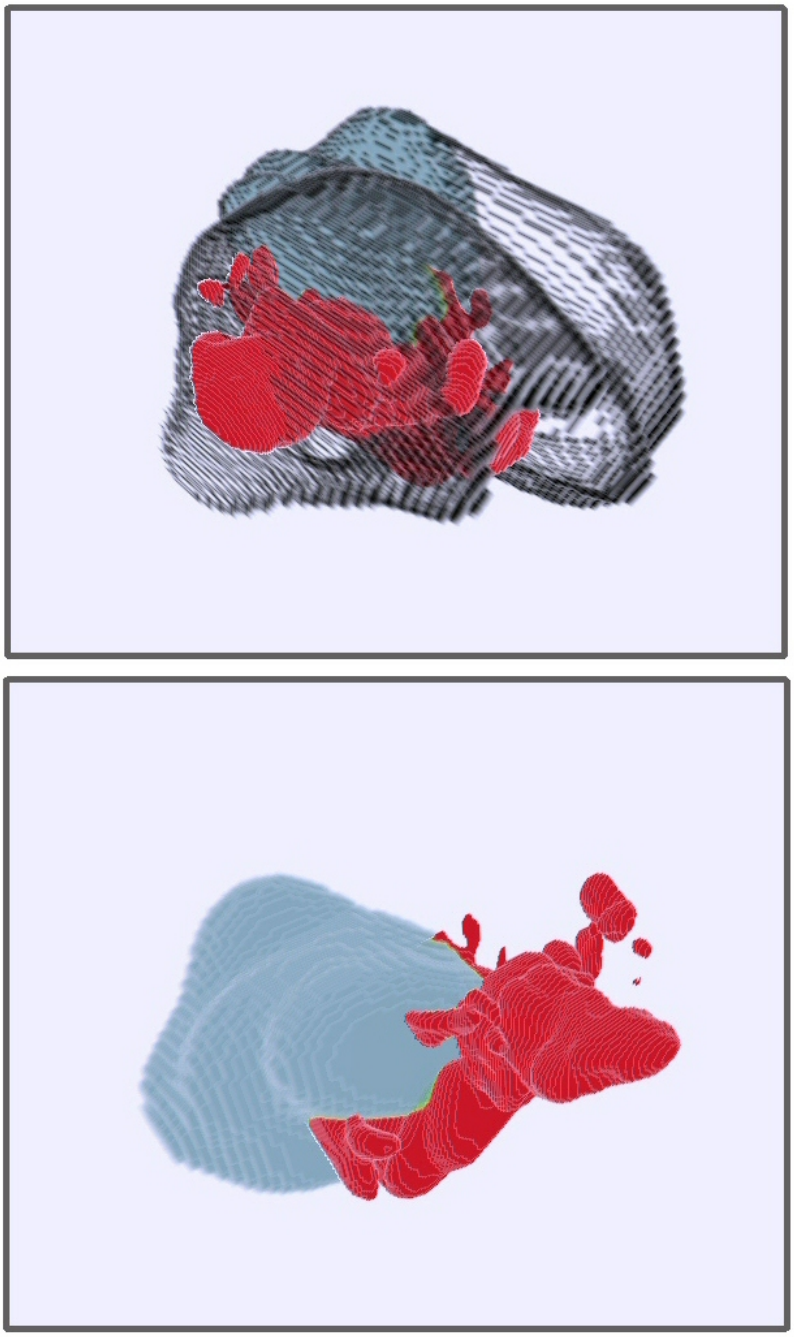


Figure 8.9: Case 3: volume rendering of cancer (blue) and extrinsic muscles (red).

Chapter 9

General Discussion and Conclusion

This thesis presents the intrinsic details of the system that I developed for modelling the tongue and extrinsic muscles of the tongue on MR images in order to assist the clinician in tongue cancer staging. The methodology that I followed was to split the system into discrete units which were described, tested and discussed in the relevant chapters. I then tested the operation of the integrated system with volunteer and clinical medical images and proved that my system works. As such the work itself contributes directly to the not yet fully explored field of tongue imaging, and indirectly to the field of staging of tongue cancer. There is a distinction that must be made between the two, which is addressed in the next section.

9.1 Contributions of this thesis

The contributions of this thesis to clinical medicine are best described by a list of the objectives met in the course of this work, as below.

9.1.1 Construction of a 3D atlas of tongue musculature

A methodology was defined and used in order to construct a 3D model of the human tongue and extrinsic musculature based on the Visible Human Project and manual interaction. The building of the model relied on a software system that I wrote. Using the same system it is possible to model *any* anatomical feature of the human body in a fast and intuitive fashion. Also it is possible to extend the atlas to include more information per atlas voxel, such as, for example, muscle fibre direction.
My system also provided the means to compare the model in real-time with existing models from the background literature and hence validate the construction process. With some extra work it is, therefore, possible to extend my system to work on a universal database of Visible Human Project based "anatomical models" which research facilities could have access to and update.

9.1.2 Cavitatory enhancement with Gadolinium contrast agent in solution

Part of the experimental procedure was to use cavitatory enhancement with Gadolinium contrast agent suspended in a solution of pectin. This was a side experiment, albeit important for the work as a whole, and was described in chapter 3. The experiment was successfully tested on two different subjects in clinical conditions under T_1 and T_2 weighted imaging and provided a simple and reproducible method to improve the delineation potential between the dorsum and hard palate, which would otherwise be indistinguishable from each other, at least for robot vision. While the use of this technique in clinical images was not an objective, the results are promising for further experimentation and would justify a trial of this technique in clinical practice.

9.1.3 Construction of the 3D model of the tongue based on MRI

Throughout this research, much effort went into constructing the 3D model of the tongue based solely on subject specific MR images. I explored a number of manual and automated methods that were implemented, tested and discussed in chapter 5. The extraction of the model of the tongue from MR images proved a difficult task and my contribution to medical imaging, beyond the proposal of some new algorithms, is my methodology: I approached the task as an engineering problem. As such, I evaluated the methods based not only on their functionality but also on their efficiency and cost of operation in a clinical environment, an important issue. For the problem in hand it was shown that the hybrid manual-Active Contours solution has the best balance of these qualities if applied in a clinical environment that has certain time restrictions and image quality standards.

9.1.4 Fusion of anisotropically sampled MR volumes into a single isotropic

Chapter 4 described a method that I developed and tried in which an isotropic MRI volume of the subject was produced from a non-linear combination of two coplanar anisotropic MRI volumes of the same subject. This experiment, while it did not fully achieve its purposes, suggested a direction for further research in the future. More specifically I believe that this concept can better complement time-based (rather than volume based) studies with only minor changes made in the core algorithm, acting like a non-linear "temporal smooth" filter for high spatial/low temporal resolution studies. In any case, it would be beneficial to test whether the produced volumes (in spatial or temporal domain) have the potential to improve the operation of a robot vision segmentation algorithm. This was not attempted in this work, mainly because I considered that there would be very little return (if any) for the amount of effort spent trying to validate the concept.

9.1.5 Registration of VHP atlas tongue on MRI derived tongue

Another contribution of this thesis was the set of algorithms adapted and implemented for the purpose of registering the VHP atlas on the MRI images of the tongue of subjects. I implemented two different algorithms, one working with 2D contours and another with 3D meshes. My approach combined elements from different branches of computer science, mathematics, engineering and physics such as image processing, computer graphics, vector and function analysis, topology and mechanics. Furthermore, the visualization of results was made possible by using original software that I developed in the course of this work.

The registration algorithms were tested successfully with a number of procedural and clinical tests. By extending the concept beyond the complex structures of the tongue, it should be possible to generalize the methods so that other anatomical organs can be registered with the relevant atlas.

9.1.6 Construction of a system to identify tongue musculature on MR images

By putting all the above concepts together, I constructed a system that can identify the musculature type of the voxels of the tongue on MR images. The clinical use of this system is the primary objective of this thesis: a metric of penetration of cancer into the extrinsic muscles of the tongue can be established and used in quantifying and visualizing the surgical stage of cancer, thus providing the surgeon with more tangible information than the current TNM system can support. The system is able to visualize the relationship between disease and internal structure of the tongue in a tangible manner, with overlays of the colour-coded myoarchitecture atlas registered on the original MR images, so as to assist treatment and operative planning of the malignancy.

The system itself was the ultimate objective of my work, rather than the

series of experiments required to show that the system is or is not a valuable tool in surgical management of cancer. It was decided from the beginning that the latter would require a long run of systematic clinical research into the field that would be beyond the scope of a single research degree. One direction, for example, is that validation of extent of tumours would be possible by comparing the results obtained using my system with histological analysis. This concept was not pursued. Instead, I decided to construct this system and provide all the algorithms and concepts in order to make such undertaking feasible for future research from a third party. I am positive that any future research will only confirm the usefulness and validity of the system described by this thesis.

9.2 Limitations of the system

This section discusses the limitations of the system imposed by the quality of raw data and the design of the algorithms used.

9.2.1 The VHP atlas model

The VHP atlas model was constructed based on one specimen and it only represents the tongue in a fixed position. It would be beneficial to the design if more models could be produced based on either cryo-photography or high fidelity MRI. The first has certain logistical and practical complexities. There are only two cryo-photography based databases in the world, and the reason for that is that it is a very expensive and laborious task. Image-based models are more likely to happen in time with the advent of MRI of sufficient resolution and improvements in technology[1] and imaging techniques that would make the delineation of muscle group types possible on MR images.
As already mentioned, my tongue atlas, which was designed by an anatomy specialist using my tool, was based on a single template, the Visible Human

[1]Close-coupled receiver coils designated to specific areas could yield high resolution MR images with contiguous sections of fine spacing.

Project. I did, however, validate the end result and fine-tuned the model by consulting a limited number of other models from the background literature. With more models appearing in the literature it is possible to improve, a step at a time, the modelling techniques and end result. The key point here is that a collaborative attempt is needed from the community, which should make more tongue models available in the public domain in order to facilitate further research and improvements in existing models.

9.2.2 Limitations of MR images in segmentation

One of the problems that had to be dealt with in this work was the segmentation of the tongue on MR images. I described a number of automated and semi-automated methods to extract the tongue structure from the images, as well as new experimental procedures to assist the outlining of the structures in question. I consider that the problem is still not entirely solved on the clinical 1.5T scanner that was used throughout this experiment. This is because of the limitations in terms of voxel anisotropy, matrix resolution, slice thickness and contrast posed by the scanner that make the outlining of the tongue a difficult and often laborious task.

In addition, a factor that is common to all MRI scanners and moderates the image quality is that dental fillings and bridges can produce image artefacts that may distort the geometrical accuracy of the display or even prevent the visualization of the features under scan. Another factor is the effect of large distortion of the tongue outline due to presence of tumour. Under these circumstances, the segmentation procedure requires the attention and feedback of the user who must be able to judge, mainly according to experience, the boundaries of the tongue and separate the tongue from the surrounding structures. This can be a complex or impossible task on very low quality images.

Clearly, the only solution is to improve the source of data by updating the hardware and software of the MRI scanner, as described in the previous subsection. One direction is to use a relatively new imaging protocol on 3T

MRI scanner called fast low angle shot (FLASH) with T_1 weighting, a high resolution matrix and ultra thin slice thickness (¡1 mm), as it produces images that are superior in displaying the anatomy of the muscles of the face. However, not all clinical establishments have the ability to do so because of the installation and maintenance costs of advanced scanners.
The limitations of MR imaging of 1.5T scanners in routine clinical use are probably the factor that affects most the quality of results in this work, or future research that is based on this work.

9.3 Further clinical applications of the model and system

I have already referred to the clinical objective of this thesis which is to support improvements in surgical staging of tongue cancer by representing the extent of disease in relation to myoarchitecture. This section presents a number of additional clinical applications that this model and system can be used for.

9.3.1 Articulation disorders

The model can assist research on the connection of articulation and other speech disorders to muscle architecture. It would be possible to modify my system in order to identify deficits in muscle development by constructing a number of tongue atlases from different tongue specimens with classified disorders and finding the best fit.

The general concept of assessing, diagnosing and treating speech disorders can be further extended by introducing time-based studies - with sufficient spatial and temporal resolution - of different positions of phonation to the modelling approach.

9.3.2 Post-operative evaluation of partial glossectomy

Glossectomy in this context refers to the amputation of the whole or part of the tongue as a result of clinical treatment of tongue cancer. Glossectomy affects the shape and function of the tongue in a manner that is evident only post-operatively. With some modifications, the system presented in this thesis can be used to simulate the post-operative results as it can identify the muscles that will be directly affected. This information could then be correlated to a database that covers all possibilities and results. An estimation could then be made about the post-operative dysfunction, which could be used to develop a program of care for the patient.

9.3.3 Swallowing Disorders

Swallowing disorders are usually investigated on X-ray fluoroscopy studies with a frame rate that is beyond what can be achieved through MRI, at least today. If the segmentation of the tongue is possible on the images acquired then it is possible to register my model on every frame of the sequence. Several changes to the model must be made so as to process the forces, energies, deformations and restrictions imposed by the newly introduced dynamics.

As a result, the system would be able to visualize the internal structure of the tongue in 4 dimensions including time, which could possibly give insight into the relation between myoarchitecture and certain swallowing disorders.

9.3.4 Other areas

The model of the tongue could be further extended to simulate the biomechanics of the attachment of the tongue to the mandible, the Styloid process and the Hyoid bone. Such model can benefit from recent developments in segmentation of muscles of mastication, as proposed by Ng et al. [HN06]. An integrated system that takes all this information into account could then be used to investigate the role of the tongue in dysfunctions related to mandibular prognathism or obstruction of the upper airway, such as

obstructive sleep apnoea.

9.4 Future plans

The previous section presented some of the clinical implications of the project and how further research that is based on the present work could explore new solutions to given medical problems. This section shows how it is possible to extend the project while retaining the focus on tongue cancer staging. These are the new tangents for exploration that are directly relevant to the project and can be followed if time, resources and technology permit.

9.4.1 Diffusion Tensor MRI

One prospect is to use Diffusion Tensor MRI to extract more explicit details of tissue micro-structure ex-vivo. The extra information can then be used to enhance the existing model and attempt to answer the questions set in the following subsection. Diffusion Tensor is a new technique in MRI. Diffusion refers to the random movement of molecules, known as Brownian motion, which is a result of the exchange of thermal energy carried by the molecules. During this random motion, molecules probe tissue structure, cell membranes, fibres and macromolecules at a microscopic level which is beyond the resolution of normal MRI. Diffusion MRI reflects on the overall effect observed on a statistical basis in voxels that have relatively large volume, of the order of several mm^3. The observation is essentially the displacement distribution of molecules of water within a given voxel. The displacement vector can provide information about the structure and geometry of tissues, such as the fibre orientation. Wedeen et al. ([VJWG01]) have successfully used this technique on ex-vivo tongue to study the bovine tongue myoarchitecture, as shown in figure 9.1. Adding this technique successfully to my modelling approach could improve the visualization of individual muscles and their orientation.
However, Diffusion Tensor MRI is still in its early stages. Apart from the low

volume resolution, which far lower than the desired atlas resolution, the technique is extremely sensitive to motion, which prohibits in-vivo experimentation. Nevertheless, I believe that future developments in MRI hardware and the advent of new pulse sequences will eventually make this feasible.

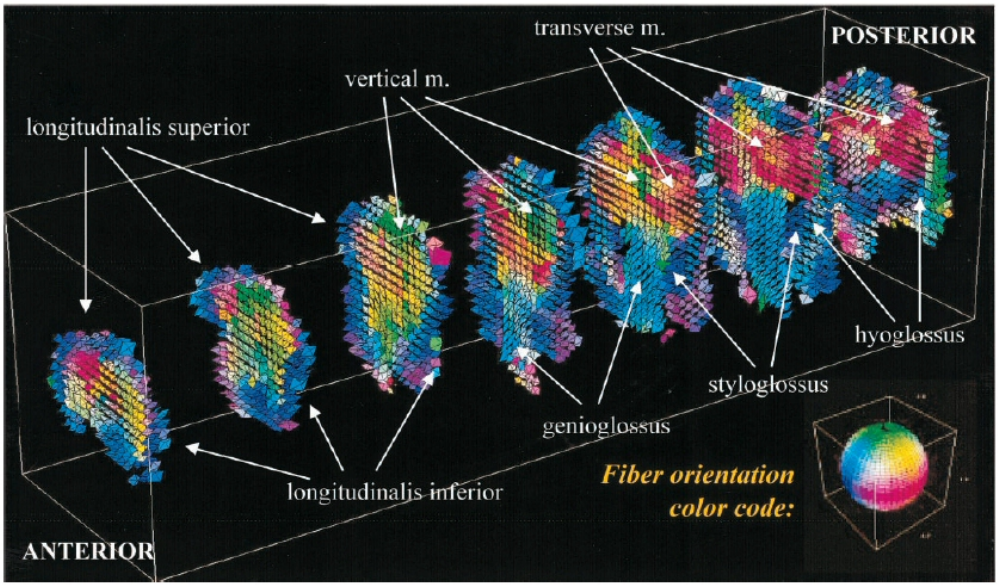


Figure 9.1: 3D multislice diffusion tensor MRI of the bovine tongue. Coronal slices comprising the bovine tongue obtained by diffusion tensor imaging, with muscle components identified and rendered with octahedra [VJWG01].

9.4.2 Direction of muscle fibres: implications in terms of staging and treatment

Another possibility for the project is to extend the model to include information on the direction of the muscle fibres [WT06a]. That would stimulate research on whether tongue cancer spreads mainly along the fibre directions of muscles or according to innervation (or both), and what the implications are in terms of staging and treatment. In order to do so, the extra informa-

tion must be added to the VHP atlas, which is technically not difficult[2]. The problem is how this information could be compiled in the first place, and what would be the source of information available to the modeller. There is some information available (most notably by the Upper airway project ([SIA02]) and the work of Tricarico et al. [WT06b]) where modelling is based on ultra high resolution images of very thin (3 mm) frontal sections of a dissected human tongue. These images are processed manually to give a coarse indication of fibre directions at a quasi-microscopic level. It is not known how valid this assessment can be since there may be substantial differences between an in-vivo and ex-vivo model. Additionally, the anatomy of the human tongue is still not completely understood, and the fibre directions are (most notably in spaces where muscles interweave) complicated or undetermined. Innervation is also complex as it involves several cranial nerves. Innervation is very difficult or impossible to acquire through photography or medical imaging. What could pragmatically happen within the next few years is that further systematic research based on photography, microscopy or Diffusion Tensor MRI could determine the accurate fibre direction and innervation patterns for most of the volume of the muscles rather than for all. The rest of the information would then be calculated through a mathematical model developed to compute the fibre orientation inside the unresolved areas. This information could then be incorporated in a model similar to mine and used in studies of tongue cancer in time. The question to answer then would be, "Is the fibre direction or innervation of muscles related to tongue cancer spread and if so should the clinical and surgical management of the malignancy take that into consideration?".

9.4.3 Image Guided Surgery and surgery simulation

Finally, this project could significantly expand into the area of image guided surgery and surgery simulation. Image guided surgery is based on augmented

[2]It would require the addition of a unit vector for every voxel in the VHP atlas to indicate fibre direction, while for innervation it would require an extra ID number.

reality during operation, in which image data and other information are superimposed onto the display of the patient. Tongue surgery could benefit from a mechanism that would augment the display of the tongue by visualizing the internal structure in real-time. The system would have the following components:

- An MRI scanner that scans the patient off-line.
- A system to track, in real-time, the position of subject and all probes/surgical instruments that interact with the subject during the operation.
- A computer system that registers the tongue off-line and then physically models the tongue model based on external influences and set constraints, in real-time.
- A VR display or a monitor that superimposes the myoarchitecture and other relevant information over the MR images.

This procedure could potentially permit surgery without direct vision or robotically guided surgery.
A similar concept could be used in surgery simulation in which a computer model simulates the operation before it is carried out. To do so, the model has to be extended to incorporate Finite Element Analysis[3] techniques that would allow the simulation of external forces that result in elastic deformation, tearing of tissue etc. The simulation could be used to predict operative results. For example, it might be possible to evaluate the results of plastic surgery in terms of how it affects the muscle architecture and overall shape of the tongue. Another use of the simulation is in lingual surgery training where the entire operation becomes virtual, without the need for a subject or a scanner.
These are all possibilities that can be pragmatically pursued in the near

[3]Finite Element Analysis is a mathematical technique that can be used to predict the behaviour of physical based system that are subject to external influences such as forces, temperatures etc.

future since the hardware and software required exists today, or should be available soon.

9.5 Conclusion

This research has presented a method to model the extrinsic muscles of the tongue on MR images for use in surgical staging of tongue cancer. The implications are that the surgeon can now visualize the affected areas and myoarchitecture concurrently in 3D and gain insight into the extent of the malignancy in a manner that was not feasible in the past. This wealth of extra information available will primarily assist the surgical management of tongue cancer and potentially be used as a solid basis for further research in tongue cancer staging and in other relevant clinical applications.

Bibliography

[AAW88] S. Tehrani A. Amini and T. Weymouth. Using dynamical programming for minimizing the energy of active contours in the presence of hard constraints. *Proceedings In'l Conf. Computer Vision*, pages 95–99, 1988.

[AG55] Kemp EH Ardran GM. A radiographic study of the movements of the tongue in swallowing. *Dent Pract*, 5:252–263, 1955.

[Ahl96] Jorgen Ahlberg. Active contours in three dimensions. Master's thesis, Linkoping University, 1996.

[AJ05] T. Murray et al. A. Jemal. Cancer statistics 2005. *CA Cancer J Clin*, 55:10–30, 2005.

[Ame04] Raid Saeed Ameen. *Comparative three-dimensional biometrics of the human head derived from CT and MRI.* PhD thesis, University of Oxford, 2004.

[And01] Christopher Jaynes And. Temporal registration using a kalman filter for augmented reality applications, 2001.

[AS72] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing.* Dover, 1972.

[AYJ96] Michael J. Ackerman, Terry Yoo, and Donald Jenkins. From data to knowledge - the visible human project® continues. 1996.

[BH81] B.G. Schunck B.K.P. Horn. Determining optical flow. *AI Memo 572. Massachusetts Institute of Technology*, 1981.

[Boo79] F.L. Bookstein. Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, 9:56, 1979.

[Boo84] F.L Bookstein. Principal warps: Thin plate splines and the decomposition of deformations. *IEEE Image Processing*, 27:321–345, 1984.

[Boo01] Carl De Boor. *A Practical Guide to Splines (Hardcover)*. Springer, 2001.

[Bos00] Paromita Bose. The encoding and fourier descriptors of arbitrary curves in 3-dimensional space. 2000.

[BR94] Carrington B.M. and Johnson R.J. *Magnetic Resonance Imaging and Computed Tomography of the Head and Neck*. Chapman and Hall, 1994.

[Bru91] M. E. Brummer. Hough transform detection of the longitudinal fissure in tomographic head images. *IEEE Trans. Med. Imag.*, 10:74–81, 1991.

[Can86] J. F. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.

[CDB96] J. Prince C. Davatzikos and R. Bryan. Image registration based on boundary mapping. *IEEE Transactions on Medical Imaging*, 15:112–115, 1996.

[CJ98] Xu C. and Prince J.L. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Proc*, pages 359–369, 1998.

[C.M92] Wu C.M. Graphics tool for 3d tongue modelling. *Journal of the acoustical Society of America*, 92(4):2390–2391, 1992.

[CM.96] Wu CM. *Computational Methods for integrating different anatomical data sets of the human tongue.* PhD thesis, Ohio State University, 1996.

[CSH89] U. Tiede C. Schiers and K.H. Hoelne. Interactive 3d registration of image volumes from different sources. *CARS 1989*, pages 667–669, 1989.

[CT98] Frédéric Cordier and Nadia Magnetat Thalmann. Comparison of two techniques for organ reconstruction using the visible human data set. In *The Second Visible Human Project Conference Proceedings*, 1998.

[Del34] Boris Delaunay. Sur la sphere vide. *Bulletin of Academy of Sciences of USSR*, 7:793–800, 1934.

[Den99a] J. Denzler. Cvonline: Geometric feature extraction methods. In R.B. Fischer, editor, *CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision.* 1999. http://www.dai.ed.ac.uk/CVonline/.

[Den99b] J. Denzler. Cvonline: Snakes, active contours. In R.B. Fischer, editor, *CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision.* 1999. http://www.dai.ed.ac.uk/CVonline/.

[DH72] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Comm. ACM*, 15:11–15, 1972.

[D.H94] D.Hawkes D.Hill, C.Studholme. Voxel similarity measures for automated image registration. *Visualization in Biomedical Computing (VBC 94)*, pages 205–216, 1994.

[DH96] Jianwu Dang and Kiyoshi Honda. An improved vocal tract model of vowel production implementing piriform resonance and

transvelar nasal coupling. *4th International Conference on Spoken Language Processing*, 1996.

[DH02] Jianwu Dang and Kiyoshi Honda. Estimation of vocal tract shapes from speech sounds with a physiological articulatory model. *Journal of Phonetics*, 30:511–532, 2002.

[DJ82] Goldstein JC Decker JL. Risk factors in head and neck cancer. *New England Journal of Medicine*, 306(19):1151–1155, 1982.

[DL03] S.Antani D.J.Lee and L.R.Long. Similarity measurement using polygon curve representation and fourier descriptors for shape-based vertebral image retrieval. *SPIE Medical Imaging, Ime processing*, 5032:1283–1291, 2003.

[DM92] Williams D.J. and Shah M. A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, 1:14–26, 1992.

[DN89] D.F.Rogers and N.G.Fog. Constrained b-spline curve and surface fitting. *Computer Aided Geometric Design*, 21:641–648, 1989.

[Dor75] G. Doran. Review of the evolution and phylogeny of the mammalian tongue. *Acta Anat.*, 92:118–129, 1975.

[DTK88] Andrew Witkin Demetri Terzopoulos and Michael Kass. Constraints on deformable models: Recovering 3d shape and nonrigid motion. *Artificial Intelligence*, 36:91–123, 1988.

[ea76] Kiritani et al. A computational model of the tongue. *Annual Bulletin of the Research Institute of Logopedics and Phoniatrics*, 10:243–251, 1976.

[ea84] VM Runge et al. Paramagnetic contrast agents in magnetic resonance imaging: research at vanderbilt university. *Physiol Chme Phy Med NMR*, 16(2):1984, 1984.

[ea86a] RB Lufkin et al. Tongue and oropharynx: findings on mr imaging. *Radiology*, 161:69–75, 1986.

[ea86b] U. Cerchiari et al. 3d correlation technique for anatomical volumes in functional stereo tactic neurosurgery. *Time-Varying Image Processing and Moving Object Recognition*, pages 147–152, 1986.

[ea92a] Gerig G. et al. Nonlinear anisotropic filtering of mri data. *IEEE Transactions on Medical Imaging*, 11(2):221–232, 1992.

[ea92b] O. Monga et al. From partial derivatives of 3d density images to ridge lines. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 354–359, 1992.

[ea92c] W.H.Press et al. *Numerical recipes in C: The art of scientific Computing*. Cambridge University Press, 1992.

[ea96a] Carl-Fredrik Westin et al. Segmentation by adaptive geodesic active contours. 1996.

[ea96b] S. Lavallee et al. Anatomy-based registration of 3-d medical images, range images, x-ray projections, 3-d models using octree splines. *Computer-Integrated Surgery*, pages 115–143, 1996.

[ea98a] Bartels et al. *An Introduction to Splines for Use in Computer Graphics and Geometric Modelling, ch. 3*. Morgan Kaufmann, 1998.

[ea98b] Jayaram K. Udupa et al. *3D Imaging in Medicine, Second edition (Hardcover)*. CRC-Press, 1998.

[ea99] Peter Dawson et al. *Textbook of contrast media*. ISIS Medical media, 1999.

[ea02a] A. Wrangsjo et al. Non-linear gaussian filtering for resampling of images. In *SSAB Symposium on Image Analysis*, 2002.

[ea02b] Roger Webster et al. Elastically deformable 3d organs for haptic surgical simulation. *Medicine Meets Virtual Reality 2002*, 2002.

[ea03a] Domink Weishaupt et al. *How does MRI work*. Springer, 2003.

[ea03b] McRobbie et al. *MRI from Picture to Proton*. Cambridge university press, 2003.

[ea04] Ian F. Tannock et al. *The Basic Science of Oncology (Paperback)*. McGraw-Hill Professional; 4th edition, 2004.

[EDC87] C. Morandi E. De Castro. Registration of translated and rotated images using finite fourier transforms. In *IEEE Transactions on pattern analysis and machine intelligence*, 1987.

[EH96] Matthias Eck and Hugues Hoppe. Automatic reconstruction of b-spline surfaces of arbitrary topological type. *Computer Graphics (Proceedings of SIGGRAPH 96)*, 1996.

[eHl05] eMedicine Health library. Cancer signs and symptoms. 2005.

[EP97] Tony Ezzat and Tomaso Poggio. Videorealistic talking faces: A morphing approach. In *AVSP 1997*, 1997.

[Eva04] Iordanis Evangelou. *3D modelling of the human tongue musculature : definition of a method based on active contours*. University of Oxford, 2004.

[FD93] James Foley and Andries Van Dam. *Introduction to computer graphics*. Addison Wesley, 1993.

[Fer01] R.S. Ferguson. *Practical algorithms for 3D computer graphics*. AK Peters, Natic Mass., 2001.

[FJ98] Matteo Frigo and Steven G. Johnson. FFTW: An adaptive software architecture for the FFT. In *Proc. 1998 IEEE Intl. Conf.*

Acoustics Speech and Signal Processing, volume 3, pages 1381–1384. IEEE, 1998.

[FJ05] Matteo Frigo and Steven G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[Fre74] Herbert Freeman. Computer processing of line-drawing images. *Computing Surveys*, 6(1):57-97, March 1974.

[Fri05] Horst Friebolin. *Basic One- and Two-Dimensional NMR Spectroscopy (Paperback)*. John Wiley & Sons, 2005.

[FS78] H. Freeman and A. Saghri. Generalized chain codes for planar curves. In *Proceedings of the 4th International Joint Conference on Pattern Recognition, pages 701-703*, November 7-10 1978.

[Gos05] A. Ardeshir Goshtasby. *2-D and 3-D Image Registration*. John Wiley & Sons, 2005.

[GPW04] C. Grigorescu, N. Petkov, and M. A. Westenberg. Contour and boundary detection improved by surround suppression of texture edges. *Image and Vision Computing*, 22(8):609–622, 2004.

[Gru05] Stephen Grund. Oral cancer. *MedLinePlus service of U.S. national library of medicine*, 2005.

[HB00] D.H. House and D.E Breen. *Cloth modelling and animation*. A K Peters, 2000.

[HFU77] Z.M. Kedem Z.M H Fuchs and S.P. Uselton. Optimal surface reconstruction from planar contours. *Communications of the ACM*, 20(10):693–702, 1977.

[HJ92] Wang H. and Brady J.M. Corner detection with sub-pixel accuracy. Technical report, Dept. Engineering Science, University of Oxford, 1992.

[HKJ94] Hirai H. Honda K. and Dang J. A physiological model of speech production and the implication of tongue-larynx interaction. *International Conference on Spoken Language Processing*, pages 175–178, 1994.

[HL77] Freeman H. and Davis L.S. A corner finding algorithm for chain code curves. *IEEE Trans. on Computers*, 26:297–303, 1977.

[HN06] Q. Hu K.W.C. Foong P.S. Goh W.L. Nowinski H.P. Ng, S.H. Ong. Muscles of mastication model-based mr image segmentation. *Int Journal CARS*, 1:137–148, 2006.

[Hor06] Joseph P. Hornak. *The Basics of MRI*. 2006.

[Hou59] P.V.C. Hough. Machine analysis of bubble chamber pictures. In *International Conference on High Energy Accelerators and Instrumentation CERN*, 1959.

[Hou67] R.A Houde. *Study of tongue motion during selected speech sounds.* PhD thesis, Speech Commun. Res. Lab., Santa Barbara, 1967.

[H.P77] Moravec H.P. Towards automatic visual obstacle avoidance. *Proc. of the International Joint Conference on Artificial Intelligence*, page 584, 1977.

[H.W03] S.R.Watt-Smith I.Evangelou H.Wright, S.J.Golding. Analysis of intralingual anatomy and its variations. In *ESHNR2003*, 2003.

[J.69] Perkell J. Physiology of speech production: Results and implications of a quantitative cineradiographic study. *Research Monograph*, 53, 1969.

[J.74] Perkell J. *A Physiologically-oriented Model of tongue activity in speech production.* PhD thesis, Massachusetts Institute of Technology, 1974.

[JB03] ImPACT Julia Barrett, Nicholas Keat. Artefacts in ct: recognition and avoidance. In *RSNA 2003*, 2003.

[JB05] B.N. Jacob J.M. Bourgery. *Atlas of human anatomy and surgery. The complete coloured plates of 1831-1854.* Tachen, 2005.

[JD91] D. Sorensen J. Dongarra, I. Duff. Solving linear systems on vector and shared memory computers. *PA: SIAM*, 1991.

[JF95] Steven Feiner John F. Hughes James Foley, Andries Van Dam. *Computer Graphics, reissued 2nd Ed.* Addison Wesley, 1995.

[JJ93] C. Roux J.J. Jacq. Automatic registration of 3d images using a simple genetic algorithm with a stochastic performance function. *IEEE Engineering in Medicine and Biology Society Proceedings*, pages 126–127, 1993.

[Joy00] Kenneth I. Joy. Bicubic uniform b-spline surface refinement. Online geometric modelling notes, Visualization and Graphics Research Group, Department of Computer Science, University of California., 2000.

[J.P96] J.Ph.Thirion. New feature points based on geometric invariants for 3d image registration. *International Journal of Computer Vision*, 18(2):121-137, 1996.

[JS78] P. M. Joseph and R. D. Spital. A method for correcting bone induced artifacts in computed tomography scanners. *Med. Phys.*, 2:100–108, 1978.

[JT02] Bushberg JT. *The essential physics of medical imaging.* Lippincott Williams & Wilkins, 2002.

[K.74] Miyawaki K. A study of the musculature of the human tongue. *Annual Bulletin of the Research Institute of Logopedics and Phoniatrics, University of Tokyo*, 8:23–50, 1974.

[Kep75] E. Keppel. Approximating complex surfaces by triangulation of contour lines. *IBM journal of research and development*, 19(1), 1975.

[KK00] Marija Ivanovic et al. Karin Kneaurek. Medical image registration. *Europhysics News (2000) Vol. 31 No. 4*, 200.

[KL96] Venkat Krishnamurthy and Marc Levoy. Fitting smooth surfaces to dense polygon meshes. 1996.

[Kle05] Lewis J. Kleinsmith. *Principles of Cancer Biology (Paperback)*. Benjamin Cummings, 2005.

[KPG05] I. Evangelou K. Pataridis and S. Golding. Bounding ellipse fitting to head images for accurate active contour initialization. In *CARS 2005*, 2005.

[Kru] CJC Kruger. Constrained cubic spline interpolation for chemical engineering applications. http://www.korf.co.uk/spline.pdf.

[KS85] X. M. Kier and K. K. Smith. Tongues, tentacles and trunks: The biomechanics of movement in muscular-hydrostats. *J. Linn. Soc. London Zool*, 83:307-324, 1985.

[KS86] Hashimoto K. and Suga S. Estimation of the muscular tensions of the human tongue by using a three-dimensional model of the tongue. *Journal of the Acoustical Society of Japan*, 7(1):39–46, 1986.

[KS88] Avinash Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, 1988.

[Kul77] Z. Kulpa. Area and perimeter measurement. *Computer Graphigs Image Process*, 6:434-451, 1977.

[KYS00] T. Nagata S. Kanda M. Ohishi K. Yuasa, T. Kawazu and K. Shirasuna. Computed tomography and ultrasonography of metastatic cervical lymph nodes in oral squamous cell carcinoma. *Dentomaxillofacial Radiology*, Vol 29, Issue 4:238–244, 2000.

[L.01] Cieplinski L. Mpeg-7 color descriptors and their applications. *Intl. Conf. on Comp. Analysis of Images and Patterns (CAIP'01)*, 2001.

[LA92] A.P. Dhawan L.K. Arata. Iterative principal axes registration: a new algorithm for retrospective correlation of mr-pet brain images. *IEEE Engineering in Medicine Biology Society (EMBS)*, pages 2776–2777, 1992.

[LF95] Smoker WR. Laine FJ. Oral cavity: anatomy and pathology. *Seminars in Ultrasound. CT MR*, 16(6):527–45, 1995.

[L.G92] L.G.Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24:325–376, 1992.

[LH02] Sobin LH. *TMN Classification of Malignant Tumours 6th edition.* Wiley, 2002.

[LK81] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679, 1981.

[LM96] Hermans R. Lenz M. Imaging of the oropharynx and oral cavity. part ii: Pathology. *Eur Radiol.*, 6(4):536–49, 1996.

[M.90] Stone M. A three-dimensional model of tongue movement based on ultra-sound and x-ray microbeam data. *Journal of the Acoustical Society of America*, 87(5):2207–17, 1990.

[Mac95] Blair Mackiewich. Intracranial boundary detection and radio frequency correction in magnetic resonance images. Master's thesis, Simon Fraser University, Computer Science Department, 1995.

[Mal39] S. Abd-El Malek. Observations on the morphology of the human tongue. *Journal of Anatomy*, 73:201–210, 1939.

[Mat04] James Matthews. Thresholding and segmentation. 2004.

[MH80] D. Marr and E. C. Hildreth. Theory of edge detection. *Proceedings of the Royal Society, London B*, 207:187–217, 1980.

[MJ95] W. Ma and J.P.Kruth. Parameterization of randomly measured points for least squares fitting of b-spline curves and surfaces. *Computer Aided Design*, 27(9):663–675, 1995.

[MK87] D. Terzopoulos M. Kass, A. Witkin. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.

[MMD83] PC Lauterbur MH Mendonca-Dias, E Gaggelli. Paramagnetic contrast agents in nuclear magnetic resonance medical imaging. *Semin Nucl Med*, 13:364–376, 1983.

[MO00] E.Guney N.Ozcan M. Ozturk, I.Yorulmaz. Masses of the tongue and floor of the mouth: findings on magnetic resonance imaging. *Eur. Radiol.*, 10:1669–1674, 2000.

[Mor98] Bryan S. Morse. Lecture notes on image processing, neighbors, connected components and distance. 1998.

[Mor04] Tim Morris. *Computer Vision and image processing.* Palgrave Macmillan, 2004.

[MP96] R.Fisher M. Pilu, A. Fitzgibbon. Ellipse-specific direct least-square fitting. In *IEEE International Conference on Image Processing, Lausanne*, 1996.

[MT79] Nagao M. and Matsuyama T. Edge preserving smoothing. *Computer Vision, Graphics, and Image Processing*, 9:394–407, 1979.

[Mur99] Shunji Murai. *GIS Work Book Volume 2.* 1999.

[MVH94] H.M. Kooy M. Van Herk. Automatic three-dimensional correlation of ct-ct, ct-mri and ct-spect using chamfer matching. *Medicine Physics*, 21:1163–1178, 1994.

[MY95] Masamune Takeda Tateo Korenaga Mayumi Yasumoto, Hitoshi Shibuya. Squamous cell carcinoma of the oral cavity: Mr findings and value of t1- versus t2-weighted fast spin-echo images. *AJR*, 164:981–987, 1995.

[NCV85] C.L. Nikias, A.P. Chrysafis, and A.N. Venetsanopoulos. The lu decomposition theorem and its implications to the realization of two-dimensional digital filters. *T-ASSP*, 33:694–711, 1985.

[Net06] Frank H. Netter. *Atlas of human anatomy 3rd edition.* Novartis medical education, 2006.

[oDD] Malene Schlaikjer Technical University of Denmark (DTU). The resonance property of the hydrogen nucleus. http://www.es.oersted.dtu.dk/ masc/resonance.htm.

[ONS02] ONS. Cancer statistics. Technical report, ONS, 2002.

[Ost86] Eben Ostby. *Describing free-form 3D surfaces for animation.* 1986.

[oV06] University of Virginia. All about cancer - diagnostic imaging. 2006.

[Pav82] Theo Pavlidis. *Algorithms for Graphics and Image Processing.* Computer Science Press, Rockville, Maryland, 1982.

[PBB93] Stephen Parry-Barwick and Adrian Bowyer. Feature technology. Technical report, University of Bath, School of Mechanical Engineering, 1993.

[Per84] Ken Perlin. Course in "advanced image synthesis". In *ACM Siggraph conference*, 1984.

[Per02] Ken Perlin. Improving noise. In *ACM Siggraph conference*, 2002.

[Pin97] R. Pinchumani. Snakes: an active model. In R.B. Fischer, editor, *CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision*. 1997. http://www.dai.ed.ac.uk/CVonline/.

[PJ90] Perona P. and Malik J. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–39, 1990.

[PJB92] N. D. McKay P. J. Bes. A method for registration of 3-d shapes. *IEEE Trans. Pat. Anal. and Mach. Intel.*, 14:239–256, 1992.

[PPE88] Abry C. Perrier P. and Keller E. Toward a model of tongue dorsum movements. *Jouranl d' Acoustique*, 2(1):69–77, 1988.

[Pre70] J. M. S. Prewitt. Object enhancement and extraction. In B. S. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psychopictorics*, pages 75–149. Academic Press, New York, 1970.

[PSRS96] Wladimir Peckar, Christoph Schnörr, Karl Rohr, and H. Siegfried Stiehl. Non-rigid image registration using a parameter-free elastic model. 1996.

[R.95] Tricarico Wilhelms R. Physiological modelling of speech production: Methods for modelling soft-tissue articulators. *Journal of the Acoustical Society of America*, 97(5):3085–3098, 1995.

[RC83] Chin RT. and Yeh CL. Quantitative evaluation of some edge preserving noise-smoothing techniques. *CVGIP*, 23:67–91, 1983.

[RCM95] A.J.Chippindale G.Wilson N.Mclean J.V.Soames R.S.D Campbell, E.Baker and M.F.Reed. Mri t staging of squamous cell carcinoma of the oral cavity: Radiological-pathological correlation. *Clinical Radiology*, 50:533–540, 1995.

[RFW00] Ashley Walker Robert Fisher, Simon Perkins and Erik Wolfart. Convolution. *The online Hypermedia Image Processing Reference*, 2000.

[RGC95] A. Zisserman T. J. Mc Donald R. Guillemaud, P. Marais and B. Crow. A 3-dimensional midsagittal plane for brain asymmetry measurement. *Schizophrenia Res.*, 18(2):183–184, 1995.

[RH91] David R.Forsey and Richard H.Bartels. Surface fitting with hierarchical spline. *Topics in the construction, manipulation, and assessment of spline surfaces SIGGRAPH course 25*, pages 700–714, 1991.

[Rob63] L G Roberts. *Machine perception of three-dimensional solids.* PhD thesis, MIT, 1963.

[RS89] R.Bajcsy and S.Kovacic. Multiresolution elastic matching. *Computer Vision, Graphics, and Image processing*, 46:1–21, 1989.

[RS96] S.Lavallee R. Szeliski. Matching 3-d anatomical surfaces with non-rigid deformations using octree-splines. *International Journal of Computer Vision (IJVC)*, 18(2):171–186, 1996.

[SF03] S.Osher and Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces.* Springer-Verlag, 2003.

[SG04] Robert Strzodka and Christoph Garbe. Real-time motion estimation and visualization on graphics cards. In *Proceedings IEEE Visualization 2004*, pages 545–552, 2004.

[SIA02] Amirali A. Sanders I., Mu L. and Iskander A. A 3-dimensional atlas of human tongue muscles. *Visible Human Project Conference, Keystone, Colorado*, 2002.

[SJ02] H Sahni and PN Jayakumar. Anaphylactoid reaction following intravenous gadolinium dimeglumine. *Ind J Radiol Imag*, 12-1:81–82, 2002.

[SP91] Stan Sclaroff and Alex Pentland. Generalized implicit functions for computer graphics. *Computer Graphics (SIGGRAPH 91 proceedings)*, 25:247–250, 1991.

[SS04] Tatiana Surazhsky and Vitaly Surazhsky. Sampling planar curves using curvature-based shape analysis. *Mathematical Methods for Curves and Surfaces*, 2004.

[Sto91] M. Stone. Imaging the tongue and vocal tract. In *British Journal of Disorders of Communication 26.11-23*, 1991.

[SVY97] Laboissiere R. Sanguineti V. and Payan Y. A control model of human tongue movements in speech. *Bio Cybern*, 77(1):11–22, 1997.

[SWC04] Edgar A. Ramos Tathagata Ray Siu-Wing Cheng, Tamal K.Dey. Sampling and meshing a surface with guaranteed topology and geometry. *Proceedings of 20th Annual Symposium in Computer Geometry*, pages 280–289, 2004.

[Tak01] H. Takemoto. Morphological analysis of the human tongue musculature for three-dimensional modeling. *J Speech Lang Hear Res*, 44:95–107, 2001.

[TLT03] Antonio Wilson Vieira Thomas Lewiner, Helio Lopes and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of Graphics Tools*, 8(2):1–15, 2003.

[Ung85] JM Unger. The oral cavity and tongue: magnetic resonance imaging. *Radiology*, 155:151–153, 1985.

[VJWG01] Vitaly J. Napadow Van J. Wedeen, Timothy G. Reese and Richard J. Gilbert. Demonstration of primary and secondary muscle fiber architecture of the bovine tongue by diffusion tensor magnetic resonance imaging. *Biophysical Journal*, 80:1024–1028, 2001.

[Wat99] Kenneth L. Watkin. Ultrasound and swallowing. *Folia Phoniatr Logop*, 51:183–198, 1999.

[Wei99] Eric W. Weisstein. Homothetic transformation. *MathWorld Wolfram Web resource*, 1999.

[WFJ01] Xiu Ying Wang, David Dagan Feng, and Jesse Jin. Elastic medical image registration based on image intensity. *ACM International Conference Proceeding Series*, 147, 2001.

[Woo00] Roger P. Woods. *Spatial Transformation Models*. CLA School of Medicine; Academic Press, 2000.

[WSH03] F. R. Rybicki R. V. Mulkern C.F. Westin W. S. Hoge, D. Mitsouras. Registration of multi-dimensional image data via subpixel resolution phase correlation. In *IEEE Intl. Conf. on Image Processing (ICIP-03)*, pages 707–710, Barcelona, Spain, 2003.

[WT06a] Reiner Wilhelms-Tricarico. June 2006 personal communications. 2006.

[WT06b] Reiner Wilhelms-Tricarico. Tongue modeling. http://webpages.charter.net/reinerwt/themodel.htm, 2006.

[WTC91] R Wilhelms-Tricarico and Wu C.M. Dynamic 3d tongue model - progress report. *Journal of the Acoustical Society of America*, 90(4):14–26, 1991.

[Y.77] Kakita Y. Computational model of the tongue: A revised version. *94th Meeting: Acoustical Society of America*, pages 15–16, 1977.

[You94] David Young. Sussex computer vision: Teach vision6. 1994.