Development of a Multi-body Statistical Shape Model of the Wrist

by

Anton Semechko

A Thesis
presented to
The University of Guelph

In partial fulfilment of requirements
for the degree of
Master of Applied Science

Guelph, Ontario, Canada
© Anton Semechko, December, 2011
ABSTRACT

DEVELOPMENT OF A MULTI-BODY STATISTICAL SHAPE MODEL OF THE WRIST

Anton Semechko
University of Guelph, 2011

Advisors:
Dr. Michele Oliver
Dr. Robert Dony

With continually growing availability of high performance computing resources, the finite element methods (FEM) are becoming increasingly more efficient and practical research tools. In the domain of computational biomechanics, FEMs have been successfully applied in investigation of biomedical problems that include impact and fracture mechanics of bone, load transmission through the joints, feasibility of joint replacements, and many others. The present research study was concerned with the development of a detailed, anatomically accurate, finite element model of the human hand and wrist. As a first step in this direction, we used a publically available database of wrist bone anatomy and carpal kinematics to construct a multi-body statistical shape model (SSM) of the wrist. The resulting model provides an efficient parameterization of anatomical variations of the entire training set and can thus overcome the major shortcoming of conventional biomechanical models associated with limited generalization ability. The main contributions of this work are:

- A robust method for constructing multi-body SSM of the wrist from surface meshes.
- A novel technique for resampling closed genus-0 meshes to produce high quality triangulations suitable for finite element simulations.

Additionally, all techniques developed in the course of this study could be directly applied to create an equivalent model of the tarsus.
Acknowledgements

I am deeply grateful to my academic advisers, Dr. Michele Oliver and Dr. Robert Dony, for providing the opportunity to work on this project. I would also like to thank my family, especially my dad, Vladimir Semechko, for continually encouraging me to pursue my academic interests. Thank you all for your patience and support.
# Contents

List of Tables .................................................................................................................. vi
List of Figures .................................................................................................................. vii
List of Abbreviations ...................................................................................................... xi

Chapter 1 ......................................................................................................................... 1
  Introduction .................................................................................................................... 1
  1.1 Osteology of the Wrist ......................................................................................... 3
  1.2 Contributions ....................................................................................................... 5
  1.3 Overview .............................................................................................................. 6

Chapter 2 ......................................................................................................................... 8
  High Quality Remeshing of Closed Genus-0 Surfaces ........................................... 8
  2.1 Motivation ........................................................................................................... 9
  2.2 Related Work ..................................................................................................... 12
  2.3 Methods ............................................................................................................ 15
    2.3.1 Removal of Thin Triangles ......................................................................... 17
    2.3.2 Conformal Parameterization ................................................................. 18
    2.3.3 Distortion Field ......................................................................................... 23
    2.3.4 Adaptive Sampling of the Sphere .......................................................... 24
    2.3.5 Subdivision & Regularization .................................................................. 26
    2.3.6 Controlling Mesh Complexity ................................................................. 27
    2.3.7 Assessing Mesh Quality .......................................................................... 28
  2.4 Results .................................................................................................................. 29
  2.5 Discussion ......................................................................................................... 39
  2.6 Conclusion .......................................................................................................... 47

Chapter 3 ......................................................................................................................... 49
  Statistical Shape Models of Carpal Bones ............................................................... 49
  3.1 Statistical Shape Models Background ................................................................ 52
    3.1.1 Shape Representation ................................................................................ 52
    3.1.2 Statistical Analysis .................................................................................... 54
    3.1.3 Correspondence ......................................................................................... 56
    3.1.4 Description Length Cost Function .......................................................... 63
    3.1.5 Optimization of Landmark Correspondences .......................................... 66
  3.2 Methods ............................................................................................................... 72
    3.2.1 Linear Surface Registration ...................................................................... 73
      3.2.1.1 Surface Registration with Distance Transforms .................................. 75
    3.2.2 Mobius Transformation Normalization ................................................... 77
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Mesh complexity, quality and reconstruction errors before and after resampling.</td>
<td>30</td>
</tr>
<tr>
<td>3.1</td>
<td>SSM quality measures.</td>
<td>62</td>
</tr>
<tr>
<td>3.2</td>
<td>Initial number of landmarks used to optimize correspondence.</td>
<td>83</td>
</tr>
<tr>
<td>3.3</td>
<td>Surface correspondence optimization summary.</td>
<td>116</td>
</tr>
<tr>
<td>3.4</td>
<td>p-values of the Kolmogorov-Smirnov test used to verify if the distributions of shape parameters were significantly different from the normal distribution.</td>
<td>117</td>
</tr>
<tr>
<td>4.1</td>
<td>List of normalized $\delta$ values, where $\delta = \delta/(\text{AGT})$. AGT stands for average gap thickness and equals twice the mean distance between the gliding surface and the cortical surface of one of the articulating bones.</td>
<td>136</td>
</tr>
<tr>
<td>4.2</td>
<td>Performance summary of three different wrist registration schemes.</td>
<td>156</td>
</tr>
<tr>
<td>4.3</td>
<td>p-values of the Kolmogorov-Smirnov test used to verify if the distributions of shape parameters were significantly different from the normal distribution.</td>
<td>162</td>
</tr>
<tr>
<td>A.1</td>
<td>Laplacian coefficients.</td>
<td>170</td>
</tr>
<tr>
<td>D.2</td>
<td>Expressions used to test if an arbitrary point is inside a planar triangle.</td>
<td>180</td>
</tr>
</tbody>
</table>
# List of Figures

1.1: Palmar view of the right hand showing the arrangement of the wrist bones. .............................................4  
1.2: Wireframe representation of one of the wrist models of the right hand in neutral posture. .........................4  
1.3: Carpal bones of the right hand. .................................................................................................................5  
2.1: Original meshes of eight carpal bones from one of the sample wrists. .........................................................10  
2.2: Example of one the original (hamate) meshes. .............................................................................................11  
2.3: Mesh resampling procedure. .......................................................................................................................16  
2.4: Half-edge collapse.........................................................................................................................................17  
2.5: Harmonic weight associated with an edge depends on the two angles opposing the edge. .........................19  
2.6: Tangential derivative.....................................................................................................................................21  
2.7: Angle weighted normal .................................................................................................................................21  
2.8: Planar and spherical triangles.......................................................................................................................23  
2.9: One iteration of triangular quadrisection for a planar triangle. ....................................................................27  
2.10: The effect of regularization. .......................................................................................................................27  
2.11: Original and resampled meshes of scaphoid. ...............................................................................................31  
2.12: Original and resampled meshes of lunate. .................................................................................................32  
2.13: Original and resampled meshes of trapezium. ...........................................................................................33  
2.14: Original and resampled meshes of hamate. ...............................................................................................34  
2.15: Original and resampled meshes of proximal end of femur. .................................................................35  
2.16: Original and resampled meshes of distal end of femur. ............................................................................36  
2.17: Original and resampled meshes of proximal end of tibia. .......................................................................37  
2.18: Original and resampled meshes of a molar. ..............................................................................................38  
2.19: One of the limitations of our remeshing algorithm in its present form is that it is unable to preserve salient features of artificial shapes such as the one shown above. Sharp edges present in the original mesh (left) are not fully retained in the resampled mesh (right). .........................................................40  
2.20: The digits of the hand could not be resampled due to large distortions produced at the fingertips in the parameter domain. The corresponding area distortion map is shown in Figure 2.21. The resampled mesh was obtained by subdividing the base mesh of 700 vertices two times. .................41  
2.21: Logarithm of area distortion field superimposed on the original mesh (top left) and conformal parameterization (top right). Note how the digits of the hand are compressed to very small regions in the parameter domain. Area distortion at the distal ends of middle and ring fingers is $>10^{-6}$ ............43  
3.1: Eighteen samples of capitate, demonstrating the degree of anatomical variability and absence of visually consistent salient features necessary to establish dense correspondence........................................59  
3.2: Illustration of why pairwise proximity based measures cannot guarantee correct correspondence. The distance-based correspondences between the landmarks on two surface profiles are shown by the black (solid) lines while the correct correspondences are shown by the green (dashed) lines. These types of incorrect matches are more likely to occur if there is a significant amount of local anatomical variation across the training set. .................................................................60
3.3: Demonstration of a symmetric theta transformation applied on a unit sphere. In each case, two views (top and bottom) are presented. The color denotes relative degree of length distortion with compression and stretching being treated equally (blue=no distortion). .................................69

3.4: Examples of deformations obtained with symmetric-theta transformations. Object on the far left is the original mesh. The remaining objects (from left to right) have been obtained by deforming the original mesh with a random configuration of 4, 6 and 10 Cauchy kernels. ........................................69

3.5: An example of four different scales of truncated Gaussian kernels uniformly distributed across the surface of the unit sphere. ...........................................................................................................71

3.6: Deformations produced by the first two kernels configuration shown in Figure 3.5 due to the displacement of 10 (left) and 8 (right) degrees in the latitudinal direction. The red dots denote the displacement of points situated at the kernel centroids (black dots) after the deformation. Edge color represents the relative amount of length distortion with compression and stretching being treated equally (blue=no distortion). ...........................................................................................................71

3.7: Color coded angular displacement map due to a single kernel with displacement of 10 degrees in the latitudinal direction. Red dot denotes the position of the point coincident with kernel centroid (black dot) after the deformation. Note the local nature of the deformation. Also observe how the vertices above the red point are pulled apart and those below it are compressed. ..................72

3.8: Scaphoid MDL parameter selection ...........................................................................................................89

3.9: Lunate MDL parameter selection ....................................................................................................................90

3.10: Triquetrum MDL parameter selection ............................................................................................................91

3.11: Pisiform MDL parameter selection ...................................................................................................................92

3.12: Trapezoid MDL parameter selection ..............................................................................................................93

3.13: Trapezium MDL parameter selection ...............................................................................................................94

3.14: Capitate MDL parameter selection ....................................................................................................................95

3.15: Hamate MDL parameter selection ....................................................................................................................96

3.16: Extremes of the first four principal components of the scaphoid SSM .............................................................100

3.17: Colormap of local variations described by the first four modes of the scaphoid SSM superimposed on the mean of the training set. ................................................................................................................100

3.18: Changes in the scaphoid SSM quality during the course of correspondence optimization ................................101

3.19: Extremes of the first four principal components of the lunate SSM ...............................................................102

3.20: Colormap of local variations described by the first four modes of the lunate SSM superimposed on the mean of the training set. ........................................................................................................102

3.21: Changes in the lunate SSM quality during the course of correspondence optimization ..................................103

3.22: Extremes of the first four principal components of the triquetrum SSM. ........................................................104

3.23: Colormap of local variations described by the first four modes of the triquetrum SSM superimposed on the mean of the training set. ........................................................................................................104

3.24: Changes in the triquetrum SSM quality during the course of correspondence optimization ....................105

3.25: Extremes of the first four principal components of the pisiform SSM ..........................................................106

3.26: Colormap of local variations described by the first four modes of the pisiform SSM superimposed on the mean of the training set. ........................................................................................................106

3.27: Changes in the pisiform SSM quality during the course of correspondence optimization ..........................107

3.28: Extremes of the first four principal components of the trapezoid SSM ........................................................108
3.29: Colormap of local variations described by the first four modes of the trapezoid SSM superimposed on the mean of the training set. .................................................................108
3.30: Changes in the trapezoid SSM quality during the course of correspondence optimization. ......109
3.31: Extremes of the first four principal components of the trapezium SSM. ..........................110
3.32: Colormap of local variations described by the first four modes of the trapezium SSM superimposed on the mean of the training set. .................................................................110
3.33: Changes in the trapezium SSM quality during the course of correspondence optimization ....111
3.34: Extremes of the first four principal components of the capitate SSM. ..............................112
3.35: Colormap of local variations described by the first four modes of the capitate SSM superimposed on the mean of the training set. .................................................................112
3.36: Changes in the capitate SSM quality during the course of correspondence optimization. ....113
3.37: Extremes of the first four principal components of the hamate SSM. ...............................114
3.38: Colormap of local variations described by the first four modes of the hamate SSM superimposed on the mean of the training set. .................................................................114
3.39: Changes in the hamate SSM quality during the course of correspondence optimization. ......115
3.40: Distribution of shape parameters for the first twelve modes of scaphoid SSM. Taken together these modes describe 94.32% of variations present in the training set of 30 samples. ..............118
3.41: Distribution of shape parameters for the first twelve modes of lunate SSM. Taken together these modes describe 90.60% of variations present in the training set of 30 samples. ..................119
3.42: Distribution of shape parameters for the first twelve modes of triquetrum SSM. Taken together these modes describe 91.64% of variations present in the training set of 30 samples. ......................120
3.43: Distribution of shape parameters for the first twelve modes of pisiform SSM. Taken together these modes describe 93.44% of variations present in the training set of 30 samples. ......................121
3.44: Distribution of shape parameters for the first twelve modes of trapezoid SSM. Taken together these modes describe 90.61% of variations present in the training set of 29 samples. .................122
3.45: Distribution of shape parameters for the first twelve modes of trapezium SSM. Taken together these modes describe 87.36% of variations present in the training set of 30 samples. ..................123
3.46: Distribution of shape parameters for the first twelve modes of capitate SSM. Taken together these modes describe 90.60% of variations present in the training set of 30 samples. ......................124
3.47: Distribution of shape parameters for the first twelve modes of hamate SSM. Taken together these modes describe 89.24% of variations present in the training set of 30 samples. .................125
3.48: a) Conformal parameterization of the hamate mesh. Color denotes the amount of area distortion, with red corresponding to expansion and blue to compression. b) Using the algorithm described in Chapter 2, we can adaptively sample the parameter domain to produce uniform distribution of landmarks in the primary domain. c) Example of how uniform sampling of the parameter domain can lead to poor initialization of landmarks in the primary domain. .........................................................127
4.1: Two principal anatomical axes of rotation of the wrist. The diagram on the right is shown in palmar view. Rotation axis for pronation/supination is not shown because these movements are a function of relative motion between radius and ulna. ..............................................133
4.2: Graph representation of inter-carpal articulations. ............................................................137
4.3: Cross-sections of two articulating surfaces, U and W. Both U and W are open surfaces. Consequently, during registration, multiple vertices from U may be matched to the boundaries on W (and vice versa). To avoid this problem, all vertices matched to the boundary points must be discarded (dashed lines). Only matches shown by solid arrows should be used to evaluate $J_{int}$..........137
4.4: Region of intersection of two DT domains ................................................................. 147
4.5: Gliding surface is a subset of the medial surface interposed exactly halfway between the cortical surfaces of two articulating bones. ................................................................. 148
4.6: An actual example of medial and gliding surfaces interposed between capitate and lunate. .......... 149
4.7: a) Unprocessed medial and gliding surfaces extracted using marching cubes algorithm. b) Medial and gliding surfaces after removal of extraneous patches and resampling with dynamic particle algorithm. ................................................................. 149
4.8: a) After projection, the point $T(g_1)$ is matched to $p_1$, and $T(g_2)$ is matched to $p_2$. b) In the example shown, $T(g_1)$ was not matched to the interior of $G$ and therefore will not be used in evaluation of (4.21). Conversely, $T(g_2)$ has a matching counterpart on $G$ and is therefore suitable for evaluation of the constraints. ................................................................. 152
4.9: Comparison of the composite SSM quality for different registration schemes. ....................... 157
4.10: Extremes of the first eight principal modes of carpus variation ........................................ 160
4.11: Colormap of local variations described by the first eight principal modes superimposed on the mean of the training set ................................................................................. 161
4.12: Distribution of shape parameters for the first twelve modes of the composite SSM. Taken together these modes describe 91.7% of the variations present in the training set of 28 samples .......... 162
A.1: Harmonic weight associated with an edge depends on the two angles opposing the edge. .......... 171
D.1: Angle weighted normal ...................................................................................................... 178
D.2: The seven cases of a distance to a triangle ........................................................................ 178
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALPP</td>
<td>Area and Length Preserving Parameterization</td>
</tr>
<tr>
<td>CF</td>
<td>Cost Function</td>
</tr>
<tr>
<td>COP</td>
<td>Constrained Optimization Problem</td>
</tr>
<tr>
<td>DF</td>
<td>Distortion Field</td>
</tr>
<tr>
<td>DL</td>
<td>Description Length</td>
</tr>
<tr>
<td>DT</td>
<td>Distance Transform</td>
</tr>
<tr>
<td>FE</td>
<td>Finite Element</td>
</tr>
<tr>
<td>GCO</td>
<td>Groupwise Correspondence Optimization</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>GS</td>
<td>Gliding Surface</td>
</tr>
<tr>
<td>ICP</td>
<td>Iterative Closest Point</td>
</tr>
<tr>
<td>LMA</td>
<td>Levenberg-Marquardt Algorithm</td>
</tr>
<tr>
<td>MDL</td>
<td>Minimum Description Length</td>
</tr>
<tr>
<td>MTN</td>
<td>Möbius Transformation Normalization</td>
</tr>
<tr>
<td>NURBS</td>
<td>Non-Uniform Rational B-Splines</td>
</tr>
<tr>
<td>PC</td>
<td>Point Correspondence</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>PDM</td>
<td>Point Distribution Model</td>
</tr>
<tr>
<td>RSI</td>
<td>Repetitive Strain Injury</td>
</tr>
<tr>
<td>SDT</td>
<td>Signed Distance Transform</td>
</tr>
<tr>
<td>SPHARM</td>
<td>Spherical Harmonics</td>
</tr>
<tr>
<td>SQP</td>
<td>Sequential Quadratic Programming</td>
</tr>
<tr>
<td>SSM</td>
<td>Statistical Shape Model</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

According to the Canadian Community Health Survey carried out by Statistics Canada in 2000/2001 approximately 1 out 10 Canadian adults had a repetitive strain injury (RSI) serious enough to limit their normal activities [1]. By general definition, RSI is an injury of the musculoskeletal and/or nervous systems that may be caused by any single or a combination of the factors which include but are not limited to repetitive movements, sustained static or awkward postures, forceful exertions and vibrations [1,2]. RSIs affect millions of people worldwide and cost private businesses hundreds of billions of dollars every year [2-4]. In fact, it is estimated that approximately 23% of all RSIs occur in the hand and wrist [1]. Examples of common RSIs of the upper limb are the carpal tunnel syndrome, tendonitis and osteoarthritis of the wrist. Presently, the mechanisms which govern the onset of RSI in the hand and wrist are not fully understood and remain active areas of research [2]. Knowledge gained from these studies can be used to develop preventative methods and tools to reduce the effects of exposure to various risk factors associated with the RSIs.

One approach of gaining insight about the pathophysiological mechanisms of RSIs in the hand and wrist is to study hand and wrist biomechanics in response to precisely controlled loading scenarios. Experiments of this sort, however, cannot be performed in vivo, thus leaving the cadaveric and computer simulation studies as the remaining alternatives. The former offer the benefit of working with real tissues, but require
invasive disruption of the joints to insert tracking devices and/or pressure sensors. The later also have specific shortcomings but offers the advantage of non-invasive load bearing simulation and parametric study [5]. The utility of computational biomechanics has been explored widely in the analysis of several joints including the knee, hip, shoulder, and spine [6]. More recently, a number of hand and wrist finite element (FE) studies have also been reported [5-15].

The accuracy and predictive ability of all computer based simulations depend on at least two factors. The first is related to the assumptions about the mechanical properties and constitutive equations appropriate for different tissues, an issue beyond the scope of this research study. The second consideration is related to the accuracy of the physical model. Such a model should be a faithful anatomical approximation, thereby capturing the geometry and relative position of as many relevant components as possible. This is an especially imperative requirement for a multi-body structure such as the wrist because its behaviour is determined by the interaction of multiple rigid and soft tissue constituents [5].

Fortunately, the information required to create accurate anatomical models is more accessible now than ever before in large part due to the increased availability of high resolution medical imaging systems (CT and MRI). Given the abundance of data, the task of creating accurate, detailed biomechanical models, representative of a particular segment of the human population, is far from a trivial undertaking, and this study is meant to be a first step in that direction. Since skeletal structures provide the supporting framework for all soft tissues in the human body, creating a model of the bony geometry is a solid place to start. In all, there are 27 bones in the hand and wrist. A recently
developed open-access database of the wrist anatomy and carpal kinematics [16] contains sufficient data to create a parametric model for 8 of those bones. In the course of further study it is hoped that more data will be acquired to complete the skeletal model using the same methods that were developed in the course of this research.

1.1 Osteology of the Wrist

The wrist is interposed between the distal aspect of the forearm and the proximal aspect of the hand, and acts as a pivot to enable a continuous range of motions between the three anatomical directions. As illustrated in Figure 1.1, the wrist is composed of eight carpal bones as well as their articulations to the two forearm bones (radius and ulna) and the five metacarpals. In the remainder of this manuscript, the terms carpus (referring to the complex of eight carpal bones) and wrist will be used interchangeably.

The carpal bones exhibit a physical arrangement that naturally allows them to be grouped into proximal and distal rows. In radial to ulnar order, the bones in the proximal row are the scaphoid, lunate, triquetrum and pisiform. The pisiform articulates with the palmar surface of the triquetrum, and is thus separated from the other carpal bones, all of which articulate with their neighbours [17]. The other three proximal bones form an arch which articulates with the radius and articular disc of the distal radioulnar joint [17]. In the similar order, the bones of the distal row are the trapezium, found at the base of the thumb, trapezoid, capitate, and hamate. Figures 1.1 and 1.2 show the physiological arrangement of the wrist bones of the right hand. The individual carpal bones are shown in Figure 1.3.
Figure 1.1: Palmar view of the right hand showing the arrangement of the wrist bones. Image adapted from [17].

Figure 1.2: Wireframe representation of one of the wrist models of the right hand in neutral posture (palmar view). Metacarpals not shown.
The main objective of the present study was the development of a method that could be used to construct of parametric models of the skeletal structures found in the hand and the wrist. To this end, we chose to pursue a population-based modeling strategy based on the statistical theory of shape. The resulting statistical shape models incorporate the anatomical variability of all training instances, and thus in addition to providing parametric representation, they circumvent the major drawback of the traditional, subject-specific models of having limited generalization ability. Since the hand and especially the wrist, are compact structures whose biomechanics depend on the interaction of multiple components [5], to ensure accurate physical description, the geometries of the individual structures must be coupled within a single model. Hence the primary contribution of this study is:
• Development and implementation of a robust method to create statistical shape models of compact multi-body structures.

The carpal bone data used in this study came in the form of triangular surface meshes [16]. Unfortunately most of the meshes were not readily suitable for use during the model building stages due to a large number of nearly degenerate triangles that corrupted mesh quality and increased their complexity by a considerable amount. The lack of easily implementable mesh resampling algorithms that concurrently addressed the issues of quality and complexity inspired us to develop our own technique. Consequently, the second contribution is:

• A novel procedure for improving the overall quality of closed genus-0 surface meshes.

1.3 Overview

The remainder of this manuscript is organized as follows. Chapter 2 describes the motivation for developing a new mesh resampling procedure and the methods by which it is realized. The effectiveness of the remeshing technique is demonstrated on 239 carpal bones [16] and a small set of anatomical objects downloaded from public databases [18,19]. In Chapter 3, we review previously reported model construction techniques pertinent to the creation of the computer-based models of the upper limb and contrast them with the merits of the statistical shape modeling framework adopted in this study. One of the objectives of this chapter is to introduce the reader to statistical shape models

1 Closed genus-0 surfaces are topologically equivalent to a sphere. Such surfaces are characteristic of various everyday objects as well as many organs in the human body (brain, kidneys, liver, lungs, prostate, etc.) including the bones in the hand and wrist.
(SSM). Therein we provide the mathematical theory of SSMs, and discuss concepts and methods related to their construction, with specific emphasis on approaches based on the minimum description length principle. We also describe a robust procedure for establishing dense point correspondences across a set of training shapes and evaluate the quality of the resulting models using the standard SSM quality measures of compactness, specificity and generalization ability. Chapter 4 addresses the core problem, concerned with the development of a multi-body SSM of the entire carpus. We review the existing methods for coupling multiple bones within a single model and build upon them to develop a constrained registration technique that can be used for constructing composite SSM for ensembles of tightly packed, interdependent rigid structures. Finally, the last chapter of this manuscript provides a concise summary of the findings, their significance in the context of biomechanical modeling, their potential for supplementary applications as well as reflection on the direction for future work.
Chapter 2

High Quality Remeshing of Closed Genus-0 Surfaces

Triangle meshes are one of the simplest and perhaps the most widely used surface representations. They are central to various methods in computer graphics, computer assisted design, finite element (FE) simulations and related areas. Meshes are generally subject to two common considerations of quality and complexity. For instance, in applications where meshes are used to discretize boundary value problems, to ensure good convergence and accuracy of the solution, aspect ratios of triangle faces should fall within a relatively narrow range [20]. Additionally, in applications where time performance is important, meshes representative of the underlying geometry, reduced complexity and good quality may be desirable. The main objective of the current study is to create statistical shape models of the wrist suitable for FE simulations. Therefore, the issues of mesh quality and complexity were equally important. As one of the contributions of this research, we present a novel, modular algorithm that can be used to address either one of the above issues separately or simultaneously. The proposed method is realized by combining the properties of conformal maps (see Appendix A) with the physical principles governing the dynamics of charged particle systems.

The remainder of this chapter is organized as follows. Section 1 provides motivation for the development of a new resampling technique. In Section 2 we provide a review of mesh resampling procedures most similar to ours, which rely on parameterizations and/or
physical principles to obtain high quality meshes. The specific details of our algorithm are the subject matter of Section 3. In Section 4 we demonstrate the efficacy of our resampling procedure by applying it to a set of carpal bones and a number of other anatomical objects with intricate geometries. The attributes of our method, its known limitations and provisional solutions to these limitations are discussed in Section 5. Section 6, concludes this chapter with the summary of the findings and an outline of future research objectives.

2.1 Motivation

The carpal bone data used in the present study was obtained from a publically available database of wrist bone anatomy [16] and came in the form of triangular surface meshes. Unfortunately most of the meshes were not readily suitable for use during the model building stages due to a large number of nearly degenerate triangles (i.e. triangles with almost collinear vertices) that corrupted mesh quality and increased its complexity by a considerable amount. Although these artefacts did not have any effect on structural integrity of the shapes, they did have an adverse effect on the computational efficiency of all subsequent mesh-based processing operations.

The most probable cause of these artefacts was the fact that the surfaces were fitted “with a mosaic of individual NURBS patches” [16]. Non Uniform Rational B-Splines (NURBS) are common surface representations and it is indeed possible to produce high quality surface meshes using NURBS as input. However, most meshing algorithms triangulate the patches individually, thus generating vertices along the boundaries of the patch [21]. In the presence of thin NURBS patches the resulting triangulation tend to
contain small, distorted triangles with very acute angles [22,23]. As an example, Figure 2.1 shows a wireframe model of eight carpal bones from one of the wrist and Figure 2.2 shows the close-up the hamate mesh. In both images, note how the surfaces are composed of multiple patches and observe the lines of dense triangulations along the seams of these patches. It is those thin regions that ultimately corrupt the quality of the meshes and increase their complexity by a considerable amount.

*Figure 2.1:* Original meshes of eight carpal bones from one of the sample wrists.
Figure 2.2: Example of one the original (hamate) meshes obtained from the wrist database [16].
2.2 Related Work

Remeshing of a surface refers to the procedure of resampling the input mesh to create another that has an overall better quality\(^2\) but also retains the topology and salient features of the original surface. A literature survey of remeshing techniques reveals an enormous amount of research done to address this problem and interested readers may refer to [24] for more detailed discussion of this subject. In the remainder of this section we provide a brief overview of remeshing techniques most similar to ours, namely the methods that rely on parameterization or physics based principles to obtain high quality triangulations.

Parameterization of a surface mesh is the problem of establishing a bijective (i.e. invertible) mapping between the original piecewise linear surface and a continuous parameter domain which is a primitive topological analog of the surface (see Appendix A). The methods described herein are tailored specifically for closed genus-0 surfaces, which are characteristic of various everyday objects as well as many organs in the human body (brain, kidneys, liver, lungs, prostate, etc.) including the bones in the wrist. For such surfaces, the unit sphere is the most natural parameter domain [25,26]. Planar parameterizations of closed genus-0 surfaces, however, are also common [92-95] and may be advantageous in certain situations (e.g. texture mapping) but are less natural from a global point of view, as they tend to produce higher distortions. Remeshing algorithms that rely on parameterizations fall into two categories. One approach consists of tessellating the parameter domain and then mapping the result back to the original surface [21,26-29,99]. This operation is very intuitive but usually requires maps that minimize

\(^2\) Mesh quality can refer to any of one or a combination of criteria that include the sampling, grading, regularity, size and shape of elements [2].
global area and length distortions [27,28,96-98] or on other hand some type of mechanism that can correct these distortions [38]. An alternative use of parameterization is to provide surface constraints while locally adaptive optimization procedures modify the mesh and steer it toward some ideal configuration [30-32]. In our implementation we rely on the former approach to resample the surface.

A completely different class of methods used to produce high quality meshes is based on the observation that minimization of the potential energy of physical systems, composed of charged particles or a network of nodes interconnected by springs, tends produce compact descriptions of the underlying object geometry. In these ground energy states particle/node conformations have ideal (hexagonal) or close to ideal packing [33], producing regular Voronoi cells that in turn generate regular triangular tessellation. For example, uniform sampling of a sphere using a system of charged particles is a well-known technique with many practical applications in computer graphics, physical chemistry and numerical analysis [34,35]. Not surprisingly, meshing/remeshing methods described in [33,36] rely precisely on the interaction of charged particle systems to produce high quality triangulations of arbitrary surfaces suitable for FE analysis. By far the most general method in this family is presented in [37] and obtains optimal triangulation by solving a set of 2\textsuperscript{nd} order (Euler-Lagrange) ordinary differential equations:

\[ m_i \frac{d^2 x_i}{dt^2} + b \frac{dx_i}{dt} = \overrightarrow{F}(x_i) \]

where \( x_i \) is the position of the i-th particle, \( m_i \) is the mass of the particle, \( b \) is a damping parameter and \( \overrightarrow{F}(x_i) \) is the net force exerted on the particle due to interaction with its
neighbours and can be defined using arbitrary physical laws (e.g. Hooke's law, Coulomb's law, etc.). The equation above can be solved using any finite difference scheme and according to [37] “results in evenly distributed points that very closely follow the surface features”. Moreover, $\vec{F}(x_i)$ can be modified to include curvature information to obtain adaptive sampling of the surface. The main disadvantage of this method and those presented in [33, 36] is the need to consistently enforce surface constraints, which is a computationally expensive operation.

The algorithm we present herein is a hybrid of the above approaches. It was inspired by the remeshing procedure of Alliez et al. [38] that employs planar parameterization and sampling density maps. Alliez’s resampling procedure is very flexible and as such can produce triangulations of arbitrary complexity with uniform or adaptive vertex configurations for meshes of any topology. Its main drawback is that it requires closed surfaces to be partitioned into a number of patches (to enable 2D parameterization) that must afterwards be reconnected. For instance, in case of closed genus-0 surfaces, the remeshing process would require partitioning of the surface into (at least) two patches.

The algorithm we propose was tailored specifically for closed genus-0 surfaces and was designed to produce multi-resolution mesh representations with high quality. In contrast to Alliez's remeshing procedure [38], our technique relies on a fundamentally different method to sample the parameter domain and uses spherical parameterizations (instead of planar), which do not require partitioning of the surface. Overall, our method is conceptually very simple and unique.

In [39], Gu et al. make an important point that for genus-0 surfaces, spherical embedding based on minimization of the harmonic energy is conformal. The characteristic property
of conformal maps is that they preserve angles [39,40]. For closed genus-0 surfaces that are geometrically dissimilar to a sphere, however, conformal maps produce significant area distortions. Defining area distortion simply as the ratio of areas of the spherical triangle and its planar counterpart, we show how to sample the sphere (i.e. parameter domain) adaptively in a manner that promotes dense aggregation of points in regions of low distortion and sparse distribution in zones of high distortion. This type of adaptive sampling of the sphere is equivalent to approximately uniform sampling of the original surface. We control the complexity of the mesh by initializing the system with a small number of particles to produce a base mesh in the parameter domain. The surfaces of increasing complexity are recovered by subdividing the base mesh with a standard subdivision scheme (triangular quadrisection [41]) and then mapping the result back to the original surface. The method works because subdivision of equilateral triangles produces approximately self-similar triangles at finer scales while conformal mapping ensures that the triangles retain their aspect ratios.

### 2.3 Methods

The purpose of this subsection is to describe in detail the method to resample closed genus-0 surfaces to obtain high quality triangulations of arbitrary complexity. The resampling procedure consists of five consecutive stages (see Figure 2.3) that include: (1) removal of triangles with highly acute angles, (2) calculation of the conformal map of the simplified mesh from step 1, (3) calculation of the distortion field induced by the mapping from step 2, (4) adaptive sampling of the parameter domain guided by the distortion field from step 3, (5) subdivision and minor regularization.
Figure 2.3: Mesh resampling procedure.
2.3.1 Removal of Thin Triangles

As previously mentioned, most meshes used in this study contain large numbers of nearly degenerate triangles which pose a hindrance in obtaining conformal parameterization in an efficient manner. To improve the efficiency of subsequent operations, during the first pre-processing step, we remove a significant portion of triangles with highly acute angles. It should be noted that this operation does not alter the geometry of the surfaces in any significant way because the concerned elements contain little or no geometric importance.

The mesh simplification algorithm used in this study is based on the half-edge collapse scheme described in [44]. Figure 2.4 demonstrates the mechanism of the simplification process.

Figure 2.4: Half-edge collapse.

Note how the deletion of a single edge, collapses two vertices (bounding the edge) into a single vertex, thereby eliminating one vertex, three edges and two faces from the mesh data structure. Since the objective is to remove the faces along the seams of otherwise regularly sampled surface patches, only edges of triangles with very small angles are considered as candidates for collapse. The list of edge candidates is further narrowed by considering only the edges whose opposing angles are below some user defined threshold (e.g. 10 degrees). The two distortion metrics used to ensure that the collapse does not
produce triangle intersections and does not significantly alter the geometry of the mesh, were the so called local tessellation error (LTE) and local geometric error (LGE) [44], respectively. LTE is a measure of the maximum angular deviation of the faces in the local neighbourhood of the collapsing edge before and after edge removal. For example, if LGE exceeds 90°, the edge should not be deleted. The second measure of distortion (LTE) is defined as the maximum spatial displacement of the collapsing vertex relative to its “zone” after edge collapse. In the course of the simplification process, to stop the accumulation of errors during progressive simplification of the mesh, the plane equations of the eliminated faces are added to the zone of the retained edge vertices. The edge deletion is repeated iteratively until the list of edge collapse candidates is exhausted.

### 2.3.2 Conformal Parameterization

Let \( M = [V, E] \) be a triangular surface mesh representing the underlying continuous surface \( S \), where \( V = \{ \mathbf{v}_i \in S \mid i = 1, \ldots, N_v \} \) and \( E = \{ (\mathbf{v}_j, \mathbf{v}_k) \mid m = 1, \ldots, N_E \} \) are sets of vertices and edges, respectively. Parameterization of \( M \) is defined as an isomorphism \( f : S \to S^2 \), where \( S^2 = \{ \mathbf{x}_i \subset \mathbb{R}^3 \mid \|\mathbf{x}_i\| = 1 \} \). We are most interested in conformal maps that by definition minimize the distortion in angles, a property crucial to the method. In [39], Gu et al. make an important point that for genus-0 surfaces harmonic maps are also conformal. A map is harmonic if it is a stationary function of the Dirichlet energy functional [40,42]. For piecewise linear surfaces, the harmonic energy is a quadratic function:

---

3 In this study, we define “zone” around a collapsing vertex as a set of faces its 1-ring neighbourhood.

4 In this study, 0.01 mm was used as a LTE tolerance.
\[ \Phi(f, M) = \frac{1}{2} \sum_{(v_i, v_j) \in E} k_{ij} \| f(v_i) - f(v_j) \|^2 \]  
(2.1)

where \( k_{ij} \) coefficient is termed the edge weight, derived from the opposing angles (\( \alpha \) and \( \beta \)) of the two faces sharing the edge \((v_i, v_j)\) (see Figure 2.5).

\[ k_{ij} = [\cot(\alpha) + \cot(\beta)]/2 \]  
(2.2)

For practical reasons it is more convenient to use a matrix formulation of \( \Phi(f, M) \).

Defining \( X \) as \( N_v \times 3 \) matrix whose rows are vertex coordinates in the parameter space, (2.1) can be rewritten as:

\[ \Phi(f, M) = \frac{1}{2} \sum_{k=1}^{3} (AX_k)^T K AX_k = \frac{1}{2} \sum_{k=1}^{3} X_k^T (A^T K A) X_k = \frac{1}{2} \sum_{k=1}^{3} X_k^T L X_k \]  
(2.3)

where \( X_k \) is the k-th column of \( X \), \( K \) is \( N_e \times N_e \) diagonal matrix of edge weights (2.4), \( A \) is \( N_e \times N_v \) connectivity matrix given by (2.5), and \( L = A^T K A \) is \( N_v \times N_v \) (sparse) symmetric matrix.
Conformal parameterization of closed genus-0 meshes can therefore be obtained by optimizing $\Phi (f, M)$ subject to the surface and zero-mass center constraints [39] (see step 3 of Algorithm 1). One of the most efficient ways to address this constrained quadratic programming problem is to recast it into a dual optimization problem as prescribed by the method of Lagrange multipliers and solve it using Lagrange-Newton (LN) method as was demonstrated in [45]. The LN method uses a second order approximation of the objective function and requires calculation of the Hessian. For large scale optimization problems, such as the one discussed here, the size of the Hessian is prohibitively large (for $10^4$ vertices the memory required to store the Hessian is $\sim 6.7$ GB). In [39], Gu et al. describe a less memory intensive optimization scheme based on the gradient descent. Adopting Gu’s approach, the gradient of the objective function with respect to a single vertex is

$$\nabla_i \Phi = \sum_{j \in D(i)} k_{ij} [f(v_i) - f(v_j)] = \sum_{j \in D(i)} k_{ij} (x_i - x_j) \quad (2.6)$$

and the gradient with respect to all mesh vertices is

$$\nabla_X \Phi = LX \quad (2.7)$$

In equation (2.6), the variable $D(i)$ is the set of vertices that share an edge with the $i$-th vertex. In equation (2.7), $\nabla_X \Phi$ is a $N_v \times 3$ matrix with its rows as vector updates of the vertex positions in the parameter space, therefore $\nabla_X \Phi = [\nabla_i \Phi]$. $\nabla_i \Phi$ can be decomposed into two orthogonal components, one tangential to the surface $\nabla_i \Phi_{\parallel}$ and the other one
normal to the surface $\nabla_i \Phi_{\perp}$ (see Figure 2.6). When updating vertex positions in the parameter space only tangential components are useful and can be computed using (2.8). The algorithm described below provides detailed instructions to obtain conformal parameterization of orientable, closed genus-0 triangular surface meshes.

$$\nabla_i \Phi_{\parallel} = \nabla_i \Phi - (x_i \cdot \nabla_i \Phi)x_i$$ (2.8)

---

**Figure 2.6**: Tangential derivative. **Figure 2.7**: Angle weighted normal. Adapted from [46].

---

**ALGORITHM 1: Conformal Parameterization of Closed Genus-0 Meshes**

1. Begin by computing the Gauss map (GM) of the mesh, $GM : M \rightarrow S^2$. GM is defined as the set of unit normals evaluated at the vertices of the mesh, $M$.

$$GM(\nu) = \bar{N}(\nu), \nu \in V$$ (2.9)

$M$ is a piecewise linear surface and therefore non-differentiable at the edges and vertices. This means that the tangent vectors needed to compute the normals at the vertices are not defined. This complication can be circumvented by estimating the normals at the vertices using an angle weighted approximation of the normal [46,47] given by (2.10) and illustrated in Figure 2.7.

$$\bar{N} = \sum_i \alpha_i \bar{n}_i / \left\| \sum_i \alpha_i \bar{n}_i \right\|$$ (2.10)
2. In this step GM is used to compute a Tutte map\(^5\), TM, which a precursor to the conformal map. The specific steps involved in finding the Tutte map are described in the following sub-algorithm.

**Input:** - Gauss map, \(X_{GM}\)  
**Output:** - Tutte map, \(X_{TM}\)  

Let \(\Phi_{TE}\) be Tutte energy as defined by (2.1) with all \(k_{ij} = 1\).

a. Initialize \(X_{TM} = X_{GM}\)  
b. \(\Phi^{0}_{TE} \leftarrow \Phi_{TE}(X_{TM})\)  
c. Evaluate \(\nabla \Phi_{TE}\) (2.7)  
d. Compute \((\nabla \Phi_{TE})|| (2.8).\) Recall \(\nabla \Phi_{TE} = [\nabla_i \Phi_{TE}].\)  
e. Update \(X_{TM} \leftarrow X_{TM} - \alpha \nabla ||\Phi_{TE}, where \alpha\) is a user specified step-size.  
f. Enforce the surface constraints: \(v \leftarrow v/||v||\)  
g. \(\Phi^{+}_{TE} \leftarrow \Phi_{TE}(X_{TM})\)  
h. Evaluate \(\delta \Phi = 1 - \Phi^{+}_{TE}/\Phi^{0}_{TE}\)  
i. Repeat steps b to h until the maximum number of iterations has been reached OR \(\delta \Phi < \epsilon, where \epsilon\) is a user specified convergence tolerance.

3. Finally, the conformal map can be computed from the Tutte map. The specific steps involved in finding the conformal map are described in the following sub-algorithm.

**Input:** - Tutte map, \(X_{TM}\)  
**Output:** - Conformal map, \(X_{CM}\)  

Let \(\Phi_{HE}\) be the harmonic energy as defined by (2.1).

a. Compute the harmonic weights \(k_{ij}\) using (2.2).  
b. Initialize \(X_{CM} = X_{TM}\)  
c. \(\Phi^{0}_{HE} = \Phi_{HE}(X_{CM})\)  
d. Evaluate \(\nabla \Phi_{HE}\) (2.7)  
e. Compute \((\nabla \Phi_{HE})|| (2.8). Recall \(\nabla \Phi_{HE} = [\nabla_i \Phi_{HE}].\)  
f. Update \(X_{CM} \leftarrow X_{CM} - \alpha \nabla ||\Phi_{HE}, where \alpha\) is a user specified step-size.  
g. Enforce the zero-mass center constraints:  
i. Approximate the mass center, \(C = \sum_i^M f(c_i) \Delta A_i / \sum_i \Delta A_i, where c_i\) and \(\Delta A_i\) is the centroid and the area of the i-th triangle of \(M, respectively.\)  
ii. For all \(v \in V, f(v) \leftarrow f(v) - C\)  
h. Enforce the surface constraints: \(v \leftarrow v/||v||\)  
i. \(\Phi^{+}_{HE} \leftarrow \Phi_{HE}(X_{CM})\)  
j. Evaluate \(\delta \Phi = 1 - \Phi^{+}_{HE}/\Phi^{0}_{HE}\)  
i. Repeat steps c to h until the maximum number of iterations has been reached OR \(\delta \Phi < \epsilon, where \epsilon\) is a user specified convergence tolerance.

\(^5\)Tutte map is also known as barycentric map because it positions mesh vertices at the centroid of the local neighbourhood surrounding each vertex.
2.3.3 Distortion Field

The proposed mesh resampling technique uses conformal maps to establish the connection between the original surface and the parameter domain. Conformal maps do not preserve areas in general and induce large area deformations for surfaces that are dissimilar to a sphere. To maintain uniform sampling of the original surface, mapping induced distortions have to be taken into consideration. For every face in the mesh, define area distortion (AD), as the ratio of the normalized areas of the spherical triangle and its planar image (2.11). Similarly, for every edge in the mesh, define length distortion (LD), as the ratio of edge length in the parameter domain and its length on the original surface (2.14). Next, for every vertex define $A_{D_{V}}$ ($L_{D_{V}}$) as the minimum AD (LD) of the faces (edges) incident on that vertex. The net distortion field (DF) used to guide the distribution of points on the sphere is simply the product of $L_{D_{V}}$ and $A_{D_{V}}$. Finally, the spherical barycentric coordinates can be used to estimate the value of DF at an arbitrary point on the unit sphere.

$$AD_{i} = \frac{Area(T_{i}^{*})}{Area(T_{i})} \sum \frac{Area(T_{i})}{4\pi}$$

$$Area(T_{i}^{*}) = (\alpha_{i} + \beta_{i} + \gamma_{i}) - \pi$$

Figure 2.8: Planar and spherical triangles.
\[
\text{Area}(T_i) = \|(C - B) \times (A - B)\|/2
\]

(2.13)

\[
LD_{ab} = \frac{\cos^{-1}(a \cdot b) \sum |E_j|}{\|A - B\| \sum |e_j|}
\]

(2.14)

Note that the sum in (2.11) represents the total surface area of the mesh in the primary domain. Likewise, the sums, \(\sum|E_j|\) and \(\sum|e_j|\), represent to total edge length in the primary and the parameter domains, respectively.

### 2.3.4 Adaptive Sampling of the Sphere

A system of charged particles provides an efficient means of sampling a surface. We use a slightly generalized version of the electrostatic potential energy, Reisz s-energy (2.15) [48], to describe dynamics of a particle system with unequal positive charges.

\[
R_s = \sum_{i=1}^{N} \sum_{j \neq i}^{N} \frac{q_i q_j}{d_{ij}^s}
\]

(2.15)

where \(N\) is the total number of particles \(q\) is the particle charge (defined as the product of area and length distortion), \(d\) is the geodesic distance between the particles and \(s \in (0,1]\) is a free parameter that modulates the strength of particle interactions. For smooth surfaces such as the bones of the wrist, it was observed that the quality of the resulting triangulation is almost independent of \(s\). In this study, all surfaces were resampled with \(s = 0.5\).

To adaptively distribute the particles over the surface of the sphere, we modify particle charges according to their position. Let \(q_i = q(x_i)\), where \(q : \mathbb{R}^3 \to \mathbb{R}^+\). For a particle at an arbitrary position on the sphere, \(q(x_i)\) can be approximated using linear interpolation:
\[ q(x_i) = uq_1^i + vq_2^i + wq_3^i \] (2.16)

where \( q_k^i \) are the charges defined on the mesh vertices and \( x_i = (x_{i1}, x_{i2}, x_{i3})^T \) is the position vector of the \( i \)-th particle on the surface of the sphere. Given \( \{x_k^i \mid k = 1,2,3\} \) as the vertex coordinates of the spherical triangle containing the particle at \( x_i \), then \( u, v \) and \( w \) are the spherical barycentric coordinates of \( x_i \),

\[ A_i[u \quad v \quad w]^T = x_i, \quad u + v + w \geq 1, \quad u, v, w \geq 0 \] (2.17)

where \( A_i = [x_1^i \quad x_2^i \quad x_3^i] \). We use a ray-triangle intersection algorithm described in [49] to keep track of particle positions relative to the mesh faces. Finally, the derivative of \( R_s \) with respect to \( x_i \) can be approximated as:

\[ \frac{\partial R_s}{\partial x_i} \approx 2 \sum_{j \neq i}^{\Omega_i} \frac{q_j}{d_{ij}^3} \begin{bmatrix} \text{Adj}(A_i)[q_1^i & q_2^i & q_3^i]^T + \frac{q_is}{d_{ij}} \frac{x_i \cdot x_j}{1 - (x_i \cdot x_j)^2} x_j \end{bmatrix} \] (2.18)

\[ \Omega_i = \{ j \mid x_i \cdot x_j > \cos \theta_{\text{max}} \} \] (2.19)

where \( \text{Adj}(A) \) and \( \text{det}(A) \) are the adjoint matrix and determinant of \( A \), respectively. \( \theta_{\text{max}} \) is the angle separation threshold, measured in radians (e.g. \( \theta_{\text{max}} = \pi / 4 \)).

An important attribute of the expression in (2.18) is that the vector \( \frac{\text{Adj}(A_i)}{\text{det}(A_i)}[q_1^i & q_2^i & q_3^i]^T \) does not vary in the course of the optimization as it is intrinsic to the input mesh (in the parameter domain) and the deformation field defined at its vertices. In order to speed-up the optimization, we pre-compute and save these vectors. The overall optimization procedure is summarized below.
ALGORITHM 2: Adaptive Sampling of the Sphere

1. Initialize the system with $N^0_V$ particles randomly distributed across the surface of the sphere.
2. Compute the contribution of the individual particles to (2.15) and sort them in descending order. This establishes the so called priority queue.
3. Update the position of the individual particles in the same order as they appear in the priority queue using (2.20) and then (2.21). The first operation modifies the particle position along the negative gradient of $R_s$, where $\alpha > 0$ is a user defined step size (e.g. $10^{-3}$). The second operation re-projects the particles back onto the surface of the unit sphere.

\[
x_i \leftarrow x_i - \alpha \left[ \frac{\partial R_s}{\partial x_i} - \left( x_i \cdot \frac{\partial R_s}{\partial x_i} \right) x_i \right]
\]

(2.20)

\[
x_i \leftarrow x_i / \| x_i \|
\]

(2.21)

4. Repeat 2 and 3 until convergence.

2.3.5 Subdivision & Regularization

Once the particles have been fixed on the unit sphere, their positions determine the vertices of the base mesh in the parameter domain. Surfaces of increasing complexity can be recovered by iteratively subdividing this mesh with any standard subdivision scheme and mapping the result back to original surface. Triangular quadrisection [41] is the subdivision method used in this study. Figure 2.9 demonstrates one iteration of this procedure. The method works because triangular quadrisection of equilateral triangles produces self-similar triangles at finer scales while conformal mapping ensures that the triangles retain their aspect ratios. Every subdivision quadruples the number of vertices thus resulting in an exponential increase in mesh complexity. After two successive subdivisions, the resulting surfaces may begin to acquire visual artefacts as shown in Figure 2.10. This problem is easily remedied by regularizing the subdivided mesh in the

---

\[6\] In this study, triangulations were obtained using MATLAB's `convhulln` routine.
parameter domain just prior to mapping it back onto the original surface. Regularization is achieved by relaxing the mesh with a few iterations of barycentric mapping (a.k.a. Tutte mapping) described in subsection 2.3.2. The step size (e.g. $10^3$) and the number of iterations of barycentric mapping (e.g. 40) are two free parameters of the regularization procedure.

![Figure 2.9: One iteration of triangular quadrisection for a planar triangle. New vertices are inserted at edge midpoints. Note that in case of spherical triangles, the newly inserted vertices must be re-projected to the surface of the sphere (i.e. $x \leftarrow x/\|x\|$).]

![Figure 2.10: The effect of regularization. Left - mesh without regularization. Right - mesh after regularization. The thick black lines correspond to the edges of the base mesh.]

### 2.3.6 Controlling Mesh Complexity

One of the main attributes of the mesh resampling approach developed is this study is its intrinsic multiresolution construction, a property that can be exploited to improve computational efficiency of various mesh processing operations. For instance, in Chapter
we rely on the distance transforms (DT) to calculate optimal similarity transformations that we subsequently employ to align multiple surfaces into a common pose. Computing DTs requires approximately $MN$ operations where $M$ is the number of grid points in the domain of DF and $N$ is the number of mesh vertices (see Appendix D). Determining DT of high resolution for meshes of full complexity can be very time consuming, but performing the same operation for meshes of lower complexity can significantly speed-up the process. Ideally, it is desirable to have an expression that relates the number of vertices of the base mesh ($N_v^0$) and the final number of mesh vertices ($N_v^n$) after a certain number of subdivisions ($n$). Such a relationship also would be useful, because the dimensionality of the optimization problem concerned with adaptive sampling of the sphere could be regulated almost precisely. Indeed, as shown in Appendix C, it is possible to derive such an expression by considering the changes in the numbers of vertices ($N_v$), faces ($N_f$) and edges ($N_e$) between two consecutive subdivisions as well as the Euler-Poincaré formula that relates $N_v$, $N_f$, $N_e$ and the genus of the surface [21]. For meshes whose vertex distribution minimize the Reisz s-energy, the relationship can be approximated simply as $N_v^n \approx 4^n N_v^0$.

2.3.7 Assessing Mesh Quality

With the resampling procedure in place it is desirable to know how much it improves the quality of the mesh. The definition of mesh quality used in this study is based on the minimum and mean triangle aspect ratios (2.22). By this standard, the equilateral triangles are the most desirable and have $\eta = 1$. 
\[
\eta = 2 \frac{\text{inscribed radius}}{\text{circumscribed radius}}
\]  

(2.22)

There are number of alternative measures of quality and the interested reader is referred to [20] for more discussion on this subject. The reason aspect ratio was chosen as a benchmark of quality was because the stability of FE simulations (for which these models are ultimately designed) largely depend on this specific measure [20].

\section{2.4 Results}

To demonstrate the effectiveness of our algorithm in generating high quality triangulations, we used it to resample 239 meshes\textsuperscript{7} of the carpal bones of the right hand [16] as well as a small number of objects that were downloaded from open-access databases [18,19]. Figures 2.11 to 2.18 show side by side comparisons of 8 different anatomical objects (scaphoid, lunate, trapezium, hamate, molar, proximal and distal ends of the femur, proximal end of tibia) before and after resampling. Table 2.1 provides a corresponding summary of the changes in mesh complexity, quality and approximation errors as measured by the discrepancy of surface area and volume. As described in subsection 2.3.7, mesh quality is assessed in terms of minimum and mean triangle aspect ratios (\(\eta\)) (2.22). By this definition, the equilateral triangles are the most desirable and have \(\eta = 1\).

It should be noted that although the wrist is composed of eight bones, only the four bones shown in Figures 2.11-2.14 had sufficiently elaborate geometry to convincingly

\textsuperscript{7} The wrist database reported in [16] was based on the study of 30 subjects. Because the wrist contains 8 bones, there were supposed to be a total of 240 bone meshes. For an unknown reason, a trapezoid from one of the subjects was not included in the database.
demonstrate the effectiveness of our resampling procedure. The results obtained for these meshes are entirely representative of the results for the other 235 carpal bones used in this study (that obviously included pisiform, triquetrum, trapezoid and capitate). The remaining test objects shown in Figures 2.15-2.18, were also chosen because of their intricate geometry.

All of the mesh processing operations described in this chapter were performed in a MATLAB environment running on a desktop with 6 GB RAM and 2.80 GHz x6 AMD processor. Typical run time for an average mesh consisting of 13500 vertices and for the base mesh of 500 vertices was 19.5 min, with conformal parameterizations accounting for approximately 60% of the total execution time. Although at first this performance may appear inefficient, it should be noted that implementations of iterative algorithms such as the one presented here, are inherently much slower in MATLAB than in 3rd generation programming languages such as C++ [101].

<table>
<thead>
<tr>
<th>Object</th>
<th># verts</th>
<th># faces</th>
<th>( \eta_{\text{ave}} )</th>
<th>( \eta_{\text{min}} )</th>
<th>AE (%)</th>
<th>VE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scaphoid O</td>
<td>31068</td>
<td>15936</td>
<td>0.6218</td>
<td>0.0046</td>
<td>-0.1904</td>
<td>-0.1022</td>
</tr>
<tr>
<td>R 15536</td>
<td>7970</td>
<td>0.9914</td>
<td>0.8831</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lunate O</td>
<td>34822</td>
<td>17413</td>
<td>0.6473</td>
<td>0.0058</td>
<td>-0.1820</td>
<td>-0.0967</td>
</tr>
<tr>
<td>R 15936</td>
<td>7970</td>
<td>0.9935</td>
<td>0.8599</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trapezium O</td>
<td>32820</td>
<td>16412</td>
<td>0.6635</td>
<td>0.0008</td>
<td>-0.2002</td>
<td>-0.0999</td>
</tr>
<tr>
<td>R 15936</td>
<td>7970</td>
<td>0.9937</td>
<td>0.8860</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hamate O</td>
<td>39548</td>
<td>19776</td>
<td>0.5798</td>
<td>0.0017</td>
<td>-0.2764</td>
<td>-0.1160</td>
</tr>
<tr>
<td>R 15936</td>
<td>7970</td>
<td>0.9865</td>
<td>0.7974</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proximal Femur O</td>
<td>15002</td>
<td>30000</td>
<td>0.8941</td>
<td>0.2678</td>
<td>-0.9194</td>
<td>-0.0788</td>
</tr>
<tr>
<td>R 14370</td>
<td>28736</td>
<td>0.9897</td>
<td>0.7776</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distal Femur O</td>
<td>38752</td>
<td>19378</td>
<td>0.8679</td>
<td>0.0563</td>
<td>-0.3567</td>
<td>-0.0635</td>
</tr>
<tr>
<td>R 35136</td>
<td>17570</td>
<td>0.9908</td>
<td>0.7273</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proximal Tibia O</td>
<td>36164</td>
<td>18084</td>
<td>0.8781</td>
<td>0.0887</td>
<td>-0.3693</td>
<td>-0.0577</td>
</tr>
<tr>
<td>R 31936</td>
<td>15970</td>
<td>0.9889</td>
<td>0.7630</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Molar O</td>
<td>21900</td>
<td>43796</td>
<td>0.8685</td>
<td>0.2540</td>
<td>-0.3674</td>
<td>-0.0720</td>
</tr>
<tr>
<td>R 15970</td>
<td>31936</td>
<td>0.9890</td>
<td>0.8351</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1: Mesh complexity, quality and reconstruction errors before and after resampling for objects shown in Figures 2.11-2.18. ‘O’ stands for original and ‘R’ for resampled. \( \text{AE} = \text{surface area error} = (A_e/A_o - 1)\times100 \). \( \text{VE} = \text{volume error} = (V_e/V_o - 1)\times100 \). \( \eta = \text{triangle aspect ratio} \).
Figure 2.11: Original and resampled meshes of scaphoid. Two views are shown. Base mesh of the resampled object contained 500 vertices. The final mesh was obtained by subdividing the base mesh two times.
Figure 2.12: Original and resampled meshes of lunate. Two views are shown. Base mesh of the resampled object contained 500 vertices. The final mesh was obtained by subdividing the base mesh two times.
Figure 2.13: Original and resampled meshes of trapezium. Two views are shown. Base mesh of the resampled object contained 500 vertices. The final mesh was obtained by subdividing the base mesh two times.
Figure 2.14: Original and resampled meshes of hamate. Two views are shown. Base mesh of the resampled object contained 500 vertices. The final mesh was obtained by subdividing the base mesh two times.
Figure 2.15: Original and resampled meshes of proximal end of femur. Base mesh of the resampled object contained 900 vertices. The final mesh was obtained by subdividing the base mesh two times. (b-e) are the close-ups of the rectangular contours in (a).
Figure 2.16: Original and resampled meshes of distal end of femur. Two views are shown. Base mesh of the resampled object contained 1100 vertices. The final mesh was obtained by subdividing the base mesh two times.
Figure 2.17: Original and resampled meshes of proximal end of tibia. Base mesh of the resampled object contained 1000 vertices. The final mesh was obtained by subdividing the base mesh two times. (b-e) are the close-ups of the rectangular contours in (a).
Figure 2.18: Original and resampled meshes of a molar. Base mesh of the resampled object contained 1000 vertices. The final mesh was obtained by subdividing the base mesh two times. (b-e) are the close-ups of the rectangular contours in (a).
2.5 Discussion

The results summarized in Table 2.1 show a dramatic improvement in mesh quality at the cost of minor errors in volume and surface area (except for the proximal femur, errors for all other objects are less than 0.40%). For the four carpal bones shown in Fig 2.11-2.14, the reduction in mesh complexity was greater than 50% (on average) while the surface area and volume errors were less than 0.28% and 0.12%, respectively. In fact, these errors remained below 0.6% when the complexity was reduced to 25% of the original. The results shown for the remaining anatomical objects are also noteworthy because all of their intricate features were fully retained after resampling.

The surface area and volume errors mentioned above, however, are global measures of similarity. As such, they do not provide any indication of local structural changes made during resampling and therefore, are not the best indicators of the trade-off between mesh complexity and preservation of visually salient features. In the present study, the acceptable sampling density\(^8\) used to reconstruct the surface was determined using a visual comparison of the resampled models relative to the original appearance of the mesh. Although the human visual system is quite reliable in detecting perceptually acute discrepancies in geometric form, it is still very subjective. For this reason, future work should establish similarity measures (between the original and resampled surfaces) that can be used to guide the selection of optimal sampling density. This last point directly relates to the major limitation of our resampling procedure, which in its present form only allows the generation of uniform triangulations. In instances where objects have

---

\(^8\)Average sampling density is defined as the total number of mesh vertices per surface area of the entire object.
extensive surface patches with low (Gaussian) curvature, uniform meshes are not necessarily ideal as more compact representations may be obtained with adaptive triangulations whose local vertex density is a function of curvature [63-66].

Another limitation of our remeshing procedure is that it cannot accurately resample meshes that have artificial features or whose conformal parameterizations produce locally “large”\(^9\) area distortions. Examples for both of these cases are illustrated in Figures 2.19 and 2.20.

\[\text{Figure 2.19: One of the limitations of our remeshing algorithm in its present form is that it is unable to preserve salient features of artificial shapes such as the one shown above. Sharp edges present in the original mesh (left) are not fully retained in the resampled mesh (right).}\]

\(^9\) As an example see the distortion map shown in Figure 2.21. At the tips of the middle and ring fingers the normalized area of some triangles was reduced by more than \(10^6\) times! This example is highly uncharacteristic of the carpal bone meshes used in this study whose parameterizations had much lower distortions.
Figure 2.20: The digits of the hand could not be resampled due to large distortions produced at the fingertips in the parameter domain. The corresponding area distortion map is shown in Figure 2.21. The resampled mesh was obtained by subdividing the base mesh of 700 vertices two times.

One should note that in the context of the present study, which deals exclusively with relatively smooth natural surfaces, these deficiencies are not problematic. However, since we aim to present a general resampling strategy, these shortcomings will have to be addressed in the future. For now we will only outline a tentative course of action that should resolve these concerns.

The limitation related to resampling of meshes with artificial features is closely related to the aforementioned issue of sampling density and surface similarity measures. Strong salient features, such as the sharp edges shown in Figure 2.19, are characterized by large curvatures. In remeshing algorithms based on particle systems, these features are usually
preserved using the so called “sizing field” [63,91] (this is the same principle that is already employed in our method where we use distortion field to produce adaptive particle distributions in the parameter domain). When these fields are combined with adaptive sampling strategies they tend to ensure preservation of important features. In theory, it appears that the essential mechanism to produce triangulations, which preserve artificial features, is already in place. What remains to be done is to identify a means of combining the distortion field already in use with curvature information (e.g. Gaussian or mean curvature) to compute optimal sampling density maps and then verify if the current adaptive sampling strategy (that relies on Reisz s-energy and linear interpolation) could use these maps to produce high quality, adaptive triangulations.

The means to resolve the undersampling that occurs in the presence of large distortions in the parameter domain are less obvious because the underlying cause of this problem is less clear. One possible explanation is that the observed undersampling of regions, such as the fingertips shown in Figure 2.20, is a consequence of poor initialization of vertex positions (which are random) in the parameter domain. Due to the nonlinear nature of the optimization problem, the presently used gradient descent scheme likely produces only a locally optimal solution. Assuming this hypothesis is valid and the sought configuration of mesh vertices is a minimizer of Reisz s-energy\(^ {10}\), then it should be possible to rectify the undersampling problem by optimizing the conformation of the (base mesh) vertices using consecutively better initializations. The procedure by which this can be accomplished is summarized in Algorithm 3.

\(^{10}\) In other words it is assumed that the correct solution is embedded in the energy function as a minimum. This assumption is reasonable because it is supported by experiments on models that did not have features with large deformations in the parameter domain (for example see Figures 2.11-2.18).
Figure 2.21: Logarithm of area distortion field superimposed on the original mesh (top left) and conformal parameterization (top right). Note how the digits of the hand are compressed to very small regions in the parameter domain. Area distortion (as defined by equation (2.11) in subsection 2.3.3) at the distal ends of middle and ring fingers is $>10^{-6}$.
ALGORITHM 3: Reducing Undersampling of Features With Large Distortions in the Parameter Domain

1. Adaptively sample the parameter domain using standard procedure.
2. Map the base mesh to the primary domain and calculate triangle areas.
3. Search for triangles that have disproportionately large areas with respect to their neighbours. As can be seen in Figure 2.20 these triangles are likely to occur in undersampled regions.
4. Suppose there are N triangles that were found in step 3. Calculate the centroid of each triangle in the parameter domain. This will add N vertices to the base mesh. To maintain the same number of vertices, delete N vertices whose 1-ring neighbourhoods have the smallest areas.
5. Use vertex positions of modified base mesh from step 4 as initialization in step 1.
6. Repeat steps 1 to 5 until all triangle areas fall within some acceptable range.

Another issue worth mentioning is the complexity of our resampling procedure. In the Results section it was pointed out that typical run time for an average mesh consisting of 13500 vertices and for the base mesh of 500 vertices was 19.5 min, with conformal parameterizations accounting for approximately 60% of the total execution time. With respect to other mesh resampling methods based on conformal parameterizations [29,38,100] that can process surfaces of moderate complexity at nearly interactive rates (< 1 sec), our procedure may appear highly inefficient. At this point, however, it is premature to make this conclusion for two reasons. First of all, we wish to note that implementations of iterative algorithms such as the one presented here, are inherently much slower in MATLAB than in 3rd generation programming languages such as C++ [101]. This suggests that conformal parameterizations of moderately complex meshes, which accounted for 60% of the total run time, can in fact be computed in under a second [29,38,100]. Secondly, in addition to possible gains that can be made due to implementation on other platforms, there is also a potential for parallelization of the adaptive sampling procedure. The most expensive operation involved in optimizing the
state of the charged particle system is the tracking of individual particle positions with respect the mesh faces. At the present moment these positions are modified sequentially using a Gauss-Seidel update scheme. Alternatively, all particle positions can be modified simultaneously with a Jacobi scheme and the look-up process can be parallelized on a GPU. While it is well known that Jacobi updates generally result in slower convergence rates than Gauss-Seidel updates [102], it is nonetheless possible that this difference can be offset by a parallel implementation of the currently used ray-triangle intersection (i.e. particle tracking) algorithm [49] on a GPU.

The present formulation of our algorithm was first and foremost motivated by the need to resample closed manifold meshes which are topologically equivalent to a sphere. Although such shapes are very common and carpal bones of the wrist are prime examples, at the present moment our algorithm is not suitable for processing objects of any other topology. From theoretical point of view, however, our procedure is generalizable to surface with boundaries and higher genera because at its foundation it relies only on conformal parameterizations and geodesic distances in the parameter space. Fortunately, some methods of finding conformal maps of high genus zero surfaces rely on planar parameterizations\textsuperscript{11} [38], which means that the inter-particle distances can be evaluated using a simple Euclidean metric. The boundaries of these maps can be further parameterized using 1D splines so the sampling can be first performed along the boundaries and then in the interior of the parameter domain.

On a final note we would like to point out one interesting attribute of our resampling procedure that could establish a novel way of computing area and length preserving

\textsuperscript{11} Such parameterizations are possible by making cuts in the surface and then flattening it.
parameterizations (ALPP) of closed genus-0 surfaces\textsuperscript{12}. Aside from remeshing, one important application of such parameterizations is calculation of spherical harmonics [103] commonly used in morphometric analysis\textsuperscript{13} [82,104]. ALPPSs generally involve optimization of non-convex cost functions which are substantially more computationally expensive to minimize [97-99] in comparison to the harmonic energy (which is quadratic and in case of all positive harmonic weights, guaranteed to have a global minimum\textsuperscript{14}). In contrast to the traditional approach, we hypothesize that it is possible to obtain these parameterizations without explicit optimization of area and length/angle preservation cost functions, based on the following observations:

a. Surfaces resampled with our procedure have approximately uniform triangulations with most triangles having nearly ideal aspect ratios (i.e. $\eta \approx 1$).

b. Minimization of Reisz $s$-energy for a system of particles with equal charges can produce uniform triangulations in the parameter domain with most triangles having nearly ideal aspect ratios (see Appendix B).

Taken together these observations suggest that it may be possible to compute ALPPs for resampled meshes without explicitly optimizing for area and length/angle preservation criteria. Specifically there are two possible alternatives, both consisting of two stages. In the first scenario, the standard resampling procedure described in this chapter would be performed first, followed by the deformation of the mesh in the parameter domain using constrained relaxation of vertex positions. In the last step, the vertex positions of the

\textsuperscript{12} Note, ALPPs are not isometric. Isometric parameterizations require preservation of Gaussian curvature at every point on the surface and therefore do not exist for general types of surfaces [40] (see Appendix A).

\textsuperscript{13} Spherical harmonics are very useful in quantifying shape differences and classifying shapes. For instance they can be used to analyze structural changes that are a cause or a consequence of pathology [105].

\textsuperscript{14} If all $k_{ij} > 0$, matrix $L$ in (2.3) is positive definite [39], hence $\Phi(f,M)$ is convex and must have a global minimum [109].
parameter mesh would be modified to minimize Reisz s-energy (with all particle charges set a constant value) while preserving the orientability of the faces with respect to the original connectivity of the mesh. In the other scenario, the two steps would be reversed so that the uniform triangulation of the sphere would be obtained first and then, while it is embedded in the distortion field (produced by conformal mapping), it would be deformed along the negative gradient of the Reisz s-energy functional. In the course of optimization, similarly to the former approach, one would also have to ensure that the mesh does not develop any foldovers. From computational point of view, however, the first scenario would be easier to implement because there would be no need to keep track of particle positions with respect to the conformal parameterization during the deformation stage.

2.6 Conclusion

In summary, we have presented a novel and unique remeshing technique that unifies conformal parameterizations with the principles governing the dynamics of charged particle systems, to enable recovery of high quality triangulations suitable for FE simulations. The performance of the proposed approach was tested on a number of anatomical objects and preliminary results indicate that the method does indeed greatly improve the quality of the input triangulation at the cost of minor decrease (<1%) in surface area and volume. The resulting meshes have the so called subdivision connectivity, a property that can be further exploited in compression as well as in mesh processing applications that use hierarchal mesh representations. Additionally, our method provides a novel mechanism of computing area and length preserving
parameterizations of closed surfaces without explicit optimization of area and length/angle preservations cost functions.

In the present form, however, the method contains a number of limitations that include: (a) absence of robust measures to assess the reconstruction accuracy, (b) absence of a mechanism to control local sampling density, (c) undersampling of features with large area distortions in the parameter domain and, (d) restriction to closed genus-0 meshes. The tentative means to resolve these issues have been suggested and will be the subject of future work along with implementation on a different platform.
Chapter 3

Statistical Shape Models of Carpal Bones

All able-bodied humans interact with the physical world through a seemingly ordinary function of their hands, which is enabled by the continuous range of motion of the wrist. Occasionally, some individuals sustain an injury to the proximal end of their upper extremity that diminishes their ability to perform their job and in severe cases prevents them from executing routine tasks of everyday life. Keenly aware of these problems, researchers in the medical community are constantly in search for new treatment methods and therapies to restore the normal function to patients with debilitating wrist injuries. Investigations of this sort can often benefit from finite element (FE) analysis, a tool that has the potential to provide invaluable insight into previously intractable problems.

A brief literature survey will reveal a number of FE studies of the hand and wrist [5-15]. FE analyses of these anatomical structures are motivated by various research objectives that include the investigation of load transmissions through bones and joints, evaluation of contact analysis at the joints, study of joint injury mechanisms, assessment of feasibility of joint replacements and design of prosthetic/fixation devices [15]. For example in [8], [9] and [10] the effectiveness of a specific surgical intervention was assessed on the treatment of Kienbocks disease, carpal tunnel syndrome and scaphoid non-union with osteoarthritis, respectively. Out of these studies the model used in [6] was the most anatomically complete and included approximations of joint contact areas, ligaments, complete representations for all metacarpal and carpal bones as well as
substantial proximal extensions of the radius and ulna. Despite the significant contributions of these studies, they all relied on models derived from a single individual. Therefore, the results derived from them cannot be readily generalized to a larger population set.

Three notable exceptions to the conventional hand and wrist model construction approaches mentioned above were described in [50-52]. In these studies, the models were built using the same population-based principles adopted in the current work. In contrast to [50-52], however, we implement a more robust method for statistical shape model (SSM) building that produces higher quality models\textsuperscript{15}. Furthermore, unlike [50-52] that generate a set of SSMs of the individual bones, we develop a multi-body SSM of the entire wrist (see Chapter 4) that couples the geometry of the individual carpal bones. This is the first reported instance of such a wrist model. Finally, we would like to note that in addition to [50-52], a small number of biomechanical studies incorporating the SSM method have also been reported by other researchers, but were restricted to the knee joint [53], femur [54-56] and clavicle [57].

Based on extensive review by Heimann and Meinzer [58], it appears that SSMs have been used to model all kinds of anatomical objects for many years in the medical image processing community. However, the primary application of these models was directed at the task of unsupervised segmentation of medical images. Within the domain of biomechanics, the statistical shape modeling framework provides numerous advantages over traditional, subject-specific models, not the least of which are parametric

\textsuperscript{15} SSM quality strongly depends on the methods used to establish dense pointwise correspondence across a set of training shapes [60]. In Section 3.1.3 it is explained why distance based similarity measures, as used in [50-52], are inappropriate for creating compact SSMs with good specificity and generalization properties.
representation and groupwise generalization ability. The power of SSMs stems from their generative ability, a trait that makes SSMs exceptionally useful in machine vision and image processing tasks that often rely on interpretation of previously unseen data. Furthermore, the basic mathematical formulation of SSMs is very appealing from an intuitive point of view because within the SSM framework the notions of “mean shape” and “principal variations of shape” are explicitly defined. In the context of biomechanical modeling and biomedical applications where one seeks to elucidate a relationship between the physical structure and anatomical function (e.g. understanding the structural changes in anatomy at various stages of growth or disease [59]) such an intuitive and parametric description is ideal.

Development of a multi-body SSM of the wrist is a relatively involved process. For the sake of clarity, the problem was subdivided into two parts. The first is addressed in this chapter and deals with construction of SSMs of the individual wrist bones. The second part is dealt with in Chapter 4 and is concerned with the issue of combining the individual carpal SSMs into a composite SSM of the entire wrist. The remainder of this chapter is organized as follows. Section 2 describes the mathematics of SSMs with particular attention given to the most challenging aspect of SSM construction known as the point correspondence problem. In Section 3, we describe a principled and robust three step procedure for constructing SSMs of the individual carpal bones. The results and discussion pertaining to the process of SSM construction as well as the quality of the final models are provided in Section 4. Section 5 concludes this chapter with an overview of the major findings.
3.1 Statistical Shape Models Background

The basic construct of statistical shape models (SSMs) is based on an idea that any shape can be represented as a point in a high-dimensional vector space. By extending this notion to an ensemble of shapes, it becomes possible to study the distribution of shapes in the so-called shape space to uncover patterns of geometric variations and spatial relationships between the structures of interest. In essence, SSMs are compact, mathematical descriptions of objects that provide efficient parameterization of their variability. This section is meant to provide an overview of the concepts and methods related to the SSM construction.

3.1.1 Shape Representation

One of the simplest and perhaps most popular ways of representing a shape, $S_i \subset \mathbb{R}^3$, is by a set of points sampled from its surface, $V_i = \{\mathbf{v}_j \in S_i \mid j = 1, 2, \ldots, N\}$. Triangulation of $S_i$ defines a mesh which provides information required for the reconstruction of the underlying surface in-between the points and calculation of the normal vectors. In the context of SSM this type of representation is termed the point distribution model (PDM). Alternate shape representations include but are not limited to: medial axes, spherical harmonics, wavelets, non-uniform rational B-splines (NURBS) and signed distance fields. The main distinction between the PDM and all other representations is that it is explicit, which means that its parameters can be manipulated locally in a manner that is independent of shape geometry [60], whereas all other descriptions are implicit and are a direct function of shape geometry. Although it is arguable which type of representation is superior, the merits of each depend on the specific application. In light of the fact that
PDM is the most intuitive and the most widely used shape representation, it is then natural for the SSM to assume the same format. Moreover, all the previously mentioned representations can be recovered from the PDMs.

Provided that an object of interest has already been segmented from the medical image, the PDM can be obtained in a number of ways. A typical approach is to use a marching cubes algorithm [61] to tessellate the surface. Triangulations produced in this manner however, often have an unreasonably large number of elements [62,63] that may lead to extensive processing times down the road. A more recent technique is to sample the object using a system of charged particles constrained to its surface [63-66]. Within this formulation, the dynamics of particle interactions can be modified according to the local features to promote uniform or adaptive sampling of the surface. The major benefit of this approach is that the resulting triangulation is not only a compact and accurate description of the object but also has good quality (see Chapter 2, subsection 2.3.7).

The data used in the present study already came in the form of triangular meshes[16]. Unfortunately most of the meshes were not readily suitable for use during the model building stages due to a large number of nearly degenerate triangles that corrupted the quality of the mesh and increased its complexity by a considerable amount. As discussed in Chapter 2, one of the contributions of this study is a novel method for resampling closed genus-0 surface meshes to produce high quality triangulations. In fact, the principle at the core of our method was based on the dynamics of a charged particle

[16] Refer to [16] for more information regarding specific methods that were used to generate the triangular meshes from the medical images.
system, thus demonstrating the wide applicability of this approach to the problems of surface representation.

### 3.1.2 Statistical Analysis

PDMs provide a foundation on which statistical relationships between different parameters describing the morphology of the object can be quantified. The space of geometric variations of an object is learned from a training set of sample PDMs, \( T = \{ \mathbf{V}_m \mid m = 1, 2, ..., M \} \), using principal component analysis (PCA) \([60]\). To derive meaningful and accurate information regarding the shape variability, the correspondence between all points in \( T \) must be established first. This issue is addressed in the following subsection. Assuming for a moment this requirement has been satisfied, the shapes must thence be aligned to a common coordinate system. This operation is accomplished by normalizing the PDMs with respect to the similarity transformation group composed of global scaling, rotation and translation parameters. Typically this problem is solved using the so called Generalized Procrustes Analysis \([60]\) that registers the shapes to an unbiased coordinate system by iteratively minimizing the pairwise least squares difference between \( \mathbf{V}_m \) and an average PDM. Once the similarity transformations have been removed, the SSM can be easily constructed.

Let \( \mathbf{V}_m \) be a shape vector composed of an ordered list of PDM vertices (normalized with respect to the similarity transformation group) such that \( \mathbf{V}_m = (v_1, v_2, v_3, ..., v_N)^T \) where \( v_j = (x_j, y_j, z_j) \) and \( N \) is the total number of vertices. Then the shape covariance matrix can be defined as follows:

\[
\mathbf{C} = \frac{1}{M-1} \sum_{m=1}^{M} (\mathbf{V}_m - \overline{\mathbf{V}})(\mathbf{V}_m - \overline{\mathbf{V}})^T
\]  

\[ (3.1) \]
A linear SSM is simply a weighted sum of the mean shape vector ($\bar{V}$) and the first $k$ most important eigenvectors (a.k.a. eigenmodes) of $C$,

$$V = \bar{V} + \sum_{m=1}^{M} b_m p_m , \quad 1 \leq k \leq M - 1$$

(3.3)

where \{p_m\} are the principal modes of shape variation$^{17}$ and \{b_m \in \mathbb{R}\} are the shape parameters used to control the geometry of the object by modifying the contribution of the individual eigenmodes. $p_m$ can be obtained directly by performing eigen-decomposition of $C$, a procedure that requires solving the linear system $$(C - \lambda_m I)p_m = 0,$$

where $\lambda_m$ is the eigenvalue of the m-th mode. The amount of variance captured by any single eigenmode is equal to $\lambda$. For ill-conditioned $C$, singular value decomposition (SVD) provides a more efficient and numerically stable means for computing $p_m$ [67].

The major attribute of the SVD based approach is that it does not require explicit calculation of the covariance matrix of training shape vectors but instead infers the principal shape components and corresponding eigenvalues from the left orthogonal and diagonal matrices of $S = (L - \bar{L})/\sqrt{M-1}$, respectively, where $L$ is a $3N \times M$ matrix whose column entries are the training shape vectors and $\bar{L}$ is a matrix with all columns set to $\bar{V}$, the mean shape vector. The shape matrix $S$ can be decomposed into a matrix product $S = U D W^T$ where $D$ is a $M \times M$ diagonal matrix of singular values, while $U$ and $W$ are left and right column orthogonal $3N \times M$ and $M \times M$ matrices, respectively. The

$^{17}$ It is assumed that the eigenvectors \{p_m\} are sorted in the decreasing magnitude of energy of the corresponding eigenvalue, so that $\lambda_m > \lambda_{m+1}$. Note that because $C$ is symmetric, positive definite matrix, $\lambda_m \geq 0$. 

55
columns of \( U \) are the eigenvectors of \( C = SS^T \) and the corresponding eigenvalues are the diagonal entries of \( D \) squared \( (\lambda_m = D_{mm}^2) \).

Having established the means to perform PCA on the training set of shape vectors, the SSM is essentially complete. The only remaining question is how to constrain the shape parameters, \( \{b_m\} \), so that the model produces valid instances of the object? The main assumption at the core of linear SSMs is that shape vectors have a Gaussian distribution in the shape space. Consequently, the individual shape parameters may restricted to the range \( -3\sqrt{\lambda_m} \leq b_m \leq 3\sqrt{\lambda_m} \). In cases where the distribution of shape parameter vectors from the training data is non-Gaussian, the simple “min-max” interval constraint may produce invalid instances of the shape. In those situations more robust procedures have to be used to determine feasible shape parameter values. The four common approaches are Gaussian mixture modeling, kernel density estimations, kernel PCA and independent component analysis [60,68]. In this study, we consider only linear SSMs and verify the validity if this approach in the Results section.

3.1.3 Correspondence

In the previous subsection it was assumed that all of the points comprising the PDMs of the training shapes have accurate correspondence, a condition which implies that all PDM landmarks\(^\text{18}\) are located in the semantically equivalent anatomical positions. This requirement is the most challenging aspect of 3D model construction and ultimately

---

\(^{18}\) In [69] landmarks are defined as “points situated on the surface of the object for which objectively meaningful and reproducible biological counterparts exist in all other objects of a data set”. In practice, however, the number of points that meet this requirement is usually too sparse to accurately reconstruct the surface. For this reason we drop the objectivity constraint and instead consider pseudo-landmarks that can be defined as “points situated on the surface of the object for which reproducible biological counterparts exist in all other objects of a data set”.

56
determines the quality (a property that will be defined shortly) and the accuracy of the resulting model.

Consider for a moment a set of identical shapes that have been sampled with an equal number of points but the positions of the individual points differ by some amount across the set. It is obvious that although in reality there are no geometric variations to begin with, the inconsistent sampling procedure will inadvertently introduce apparent variations across the set. As pointed out in the previous subsection, we assume that linear SSMs could produce valid approximations of observed shape variations, however, according to [68] the utility of such models depends strongly on the accuracy of the correspondence established between the members of the training set. According to [60] poor correspondence can result in the need for a large set of modes of variation (and corresponding shape parameters) to approximate the training shapes to a given accuracy, and may lead to 'legal' values of \( \beta_i \) generating 'illegal' shape instances. This fact clearly necessitates a robust correspondence methodology.

The common methods used to establish point correspondence can be organized according to the underlying mechanics of the registration procedure. According to [58], most registration techniques can be classified into either one or a combination of the five categories\(^\text{19}\) that include:

- Mesh-to-mesh registration
- Mesh-to-volume registration
- Volume-to-volume registration

\(^{19}\)Refer to [58] for additional description of these methods and discussion of related work.
- Parameterization-to-parameterization registration
- Population based optimization

The fundamental problem associated with the first four approaches is the fact that reliable correspondence can be obtained only through identification of features with known or at least assumed correspondences. One way of addressing this problem is to use manual annotation to guide the registration process. Unfortunately, although user-dependent landmark selection often leads to acceptable results, it is an extremely time-consuming and subjective process that lacks reproducibility, and cannot be guaranteed to produce good models [60,68]. In 3D, these shortcomings are significantly more detrimental. Considering the fact that this study used a total of 239 carpal bones, the majority of which did not have strong salient features (as an example see Figure 3.1), manual landmarking was immediately discounted as an impractical approach for selecting dense correspondences.

In light of the complexity of the task at hand, the automatic correspondence techniques were clearly preferred. Such methods are typically formulated as optimization problems that seek point correspondences that minimize some type of objective measure. Most commonly used objective functions are defined in terms of pairwise distance-based proximity measures between presumed landmarks. In reality, however, the relative position of homologous (i.e. corresponding) points may vary substantially over the training set, thus rendering pairwise distance-based comparisons an unsatisfactory basis for establishing correspondence [58,60,68,70-72] (see Figure 3.2). In fact, this is the
primary reason why the previously reported SSM models of the carpal bones [50-52] may not be adequate descriptions of these biological shapes\textsuperscript{20}.

\textbf{Figure 3.1}: Eighteen samples of capitate, demonstrating the degree of anatomical variability and absence of visually consistent salient features necessary to establish dense correspondence.

\textsuperscript{20} The studies in [50-52] used a variant of iterative closest point algorithm [73] to establish correspondence.
Figure 3.2: Illustration of why pairwise proximity based measures cannot guarantee correct correspondence. The distance-based correspondences between the landmarks on two surface profiles are shown by the black (solid) lines while the correct correspondences are shown by the green (dashed) lines. These types of incorrect matches are more likely to occur if there is a significant amount of local anatomical variation across the training set.

Alternative and also common measures of correspondence are based on surface deformation energies. Although quite appealing from an intuitive point of view, such measures are problematic for a number of reasons, not the least of which is the fact the observed variations of form do not arise as a result of physical deformation, but as consequence of a whole process of biological development [60]. Another major problem with this approach is that it tends to penalize large or non-uniform deformations, but it may precisely be such deformations that are needed to produce correct correspondences [60]. Moreover, due to the inherently pairwise nature of deformation-based registration strategies they may be potentially inconsistent when applied to the problem of establishing correspondence across a group of shapes. As an example, consider three
shapes, A, B and C. It is possible to match A to C and B to C, however, it is not guaranteed that the inferred correspondence would be the same if A and B were matched directly [60]. These critiques extend to the pairwise techniques that combine any of the proximity and deformation measures in some way or another. Such ad-hoc and heuristic correspondence measures are also inconsistent from a theoretical point of view because they are based on arbitrary combinations of incommensurate similarity terms [60,70].

In light of these drawbacks and the lack of general measure to define good correspondence, population-based optimization strategies have been gaining popularity. The key idea behind these approaches is that the quality of the correspondence can be evaluated by measuring the quality of the SSM built using that correspondence. Thus parameterizations of training shapes which produce an optimal SSM also establish correspondence between them. Optimality in this case refers to any objectively quantifiable, quality related property of a SSM. One of the very first studies [74] to adopt this methodology defined an optimal model which was the most compact and could therefore be described by as small a number of eigenmodes as possible. In mathematical terms this measure translates into a product of SSM eigenvalues, which could be interpreted as the square of the hyper-volume spanned by the eigenmodes in the shape space. Later, Davies [60,70] introduced three general measures that could be used to assess the SSM quality. These measures are described in Table 3.1 and their mathematical formulation is provided in the Methods section.

---

21 Such a measure does not exist because true correspondences between biological shapes are generally not known [58,60].
Table 3.1: SSM quality measures.

<table>
<thead>
<tr>
<th>MG</th>
<th>Generalization Ability</th>
<th>Ability of the model to represent new instances of the shape and not just those seen in the training set.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS</td>
<td>Specificity</td>
<td>Validity of the shapes generated by the model with respect to the training set.</td>
</tr>
<tr>
<td>MC</td>
<td>Compactness</td>
<td>Total variance captured by the model.</td>
</tr>
</tbody>
</table>

Given the above definitions of SSM quality, it soon becomes obvious that the first two measures (MG and MS) are mutually exclusive and optimizing either one of them independently will reduce the SSM quality with respect to the other. Furthermore, it is not clear how they should be combined into a single objective function because they quantify distinctly different properties of the model. It seems that to reconcile MG and MS, the SSM must be able to simultaneously interpolate and to some extent, extrapolate the training set. In order to satisfy these conditions, Davies [70] adopted the principle of Occam's razor that in the present context could be paraphrased as “the simplest description of the training set will interpolate/extrapolate the best” [60]. Remarkably, this notion could be formalized using the principle of minimum description length (MDL). Within this information theoretic framework, the description length refers to the information content of an encoded message that must be transmitted to a receiver. The message consists of two parts, the parameters of the model that would allow the receiver to reconstruct the model and the training set encoded using via this model [60]. An outstanding attribute of this approach is that it can take any number of incommensurate data terms and subsequently reduce them to a simple message length measured in bits (or any other unit of information) [60], which is exactly what we aim to achieve in constructing a SSM that optimally satisfies all of the quality measures listed in Table 3.1. In fact, it has been demonstrated that optimization of the MDL based objective function
produced models which were better than those produced with manual annotation [60,68,71,72]. For this reason and all previous considerations, the SSMs built in this study were based on the correspondences obtained with MDL objective functions, which will be described shortly. The methods of deriving the general description length function are, however, beyond the scope of this study and interested readers are referred to [60,68,70-72] for additional information regarding this subject.

3.1.4 Description Length Cost Function

In [60,70], relying on the assumption that the distribution of shapes can be approximated as a multivariate Gaussian, Davies derived a general description length (DL) cost function suitable for establishing correspondence between objects that belong to the same shape class. The full DL functional is given in (3.4) with constituent terms defined in equations (3.5) to (3.13):

\[ \mathcal{L} = \sum_{m=1}^{M} \left( \frac{L_m^{(A1)}}{L_m^{(B1)}} + \frac{L_m^{(A2)}}{L_m^{(B2)}} \right), \sigma_m \geq \sigma_{cut} \]

\[ L_m^{(A1)} = 1 + \ln \left( \frac{\sigma_{max} - \sigma_{cut}}{\delta_m^{(A)}} \right) + \left| \ln \left( \delta_m^{(A)} \right) \right| \]  
\[ \delta_m^{(A)} = \min \left( 1, \sigma_m \sqrt{12/M} \right) \]  
\[ L_m^{(B1)} = 1 + \ln \left( \frac{\sigma_{max} - \sigma_{cut}}{\delta_m^{(B)}} \right) + \left| \ln \left( \delta_m^{(B)} \right) \right| \]  
\[ L_m^{(A2)} = \frac{M}{2} \left[ 1 + \ln \left( \frac{2 \pi \sigma_m^2}{\Delta^2} \right) + \frac{1}{6} \left( \frac{\delta_m^{(A)}}{\sigma_m} \right)^2 \right] \]  
\[ \delta_m^{(B)} = \min \left( 1, \sigma_m \sqrt{12/M} \right) \]  

63
\[
L_{m}^{(B2)} = \frac{M}{2} \left[ \ln \left( \frac{2n \sigma_{cut}^2}{\Delta^2} \right) + \left( \frac{\sigma_m}{\sigma_{cut}} \right)^2 \left( 1 + \frac{1}{4} \left( \frac{\delta^{(B)}}{\sigma_{cut}} \right)^2 \right) - \frac{1}{12} \left( \frac{\delta^{(B)}}{\sigma_{cut}} \right)^2 \right]
\]  
(3.9)

\[
\delta^{(B)} = \min \left( 1, \sigma_{cut} \sqrt{\frac{12}{M}} \right)
\]  
(3.10)

\[
\sigma_m^2 = \lambda_m
\]  
(3.11)

\[
\sigma_{cut} = 2\Delta
\]  
(3.12)

\[
\sigma_{\max} = r\sqrt{N}
\]  
(3.13)

In the above equations, \(M\) is the total number of shapes in the training set, \(\lambda_m\) is the eigenvalue of the \(m\)-th eigenmode, \(r\) is the radius of a sphere that encloses all of the aligned training shapes and \(\Delta\) is the point precision used to encode the training set. The only free parameters are the number of shape points (\(N\)) and the data precision (\(\Delta\)). For shape boundaries obtained from image data, \(\Delta\) is typically on the order of the voxel size [60].

At this junction we would like to point out an essential feature of Davies' correspondence algorithm. Specifically, Davies \(et\ al\). compute the covariances using continuous surface representations\(^{22}\) so that \(mn\)-th element of the matrix is defined as:

\[
C_{mn} = \frac{1}{A} \iint \left( S_m(u, v) - \bar{S}(u, v) \right) \cdot \left( S_n(u, v) - \bar{S}(u, v) \right) \, du \, dv
\]  
(3.14)

where \(u\) and \(v\) are the tangential components of the parametric surface representation, \(\bar{S}\) is the mean shape and \(A\) is the surface area of \(\bar{S}\). One should note that in this formulation \(N\) is not an explicit parameter. Furthermore, although (3.14) provides a more accurate measure of similarity (or dissimilarity) between the surfaces it is also significantly more

\(^{22}\) Although continuous surface representations are infinitely dimensional, a training sets consisting of \(M\) objects can have at most \(M - 1\) non-zero eigenvalues [60]. In fact, these eigenvalues can be inferred from the dual space of the covariance matrix of dimension \(M\), whose entries are specified in (3.14).
computationally expensive to evaluate. For the remainder of this chapter we assume that eigenvalues used in calculating the DL of statistical shape models are derived from the discrete (i.e. point-based) surface representations (see subsection 3.1.2).

The general DL function is quite voluminous. By considering the special case of infinitely small data precision ($\Delta \rightarrow 0$) and an infinitely large dataset ($M \rightarrow \infty$) [71], one can obtain an approximate, though much simpler expression of the DL:

$$\mathcal{L} \approx \mathcal{L}_D = f(r, \Delta, M) + \sum_{\lambda_m \geq \lambda_{cut}} \mathcal{L}_m^{(1)} + \sum_{\lambda_m < \lambda_{cut}} \mathcal{L}_m^{(2)}$$

(3.15)

$$\mathcal{L}_m^{(1)} = \frac{(M - 2)}{2} \ln(\lambda_m)$$

(3.16)

$$\mathcal{L}_m^{(2)} = \frac{M - 2}{2} \ln(\lambda_{cut}) + \frac{M + 3}{2} \left[ \frac{\lambda_m}{\lambda_{cut}} - 1 \right]$$

(3.17)

where $f(r, \Delta, M)$ is some fixed function of the training set and model parameters. Because $f(r, \Delta, M)$ does not depend on $\sigma_m$, its contribution to $\mathcal{L}_D$ does not vary in the course of the optimization. More importantly, it should be noted that that $\mathcal{L}_D$ is continuous as $\lambda_m \rightarrow \lambda_{cut}$, even in the limit $\lambda_m \rightarrow \infty$ [72].

Taking an additional step, Thodberg [75] simplified the DL formula in (3.15) even further by assuming that the range of the data in any direction is always greater than quantization parameter ($\Delta$) [60], so that:

$$\mathcal{L}_D \approx \mathcal{L}_T = \sum_m \mathcal{L}_m$$

(3.18)

$$\mathcal{L}_m = \begin{cases} 
\ln \left( \frac{\lambda_m}{\lambda_{cut}} \right) + 1 & ,\lambda_m \geq \lambda_{cut} \\
\frac{\lambda_m}{\lambda_{cut}} & ,\lambda_m < \lambda_{cut}
\end{cases}$$

(3.19)
where $\lambda_{\text{cut}} = (\Delta/r)^2$ is the degree of uncertainty in the measurement [67,75]. As defined previously, $r$ is the radius of a sphere that encloses all of the aligned training shapes and $\Delta$ is the measurement precision (commonly used setting for $\Delta$ is 0.3 times the pixel/voxel resolution [67,75]). An interesting difference between (3.15) and (3.18) is that for training sets of moderate size (e.g. $M=30$), the latter equation is differentiable in the limit $\lambda_m \mapsto \lambda_{\text{cut}}$ whereas the former is not. In this study both (3.15) and (3.18) were used to establish correspondence. We evaluate the capacity of the two cost function to produce good models (as measured by compactness, specificity and generalization ability) in the Results section.

One important point about the cost functions presented above is that they are typically implemented with PDMs and thus use proximity-based measures to optimize correspondence$^{23}$. However, the idea of minimizing these measures within a groupwise context so as to find the most compact description of the training set is fundamentally different from pairwise registration process that was discussed in the preceding subsection.

### 3.1.5 Optimization of Landmark Correspondences

With the measure of correspondence in place, we still require a mechanism to update the landmark configuration while constraining them to their proper surfaces. One way of addressing this issue is to use what is called parametric representation of shape. This approach involves embedding the mesh into a canonical space (where surface constraints

\[23\] Although shape vectors are typically assembled using only positional information, in some situations it may be beneficial to augment these representations with other shape information such as curvature [132,133].
can be easily enforced) and iteratively modifying the landmark positions until the DL can be no longer decreased. Appendix A provides a review of mesh parameterization and related concepts.

During the optimization, there are two possible ways of manipulating landmark correspondences. The first is based on direct adjustment of landmark positions while maintaining fixed parameterizations. The second alternative is the so called re-parameterization approach and relies on the notion that parameterization can be modified independently of shape. Thus by maintaining a fixed set of landmarks on the sphere, parameterizations can be deformed to align with the landmarks in a way that produces consistent mapping to the surfaces of the training shapes. According to [67] the advantage of the latter method is that the correspondence is valid for any set of points placed on the unit sphere. Furthermore, to better adapt the triangulation to the training shapes, the number and placement of landmarks can be easily modified at any point during the optimization. For these reasons, this study used re-parameterization strategy to obtain optimal correspondences.

In [60] Davies et al. describe a number of local and global deformation schemes that can be employed during re-parameterization. Based on [67,68,71,72] it appears that the most common method relies on symmetric-theta transformations. According to this formulation, the cumulative distribution $f(\theta)$ of a warped Cauchy kernel situated at $a \in S^2$ is used to modify the position of an arbitrary point $x$ on the unit sphere as shown in (3.20).
\[ x^* = \mathbf{a} \cos f(\theta) + \frac{\sin f(\theta)}{\sin \theta} (x - \mathbf{a} \cos \theta) \]  

(3.20)

\[ \theta = \cos^{-1} \mathbf{a} \cdot \mathbf{x} \]  

(3.21)

\[ f(\theta) = \frac{1}{1 + A} \left[ \theta + A \cos^{-1} \left( \frac{(1 + \Omega^2) \cos \theta - 2\Omega}{1 + \Omega^2 - 2\Omega \cos \theta} \right) \right] \]  

(3.22)

One should note that the symmetric theta transformation is undefined at \( \mathbf{a} \) and its antipodal point \((-\mathbf{a})\), and in those instances \( x^* = x \).

By combining the effects of multiple Cauchy kernels, with varying width \( (\Omega) \), amplitude \( (A) \) and position \( (\mathbf{a}) \), it becomes possible to obtain arbitrary deformations of the parameter mesh. Optimizing over the parameters of the individual kernels (i.e. \( \Omega, A \) and/or \( \mathbf{a} \)), the parameterization can be steered towards conformation that minimizes the description length of the model. A noteworthy attribute of this approach is that it is indirect. As such, it avoids explicit calculation of the displacement vector of the individual parameter mesh vertices and instead optimizes a parameter set of much smaller cardinality, effectively reducing the dimensionality of the optimization problem.

Figure 3.3 shows the result of applying a single symmetric theta transformation due to a kernel situated at the north pole of the unit sphere. Observe how the mesh vertices are forced apart at the north pole and are compressed at the antipodal point (i.e. south pole).

One can also imagine arbitrary arrangements of Cauchy kernels with random or uniform distributions. As shown in [60,67,68,71,72], by optimizing over the width and/or amplitude parameters of these kernels, parameterizations can be deformed to minimize
any type of objective function\textsuperscript{24}. For example, Figure 3.4 provides an illustration of deformations obtained with random arrangement of Cauchy kernels.

\begin{figure}
\centering
\begin{tabular}{ccc}
\textbf{ORIGINAL MESH} & \textbf{$\Omega=0.10$} & \textbf{$\Omega=0.25$} & \textbf{$\Omega=0.50$} \\
\includegraphics[width=0.25\textwidth]{original_mesh}
\includegraphics[width=0.25\textwidth]{omega_0.10}
\includegraphics[width=0.25\textwidth]{omega_0.25}
\includegraphics[width=0.25\textwidth]{omega_0.50}
\end{tabular}
\caption{Demonstration of a symmetric theta transformation applied on a unit sphere. In each case, two views (top and bottom) are presented. The color denotes relative degree of length distortion with compression and stretching being treated equally (blue=no distortion).}
\end{figure}

\begin{figure}
\centering
\includegraphics[width=0.6\textwidth]{symmetric_theta_transformations}
\caption{Examples of deformations obtained with symmetric-theta transformations. Object on the far left is the original mesh. The remaining objects (from left to right) have been obtained by deforming the original mesh with a random configuration of 4, 6 and 10 Cauchy kernels.}
\end{figure}

\textsuperscript{24} The issue of whether these optimizations converge on the globally optimal solutions is beyond the scope of this study.
The main drawback of optimizations based on symmetric-theta transformations is that they are inefficient due to their global effect on the parameterization, whereby addition of one new kernel modifies all vertex positions [67]. In [67], Heimann et al. propose an alternative method that maintains previously established landmark correspondences stable by using kernels with strictly local effects. According to this formulation the new position of a point \( \mathbf{x} \) on the sphere (e.g. a mesh vertex) with spherical co-ordinates \((\theta, \phi)\) is determined by (3.23),

\[
(\theta^*, \phi^*) = (\theta, \phi) + G(\|\mathbf{x} - \mathbf{a}\|, \Omega) \cdot \Delta(\theta, \phi)
\]  

(3.23)

\[
G(d, \Omega) = \begin{cases} 
\exp \left( -\frac{d^2}{2\Omega^2} \right), & d \leq 3\Omega \\
0, & d > 3\Omega 
\end{cases}
\]  

(3.24)

where \( \mathbf{a} \in S^2 \) and \( \Omega \in (0, 2/3] \) are the position and standard deviation of a truncated Gaussian kernel (3.24), respectively. \( \Delta(\theta, \phi) \) is the displacement direction associated with the kernel that minimizes the description length\(^{25}\).

To ensure equal treatment of all mesh vertices Heimann et al. modify the relative position of a fixed kernel configuration by a random rotation of the parameterization mesh (and corresponding landmark configuration). The optimization proceeds in a hierarchal order, with increasingly narrower kernels being used to manipulate the parameterizations at smaller scales (see Figure 3.5).

\(^{25}\) The means to compute optimal \( \Delta(\theta, \phi) \) is described in Section 3.2.3.
Figure 3.5: An example of four different scales of truncated Gaussian kernels uniformly distributed across the surface of the unit sphere. From left to right, each configuration has 3, 10, 18 and 28 kernels. For technical reasons (discussed in [67]) kernels at the poles are omitted. In this study kernel centroids were determined using the MATLAB Sphere Partitioning Toolbox [83,84].

Figure 3.6 demonstrates the effect produced by applying the first two kernel configurations to a regular mesh with displacements of 10 and 8 degrees in the longitudinal direction, respectively. The angular displacement map due to a single kernel is shown in Figure 3.7. One can see that by iteratively assigning different $\Delta(\theta, \phi)$ to the individual kernels and then modifying their position relative to the mesh, it is possible to obtain arbitrary deformations.

Figure 3.6: Deformations produced by the first two kernels configuration shown in Figure 3.5 due to the displacement of 10 (left) and 8 (right) degrees in the latitudinal direction. The red dots denote the displacement of points situated at the kernel centroids (black dots) after the deformation. Edge color represents the relative amount of length distortion with compression and stretching being treated equally (blue=no distortion).
Figure 3.7: Color coded angular displacement map due to a single kernel ($2^{nd}$ scale, $\Omega=0.167$) with displacement of 10 degrees in the latitudinal direction. Red dot denotes the position of the point coincident with kernel centroid (black dot) after the deformation. Note the local nature of the deformation. Also observe how the vertices above the red point are pulled apart and those below it are compressed.

In [67], it was reported that the convergence rate of this method far superseded that of Davies' re-parameterization approach. However, the two procedures were implemented in completely different programming environments so it is hard to gauge exactly the extent of the purported improvement. In this study, we use Heimann's method [67] to optimize the two different cost functions ((3.15) and (3.18)). The details of the algorithm are expounded in the following section.

### 3.2 Methods

This section describes the main steps involved in obtaining dense point correspondence (PC) that was subsequently used to construct statistical shape models (SSMs) of the individual wrist bones. Correct PC ensures that the landmarks of the training shapes are
situated in the semantically equivalent anatomical positions and thus the variations captured by the SSM are based on actual anatomical shape differences. The process of establishing PC is composed of three steps: 1) pairwise registration using similarity transformation, 2) normalization of conformal parameterizations with respect to the Möbius transformation group and 3) groupwise point correspondence optimization. The first step pre-aligns the surfaces using similarity transformation and was designed with the considerations of robustness and reusability. Although the utility of the first criterion is self-explanatory, the reusability feature is meant to allow parts of the procedure to be integrated into the later model building stages (see Chapter 4). The second step is meant to ensure that parameterizations of the training shapes are also pre-aligned thus providing good initialization to the re-parameterization procedure that followed. The last step was the most challenging part of the entire model building process and ultimately determined the final quality of the SSMs. For this reason it was implemented with the method [67] that has an extensive track record of application to a wide range of SSM construction problems. Additionally, subsection 3.2.4 describes three standard measures used to evaluate the SSM quality. It should be noted that all of the above operations were performed with a resampled set of surface meshes that were obtained using the remeshing algorithm described in Chapter 2.

3.2.1 Linear Surface Registration

The purpose of linear registration is to pre-align the shapes within a common frame of reference so the resulting variance in landmark positions across the training would depend only on the anatomical differences and not differences in scale and pose. In this
study, we performed this operation using a linear transformation model composed of scaling, rotation and translation. Formally, the operation is defined in (3.25), where \( \boldsymbol{v}, \mathbf{t} \in \mathbb{R}^3 \), \( s \) is a positive scalar and \( \mathbf{R} \) is a 3x3 composite rotation matrix (3.26), defined by the product of rotations about the x-, y- and z-axes,

\[
\mathbf{v}^* = s(\mathbf{R}\mathbf{v} + \mathbf{t})
\]  

where \( C(\theta) = \cos \theta \) and \( S(\theta) = \sin \theta \).

Typically, the scale and translation parameters can be easily estimated by comparing the gross volumes and centroids, respectively, of the two shapes being registered. Finding an optimal rotation on the other hand is somewhat more involved because there is no a priori geometric correspondence between the surfaces. Two common approaches of finding rotation are comparison of principal components of the objects (i.e. axes of inertia) \([76,77]\) and the iterative closest point (ICP) algorithm \([50-52,60,73,78-80]\). The former method requires the objects to have unambiguous moments of inertia, an unrealistic prerequisite for the potato-like bones in the wrist. The latter is a pairwise optimization approach that aligns the two surfaces by minimizing a distance-based proximity cost function. Major weakness of the ICP algorithm is that it is sensitive to the initialization of the search and is therefore likely to produce suboptimal results if the initial pose and/or geometry of the two objects being registered are significantly different \([78,79]\). However, in \([80]\) a variant of an ICP algorithm was used to register a set of carpuses and was reported to have registration errors on the order of voxel resolution. One crucially important implementation detail of the method was that the objective
function used point-to-surface distances instead of the common point-to-point distance measures. Because of this feature, from a theoretical point of view, the distance transform based registration method described below is very similar to the method used in [80], but provides one additional benefit which is exploited during the later model building stages (see Chapter 4).

3.2.1.1 Surface Registration with Distance Transforms

The distance transform (DT) of a surface is a continuous scalar field that for any point in its domain provides a value of the Euclidean distance to the closest point on the surface. According to this definition, the surface is defined as the zero level-set of the DT. The theory underlying DT and how it can be computed from polygonal surfaces (i.e. meshes) is discussed in Appendix D.

Let $F_i(\nu)$ and $F_j(\nu)$ be the distance transforms of the reference and source surfaces, respectively. Likewise, let $M_i$ and $M_j$ be the piece wise linear representations of the reference and source surfaces, defined by the triangulations of uniformly distributed points sets $V_i = \{\nu_{ik} | k = 1, ..., N_i\}$ and $V_j = \{\nu_{jk} | k = 1, ..., N_j\}$, respectively. Our objective is to optimally align the source mesh ($M_j$) to the reference mesh ($M_i$), a task performed by minimizing the following symmetric registration error function [80]:

$$D_{ij}(s_{ij}, R_{ij}, t_{ij}) = \frac{1}{2\tilde{N}_j} \sum_{\nu_{jk} \in \tilde{V}_j} \left| F_i \left( s_{ij}(R_{ij}\nu_{jk} + t_{ij}) \right) \right|^2 + \frac{1}{2\tilde{N}_i} \sum_{\nu_{ik} \in \tilde{V}_i} \left| F_j \left( s_{ij}^{-1}(R_{ij}^{-1}\nu_{ik} - s_{ij}t_{ij}) \right) \right|^2$$

(3.27)
where $\tilde{N}_j$ ($\tilde{N}_i$) is the total number of $V_j$ ($V_i$) vertices inside the domain of $F_j$ ($F_i$). $D_{ij}$ can be interpreted directly as the average (squared) distance between the source and reference surfaces. We optimized (3.27) using a downhill Nelder-Mead simplex algorithm\textsuperscript{26}, with the initial rotation angles set to zero, translation set to the centroid of $M_i$ and the scale parameter initialized using (3.28).

$$s_{ij} = 3\sqrt{\frac{\text{Volume}(M_j)}{\text{Volume}(M_i)}}$$  \hspace{1cm} (3.28)

The pairwise registration procedure described above can be applied to transform the entire training set of shapes into the same frame of reference. In this study, the “optimal” coordinate system was aligned with the pose of the training shape that minimized the following quantity:

$$D_i = \left[ \max_{j \in [1,N_S]} (D_{ij}) \right] \sum_{j=1}^{N_S} D_{ij}$$ \hspace{1cm} (3.29)

where $N_S$ is the total number of objects in the training set. The shape that minimized (3.29) had an overall low registration error to all other shapes in the training set and also minimized the maximum pairwise registration score. In terms of implementation, computing $D_{ij}$ for the entire training set would require $N_Sx(N_S - 1)$ pairwise registrations, however, recognizing that $D_{ij} = D_{ji}$, this number can be halved\textsuperscript{27}. An important point about the above procedure is that it is biased with respect to the shape chosen as the reference. This problem was rectified during the groupwise optimization.

\textsuperscript{26} Nelder-Mead algorithm is implemented as 'fminsearch' function in MATLAB.

\textsuperscript{27} Note because $D_{ij} = D_{ji}$ we have $s_{ji} = s_{ij}^{-1}$. $\bf{t}_{ji} = -s_{ji} \bf{t}_{ij}$ and $\bf{R}_{ji} = \bf{R}_{ij}^{-1}$. 

76
procedure (see subsection 3.2.3) designed to incorporate minimization of the DL with respect to scale, translation and rotation of the individual training shapes.

3.2.2 Möbius Transformation Normalization

Recall that optimization of the description length is performed with the use of spherical parameterizations. One of the benefits of this approach is that it provides a very efficient means of enforcing surface constraints. All parameterizations employed in this study are conformal, that by definition minimize distortion in angles. These parameterizations were computed during the mesh resampling procedure described in Chapter 2. Conformal maps are not unique, however, as they can be mapped onto themselves by an arbitrary conformal transformation \[39,81\]. The set of such mappings forms a 6-parameter family termed the Möbius group. Let \( f : M \to S^2 \) be the initial conformal mapping of piecewise linear surface, \( M \), onto the unit sphere, then \( f \) can be mapped conformally onto itself via the following transformation (a.k.a. automorphism):

\[
F = \varphi^{-1} \circ m \circ \varphi \circ f_o
\]  

(3.30)

where \( \varphi : S^2 \to \mathbb{C} \) is the stereographic projection (3.31) of the sphere onto a complex plane, \( m \) is the Möbius transformation (3.32) and \( \varphi^{-1} : \mathbb{C} \to S^2 \) is the inverse stereographic projection (3.33) from the complex plane back to the spherical parameter domain.

---

28 The DL function cannot be used to concurrently optimize scale of all training shapes simultaneously because it can cheat by scaling all shape vectors to 0, thereby eliminating all variance between the training shapes and producing a trivial solution. However, sequential optimization of the DL with respect to scale of one object at a time will avoid degenerate solutions, because any attempts to significantly decrease \( s_{ij} \) with respect to the scale of the \( N_x - 1 \) remaining shapes in the training set will be penalized by substantial increase in the DL.
Given a set of training shapes \( \{ M_i \mid i = 1, 2, \ldots, N_S \} \) and their corresponding conformal parameterizations, \( \{ f_i \} \), the objective is to deform individual spherical meshes under the action of the Möbius transformation group so that the points in the parameter domain that have the same spherical coordinates are mapped to approximately the same regions in the primary domain of the training shapes. To clarify this requirement, consider the following example.

Let \( M = [V, E] \) be a mesh of an orientable, closed genus-0 surface, where \( V = \{ v_i \mid i = 1, \ldots, N_v \} \) and \( E \) are the sets of vertices and edges, respectively. Suppose \( f_A \) is a conformal parameterization of \( M \), such that \( f_A(v_i) = x_{Ai} \). Let \( f_B \) be an alternative parameterization of \( M \) such that \( f_B(v_i) = x_{Bi} \) and \( x_{Bi} = Rx_{Ai} \), where \( R \) is an unknown rotation matrix. Rotation does not deform the parametric mesh and therefore preserves the conformal relationship between the parameter and primary domains. For nontrivial rotations we have \( R \neq I \) which means that \( f_A^{-1}(x_{Bi}) \neq f_B^{-1}(x_{Ai}) \neq v_i \). The objective is to deform \( f_A \) so that \( F_A^{-1}(x_{Bi}) = v_i \) where

\[
F_A(x_i, a, b, c, d) = \varphi^{-1} \circ m(a, b, c, d) \circ \varphi \circ f_A(x_i)
\]  

(3.34)

---

Rotation is not a deformation in the true sense, however (3.30) does treat rigid transformations of the sphere as deformations in the complex plane.
and $a,b,c,d$ are the Möbius transformation parameters. Since the three parameter rotation group is a subset of the Möbius group [39,81] there exist $a,b,c,d$ that satisfy (3.34).

The above argument can be extended to two polygonal surfaces, $M_A$ and $M_B$, that represent different instances of the same type of object with the corresponding conformal parameterizations $f_A$ and $f_B$. The optimal parameters of the Möbius transformation that deforms $f_A$ to $f_B$ can be computed using the least-squares approach described below.

Let $M_{ref}$ be a reference triangulation and $f_{ref}$ be its corresponding conformal parameterization. Define $\{x_k\}$ be a set of points in the parameter domain such that $\{v_k = f_{ref}^{-1}(x_k)\}$ is a set of landmarks uniformly distributed on the surface of the reference shape$^{30}$ (see subsection 2.3.4). For each training shape, $M_i$, we seek a set of Möbius parameters $a^*,b^*,c^*,d^* \in \mathbb{C}$ that minimize the sum of squared Euclidean distances between $\{F_i^{-1}(x_k)\}$ and $\{v_k\}$, subject to $a^*d^* \neq b^*c^*$ constraint as in (3.35).

We solve (3.35) for all training shapes, $\{M_i | i = 1, ..., N_S\}$, using interior-point algorithm implemented as 'fmincon' function in MATLAB’s Optimization Toolbox.

\[
(a^*, b^*, c^*, d^*)_i = \arg\min_{a,b,c,d \in \mathbb{C}} \sum_k \|F_i^{-1}(x_k, a, b, c, d) - v_k\|^2
\]

subject to $|ad - bc| > 0$  

(3.35)

It is important to note that since we are dealing with discrete representations of continuous surfaces, the spherical parameterizations we employ are approximations of ideal conformal maps and therefore the Möbius transformation will only approximately register two parameterizations of different shape instances. The following subsection

$^{30}$ In this study, all parameterization were normalized with respect to the same reference that was used previously to rigidly align the training shapes (see Section 3.2.1.1).
describes a more flexible re-parameterization strategy that is not constrained to the conformal space and uses normalized parameterizations as initializations.

3.2.3 Groupwise Correspondence Optimization

In subsection 3.1.5, it was pointed out that the optimization of the MDL cost function can be performed with the use of parametric representation of shape. During optimization there are two possible ways of manipulating correspondence. The first involves direct adjustment of landmark positions while maintaining the parameterizations fixed. The second alternative, which is described in this subsection, is the so called re-parameterization approach. According to [67] the advantage of the latter method is that it produces valid correspondence for any set of points placed on the unit sphere. Furthermore, at any point in the optimization the number and placement of landmarks can be easily modified to better adapt the triangulation to the training shapes.

Let \( \{M_i \mid i = 1, ..., N_S\} \) be a set of training shape meshes and let \( \{f_i\} \) be a set of corresponding conformal parameterizations (normalized with respect to the Möbius transformations). At the beginning, one shape is chosen as the reference (could be the same as the reference used to rigidly align the training set) and is adaptively sampled in its parameter domain with \( N_P \) points, \( \{x_j \mid j = 1, ..., N_P\} \), so as to produce a uniform sampling in the primary domain (see subsection 2.3.4). This configuration of points is maintained fixed (in the parameter domain) during the optimization. Bearing in mind that \( f_i^{-1}(x_j) \) maps the j-th landmark to the surface of the i-th training shape, our objective is to find a set of optimal parameterizations, \( \{F_i\} \), such that the inverse mapping of landmarks to the primary domain, \( F_i^{-1}(x_j) \), distributes them uniformly across the training
surfaces while minimizing the description length of the model. As described in subsection 3.1.5, one of the ways to obtain \( \{ F_i \} \) is to iteratively deform the initial parameterizations, \( \{ f_i \} \), using kernels with strictly local effects (see Figure 3.5) [67].

Following Heimann et al. [67], we initialize \( F_i \) to \( f_i \) and repeatedly warp \( F_i \) along the negative gradient of the DL using a warping function \( g_k : S^2 \to S^2 \)

\[
g_k(r, \Delta \theta_k, \Delta \phi_k) = \psi^{-1} \circ (\psi(r) + G(\|\psi(r) - p_k\|, \sigma_k) \cdot (\Delta \theta_k, \Delta \phi_k)) \tag{3.36}
\]

\[
G(d, \sigma) = \begin{cases} 
\exp \left( \frac{-d^2}{2\sigma^2} \right), & d \leq 3\sigma \\
0, & d > 3\sigma
\end{cases} \tag{3.37}
\]

where \( \psi \) is the transformation from the rectangular to spherical coordinate system, \( p_k \in S^2 \) is the position of \( k \)-th truncated Gaussian kernel (3.37) and \( \sigma_k \) is the kernel's bandwidth\(^{31}\). More importantly, \( (\Delta \theta_k, \Delta \phi_k) \) is the displacement direction associated with the \( k \)-th kernel that minimizes the description length and is computed as:

\[
(\Delta \theta_k, \Delta \phi_k) = -\Delta \cdot \left( \sum_{j \in \Omega_k} \frac{\partial L}{\partial s_{ji}} \cdot \frac{\partial s_{ji}}{\partial \Delta \theta} + \sum_{j \in \Omega_k} \frac{\partial L}{\partial s_{ji}} \cdot \frac{\partial s_{ji}}{\partial \Delta \phi} \right) \tag{3.38}
\]

where \( \{ s_{ji} = F_i^{-1}(x_j) - 1/N_s \sum_i F_i^{-1}(x_j) \mid j = 1,2,\ldots,N_p, \quad i = 1,2,\ldots,N_s \} \) is the set of unbiased and centered landmark positions in the primary domain (note \( F_i \) is based on the most recently updated positions of parameter mesh vertices). Additionally, \( \Omega_k \) is a set of landmarks under the influence of the \( k \)-th kernel, \( \Delta \) is a user defined step size (e.g. \( \Delta = \pi/180 \) rad which corresponds to the shift of one degree), and the sums are the net changes in the DL cost function due to kernel movements in \( \theta \) and \( \phi \) directions,

\(^{31}p_k \) and \( \sigma_k \) are fixed for a given configuration of kernels (see Figure 3.5) and can be determined using the Sphere Partitioning Toolbox [83,84].
respectively. The dot products in (3.38) can be interpreted as “work” being done against the DL gradient, resulting in negative values when the displacement due to warping is in the direction opposing the gradient. The leftmost derivative terms that appear in the sums of (3.38) can be evaluated analytically. For example, the derivatives of the two cost functions, (3.15) and (3.18), with respect to \( s_{ji} \) are shown in (3.39) and (3.40), respectively. In Appendix E, we describe how to evaluate \( \partial \lambda_m / \partial s_{ji} \).

\[
\frac{\partial L_D}{\partial s_{ji}} = \frac{1}{2} \sum_m \left\{ \begin{array}{ll}
N_S - 2 \frac{\partial \lambda_m}{\partial s_{ji}}, & \lambda_m \geq \lambda_{cut} \\
N_S + 3 \frac{\partial \lambda_m}{\partial s_{ji}}, & \lambda_m < \lambda_{cut}
\end{array} \right. 
\] (3.39)

\[
\frac{\partial L_T}{\partial s_{ji}} = \sum_m \left\{ \begin{array}{ll}
\frac{1}{\lambda_m} \frac{\partial \lambda_m}{\partial s_{ji}}, & \lambda_m \geq \lambda_{cut} \\
\frac{1}{\lambda_{cut}} \frac{\partial \lambda_m}{\partial s_{ji}}, & \lambda_m < \lambda_{cut}
\end{array} \right. 
\] (3.40)

The remaining terms in (3.38) can be estimated numerically. For example, using the forward difference approximation, the DL derivative of the normalized position of the \( j \)-th landmark on the \( i \)-th training shape with respect to the displacement of the \( k \)-th kernel in the \((\Delta \theta, 0)\) direction is:

\[
\frac{\partial s_{ji}}{\partial \Delta \theta} \approx g_k^{-1}(F_i, \Delta \theta, 0)(x_j) - F_i^{-1}(x_j) 
\] (3.41)

To ensure equal treatment of all mesh vertices, at every iteration, the relative position of an otherwise fixed kernel configuration must be modified by a random rotation [85] of the parameterization mesh. Additionally, due to local effects of the transformations, the positions of the (parameter) mesh vertices under the influence of different kernels can be optimized simultaneously.
Recall that one of the advantages of using re-parameterization based optimization of the DL is that at any point in the optimization, the number and placement of landmarks can be modified to improve the correspondence. The main downside of the method, however, is that it is computationally very expensive. Fortunately, substantial performance gains can be achieved by initializing the optimization of the DL with a small number of landmarks and then increasing this number over subsequent optimization passes. Suppose \( L = \{ x_j | j = 1, ..., N_p \} \) is an initial set of landmarks (in the parameter domain) and \( M_L \) is its triangulation. In our implementation, after the initial optimization pass, we obtain correspondence for \( L \). During the second pass, we subdivide \( M_L \) and use deformed parameterizations from the previous pass to initialize the new search. Finally, we repeat this process once more for a total of three passes. At every pass, the optimization proceeds sequentially (i.e. parameterizations of the individual shapes are deformed independently) in a hierarchical order, as described in [67], with increasingly narrower kernels being used to manipulate the parameterizations at smaller scales (see Figure 3.5).

The initial number of landmarks \( (N_p) \) used for each bone is given in Table 3.2.

<table>
<thead>
<tr>
<th>Bone</th>
<th>( N_p )</th>
<th>Bone</th>
<th>( N_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scaphoid</td>
<td>295</td>
<td>Trapezium</td>
<td>172</td>
</tr>
<tr>
<td>Lunate</td>
<td>197</td>
<td>Trapezoid</td>
<td>206</td>
</tr>
<tr>
<td>Triquetrum</td>
<td>156</td>
<td>Capitate</td>
<td>270</td>
</tr>
<tr>
<td>Pisiform</td>
<td>65</td>
<td>Hamate</td>
<td>241</td>
</tr>
</tbody>
</table>
An important implementation detail is that the convergence rate and stability of the above procedure depends on the user specified step size, $\Delta$. In preliminary experiments it was observed that as the optimization approached the solution, a static value of $\Delta$ eventually resulted in fluctuations of the DL. This issue was addressed by assigning a set of $\{\Delta_i \mid i = 1, \ldots, N_s\}$ to all training shapes and decreasing them independently (for example, by a factor of 2) when an increase in the cost function was detected with respect to the previous iteration.

As a last note, in subsection 3.2.1 it was pointed out that pairwise registration used to pre-align the surfaces is biased with respect to the shape chosen as the reference. This problem was rectified by periodically adjusting the pose of the individual training shapes in a manner that minimized the DL cost function. The scale, rotation and translation were determined separately for each shape by optimizing (3.15) or (3.18) with the Nelder-Mead downhill simplex algorithm. To avoid preferential treatment of shapes with low indices, the order of the optimization was established by a random permutation of a number sequence from 1 to $N_s$. The overall groupwise correspondence optimization procedure described in this subsection is summarized in Appendix F.

### 3.2.4 Parameter Selection & SSM Quality

In this study we sought to construct compact SSMs of carpal bones with good generalization ability and specificity (definitions of these measures are provided below).

In subsection 3.1.4 we identified two MDL based cost functions for establishing correspondence between the training shapes. These cost functions have one free parameter, $\lambda_{cut}$, that marks the transition between the determinant-like term and the trace
term of the covariance matrix. Typical choice of $\lambda_{cut}$ depends the measurement precision ($\Delta$) of the data. For Davies' and Thodberg's cost functions $\lambda_{cut}$ is typically estimated using (3.42) and (3.43) [60,75], respectively.

$$\lambda_{cut, D} = (2\Delta)^2 \quad (3.42)$$

$$\lambda_{cut, T} = (\Delta/r)^2 \quad (3.43)$$

There is, however, no guarantee that these specific choices of $\lambda_{cut}$ will lead to SSMs with optimal compactness, specificity and generalization properties. Furthermore, $\Delta$ may not be necessarily the same as the resolution of the volumetric scans\(^{32}\) from which the surface models were derived, as additional errors could have been introduced during the segmentation process and subsequent editing of the individual meshes [16]. For these reasons we first investigated the effect of $\lambda_{cut}$ on the SSMs quality measures (defined below). Since the optimization of the MDL cost functions is a computationally expensive procedure, we used only three different settings of $\lambda_{cut}$. The first ($\lambda_{cut_1}$) was determined using (3.42) and (3.43), and the remaining two were set to 10 and 0.1 times $\lambda_{cut_1}$. In these experiments, the MDL cost functions were optimized using a relatively small set of landmarks that were uniformly distributed on the surface of the reference shape (in the primary domain). The total number of landmarks used for each bone is given in Table 3.2. Additionally, to ensure equal treatment of the three cases, all optimizations were performed using the same number of iterations ($5 \times 10^3$ at the $1^{st}$ scale, $3 \times 10^3$ at the $2^{nd}$, $2.5 \times 10^3$ at the $3^{rd}$ and $4^{th}$ scales).

\(^{32}\)Carpal bone meshes were derived from the CT scans acquired with ~0.25 mm voxel resolution [16].
3.2.4.1 Compactness

Compactness is a measure of cumulative model variance (3.44), and is commonly defined as a function of the number of modes \((m = 1, 2, \ldots, N_s - 1)\) used by the model. A compact model is one that has an overall small variance and requires as few parameters as possible to reconstruct the training set.

\[
C(m) = \sum_{i=1}^{m} \lambda_i
\]

(3.44)

3.2.4.2 Specificity

Specificity is a quantitative measure of similarity between the model and the training set from which the model was constructed. A specific model should generate shapes similar to those in the training set, hence low specificity values are desirable. In practice, this measure is estimated by generating random shapes, \(\{V_i\}\), from the distribution of the training shape parameters\(^{33}\), and comparing them to the closest match in the training set, \(V_j\). Formally, specificity is defined as a function of \(m\) reconstruction modes \((m = 1, \ldots, N_s - 1)\)

\[
S(m) = \frac{1}{N} \sum_{i=1}^{N} |V_i(m) - V_j^*|^2
\]

(3.45)

\[
j = \arg\min_{k \in [1,N_s]} |V_i - V_k^*|^2
\]

where \(N\) is the number of random samples (e.g. \(N=10^4\)). The standard error of \(S(m)\) is given by \(\sigma_S(m) = \sigma/\sqrt{N - 1}\), where \(\sigma\) is the sample standard deviation of \(S(m)\) [68,86].

\(^{33}\) This distribution was assumed to be a zero-centered Gaussian with a diagonal covariance matrix whose entries are equal to the eigenvalues of the SSM. Also recall that the shape vector \(V_i\) is simply a 'string' of concatenated landmark coordinates (see Section 3.1.2).
3.2.4.3 Generalization Ability

The generalization ability is a measure of the model's capacity to represent unseen instances of objects that belong to the same class as the training shapes. Generalization is a reflection of how well a model can learn the characteristics of an object class from a limited training set [86]. In cases where the model is over-fitted to the training set, it will be unable to generalize to unseen examples [86]. In practice, generalization is estimated by conducting a series of leave-one-out experiments on the training set. The procedure involves constructing a SSM with $N_s - 1$ training shapes and then reconstructing the omitted object with $m$ eigenmodes ($m = 0, 1, ..., N_s - 2$)\(^{34}\). After the process had been repeated for all training shapes, the generalization ability is defined as the average reconstruction error between the reduced model and the omitted example,

$$G(m) = \frac{1}{N_S} \sum_{i=1}^{N_S} |\mathbf{V}_i^* (m) - \mathbf{V}_i|^2$$

(3.46)

where $\mathbf{V}_i^* (m)$ is the model reconstruction of $\mathbf{V}_i$ using $m$ eigenmodes and $\mathbf{V}_i$ is the excluded training instance. The standard error of $G(m)$ is defined as $\sigma_G(m) = \sigma / \sqrt{N_S - 1}$, with $\sigma$ being the sample standard deviation of $G(m)$ [68].

3.3 Results and Discussion

In the first set of experiments, we were interested in identifying the best combination of the cost function (CF) and $\lambda_{cut}$ parameter. According to the plots shown in Figures 3.8-3.15, out of the thee quality measures, generalization ability is the least sensitive to the

\(^{34}\) $m=0$ corresponds to the model average.
choice of $\lambda_{cut}$. Furthermore, in comparison to Davies's CF (3.15), the quality measures of the SSMs based on correspondences determined with Thodberg's CF (3.18) appear to be more stable with respect to $\lambda_{cut}$. One can see that on average, Thodberg’s CF produced slightly more compact models than Davies' CF. One possible explanations for the latter observation is that $\lambda_{cut}$ defined using (3.43) is orders of magnitude smaller than the one defined using (3.42). To understand the significance of this difference, recall that (a) $\lambda_{cut}$ marks the transition between the determinant-like term and the trace of the covariance matrix and (b) minimizing the determinant of the covariance matrix tends to minimize the hyper-volume spanned by the eigenmodes of the SSM in the shape space. Consequently, reducing the value of $\lambda_{cut}$ should also reduce the total variance of the model as more eigenmodes are included in the determinant-like part of the CF. Regardless of the exact cause for the minor differences in the total variance, both CFs produced models with very similar generalization, specificity and compactness properties for different settings of $\lambda_{cut}$. In this study, all final SSMs were based on correspondences established with Thodberg's CF and $\lambda_{cut}$ computed with (3.43).
Figure 3.8: Scaphoid MDL parameter selection. $\lambda_{cut}$ has no noticeable effect on quality of the SSMs optimized with either cost function (CF). Davies' CF produced slightly more compact model with smallest cumulative variance of 90.57, compared to 92.38 for Thodberg's CF. Davies: $\lambda_{cut} = 0.04$. Thodberg: $\lambda_{cut} = 3.78 \cdot 10^{-5}$. 
Figure 3.9: Lunate MDL parameter selection. $\lambda_{cut}$ has no noticeable effect on the quality of SSM optimized with either cost function (CF). Thodberg’s CF produced slightly more compact model with smallest cumulative variance of 27.79, compared to 29.92 for Davies’ CF. Davies: $\lambda_{cut} = 0.04$. Thodberg: $\lambda_{cut} = 8.73 \cdot 10^{-5}$.
Figure 3.10: Triquetrum MDL parameter selection. $\lambda_{cut}$ has significant effect on specificity and compactness of a SSM optimized with Davies' cost function (CF). Thodberg's CF produced slightly more compact model with smallest cumulative variance of 23.38, compared to 25.41 for Davies' CF. Davies: $\lambda_{cut} = 0.04$. Thodberg: $\lambda_{cut} = 10.54 \cdot 10^{-5}$. 
Figure 3.11: Pisiform MDL parameter selection. $\lambda_{cut}$ has significant effect on the quality of SSM optimized with Davies' cost function (CF) and minor effect on specificity and compactness of SSM optimized with Thodberg's CF. Thodberg’s CF produced slightly more compact model with smallest cumulative variance of 5.38, compared to 5.60 for Davies’ CF. Davies: $\lambda_{cut} = 0.04$. Thodberg: $\lambda_{cut} = 23.71 \cdot 10^{-5}$. 
Figure 3.12: Trapezoid MDL parameter selection. $\lambda_{\text{cut}}$ has significant effect on specificity and compactness of a SSM optimized with Davies’ cost function (CF). Thodberg’s CF produced slightly more compact model with smallest cumulative variance of 27.81, compared to 28.45 for Davies’ CF. Davies: $\lambda_{\text{cut}} = 0.04$. Thodberg: $\lambda_{\text{cut}} = 8.72 \cdot 10^{-5}$. 
Figure 3.13: Trapezium MDL parameter selection. The quality of the SSMs produced with either cost function (CF) is only slightly affected by the choice of $\lambda_{\text{cut}}$. The smallest cumulative variance of SSMs for Davies’ CF is 30.67 and 29.82 for Thodberg’s CF. Davies: $\lambda_{\text{cut}} = 0.04$. Thodberg: $\lambda_{\text{cut}} = 7.07 \cdot 10^{-5}$. 
Figure 3.14: Capitate MDL parameter selection. \( \lambda_{cut} \) has no noticeable effect on the quality of SSM optimized with either cost function (CF). Thodberg’s CF produced slightly more compact model with smallest cumulative variance of 57.29 compared to 60.31 for Davies’ CF. Davies: \( \lambda_{cut} = 0.04 \). Thodberg: \( \lambda_{cut} = 5.56 \cdot 10^{-5} \).
Figure 3.15: Hamate MDL parameter selection. $\lambda_{\text{cut}}$ has minor effect on the specificity and compactness of SSM optimized with Davies' cost function (CF). The models obtained with Thodberg's CF are not sensitive to the choice of $\lambda_{\text{cut}}$ parameter. Davies' CF produced slightly more compact model with smallest cumulative variance of 47.53, compared to 50.56 for Thodberg's CF. Davies: $\lambda_{\text{cut}} = 0.04$. Thodberg: $\lambda_{\text{cut}} = 6.40 \cdot 10^{-5}$. 
Having identified the appropriate cost function for establishing correspondence, we proceeded to create the SSMs of the individual carpal bones. Figures below show the extremes of the first four principal modes of variation of the final SSMs as well as the changes in SSM quality measures during different stages of correspondence optimization. Table 3.3 provides a summary of the optimization progress.

It is interesting to note that during the final, groupwise correspondence optimization (GCO) stage, minimization of the DL using an increasingly larger number of landmarks did not considerably improve the SSM quality. This suggests that DL could have been minimized with a much smaller set of landmarks than the one used to construct the final SSM. Since GCO is the most expensive procedure, by eliminating the 2nd and 3rd GCO passes, the total execution time could be reduced by approximately 56% (on average). This observation, however, does not apply to all carpal bones, with pisiform being the sole exception.

Of the eight carpal bones, pisiform was the smallest, most pebble-like bone that had the least amount of salient features. Following optimization of pisiforms' correspondences we observed a somewhat unusual trend. For example, the pattern of quality improvement shown in Figure 3.18 is intuitive and very similar to all carpal SSMs (other than pisiform). According to this trend one would expect progressive improvement in the SSM quality after completion of consecutive correspondence optimization stages. The trend shown in Figure 3.27, however, is very different. According to these results, the most compact SSM was not a SSM with the lowest specificity and generalization ability, thus

---

35 All operations described in the Methods section were performed in MATLAB running on a desktop with 6 GB RAM and 2.80 GHz x6 AMD processor.
36 Recall that initial landmark mesh in the parameter domain was subdivided at the beginning of the 2nd GCO pass and then once more during the 3rd pass (see Section 3.2.3).
contradicting the notion that the most compact models necessarily have the lowest generalization and specificity scores. Although at first this observation may appear counterintuitive, it is nonetheless fully consistent with the formulation of the DL cost function. To understand how this is possible, recall that correspondence was established by minimizing

\[ L_T = \sum_{m=0}^{M-1} \left\{ \ln \left( \frac{\lambda_m}{\lambda_{\text{cut}}} \right) + 1 \right\} , \lambda_m \geq \lambda_{\text{cut}} \]

\[ , \lambda_m < \lambda_{\text{cut}} \]  

Assuming that \( \lambda_m \geq \lambda_{\text{cut}} \) for all \( m \) then (3.47) can be rewritten as

\[ L_T = (M - 1) \left( 1 - \ln(\lambda_{\text{cut}}) \right) + \ln \left( \prod_{m}^{M-1} \lambda_m \right) \]

Since \((M - 1) \left( 1 - \ln(\lambda_{\text{cut}}) \right)\) is constant, only the rightmost term\(^{37}\) is minimized during correspondence optimization. By contrast, total SSM variance is defined as \( \sum_{m=1}^{M-1} \lambda_m \); an objective measure very different from \( \prod_{m}^{M-1} \lambda_m \). The optimization performed during the Möbius transformation normalization (MTN) stage, implicitly minimized \( \sum_{m=1}^{M-1} \lambda_m \).

Conversely, as shown below, under certain conditions, optimization of \( L_T \) that follows the MTN stage could decrease \( L_T \) while simultaneously increasing \( \sum_{m=1}^{M-1} \lambda_m \).

Let \( \lambda_i \) and \( \lambda_{i-1} \) be the SSM eigenvalues such that \( \lambda_i > \lambda_{i-1} > \lambda_{\text{cut}} \). Define \( V_k = \prod_{m=1}^{M-1} \lambda_m \) and \( C_k = \sum_{m=1}^{M-1} \lambda_m \). Suppose that during groupwise correspondence optimization (GCO), \( \lambda_i \) and \( \lambda_{i-1} \) are modified by \( \Delta_i \) and \( \Delta_{i-1} \), respectively, while all other eigenvalues remain approximately the same. Therefore

\(^{37}\) \( \prod_{m}^{M-1} \lambda_m \) equals the square of the hyper-volume spanned by the SSM eigenmodes in shape space.
\[ V_{k+1} = \frac{V_k}{\lambda_i \lambda_{i-1}} (\lambda_i + \Delta_i)(\lambda_{i-1} + \Delta_{i-1}) \]  

(a)

\[ C_{k+1} = C_k + \Delta_i + \Delta_{i-1} \]  

(b)

Since \( V \) is known to decrease from one iteration to the next, we have \( V_{k+1} < V_k \), which implies that

\[
(\lambda_i + \Delta_i)(\lambda_{i-1} + \Delta_{i-1}) < \lambda_i \lambda_{i-1} \\
\lambda_i \Delta_{i-1} + \lambda_{i-1} \Delta_i < -\Delta_i \Delta_{i-1}
\]  

(c)

For \(|\Delta_{i-1}| \ll \lambda_{i-1} \Rightarrow \Delta_i \Delta_{i-1} \approx 0\), so (c) can be simplified to

\[ \Delta_i < -\frac{\lambda_i}{\lambda_{i-1}} \Delta_{i-1} \]  

(d)

From (b), when \( C_{k+1} > C_k \) we have

\[ \Delta_i > -\Delta_{i-1} \]  

(e)

Because \( \frac{\lambda_i}{\lambda_{i-1}} > 1 \), both \( V_{k+1} < V_k \) and \( C_{k+1} > C_k \) are satisfied for \( \Delta_{i-1} < 0 \) on the interval

\[ -\Delta_{i-1} < \Delta_i < -\frac{\lambda_i}{\lambda_{i-1}} \Delta_{i-1} \]  

(f)

When this condition is met, in the process of GCO, with an increasing number of iterations, the smallest SSM modes become smaller, while the influence of already dominating eigenmodes becomes even greater (because the decrease in \( \lambda_{i-1} \) is offset by a larger increase in \( \lambda_i \)). This result is supported by the plot of pisiform's compactness in Figure 3.27 which shows that as correspondence optimization transitions from the MTN to the GCO stage, compactness increases as more variance becomes “packed” into the principal modes. The observed increase in compactness, however, does not necessitate concurrent increase in the DL. As shown in Table 3.3, pisiform's DL decreased steadily from one correspondence optimization stage to the next.
Figure 3.16: Extremes of the first four principal components of the scaphoid SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.17: Colormap of local variations described by the first four modes of the scaphoid SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.18: Changes in the scaphoid SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.19: Extremes of the first four principal components of the lunate SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.20: Colormap of local variations described by the first four modes of the lunate SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.21: Changes in the lunate SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2nd and 3rd GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.22: Extremes of the first four principal components of the triquetrum SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.23: Colormap of local variations described by the first four modes of the triquetrum SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.24: Changes in the triquetrum SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.25: Extremes of the first four principal components of the pisiform SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.26: Colormap of local variations described by the first four modes of the pisiform SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.27: Changes in the pisiform SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2nd and 3rd GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.28: Extremes of the first four principal components of the trapezoid SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.29: Colormap of local variations described by the first four modes of the trapezoid SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.30: Changes in the trapezoid SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.31: Extremes of the first four principal components of the trapezium SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.32: Colormap of local variations described by the first four modes of the trapezium SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.33: Changes in the trapezium SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.34: Extremes of the first four principal components of the capitate SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.35: Colormap of local variations described by the first four modes of the capitate SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.36: Changes in the capitate SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Figure 3.37: Extremes of the first four principal components of the hamate SSM. The amount of variance captured by each mode is given on the right as a percentage of the total variance found in the training set.

Figure 3.38: Colormap of local variations described by the first four modes of the hamate SSM superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 3.39: Changes in the hamate SSM quality measures during the course of correspondence optimization. LR = linear registration. MTN = Möbius transformation normalization. GCO = groupwise correspondence optimization. At the beginning of the 2\textsuperscript{nd} and 3\textsuperscript{rd} GCO passes the number of landmarks used in optimization of the DL was approximately quadrupled (using triangular quadrisection subdivision scheme described in subsection 3.2.5).
Table 3.3: Surface correspondence optimization summary. LR = Pairwise Linear Registration (subsection 3.2.1). MNT = Möbius Transformation Normalization (subsection 3.2.2). GCO = Groupwise Correspondence Optimization (subsection 3.3.3). All times are given in minutes. For the LR stage, the times include computation of the distance transforms (also used in Chapter 4) and optimization of similarity transformations (~ 51 min per training set). At every stage, the description length (DL) was computed using $N_p$ landmarks given in the last row.

<table>
<thead>
<tr>
<th></th>
<th>Sca</th>
<th>Lun</th>
<th>Trq</th>
<th>Pis</th>
<th>Tpd</th>
<th>Tpm</th>
<th>Cap</th>
<th>Ham</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DL</td>
<td>400.80</td>
<td>366.03</td>
<td>382.20</td>
<td>331.09</td>
<td>385.11</td>
<td>407.80</td>
<td>421.14</td>
<td>397.81</td>
</tr>
<tr>
<td>Time</td>
<td>423.13</td>
<td>281.62</td>
<td>254.82</td>
<td>147.77</td>
<td>257.70</td>
<td>330.78</td>
<td>478.60</td>
<td>455.07</td>
</tr>
<tr>
<td>MTN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DL</td>
<td>389.84</td>
<td>358.79</td>
<td>374.88</td>
<td>324.61</td>
<td>376.40</td>
<td>397.18</td>
<td>410.54</td>
<td>389.28</td>
</tr>
<tr>
<td>Time</td>
<td>26.46</td>
<td>15.59</td>
<td>10.04</td>
<td>4.49</td>
<td>13.06</td>
<td>13.83</td>
<td>22.61</td>
<td>22.89</td>
</tr>
<tr>
<td>GCO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pass1</td>
<td>DL</td>
<td>366.93</td>
<td>344.49</td>
<td>367.64</td>
<td>321.22</td>
<td>364.38</td>
<td>383.14</td>
<td>395.90</td>
</tr>
<tr>
<td>Time</td>
<td>817.21</td>
<td>553.17</td>
<td>415.77</td>
<td>218.84</td>
<td>464.10</td>
<td>551.56</td>
<td>736.40</td>
<td>677.16</td>
</tr>
<tr>
<td>GCO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pass2</td>
<td>DL</td>
<td>366.28</td>
<td>343.62</td>
<td>365.06</td>
<td>317.28</td>
<td>362.60</td>
<td>381.97</td>
<td>395.20</td>
</tr>
<tr>
<td>Time</td>
<td>616.38</td>
<td>478.56</td>
<td>452.47</td>
<td>303.90</td>
<td>467.90</td>
<td>612.77</td>
<td>526.30</td>
<td>665.66</td>
</tr>
<tr>
<td>GCO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pass3</td>
<td>DL</td>
<td>365.97</td>
<td>343.33</td>
<td>364.54</td>
<td>314.64</td>
<td>362.09</td>
<td>381.64</td>
<td>394.96</td>
</tr>
<tr>
<td>Time</td>
<td>1291.39</td>
<td>540.80</td>
<td>402.77</td>
<td>252.44</td>
<td>536.58</td>
<td>600.33</td>
<td>643.93</td>
<td>915.07</td>
</tr>
<tr>
<td>$N_p$</td>
<td>4690</td>
<td>3122</td>
<td>2466</td>
<td>1010</td>
<td>2722</td>
<td>3266</td>
<td>4290</td>
<td>3826</td>
</tr>
</tbody>
</table>

The SSM quality tests performed in this study were based on the hypothesis that the distribution of shape parameters for all carpal bones types was Gaussian (see subsection 3.1.2). Accordingly, shape parameters, $\{b_m\}$, of the linear SSM in (3.3) were constrained to the interval $b_m \in [-3\sigma_m, +3\sigma_m]$, where $\sigma_m$ is the standard deviation of the $m$-th mode. The actual distributions of shape parameters for the first twelve SSM modes are shown in Figures 3.40-3.47.

\[
V = \bar{V} + \sum_{m=1}^{k} b_m p_m, \quad 1 \leq k \leq M - 1
\]  

(3.3)

Following van de Giessen et al. [52], we performed a series of Kolmogorov-Smirnov tests\(^{38}\) [107] to estimate the probability that shape parameter distributions were Gaussian. The results of these tests are summarized in Table 3.4. Out of 96 tests, the null hypothesis

\(^{38}\)Kolmogorov-Smirnov test was performed with 'kstest' function from MATLAB's Statistics Toolbox.
was rejected only in four instances (at the 0.05 significance level), thus providing a good measure of support for the linear model and suggesting that more general, nonlinear methods of estimating shape parameter distributions (such as Gaussian mixture modeling, kernel density estimations, kernel PCA or independent component analysis [60,68]) may not be necessary to generate valid instances of the carpal bones.

Table 3.4: p-values of the Kolmogorov-Smirnov test used to verify if the distributions of shape parameters were significantly different from the normal distribution. The highlighted cells indicate the SSM modes with p-values below the 0.05 significance level. Sca = scaphoid, Lun = lunate, Trq = triquetrum, Pis = pisiform, Tpd = trapezium, Tpm = trapezoid, Cap = capitate, Ham = hamate.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Sca</th>
<th>Lun</th>
<th>Trq</th>
<th>Pis</th>
<th>Tpd</th>
<th>Tpm</th>
<th>Cap</th>
<th>Ham</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.66</td>
<td>0.88</td>
<td>0.59</td>
<td>1.00</td>
<td>0.98</td>
<td>0.72</td>
<td>0.70</td>
<td>0.14</td>
</tr>
<tr>
<td>2</td>
<td>0.89</td>
<td>0.20</td>
<td>0.62</td>
<td>0.95</td>
<td>0.86</td>
<td><strong>0.03</strong></td>
<td>0.64</td>
<td>0.85</td>
</tr>
<tr>
<td>3</td>
<td>0.25</td>
<td>0.97</td>
<td>0.13</td>
<td>0.87</td>
<td>0.95</td>
<td>0.49</td>
<td>0.34</td>
<td>0.17</td>
</tr>
<tr>
<td>4</td>
<td>0.88</td>
<td>0.41</td>
<td>0.18</td>
<td>0.14</td>
<td>0.22</td>
<td>0.57</td>
<td>0.66</td>
<td>0.67</td>
</tr>
<tr>
<td>5</td>
<td>0.46</td>
<td>0.85</td>
<td>0.16</td>
<td>0.08</td>
<td>0.29</td>
<td>0.63</td>
<td>0.85</td>
<td>0.17</td>
</tr>
<tr>
<td>6</td>
<td><strong>0.00</strong></td>
<td>0.81</td>
<td>0.90</td>
<td>0.24</td>
<td>0.29</td>
<td>0.39</td>
<td>0.08</td>
<td>0.48</td>
</tr>
<tr>
<td>7</td>
<td>0.21</td>
<td>0.89</td>
<td>0.79</td>
<td>0.05</td>
<td>0.92</td>
<td>0.98</td>
<td>0.96</td>
<td>0.44</td>
</tr>
<tr>
<td>8</td>
<td>0.49</td>
<td>0.85</td>
<td>0.52</td>
<td>0.73</td>
<td>0.08</td>
<td>0.93</td>
<td>0.20</td>
<td>0.45</td>
</tr>
<tr>
<td>9</td>
<td>0.19</td>
<td>0.10</td>
<td>0.07</td>
<td>0.67</td>
<td>0.83</td>
<td>0.95</td>
<td>0.89</td>
<td>0.50</td>
</tr>
<tr>
<td>10</td>
<td><strong>0.02</strong></td>
<td>0.38</td>
<td>0.78</td>
<td>0.83</td>
<td>0.92</td>
<td><strong>0.01</strong></td>
<td>0.44</td>
<td>0.07</td>
</tr>
<tr>
<td>11</td>
<td>0.28</td>
<td>0.18</td>
<td>0.53</td>
<td>0.13</td>
<td>0.82</td>
<td>0.82</td>
<td>0.26</td>
<td>0.42</td>
</tr>
<tr>
<td>12</td>
<td>0.08</td>
<td>0.49</td>
<td>0.88</td>
<td>0.41</td>
<td>0.90</td>
<td>0.85</td>
<td>0.97</td>
<td>0.90</td>
</tr>
</tbody>
</table>
Figure 3.40: Distribution of shape parameters for the first twelve modes of scaphoid SSM. Taken together these modes describe 94.32% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.41: Distribution of shape parameters for the first twelve modes of lunate SSM. Taken together these modes describe 90.60% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.42: Distribution of shape parameters for the first twelve modes of triquetrum SSM. Taken together these modes describe 91.64% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.43: Distribution of shape parameters for the first twelve modes of pisiform SSM. Taken together these modes describe 93.44% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.44: Distribution of shape parameters for the first twelve modes of trapezoid SSM. Taken together these modes describe 90.61% of variations present in the training set composed of 29 samples. SND = standard normal distribution.
Figure 3.45: Distribution of shape parameters for the first twelve modes of trapezium SSM. Taken together these modes describe 87.36% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.46: Distribution of shape parameters for the first twelve modes of capitate SSM. Taken together these modes describe 90.60% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
Figure 3.47: Distribution of shape parameters for the first twelve modes of hamate SSM. Taken together these modes describe 89.24% of variations present in the training set composed of 30 samples. SND = standard normal distribution.
The overall method described in Section 3.2 is essentially the same as the one presented in [89] by Heimann et al. but contains two essential distinctions that make it a) easier to implement and b) more computationally efficient.

a. The first and perhaps the most notable difference is related to the way we initialize landmark positions in the parameter domain. In order to ensure accurate representation of the training shapes, the inverse mapping of landmarks to the primary domain must produce (at least) uniform sampling of the training surfaces. In their original paper [67], Heimann et al. initialized landmark positions with uniform sampling of the parameter domain (i.e. a sphere). Although easy to compute, as shown in Figure 3.48c, after mapping to the primary domain, the resulting meshes may not be accurate representations of the underlying surfaces. Recognizing this limitation, in the follow-up [89] to their original work, Heimann et al. adapted Alliez's remeshing algorithm [38] to generate more appropriate landmark distributions. In Chapter 2 (Section 2.2), we already pointed out that the abovementioned technique (i.e. [38]) is cumbersome to implement because it requires partitioning of the surfaces into separate patches. By contrast, we initialize landmark positions using our new particle-based, adaptive sampling procedure, which was designed to deal specifically with closed, genus-0 surfaces and therefore does not require any partitioning.

b. The second difference stems from our use of the Möbius transformations to provide good initializations to the GCO stage (see subsection 3.2.2). Since the correspondences are established by registering the parameterizations (of training shapes) in an intermediate (parameter) domain, optimization can converge faster.
if the parameter meshes are already well aligned. Therefore, it is very likely that the normalization of the original conformal parameterization with respect the Möbius transformations improved the overall efficiency of our method by reducing the total amount of deformation (and hence the total number of optimization iterations) required to register the parameter meshes. Further investigation, however, is required to quantify the actual amount of the purported improvement.

![Diagram](image)

**Figure 3.48:** a) Conformal parameterization of the hamate mesh. Color denotes the amount of area distortion, with red corresponding to expansion and blue to compression. b) Using the algorithm described in Chapter 2, we can adaptively sample the parameter domain to produce uniform distribution of landmarks in the primary domain. c) Example of how uniform sampling of the parameter domain can lead to poor initialization of landmarks in the primary domain.
The algorithm presented in this chapter is by no means optimal and could be improved further by finding the Möbius transformations that minimize the DL of the training set as opposed to pairwise distances (see subsection 3.2.3) and by exploiting the multi-resolution construction of the training meshes. The latter suggestion is especially promising and could enable substantial performance acceleration. The bottleneck of Heimann's algorithm is the ray-triangle intersection procedure [49] which is used to keep track of landmark positions with respect to the parameter mesh faces. In the worst case, the time complexity of the overall GCO algorithm can be roughly estimated as $O(N_S N_P N_F)$, where $N_S$ is the number of training shapes, $N_P$ is the number of landmarks used in optimize the DL and $N_F$ is the average number of parameter mesh faces. We have already shown that for seven out of eight carpal bones, optimization of the DL using the full set of landmarks is not necessary. As a matter of fact, by only performing one optimization pass for $\sim N_P/16$ landmarks, on average, the GCO run times could be reduced by approximately 67%. Since $N_S$ is fixed, further improvement in performance can be achieved by reducing $N_F$. In the preceding chapter, we pointed out that the meshes resampled with our algorithm, by construction, have subdivision connectivity. This property can be easily exploited by initializing GCO with the base meshes, iterating until convergence, then subdividing the meshes and reinitializing the GCO with the subdivided mesh. Because every subdivision quadruples the number of faces, the triangulations of desirable complexity can be easily recovered in two subdivisions (i.e. three optimization cycles). In theory, implementation of such a multi-resolution optimization scheme should increase the convergence rate of the GCO procedure by a substantial amount. The factor of 16 will likely be unachievable, but even in the worst case, if the run times will be
reduced only by a factor of 2, it will nonetheless be a substantial improvement. We would also like to point out that in addition to accelerated convergence, multiresolution optimization has the propensity to regularize the cost function, thereby improving the chances of gradient-based optimization methods of finding a globally optimal solution [95].

As a last remark, we would like to note that in the future, if we are able to gain access to the CT images from which the training samples were derived, our models could be augmented to include the variability of internal bone density distributions\(^{39}\). The utility of such models would be twofold. Firstly, by combining geometric and material property\(^{40}\) information, we could improve the accuracy of FE simulations, which are usually performed by assigning a single set of material properties to a specific type of tissue [5,6,7,9,11,15]. Secondly, the augmented models could be used to automate the segmentation of medical images [106] of the wrist, thereby facilitating the diagnosis of potential wrist injuries in clinical settings.

### 3.4 Conclusion

The primary objective of the work described in this chapter was to identify and implement a robust method to create statistical shape models (SSMs) of the eight carpal bones. To this end, we built upon the methods and results from Chapter 2 to develop a

\(^{39}\) Of course this would require some way of interpolating the surface correspondences to the interior of the objects. One approach would be to first create a template, tetrahedral mesh of the average shape. In the next step, since the surface point correspondence are known, the template mesh would be mapped to the training shapes using a smooth deformation field based on thin plate splines [69].

\(^{40}\) Nearly all FE studies investigating joint mechanics have modeled bone as a linear elastic, isotropic material [5]. Young's modulus is one of the phenomenological parameters required by these models and can be estimated directly from the CT intensity values [12,105].
three stage procedure that enabled us to construct SSMs with good generalization, specificity and compactness properties. We have given particular attention to the fact that the quality and validity of the resulting models are highly dependent upon the accuracy of the correspondences between the training shapes. As a result, at the final stage, we have chosen to optimize correspondences using a MDL strategy, which in comparison to conventional pairwise registration algorithms provides a more principled and robust approach to establish correspondences between multiple biological shapes. We have also demonstrated that ~90% of the observed anatomical variations could be recovered with a standard, linear SSM.

Considering that all operations described in this chapter were performed in MATLAB, the SSM were still computed within a reasonable amount of time (≈ 1.4 days/model). Future implementations of our method, however, could gain substantial improvement in efficiency by exploiting the multi-resolution construction of the input meshes obtained with the resampling procedure described in Chapter 2. Additionally, further study should extend the carpal SSMs to include the variability of bone density distributions to create the so called statistical appearance models, which could be applied to perform more realistic FE simulations [54-56] and to automate the segmentation of medical images of the wrist [106].
Chapter 4

Multi-body Statistical Shape Model of the Wrist

In Chapter 3 we described a method to construct statistical shape models (SSM) of the individual wrist bones. Although these models were shown to provide an efficient parameterization of shape variability, they did not consider the context in which the modeled objects naturally occur. This is a serious limitation because the function of the wrist depends on the interaction of multiple rigid and soft tissue components [5]. For instance, in a healthy wrist, articulating bones will have congruent articulating surfaces, implying that the local geometry of a given bone will be highly correlated with adjacent bones and vice versa. In the presence of tightly coupled interactions between multiple bones, altering the geometry of a single bone will likely produce a cascading effect through the entire complex of bones. Therefore, it is obvious that the anatomical variations of carpal bones are best understood in the context of the entire wrist and not in isolation. Fortunately, the SSM framework already used to parameterize the shape of individual bones can be trivially generalized to describe the assemblies of multiple objects.

In fact, superior descriptive abilities of multi-body SSMs (also known as composite SSMs [108]) have been exploited in several studies [108-120]. For example, models based on medial representations (m-reps) were used to derive statistics of multi-object configurations of deep brain structures [109-112], pelvic bones [113] and bladder-prostate-rectum complexes [114,115]. Analogous modeling approaches using point-based
(i.e. boundary) representations of internal brain structures were also reported in [116-118]. Despite the difference in shape representations, the composite shape statistics were based on a set of corresponding landmarks (or medial atoms in case of m-reps) normalized with respect to the linear transformations that included rotation, translation and in some applications scaling\textsuperscript{41}. Typically, this requirement is realized by processing the individual constituents of multi-objects complexes independently from the neighbouring structures. Unfortunately, while this approach may be valid for the soft tissue structures of the brain [108-112,118,120] or the heart [119], it is not suitable in case of the tightly packed structure like the wrist.

The wrist can assume a continuous range of postures that can be any combination of flexion/extension and radial/ulnar deviation (see Figure 4.1). When the wrist changes posture, in general, there will be non-negligible relative motions between carpal bones [16,80]. What is more, the bones from two different wrists in the same posture will usually have different spatial relationships [16]. Consequently, applying independent, unconstrained rigid registrations to the individual carpal bones will produce collisions or other physiologically non-feasible configurations [80]. Studies related to joint modeling problems have encountered similar issues [121-127] and developed constrained registration techniques to address them. By far the most relevant solution was described in [80]. Not surprisingly, the latter study was carried out by a research group also interested in constructing a multi-body SSM of the wrist (see [50-52]).

\textsuperscript{41} Pose normalization is a prerequisite (for both single and multi-body models) which ensures that variations captured by the SSM are a function of the anatomical differences alone and not differences in pose.
In the following section, we review the constrained registration method described in [80] by van de Giessen at al.. In Section 2, we identify limitations of van de Giessen's algorithm and in so doing build upon it to develop a more robust procedure that we later employ to create a composite SSM of the carpus. The performance of the original and improved registration algorithms is evaluated and discussed in Section 3. Section 4 concludes this chapter with the summary of the findings.

### 4.1 Previous Work

This section reviews the method, described recently by van de Giessen et al. in [80], for registering two different wrists to a common posture while maintaining physiologically correct conformation of the carpal bones. The registration algorithm is based on the assumption that an anatomically valid arrangement of carpal bones can be inferred from a single set of articulating surfaces (extracted from the wrist in an arbitrary posture) as long as...
as these surfaces remain coincident. To this end, van de Giessen et al. formulated the constrained registration of two different carpuses as an unconstrained optimization problem with a cost function (4.1) that consisted of two terms,

\[
E(s, \{T_i\}) = E_{ext}(s, \{T_i\}) + \alpha E_{int}(s, \{T_i\})
\]  

(4.1)

an external energy \((E_{ext})\) that was minimized when the pose differences in all carpal bones were small, and an internal energy \((E_{int})\) that was minimal when all gliding surfaces between articulating bone pairs were touching. Both terms were balanced by an \(\alpha\) parameter, which determined the influence of the constraints on the final registration that in turn was defined by a global scale parameter, \(s\), and a set of rigid transformations \(\{T_i \mid i = 1, ..., 8\}\) of all carpal bones. The registration was performed using surface meshes as input.

Suppose \(\{M_{Ai}\}\) and \(\{M_{Bi}\}\) are the sets of carpal bone meshes of two different wrists, with individual surfaces defined by triangulation of vertices \(\{a_{ij} \mid j = 1, ..., N_{Ai}\}\) and \(\{b_{ij} \mid j = 1, ..., N_{Bi}\}\), respectively. The objective is to register the source configuration denoted by subscript \(A\) to the reference configuration indentified by subscript \(B\).

Adopting a standard, distance based measures of similarity, van de Giessen et al. defined \(E_{ext}\) as the sum of symmetric registration errors, \(J_{ext}\), of all carpal bones,

\[
E_{ext}(s, \{T_i\}) = \sum_{i=1}^{8} J_{ext}(s, T_i, M_{Ai}, M_{Bi})
\]

(4.2)

\[
J_{ext}(s, T_i, M_{Ai}, M_{Bi}) = \frac{1}{2N_{Ai}} \sum_{j=1}^{N_{Ai}} d(sT_i(a_{ij}), M_{Bi})^2 + \frac{1}{2N_{Bi}} \sum_{j=1}^{N_{Bi}} d(sT_i(M_{Ai}), b_{ij})^2
\]

(4.3)
where \( d(a_j, M_B) \) is the Euclidean distance from vertex \( a_j \) to the closest point on the surface of \( M_B \). \( J_{ext} \) is a heuristic measure of similarity whose minimizer is assumed to optimally align the \( i \)-th carpal bone in the source wrist with the \( i \)-th bone in the reference wrist\(^{42} \).

Following similar principles, van de Giessen \textit{et al.} defined \( E_{int} \) as the sum of physiological penalty terms, \( J_{int} \) from 10 gliding surface pairs (see Figure 4.2) extracted from the source wrist,

\[
E_{int}(s, \{T_i\}) = \sum_{k=1}^{10} J_{int}(s, T_k, U_k, V_k, \delta_k) \quad (4.4)
\]

\[
J_{int}(s, T, U, V, \delta) = \frac{1}{2N_U} \sum_{m=1}^{N_U} |P(s, T(u_m), V, \delta)|^2 + \frac{1}{2N_V} \sum_{m=1}^{N_V} |P(s, T(U), v_m, \delta)|^2 \quad (4.5)
\]

\[
P(s, T(u_m), V, \delta) = \begin{cases} -\delta \cdot \ln \left( 1 - \frac{d(sT(u_m), V)}{s\delta} \right) & , d(\quad) < s\delta \\ \infty & , otherwise \end{cases} \quad (4.6)
\]

where \( \{(U_k, V_k) \mid k = 1, \ldots, 10\} \) are duplicate copies of the same gliding surface interposed between a pair of articulating bones (e.g. Sca-Lun as in Figure 4.2), such that \( U_k \) is associated to only one of these bones and \( V_k \) is assigned to the other. \( T_k \) is the rigid transformation of the bone to which the source gliding surface, \( U_k \), is attached. The variables, \( u_m \) and \( v_m \), denote the vertices of \( U \) and \( V \). It is important to note that unlike \( J_{ext} \), which is based entirely on the sum of squared distances, \( J_{int} \) is based on the sum of

\(^{42}\) Not coincidentally, in Chapter 3 (subsection 3.2.1.1), we already minimized \( J_{ext}(\quad) \) to establish preliminary alignment between the individual carpal bones.
logarithmic barrier functions, $P(\cdot)$, defined in (4.6), that increase to infinity when the closest distance, $d(\cdot)$, between the gliding surfaces approaches $\delta$. $\delta$ is an empirically determined threshold that takes on different values for every arthrodial (i.e. gliding) joint. The normalized values of $\delta$, as reported in [80], are listed in Table 4.1 for 10 out of 11 intra-wrist joints. Finally, the variable denoted $\bar{N}_U$ in (4.5) is the total number of vertices, $\mathbf{u}_m$, from $U$ that have been paired with non-boundary points in $V$ (see Figure 4.3). $\bar{N}_V$ is defined similarly.

\begin{table}
\centering
\begin{tabular}{|c|c|c|}
\hline
Joint & $\bar{\delta}$ & Joint & $\bar{\delta}$ \\
\hline
Sca-Lun & 0.14 & Trq-Pis & 0.27 \\
Sca-Tpd & 0.15 & Trq-Ham & 0.42 \\
Sca-Cap & 0.14 & Tpd-Tpm & 0.08 \\
Lun-Trq & 0.10 & Tpd-Cap & 0.12 \\
Lun-Cap & 0.10 & Cap-Ham & 0.17 \\
\hline
\end{tabular}
\caption{List of normalized $\delta$ values [80], where $\bar{\delta} = \delta/(AGT)$. AGT stands for average gap thickness and equals twice the mean distance between the gliding surface and the cortical surface of one of the articulating bones. See Figure 4.2 for bone abbreviations.}
\end{table}

\footnote{For some reason scaphoid-trapezium articulation was not considered in [80]. In this study we use $\bar{\delta} = 0.3$ for the scaphoid-trapezium joint.}
Figure 4.2: Graph representation of inter-carpal articulations. Sca = scaphoid, Lun = lunate, Trq = triquetrum, Pis = pisiform, Tpm = trapezium, Tpd = trapezoid, Cap = capitate, Ham = hamate. Entry denoted by ‘o’ only occurs for type II lunate [128]. Lunate-hamate articulations were not considered in this study as 22=2x11 other gliding surfaces provided sufficient constraints to register the wrists.

<table>
<thead>
<tr>
<th></th>
<th>Sca</th>
<th>Lun</th>
<th>Trq</th>
<th>Pis</th>
<th>Tpm</th>
<th>Tpd</th>
<th>Cap</th>
<th>Ham</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sca</td>
<td>XX</td>
<td>XX</td>
<td></td>
<td>XXX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lun</td>
<td>XX</td>
<td>XX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trq</td>
<td>XX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tpm</td>
<td>XX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tpd</td>
<td>XX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cap</td>
<td>XX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ham</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.3: Cross-sections of two articulating surfaces, U and W. Both U and W are open surfaces. Consequently, during registration, multiple vertices from U may be matched to the boundaries on W (and vice versa). To avoid this problem, all vertices matched to the boundary points must be discarded (dashed lines). Only matches shown by solid arrows should used to evaluate $J_{int}$. 
The expression of $J_{int}$ given in (4.5) is flawed because it penalizes the deviations of gliding surfaces (and the bones to which they are attached) from their original poses. Although this may be an alternative approach to enforce physiological conformation of the wrist bones, it is not consistent with the constraint mechanism intended by the authors in [80]. The correct formulation of $J_{int}$ should minimize relative distances between the gliding surfaces and would therefore be expressed as

$$\tilde{J}_{int}(s, T_U, T_V, U, V, \delta) = \frac{1}{2N_U} \sum_{m=1}^{N_U} |P(s, T_U(u_m), T_V(V), \delta)|^2 + \frac{1}{2N_V} \sum_{m=1}^{N_V} |P(s, T_U(U), T_V(v_m), \delta)|^2$$

(4.7)

where $P(s, T_U(u_m), T_V(V), \delta)$ is defined similarly.

According to the revised definition, $J_{int}$ is invariant to the global similarity transformations because

$$d(sT_U(u_m), sT_V(V))/s = d(T_U^{-1}T_U(u_m), V) = d(u_m, V) = d(u_m, U) = 0$$

so naturally, in the absence of the relative movement between $U$ and $V$ (i.e. $T_U = T_V$), all of the internal penalties will remain at zero. Despite this modification, the overall method described above still contains some shortcomings that diminish its capacity in obtaining accurate registration results. In the following section, we identify and address the major
limitations of van de Giessen's algorithm and in so doing develop an alternative method to perform constrained registration of multiple wrists.

4.2 Methods

The primary objective of the present study was to create a multi-body SSM of the wrist. Prior to SSM construction, however, the training wrists must be preprocessed to eliminate unimportant postural information while maintaining a physiologically valid conformation of carpal bones. In the preceding section we reviewed a registration method designed by van de Giessen et al. [80] to deal specifically with this type of a problem. Therein we also mentioned that the method contains some weak points that compromise its ability to obtain good registration results. In what follows, we describe these limitations in some detail and in the process flesh out, what we later show to be, a more robust registration scheme.

First, we would like to draw attention to the fact that van de Giessen et al. approached the problem of constrained registration of the wrist as an unconstrained optimization problem. In fact, their method belongs the family of the so called penalty methods [129,130] that attempt to determine solutions of constrained optimization problems (COP) using unconstrained optimization techniques. The difference in the two formulations can be seen by comparing (4.9) with (4.10),

\[
\begin{align*}
\text{Constrained Optimization Problem} \\
\text{minimize} & \quad f(\theta) \\
\text{subject to} & \quad C_i(\theta) \leq 0
\end{align*} \quad (4.9)
\]

\[
\begin{align*}
\text{Penalty Method Approach to Solve COP} \\
\text{minimize} & \quad f(\theta) + \alpha \sum_i P(\theta, C_i)
\end{align*} \quad (4.10)
\]
where $f(\theta)$ is the objective function, $\{C_i(\theta) \leq 0\}$ is a set of constraints, $P(\theta, C_i) \in \mathbb{R}^+$ is a barrier function that takes on large values when constraints are violated and $\alpha$ is the parameter that determines the influence of the constraints. It is well known that the solution obtained with (4.10) is very sensitive to the choice of the $\alpha$ parameter; a limitation usually overcome with sequential unconstrained minimization technique (SUMT) [129]. The basic idea of SUMT is to initialize the search to $\theta$ that violates at least one constraint and then optimize (4.10) with an arbitrarily small value of $\alpha$. In the next iteration, $\alpha$ is increased (e.g. by a factor of 2) and the previous solution is used to reinitialize the search. This sequence of steps is repeated until $\theta$ converges to a stationary point ($\theta^*$). Interestingly enough, despite having themselves demonstrated the influence of $\alpha$ on the final solution, van de Giessen et al. did not use SUMT. Instead, the authors chose to optimize (4.1) only once (note (4.1) and (4.10) are equivalent), for a single value of $\alpha$. What is more, van de Giessen et al. minimized (4.1) with the Levenberg-Marquardt algorithm (LMA), which is not appropriate (though often used) for optimization of non-smooth and non-differentiable functions [130]. Fortunately, the above mentioned limitations can all be addressed by recasting (4.1) into a constrained optimization framework. Such a change, however, would necessitate further revision of the current definition of physiological constraints (see equation (4.7)). To understand why, consider that on one hand, the number of constraints (expressed in terms of penalties, $P(\ )$) varies with the number of points ($N_U$ and $N_V$) contained in the overlapping regions of the

---

44 When the minimizer of (4.1) is situated close to the constraint boundary, LMA will have difficulty converging on the optimal solution because in those regions the barrier functions will have very large and unstable gradients [133]. Also note that $P(\ )$ is non-differentiable when $d(\ ) \geq \delta$ (4.6).

45 a) Sensitivity of the solution to $\alpha$ parameter, b) susceptibility of LMA to be trapped in the local minima and c) inability to find solutions close to or on the constraint boundary.
gliding surfaces \((U\text{ and } V)\), while on the other hand, standard constrained optimization techniques require the constraint functions (as well as their number) to remain the same in the course of the entire optimization procedure. Therefore, to reformulate (4.1) as a COP, we would have to substitute the dynamically varying constraint functions with more "permanent" measures. Assuming for a moment that \(\{C_p(\cdot) \mid p = 1, \ldots, N_C\}\) is a set of such constraints, where \(C_p(\cdot)\) is a constraint function associated with the \(p\)-th arthrodial joint, then (4.1) can be rewritten as

\[
\{s^*, T^*_i\}_{i=1}^{8} = \text{argmin}_{s \in \mathbb{R}, T_i \in SE(3)} E_{\text{ext}}(s, \{T_i\})
\]

subject to \(C_p(s, \{T_i\}) \leq 0 \ \forall \ p \in [1,11]\)

which brings us to the second limitation of van de Giessen's registration algorithm.

In [80], van de Giessen et al. assumed that successive minimization of pairwise distances (PD) will produce correct correspondences that could in turn be used to determine the optimal similarity transformation to align the wrist bones within a common frame of reference. As shown in the preceding chapter, most carpal bones have isolated surface patches where the relative positions of homologous points vary substantially over the training set (see Figures 3.17, 3.20, 3.29, 3.32, 3.35, 3.38). In these cases, as illustrated in Figure 3.2, PDs provide an inadequate measure of correspondence and thus may lead to inaccurate inference of similarity transformations. Although true correspondences between biological shapes are generally not known [58,60], as explained in the preceding chapter, optimization of the SSM description length (DL) provides a more principled and robust approach to estimate them. Ideally then, the transformations of all carpal bones, in all sample wrists should be determined by minimizing the DL of the composite SSM. To be more precise, let \(S = \{S_w \mid w = 1, \ldots, N_S\}\) be a training set composed of \(N_S\) sample
wrist, where $S_w = \{M_{wi} | i = 1, ..., 8\}$ is a set composed of eight carpal bone meshes in original poses. Additionally, suppose each bone mesh is defined by the triangulation of vertices $V_{wi} = \{v_{wik} | k = 1, ..., N_i\}$, such that the points $v_{mik}$ and $v_{nik}$ for all $m, n \in [1, N_3]$ are corresponding to the optimal transformations should be obtained from

$$\{s^*_w, T^*_w\}_{w=1, i=1}^{N_S, 8} = \arg\min_{s_w \in \mathbb{R}, T_{wi} \in SE(3)} \mathcal{L}(S, \{s_w T_{wi}\})$$

subject to $C_{wp}(s_w, \{T_{wi}\}) \leq 0 \ \forall \ w \in [1, N_3], p \in [1, N_c]$

where $\mathcal{L}$ is the description length of the composite SSM (see Chapter 3),

$$\mathcal{L}(S, \{s_w T_{wi}\}) = \sum_{w}^{N_S-1} \left\{ \ln \left( \frac{\lambda_w(S, \{s_w T_{wi}\})}{\lambda_{cut}} \right) + 1, \lambda_w \geq \lambda_{cut} \right\}$$

Unfortunately (4.12) is very difficult to solve because it is extremely nonlinear, has high dimensionality (there are a total of $49 \times N_S$ unknowns) and contains a large number of constraints ($N_c \times N_S$). To make this problem more tractable, we decompose it into $N_S$ independent, constrained optimization sub-problems of the same form as (4.11)

$$\{s^*_w, T^*_w\}_{i=1}^{8} = \arg\min_{s \in \mathbb{R}, T_{wi} \in SE(3)} \bar{E}_{ext}(V_{wi}, T^R_{wi}, s_w T_{wi})$$

subject to $C_{wp}(s_w, \{T_{wi}\}) \leq 0 \ \forall \ p \in [1, N_c]$

$$\bar{E}_{ext}(V_{wi}, T^R_{wi}, s_w T_{wi}) = \sum_{i=1}^{8} \left( \frac{1}{N_i} \sum_{k=1}^{N_i} \|T^R_{wi}(v_{wik}) - s_w T_{wi}(v_{wik})\|^2 \right)$$

where $\{s^R_w T^R_{wi}\}$ is a set of neutral, reference transformations. Specifically, the elements of $\{s^R_w T^R_{wi}\}$ are computed by sequentially minimizing $\mathcal{L}(S, \{s_w T_{wi}\})$ with respect to the

---

46 The correspondences between the individual carpal bones were established in Chapter 3.

47 For every wrist we seek one global scale parameter and $48 = 8 \times 6$ rigid transformation parameters; there are 8 carpal bones, while the rigid transformation of each bone is described by 3 Euler angles and a 3D translation vector.
scale and rigid transformation parameters of one carpus at a time (constraints are not applied during computation of \( s_w^R T^R_{wl} \)). The choice of sequential optimization is tremendously important for two reasons. First, it avoids degenerate solutions associated with zero scaling factors. Second, by maintaining \( N_S - 1 \) carpal configurations fixed and only optimizing the scale and posture of one wrist at a time, we ensure that the relative positions of carpal bones in all remain as close as possible to the state which is known to be physiologically correct. Although at this point we are unable to provide definitive proof that the transformations obtained in this manner will always minimize \( L(\cdot) \), in the Results section, we provide empirical evidence to validate this approach.

The crucial difference between the cost functions used in (4.11) and (4.14) is that unlike \( E_{ext}(\cdot) \), \( \bar{E}_{ext}(\cdot) \) is computed with respect to the configuration which is known to minimize the DL of the entire training set. Consequently, as we will demonstrate later, in contrast to the pairwise registration approach, solutions obtained via minimization of (4.14) produce models with much better specificity and generalization properties. An additional benefit of our approach is that it does not require selection of a specific reference and is therefore completely unbiased.

### 4.2.1 Constrained Registration of Multiple Wrists

Bringing together the recommendations from above, we can now provide a coherent overview of the actual method used in this study to register multi wrists to the same posture prior building the multi-body SSM. Let \( T = \{W_w \mid w = 1, \ldots, N_S\} \) be a training set composed of \( N_S \) sample wrists, where \( W_w = \{M_{wi} \mid i = 1, \ldots, 8\} \) is a set of 8 carpal bone meshes in original poses. Drawing on the results from Chapter 3, we define each
bone mesh by the triangulation of vertices $V_{ij} = \{v_{ijk} \mid k = 1, \ldots, N_j\}$, such that the points $v_{mjk}$ and $v_{njk}$ for all $m, n \in [1, N_j]$ are corresponding. Since there are substantial pose differences between the wrists, for every wrist, we seek a global scaling parameter, $s^*_i$, and a set of rigid transformations for every carpal bone, $\{T^*_i \mid j = 1, \ldots, B\}$, such that

$$\{s^*_i, T^*_i\}_{i=1}^B \approx \arg \min_{s \in \mathbb{R}, T_i \in SE(3)} \bar{E}_{\text{ext}}(V_{wi}, T^R_{wi}, s_w T_{wi})$$

subject to $C_{wp}(\{T_{wi}\}, \varepsilon_p) \leq 0 \forall p \in [1, N_c]$

$$\bar{E}_{\text{ext}}(V_{wi}, T^R_{wi}, s_w T_{wi}) = \sum_{i=1}^{B} \left( \frac{1}{N_i} \sum_{k=1}^{N_i} \| T^R_{wi}(v_{wik}) - s_w T_{wi}(v_{wik}) \|^2 \right)$$

where $\{s^R_i T^R_{wi}\}$ is a set of neutral, reference transformations computed by sequentially minimizing $\mathcal{L}(S, \{s_w T_{wi}\})$ with respect to the scale and rigid transformation parameters of one carpus at a time. $C_{wp}(\_)$ is a physiological constraint function assigned to the p-th arthrodial joint in the w-th wrist. For example, if $G_{wp}$ is a gliding surface interposed between m-th and n-th carpal bones in the w-th wrist and $V_{wp} = \{g_{wpk} \mid k = 1, \ldots, N_{wp}, g_{wpk} \in G_{wp}\}$ is a set of uniformly sampled points on $G_{wp}$ (see subsection 4.2.1.1), then we define $C_p(\_)$ as

$$C_{wp}(\{T_{wi}\}, \varepsilon_p) = \left\{ \begin{array}{ll} c_{wp}(\{T_{wi}\}, \varepsilon_p), & N_{wpm} + N_{wpn} \geq 2N_{MIN} \\ p, & \text{otherwise} \end{array} \right.$$  

$\left| d \left( T_{wm}(g_{wpr}), T_{wn}(G_{wp}) \right) \right|$

$$c_{wp}(\{T_{wi}\}, \varepsilon_p) = \frac{1}{2\varepsilon_p \Delta_{wp} N_{wp}} \sum_{r=1}^{N_{wp}} \left| d \left( T_{wm}(g_{wpr}), T_{wn}(G_{wp}) \right) \right|$$

$$+ \frac{1}{2\varepsilon_p \Delta_{wp} N_{wp}} \sum_{r=1}^{N_{wp}} \left| d \left( T_{wn}(g_{wpr}), T_{wm}(G_{wp}) \right) \right| - 1$$

$$\Delta_{wp} = \frac{1}{N_{wp}} \sum_{k=1}^{N_{wp}} d(g_{wpk}, M_{wm}) + \frac{1}{N_{wp}} \sum_{k=1}^{N_{wp}} d(g_{wpk}, M_{wn})$$
where \( d\left( T_{wm}(g_{wpr}), T_{wn}(g_{wp}) \right) \) is closest distance between \( T_{wm}(g_{wpr}) \) and \( T_{wn}(g_{wp}) \). The variable \( N_{wpm} \) is the number of points in \( V_{wp} \) that were modified by \( T_{im} \) and matched to the interior of \( T_{wn}(g_{wp}) \). This selection of points is the same as the one used in calculating \( J_{int} \) by van de Giessen et al. (4.5) and is illustrated in Figure 4.3 (see subsection 4.2.1.1 for more details). \( N_{wpn} \) is defined similarly. \( N_{MIN} \) is the smallest number of non-boundary point matches deemed sufficient for reliable evaluation of the constraints (e.g. \( [0.01N_{wp}] \)) and \( P \) is an arbitrarily large positive scalar (e.g. 100) used to indicate that the separation between the gliding surfaces is severely exceeded. Finally, \( \Delta_{wp} \) is the average gap thickness of \( p \)-th arthrodial joint in the \( w \)-th wrist.

To understand the rationale for our choice of (4.16) as a constraint function, consider that Euclidean distances are invariant under rigid transformations. Consequently, (4.17) can be rewritten as

\[
c_{wp}\left(\{T_{wi}\}, e_{p}\right) = \frac{1}{\varepsilon_{p} \Delta_{wp} N_{wpm}} \sum_{r=1}^{N_{wpm}} \left| d\left( T_{wn}^{-1} T_{wn}(g_{wpr}), g_{wp} \right) \right| \\
+ \frac{1}{\varepsilon_{p} \Delta_{ip} N_{wpm}} \sum_{r=1}^{N_{wpm}} \left| d\left( T_{wm}^{-1} T_{wm}(g_{wpr}), g_{wp} \right) \right| - 1
\]  

(4.19)

\( c_{p}(\ ) \) has the lowest possible value of -1 when the intercarpal relationships are unchanged and the relative poses of articulating bones remain in the state know to be physiologically correct. This condition is met when there is no relative displacement between two articulating bones, \( M_{wm} \) and \( M_{wn} \).

\[
d\left( T_{wn}^{-1} T_{wm}(g_{wpr}), g_{wp} \right) = d\left( T_{wm}^{-1} T_{wn}(g_{wpr}), g_{wp} \right) = d\left( g_{wpr}, g_{wp} \right) = 0
\]

which is only possible if \( M_{wm} \) and \( M_{wn} \) move in concert (i.e. \( T_{wm} = T_{wn} \)). For registration purposes, however, this requirement is too strict, so following van de Giessen
et al., we relax it by allowing the joint gap to expand or contract to within $|\epsilon_p|$ percent of its original thickness. In this study we set $\epsilon_p$ to the corresponding $\delta$ values listed in Table 4.1. The actual method used to evaluate $c_p(\ )$ is described in subsection 4.2.1.2.

We would like to emphasize that in contrast to van de Giessen’s approach, our definition of internal constraints is not only invariant to global similarity transformations but is also based on the average displacement between the gliding surfaces. The latter distinction is equally important as the first because in addition to allowing us to solve the constrained registration problem with standard constrained optimization techniques, it is also more regularized and thus more amenable to optimization with gradient based optimization methods. In fact, we use a sequential quadratic programming (SQP) algorithm\(^{49}\) to solve (4.14).

### 4.2.1.1 Gliding Surface Estimation

The gliding surfaces interposed between pairs of articulating bones provide the basis for enforcing internal constraints. In this study, we estimate these surfaces by adapting the procedure described by van de Giessen et al. [80] to our data, which came in the form of triangular surface meshes (as opposed to segmented volumetric images). Additionally, in order to facilitate evaluation of the constraint functions, we pre-compute the distance fields of the medial surfaces (see Algorithm 5) and also use a particle system approach described by Meyer [63] to uniformly sample these surfaces.

---

\(^{48}\) For example, for scaphoid-lunate joint we set $\epsilon_p$ to 0.14. For scaphoid-trapezium joint, for which $\delta$ was not provided in [80], we use $\epsilon_p = 0.3$.

\(^{49}\) SQP algorithm is implemented as ‘fmincon’ function in MATLAB’s Optimization Toolbox.
ALGORITHM 5: Gliding Surface Estimation

1. Suppose $F_A$ and $F_B$ are the signed distance transforms (DTs) of two articulating bones A and B, respectively. Furthermore, suppose $D_A$ and $D_B$ are the corresponding domains on which $F_A$ and $F_B$ are defined. Retrieve parts of $F_A$ and $F_B$ restricted to $D_A \cap D_B$ and call them $\tilde{F}_A$ and $\tilde{F}_B$, respectively. **Note:** The distance transforms of A and B were already computed, in Chapter 3, for the purpose of initial rigid alignment of carpal bones (see subsection 3.2.1). Additionally, as illustrated in Figure 4.4, the gliding surface (GS) interposed between A and B is situated in the intersection of $D_A$ and $D_B$.

![Figure 4.4: Region of intersection of two DT domains.](image)

2. Compute a composite DT of A and B as $F_{AB}(x) = \min\{\tilde{F}_A(x), \tilde{F}_B(x)\}$ for $x \in D_A \cap D_B$.

3. Extract the medial scaffold ($M_{AB}$) interposed between A and B on $D_A \cap D_B$ as the zero level-set of $F_{AB}(x) - \tilde{F}_A(x)$ using MATLAB’s ‘isosurface’ function. $M_{AB}$ is defined by triangulation of the isosurface vertices $V = \{v_k \mid k = 1, \ldots, n\}$.

4. The sought GS is defined by a set of points, $\mathbf{x} \subset V$, that meet the condition in (4.20)

$$\cos^{-1} \left[ \frac{(\mathbf{x} - \mathbf{x}_A) \cdot (\mathbf{x} - \mathbf{x}_B)}{||\mathbf{x} - \mathbf{x}_A|| ||\mathbf{x} - \mathbf{x}_B||} \right] - \pi \leq \Delta \theta$$

(4.20)

where $\mathbf{x}_A$ and $\mathbf{x}_B$ are the closest points to $\mathbf{x}$ on the surfaces of A and B, respectively. $\Delta \theta$ is the angle deviation threshold between the direction vectors $(\mathbf{x} - \mathbf{x}_A)$ and $(\mathbf{x} - \mathbf{x}_B)$. Following van de Giessen et al. we use $\Delta \theta = 5^\circ$. The schematic of this process is illustrated in Figure 4.5. An actual example of intercarpal medial and gliding surfaces is shown in Figure 4.6.
5. The surface meshes extracted in step 4 have highly non-uniform distribution of vertices (see Figure 4.7a). For the purpose of accurate evaluation of the constraints, however, all areas must be represented equally; a requirement which necessitates (approximately) uniform sampling of the surface. In this study, the medial surfaces were resampled with our own implementation of Meyer’s dynamic particle (DP) algorithm [63]. Since this algorithm was designed to work with implicit surfaces as input, $M_{AB}$ must first be converted into the distance transform (DT) representation using the procedure described in Appendix D. Once the DT is known and has been interpolated, the DP routine is initialized with \( N = A \rho \) particles (\( A \) = medial surface area in mm$^2$, \( \rho \) = desired number of particles per 1 mm$^2$), and their positions are iteratively optimized until the system reaches a stable configuration (for example, see Figure 4.7b). We used \( \rho = 16 \) to resample all medial surfaces. Finally, if the GS contains any small and isolated patches, they must be removed. \textbf{Note:} The DTs computed at this stage are also used during optimization of (4.14) to enable efficient evaluation of the constraints (see subsection 4.2.1.2).

![Figure 4.5: Gliding surface (GS) is a subset of the medial surface interposed exactly halfway between the cortical surfaces of two articulating bones.](image-url)
Figure 4.6: Actual example of medial and gliding surfaces interposed between capitate and lunate.

Figure 4.7: a) Unprocessed medial and gliding surfaces (GS) extracted in step 4 using marching cubes algorithm. Note highly irregular distribution of vertices and potential presence of isolated GS patches. b) Medial and gliding surfaces after removal of extraneous patches and resampling with dynamic particle algorithm [63].
4.2.1.2 Constraint Evaluation

Let $M$ and $G$ be medial and gliding surface interposed between carpal bones $A$ and $B$. Define $V = \{g_i | i = 1, ..., N\}$ as a set of uniformly sampled points on $G$. Suppose in the course of optimization, $A$ and $B$ are subjected to the rigid transformations $T_A$ and $T_B$, respectively. In order to determine the feasibility of $T_A$ and $T_B$, we have to evaluate \((4.21)\); an expression equivalent to \((4.17)\) and defined in subsection 4.2.1.

\[
c(G, \varepsilon_{AB}, \{A, B\}, \{T_A, T_B\}) = \frac{1}{2\varepsilon_{AB} \Delta_{AB} N_A} \sum_{k=1}^{N_A} |d(T_B^{-1}T_A(g_k), G)| + \frac{1}{2\varepsilon_{AB} \Delta_{AB} N_B} \sum_{k=1}^{N_B} |d(T_A^{-1}T_B(g_k), G)|
\]

Evaluation of $c(\quad)$ requires (i) calculation of point-to-surface distances, $d(\quad)$, and (ii) some way of discriminating between the points matched to the interior of $G$ from those matched to its exterior. In [80], (i) and (ii) were performed with a kd-tree parameterization algorithm described in [131]. In this study, (i) and (ii) were implemented using a different approach that exploited the distance transforms (DT) computed during the gliding surfaces extraction stage (see subsection 4.2.2.1).

Given $F_{AB} : \mathbb{R}^3 \rightarrow \mathbb{R}$ as the DT of $M$, the closest distance between an arbitrary point $q$ and $M$ is simply $d(q, M) = F_{AB}(q)$, therefore $d(T_B^{-1}T_A(g_i), M) = F_{AB}(T_B^{-1}T_A(g_i))$.

Suppose $p_i \in M$ as the closest point on $M$ to $T_B^{-1}T_A(g_i)$, then $d(T_B^{-1}T_A(g_i), M) = d(T_B^{-1}T_A(g_i), p_i)$. Relying on the property that the gradient of $F_{AB}$ (denoted as $\nabla F_{AB}$), at any point in the domain of $F_{AB}$, is orthogonal to $M$ (see Appendix D), $p_i$ can be computed as

\[
p_i = T_B^{-1}T_A(g_i) - F_{AB}(T_B^{-1}T_A(g_i)) \nabla F_{AB}(T_B^{-1}T_A(g_i)) / \| \nabla F_{AB}(T_B^{-1}T_A(g_i)) \|
\]
\( p_i \), however, can be situated inside or outside the boundary of \( G \). To distinguish between the two cases, we

1. transform \( p_i \) into the local frame of reference of \( G^{50} \) to obtain \( \tilde{p}_i = Rp_i \), then
2. find an orthogonal projection, \( P : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \), of \( \tilde{p}_i \) onto the xy-plane, and finally
3. perform point-in-polygon test [90] to see if \( P(\tilde{p}_i) \) lies inside \( P \circ R(\partial G) \), where \( \partial G \) is the boundary of \( G \).

If \( P(\tilde{p}_i) \) is confirmed to lie inside \( P \circ R(\partial G) \), then

\[
d(\mathbf{T}_B^{-1}\mathbf{T}_A(\mathbf{g}_i), M) = d(\mathbf{T}_B^{-1}\mathbf{T}_A(\mathbf{g}_i), G) = F_{AB}(\mathbf{T}_B^{-1}\mathbf{T}_A(\mathbf{g}_i))
\]

otherwise \( d(\mathbf{T}_B^{-1}\mathbf{T}_A(\mathbf{g}_i), G) \) is not used in evaluation of (4.21). The same procedure applies to test if \( d(\mathbf{T}_A^{-1}\mathbf{T}_B(\mathbf{g}_i), M) = d(\mathbf{T}_A^{-1}\mathbf{T}_B(\mathbf{g}_i), G) \). Refer to Figure 4.8 for the illustration of the process just described.

\( ^{50} \) Local basis of \( G \) is computed by performing PCA on \( V \).
Figure 4.8: a) After projection with (4.22), the point $T(g_1)$ is matched to $p_1$, and $T(g_2)$ is matched to $p_2$. b) Following the 1) change of basis and 2) orthogonal projection onto the xy-plane, we 3) perform in-polygon test to see which points are inside the boundary of the gliding surface. In the example shown, $T(g_1)$ was not matched to the interior of $G$ and therefore will not be used in evaluation of (4.21). Conversely, $T(g_2)$ has a matching counterpart on $G$ and is therefore suitable for evaluation of the constraints.
4.3 Results and Discussion

To evaluate the performance of our registration procedure (G-SQP) we compared it with the original method of van de Giessen et al. described in [80] (P-LMA). The registration results of both methods are summarized in Table 4.2. Additionally, we implemented a third algorithm that we refer to here as P-SQP. This algorithm was identical to G-SQP in all respects, except that instead of an unbiased carpal configuration, it used the arrangement of carpal bones from one of the training instances as a reference for other wrists in the training set\textsuperscript{51}. The registration results associated with this method are also included in Table 4.2.

The description length (DL) of the SSMs built using the three approaches are listed in the last row of Table 4.2. Not surprisingly, the SSM obtained with G-SQP has the lowest DL while the SSM constructed using the original method has the highest DL. The SSM obtained with P-SQP is situated between the two cases, and has a substantially higher DL than the one associated with G-SQP. We cannot overemphasize the significance of this result as it confirms our intuition that it is possible to perform groupwise registration of multiple wrist by decomposing the constrained optimization problem in (4.14) into $N_S$ independent sub-problems.

By comparing the magnitudes of the three registration approaches listed in the columns titled 'Initial Error', one should be able to recognize another important attribute of G-SQP. Specifically, in contrast to the pairwise approach, G-SQP implicitly minimizes the total displacement needed to register the wrists to a common posture, thus reducing the

---

\textsuperscript{51} Suppose that $T = \{W_i \mid i = 1, \ldots, N_S\}$ is the training set of all wrists in original poses and $\{W_R^{\text{U}}\}$ is the same training set after unconstrained registration to the training instance $W_R \in T$. In this study, we selected the reference as $R = \min_{k \in [1, N_S]} \mathcal{L}(\{W_i^k\})$. 

153
deviation of inter-carpal positions from their original state (which is known to be physiologically valid). In fact, on average, the total amount of postural deformation required by our method is approximately 4.8 times less than the one required by P-LMA and P-SQP. We would also like to draw attention to the entries in the row titled 'Average % Decrease', which indicate that P-SQP is much better at minimizing postural differences than P-LMA. We ascribe the last result to our earlier claim that P-LMA is susceptible to being trapped in the local minima.

The capacity of our method to produce better statistical models is further supported by the plots of standard measures of SSM quality (see subsection 3.2.4) shown in Figure 4.9. As expected, G-SQP produces models with significantly lower generalization and specificity errors than either of the pairwise approaches. Although, P-SQP cedes to G-SQP only slightly in terms of the overall compactness. On the whole, G-SQP is a better alternative to P-SQP and especially the original P-LMA method.

We would like to stress that in providing the comparison between the three registration methods, prior to calculating the registration scores in Table 4.2 and those shown in Figure 4.9, we removed any effects due to the differences in scale by normalizing all wrist instances to the same volume. For example, given $W_i^A$ and $W_i^B$ as two different configurations of the i-th wrist obtained after registration with methods A and B, respectively, the scaling effects were eliminated by modifying $W_i^B$ as shown below

$$
\tilde{W}_i^B = \frac{3}{\sqrt{Volume(W_i^B)}} \left[ W_i^B - Centroid(W_i^B) \right] + Centroid(W_i^B) \quad (4.23)
$$
Additionally, in our implementation of van de Giessen's original method (P-LMA), we used the revised definition of $J_{int}$ given in equation (4.7) and not the one provided by equation (4.5).

In closing of the discussion of the merits of the individual registration methods implemented in this chapter, we would like to draw attention to the issue of time performance. The total optimization times for G-SQP, P-SQP and P-LMA were 60.7 hrs, 61.0 hrs, and 15.2 hrs respectively\(^{52}\). The total G-SQP time included 11.2 hrs spent on the sequential, unconstrained minimization of the DL (see section 4.2) to compute the reference configurations of the training wrists and another 49.5 hrs to actually deform the wrists to the reference configurations using constrained optimization. Although at first these times may appear unreasonably long, one should recall that in the case of G-SQP, we were solving 28 constrained optimization sub-problems, where each problem had 11 nonlinear constraints and 49 unknowns. For P-SQP there were a total of 27 problems, but as mentioned previously this scheme required larger postural deformations and therefore the average time spent on one problem was approximately 30 minutes longer than that for G-SQP (see Table 4.2). The relatively short times recorded for P-LMA are only a reflection of the fact that it converged prematurely to a local minimum.

Future implementations of our algorithm may benefit by exploiting the fact that in our definition of $\tilde{E}_{ext}(\cdot)$ we use point-to-point distances (as opposed to point-to-surface), which means that it is possible to find closed form expressions for $\nabla \tilde{E}_{ext}$ (gradient) and $\nabla^2 \tilde{E}_{ext}$ (Hessian) to improve the efficiency of the SQP algorithm used to solve (4.14).

\(^{52}\) All times are based on MATLAB implementations running on a desktop with 6 GB RAM and 2.80 GHz x6 AMD processor.
Table 4.2: Performance summary of three different wrist registration schemes. G-SQP is the new method developed in this study. P-LMA is the original method that was presented by van de Giessen et al. in [80]. P-SQP is the same as G-SQP, but uses the arrangement of carpal bones from one of the training instances as a reference for other wrists in the training set. The description length (DL) of the SSMs assembled using the different methods was evaluated using equation (4.13) with $\lambda_{\text{corr}} = 4 \times 10^{-4}$.

<table>
<thead>
<tr>
<th>Spl #</th>
<th>Subj #</th>
<th></th>
<th>G-SQP</th>
<th></th>
<th>P-LMA</th>
<th></th>
<th>P-SQP</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Initial</td>
<td>Error</td>
<td>Residual</td>
<td>% Decrease</td>
<td>Initial</td>
<td>Error</td>
<td>Residual</td>
</tr>
<tr>
<td>1</td>
<td>13586</td>
<td>7.68</td>
<td>1.27</td>
<td>83.40</td>
<td></td>
<td>39.20</td>
<td>38.55</td>
<td>1.66</td>
</tr>
<tr>
<td>2</td>
<td>14931</td>
<td>8.26</td>
<td>1.45</td>
<td>82.41</td>
<td></td>
<td>39.66</td>
<td>32.29</td>
<td>18.59</td>
</tr>
<tr>
<td>3</td>
<td>15259</td>
<td>4.08</td>
<td>1.13</td>
<td>72.25</td>
<td></td>
<td>20.07</td>
<td>12.80</td>
<td>36.19</td>
</tr>
<tr>
<td>4</td>
<td>33508</td>
<td>9.84</td>
<td>1.04</td>
<td>89.47</td>
<td></td>
<td>57.14</td>
<td>50.94</td>
<td>10.84</td>
</tr>
<tr>
<td>5</td>
<td>39870</td>
<td>6.81</td>
<td>0.53</td>
<td>92.19</td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>43736</td>
<td>3.46</td>
<td>0.30</td>
<td>91.45</td>
<td></td>
<td>25.89</td>
<td>22.64</td>
<td>12.54</td>
</tr>
<tr>
<td>7</td>
<td>47116</td>
<td>10.01</td>
<td>1.55</td>
<td>84.49</td>
<td></td>
<td>55.05</td>
<td>51.80</td>
<td>5.91</td>
</tr>
<tr>
<td>8</td>
<td>53250</td>
<td>4.84</td>
<td>0.65</td>
<td>86.63</td>
<td></td>
<td>22.32</td>
<td>14.77</td>
<td>33.82</td>
</tr>
<tr>
<td>9</td>
<td>56762</td>
<td>4.77</td>
<td>0.76</td>
<td>84.02</td>
<td></td>
<td>31.19</td>
<td>26.76</td>
<td>14.21</td>
</tr>
<tr>
<td>10</td>
<td>72579</td>
<td>13.61</td>
<td>2.08</td>
<td>84.71</td>
<td></td>
<td>35.05</td>
<td>26.67</td>
<td>23.90</td>
</tr>
<tr>
<td>11</td>
<td>79183</td>
<td>8.74</td>
<td>3.97</td>
<td>54.59</td>
<td></td>
<td>51.39</td>
<td>38.71</td>
<td>24.68</td>
</tr>
<tr>
<td>12</td>
<td>86363</td>
<td>5.60</td>
<td>1.51</td>
<td>73.00</td>
<td></td>
<td>46.74</td>
<td>46.08</td>
<td>1.41</td>
</tr>
<tr>
<td>13</td>
<td>94239</td>
<td>3.89</td>
<td>0.49</td>
<td>87.30</td>
<td></td>
<td>21.80</td>
<td>12.13</td>
<td>44.36</td>
</tr>
<tr>
<td>14</td>
<td>98048</td>
<td>6.93</td>
<td>1.53</td>
<td>77.95</td>
<td></td>
<td>29.49</td>
<td>28.52</td>
<td>3.30</td>
</tr>
<tr>
<td>15</td>
<td>19186</td>
<td>12.75</td>
<td>1.98</td>
<td>84.45</td>
<td></td>
<td>47.22</td>
<td>30.17</td>
<td>36.09</td>
</tr>
<tr>
<td>16</td>
<td>22985</td>
<td>9.08</td>
<td>2.64</td>
<td>72.02</td>
<td></td>
<td>23.31</td>
<td>13.87</td>
<td>40.50</td>
</tr>
<tr>
<td>17</td>
<td>24117</td>
<td>3.10</td>
<td>0.50</td>
<td>83.83</td>
<td></td>
<td>33.04</td>
<td>23.55</td>
<td>28.73</td>
</tr>
<tr>
<td>18</td>
<td>28528</td>
<td>2.56</td>
<td>0.20</td>
<td>92.06</td>
<td></td>
<td>38.74</td>
<td>30.40</td>
<td>21.52</td>
</tr>
<tr>
<td>19</td>
<td>35842</td>
<td>8.23</td>
<td>1.45</td>
<td>82.41</td>
<td></td>
<td>18.87</td>
<td>12.84</td>
<td>31.92</td>
</tr>
<tr>
<td>20</td>
<td>62641</td>
<td>7.62</td>
<td>0.48</td>
<td>93.72</td>
<td></td>
<td>30.47</td>
<td>25.00</td>
<td>17.95</td>
</tr>
<tr>
<td>21</td>
<td>63899</td>
<td>4.55</td>
<td>1.53</td>
<td>66.45</td>
<td></td>
<td>24.97</td>
<td>21.80</td>
<td>12.72</td>
</tr>
<tr>
<td>22</td>
<td>64050</td>
<td>8.62</td>
<td>0.89</td>
<td>89.65</td>
<td></td>
<td>37.65</td>
<td>29.98</td>
<td>20.36</td>
</tr>
<tr>
<td>23</td>
<td>66582</td>
<td>3.59</td>
<td>0.48</td>
<td>86.50</td>
<td></td>
<td>32.19</td>
<td>24.26</td>
<td>24.62</td>
</tr>
<tr>
<td>24</td>
<td>66900</td>
<td>5.15</td>
<td>1.13</td>
<td>78.11</td>
<td></td>
<td>36.59</td>
<td>34.03</td>
<td>7.00</td>
</tr>
<tr>
<td>25</td>
<td>68134</td>
<td>7.46</td>
<td>0.95</td>
<td>87.23</td>
<td></td>
<td>30.65</td>
<td>18.79</td>
<td>38.72</td>
</tr>
<tr>
<td>26</td>
<td>77209</td>
<td>6.37</td>
<td>1.09</td>
<td>82.88</td>
<td></td>
<td>37.38</td>
<td>36.27</td>
<td>2.95</td>
</tr>
<tr>
<td>27</td>
<td>83303</td>
<td>5.72</td>
<td>1.19</td>
<td>79.13</td>
<td></td>
<td>44.56</td>
<td>42.67</td>
<td>3.80</td>
</tr>
<tr>
<td>28</td>
<td>86864</td>
<td>11.60</td>
<td>0.63</td>
<td>94.54</td>
<td></td>
<td>27.46</td>
<td>23.81</td>
<td>13.29</td>
</tr>
<tr>
<td>Ave. % Decrease</td>
<td>82.90</td>
<td></td>
<td>17.88</td>
<td></td>
<td>51.25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Time (min)</td>
<td>2974.00</td>
<td></td>
<td>914.21</td>
<td></td>
<td>3662.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ave. Time (min)</td>
<td>106.21</td>
<td></td>
<td>33.86</td>
<td></td>
<td>135.63</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final DL</td>
<td>461.17</td>
<td></td>
<td>473.98</td>
<td></td>
<td>469.29</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 4.9: Comparison of the composite SSM quality for different registration schemes. G-SQP is the new method developed in this study. P-LMA is the original method that was presented by van de Giessen et al. in [80]. P-SQP is the same as G-SQP, but uses the arrangement of carpal bones in one of the training instances as a reference for the other wrists in the training set.
The extremes of the first eight principal modes of variation of the SSM constructed with the new G-SQP algorithm are illustrated in Figure 4.10. As shown in the bottom right corner of the figure, together these modes describe approximately 86% of all variations of the training set composed of 28 samples\textsuperscript{53}. To help visualize the structural variations described by each mode, the relative magnitude of the local deformations was projected as a colormap on the mean configuration of carpal bones. Similar representation of the local variations described by combination of the first eight eigenmodes is shown in greater detail in Figure 4.11. Based on the observed pattern of variations, it can be seen that the most pronounced changes occur as a result of 'widening' and 'narrowing' of the carpal bone complex in the lateral directions. Other regions with prominent anatomical variations are the distal end of the lunate, hook of hamate, scaphoid's waist and tubercle, and scaphoid-trapezoid facet of capitate. The apparent variations of the pisiform, are mostly due to positional shifts associated with the widening' and 'narrowing' of the carpal bone complex mentioned above.

In this manuscript, we have repeatedly emphasized that the wrist is a very complex structure. We therefore did not expect to find that approximately 74% of the observed structural variations of the carpus could be described by only four eigenmodes. In fact, this compression ratio is higher than those achieved by the SSMs of the individual carpal bones (except scaphoid, see Figure 3.16). We suspect that this would not have been

\textsuperscript{53} Although, the wrist database was assembled from 30 individuals [16], as mentioned previously, one of the wrist (#97808) was missing a trapezoid bone. In another sample (#60684), we found a collision between the capitate and lunate (an error which probably occurred during the segmentation of the CT images). Since the collision of cortical surfaces is an indication of a physiologically incorrect arrangement of carpal bones, the second sample was also omitted from the training set used in construction of the composite SSM.
possible without the geometric correlations that exist between the pairs of articulating bones.

Finally, to verify the validity of the assumption that the structural variations of the wrist can be modeled as a linear combination of the principal modes, we tested a set of null hypotheses that the shape parameters associated with each mode are random variables drawn from a normal distribution. The results of these tests are summarized in Table 4.3 for the first 12 modes, while the normalized distributions of the actual shape parameters are shown in Figure 4.12. Taken together these modes account for approximately 92% of observed variations. As shown in Table 4.3, the null hypothesis was rejected only once (at the 0.05 significance level) for the 11th mode. In fact, we confirmed with 95% confidence that the shape parameter distributions of 25 out of 27 modes could be modeled as univariate Gaussians. This result is important because it suggests that 98% of structural variations of the entire carpus can be represented by a linear combination of the 25 eigenmodes.
Figure 4.10: Extremes of the first eight principal modes of carpus variation. The structural changes associated with each mode are represented by colormaps (superimposed on the average carpus) along the middle column. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Figure 4.11: Colormap of local variations described by the first eight principal modes, superimposed on the mean of the training set. Two views are shown. Areas in dark red denote the most amount of variation and areas in dark blue the least amount of variation.
Table 4.3: p-values of the Kolmogorov-Smirnov test [107] used to verify if the distributions of shape parameters were significantly different from the normal distribution. The highlighted cell indicates the SSM mode with p-value below the 0.05 significance level.

<table>
<thead>
<tr>
<th>Mode</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>p-value</td>
<td>0.78</td>
<td>0.85</td>
<td>0.69</td>
<td>0.26</td>
<td>0.61</td>
<td>0.90</td>
<td>0.30</td>
<td>0.39</td>
<td>0.87</td>
<td>0.95</td>
<td><strong>0.03</strong></td>
<td>0.45</td>
</tr>
</tbody>
</table>

Figure 4.12: Distribution of shape parameters for the first twelve modes of the composite SSM. Taken together these modes describe 91.7% of the variations present in the training set composed of 28 samples. SND = standard normal distribution.
4.4 Conclusion

The primary objective of the work described in this chapter was to develop a robust method to create a multi-body statistical shape model of the entire carpus. At the outset we identified that as a result of complex inter-carpal relationships such a model could not be created by registering the wrists using global similarity transformations. This realization prompted us to look into constrained registration algorithms that could be used to align multiple wrists to the same posture while preserving a physiologically valid arrangement of the carpal bones. We identified one such algorithm, developed recently by van de Giessen et al. [80], but subsequently showed that it contains a number of limitations that diminish its capacity in obtaining reliable registration results. Drawing on the MDL principle used in Chapter 3, we built around the core idea of van de Giessen's method, to develop a more robust, truly constrained and unbiased registration scheme. We have shown that the new method implicitly minimizes the total displacement needed to register the wrists to a common posture, while at the same time producing models with substantially better compactness, generalization and specificity properties. In fact, the composite SSM of the wrist built with our method provides a very compact parameterization of wrist variability and can describe ~74% of anatomical differences present in the training set (of 28 wrists) using only first 4 principal modes of variation. We also demonstrated that ~98% of carpal variations can be described by a standard, linear SSM.
In the last decade, increased availability of high performance computing resources has prompted a shift in the experimental approach across multiple scientific disciplines to the point where computer-based simulations are gradually becoming acceptable substitutes to traditional lab bench experiments. The field of computational biomechanics is one derivative of this technological evolution and has been successfully applied to biomedical problems such as impact and fracture mechanics of bone, load transmission through the joints, feasibility of joint replacements, investigation of joint injury mechanisms, and many others. Despite remarkable advances in this field, there remain a number of technical challenges. One of those is related to the fact that almost all biomechanical models reported in literature to date were derived from the anatomy of a single individual. This is a conceptually problematic issue because the results acquired from the simulations based on subject-specific models cannot be generalized to a wider population set. Statistical shape models (SSM) provide an intuitive approach for creating population-based parametric models. Within this framework the notion of “mean shape” and “principal components of shape” are explicitly defined and can be used as population representatives for finite element (FE) simulations.

This study was a stepping stone toward the ultimate goal of creating an anatomically accurate and complete model of the hand and wrist. In the course of this research, we overcame two technical challenges which we expect will be common to any future attempts to create a anatomically accurate statistical models of compact, multi-body
structures suitable for use with conventional FE solvers. Firstly, we built upon the method of van de Giessen et al. [80] to developed a new procedure for coupling multiple statistical bone models into a single, physiologically valid model of the entire wrist. Secondly, by combining properties of conformal maps from differential geometry with principles governing the dynamics of charged particle systems, we developed a novel procedure for resampling closed surface meshes (topologically equivalent to a sphere) to produce high quality meshes with subdivision connectivity. The inherent multi-resolution construction of the meshes allowed for substantial acceleration of various mesh processing operations used during the model building stages, and in the future could also be exploited by multigrid solvers during the FE simulation stage.

While the performance of the population based modeling approach adopted in this study was demonstrated on a publically available database of the wrist bones, the method is directly generalizable to the multi-body skeletal structures of the lower limb. Moreover, the image processing origin of SSMs provides an easy means to extend the SSMs of the wrist bones to the active shape or active appearance models that can be used to automate segmentation of the medical images thus facilitating diagnosis of potential wrist injuries associated with ligament ruptures and cartilage damage. Finally, in comparison to subject specific models, the SSM of the wrist permits a wider range of applications that include the optimal design of prosthetic and fixation devices; the clinical reference of normal carpal variations and inter-carpal relationships, and the systematic study of structural changes in bone anatomy at various stages of growth or disease. In the future, we intend to build upon the methods developed during this study to construct a statistical, FE model
of the upper limb (hand, wrist and forearm) that in addition to the bony structures will incorporate the connective and muscle tissues.
APPENDIX

A Spherical Parameterization

Parameterization of a surface mesh is the problem of establishing a bijective (i.e. invertible) mapping between the original piecewise linear surface and a continuous parameter domain which is a primitive topological analog of the surface. This study deals primarily with closed genus-0 surfaces, which are characteristic of various everyday objects as well as many organs in the human body (brain, kidneys, liver, lungs, etc.) including the bones in the hand and wrist. For such surfaces, the unit sphere is the most natural parameter domain [25,60]. Planar parameterizations of closed genus-0 surfaces are also common and may be advantageous in certain situations (e.g. texture mapping) but are less natural from a global point of view, as they tend to produce higher distortions.

Let \( M = [V,E] \) be a surface mesh where \( V \) and \( E \) are sets of vertices and edges, respectively. From a graph theoretic point of view parameterization of \( M \) is equivalent to embedding the connectivity graph of \( M \) onto the unit sphere, \( S^2 \), such that the resulting spherical triangles form a partition\(^1\) of \( S^2 \). Classical result due to Steinz guarantees the existence of such a mapping if and only if \( M \) is planar and 3-connected [25], implying that any genus-0 triangulation can be mapped to a sphere. Unfortunately, such a mapping is not unique and the problem is actually finding a “good” one. The continued discussion

\(^1\) Partition means the union of spherical triangles is the surface of the sphere and the intersection of these triangles is an empty set [25].
regarding the mappings between the surfaces and what set of properties constitute a desirable mapping requires some familiarity with fundamental concepts from differential geometry. A concise overview of the most relevant concepts is provided in the following subsection.

### A.1 Differential Geometry Background

Suppose $S$ is a $C^n$ differentiable surface embedded in 3-dimensional space. Let $x \in S$ such that $x(u, v) = (x(u, v), y(u, v), z(u, v))$, where $x, y, z \in \mathbb{R}$ are the components of the rectangular coordinate system and $u, v \in \mathbb{R}^+$ are the components of the tangent space of the surface. The element of arc of a curve on $S$, $ds$, is defined as:

$$ds = \frac{\partial x}{\partial u} du + \frac{\partial x}{\partial v} dv = x_u du + x_v dv$$  \hspace{1cm} (A.1)

The inner product of $ds$ in the tangent space of $S$ provides an intrinsic surface metric that can be used to quantify surface distortions at any point $x(u, v)$ due to some mapping $f : x \rightarrow x^*$. Based on the definition of $ds$ from (A.1) it follows that:

$$ds^2 = [du \, dv] \begin{bmatrix} x_u \cdot x_u & x_u \cdot x_v \\ x_u \cdot x_v & x_v \cdot x_v \end{bmatrix} [du \, dv] = [du \, dv] M [du \, dv]$$  \hspace{1cm} (A.2)

$M$ is the metric tensor of $S$ at $x(u, v)$, also known as the first fundamental form of $S$. $ds^2$ with its corresponding $M^*$ can be derived similarly.

The most ideal type of mapping\(^2\) is that which does not produce any distortions. The so called isometric maps have this property and are the only type of transformations that have length preserving property. Formalizing this condition using mathematical notation,

\[^2\]\(^2\) Note, parameterization is a mapping from the original surface to its canonical parameter domain. Hence the terms mapping and parameterization are interchangeable.
length preservation implies constancy of the element arc length before and after the transformation \((ds = ds^*)\) which is equivalent to \(M = M^*\). Unfortunately isometric maps exist only for planar and developable surfaces (e.g. cylinder, cone) [40]. The deformation of most commonly encountered surfaces almost always introduces some type of distortion. Two fundamental types of distortion can be attributed to changes in area and angles [40]. For most general types of surfaces, the gamut of possible mappings falls between the two special cases of equiareal and conformal maps. The former type of mappings preserves areas up to a constant factor. Returning to the definition of the surface differential, the area element of \(S\) can be defined as the magnitude of the cross product of its tangent vectors, \(x_u \times x_v\), which is equal to \(\text{det}(M)\). Hence the mapping is equiareal if \(\text{det}(M) = \beta \cdot \text{det}(M^*)\) for \(\beta > 0\). Conformal maps on the other hand preserve angles, hence at a given point \((u, v)\), \(M\) and \(M^*\) must be equal up to a constant factor, or more formally, \(M = \eta(u, v) \cdot M^*\). In this relation, \(\eta(u, v)\) is a scalar function termed the conformal factor and is non-zero for all \(u\) and \(v\) (otherwise the mapping will not be invertible).

### A.2 Conformal Maps

From the discussion above it should be noted that the maps that preserve both the areas and the angles also preserve length and are therefore isometric. However, since such mappings do not exist for most general types of surfaces, depending on the application, a mapping that provides a trade-off between the area and angle distortions must be used to

\(^3\) \(x_u\) and \(x_v\) are the principal components of \(M\) with corresponding eigenvalues equal to their respective norms, therefore \(\sin(\pi/2) \|x_u\| \|x_v\| = \lambda_u \lambda_v = \text{det}(M)\).
obtain the spherical parameterization. In the context of establishing correspondence between similar (non-planar and non-developable) surfaces, conformal maps are the best alternatives. The main advantages of conformal maps is that unlike equilateral maps they are (a) easy to compute, (b) almost unique\(^4\), (c) preserve local geometries, (d) depend only on the surface geometry and hence are invariant to changes in triangulation and mesh resolution, and (e) have low sensitivity to noise \([39,81,82]\). To understand how conformal maps can be obtained, consider for a moment a related type of transformation based on harmonic maps. Harmonic maps are transformations that minimize deformation defined in terms of the Dirichlet energy \([40,42,81]\):

\[
\Phi_D(f) = \frac{1}{2} \int_{S} \|\nabla f(u, v)\|^2 \, dudv = \frac{1}{2} \int_{S} \left\{ \left\| \frac{\partial f}{\partial u} \right\|^2 + \left\| \frac{\partial f}{\partial v} \right\|^2 \right\} \, dudv \tag{A.3}
\]

For a planar surface, such as a triangular faces element of \(M, \mathcal{T}_M^A\), \(\Phi_D(f)\) reduces to \([42]\):

\[
\Phi^A_D(f) = \frac{1}{2} \left[ \lambda_{12} \|f(v_1) - f(v_2)\|^2 + \lambda_{23} \|f(v_2) - f(v_3)\|^2 + \lambda_{31} \|f(v_3) - f(v_1)\|^2 \right] \tag{A.4}
\]

where \(v_1, v_2\) and \(v_3\) are the three vertices of the triangle, \(\mathcal{T}_M^A\), with the corresponding \(\lambda\) coefficients defined in Table A.1.

**Table A.1: Laplacian coefficients.**

<table>
<thead>
<tr>
<th>(\lambda_{12})</th>
<th>(\lambda_{23})</th>
<th>(\lambda_{31})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1 \frac{(v_1 - v_3) \cdot (v_2 - v_3)}{2 | (v_1 - v_3) \times (v_2 - v_3) |} )</td>
<td>(1 \frac{(v_2 - v_1) \cdot (v_3 - v_1)}{2 | (v_2 - v_1) \times (v_3 - v_1) |} )</td>
<td>(1 \frac{(v_3 - v_2) \cdot (v_1 - v_2)}{2 | (v_3 - v_2) \times (v_1 - v_2) |} )</td>
</tr>
</tbody>
</table>

\(^4\) All conformal mappings form the so called Möbius group \([39]\). See Section 3.2.2 for more details.
Since the main objective to identify a mapping which minimizes $\Phi_D$ globally, (A.4) has to be summed over all of the faces of $M$. It can be shown that this sum reduces to the simple from of (A.5):

$$\Phi(f, M) = \frac{1}{2} \sum_{(v_i,v_j) \in E} k_{ij} \|f(v_i) - f(v_j)\|^2$$  \hspace{1cm} (A.5)

![Harmonic weight associated with edge $(v_i,v_j)$ depends on the two angles ($\alpha$ and $\beta$) opposing the edge.](image)

Figure A.1: Harmonic weight associated with edge $(v_i,v_j)$ depends on the two angles ($\alpha$ and $\beta$) opposing the edge.

$\Phi(f, M)$ is the so called harmonic energy of the mesh, where $k_{ij}$ coefficient is termed the edge weight and is defined as $k_{ij} = \lambda^A_{ij} + \lambda^B_{ij}$; $\lambda_{ij}$ are listed in Table A.1 and are derived from the vertices of the two neighbouring faces A and B sharing a common edge $(v_i,v_j)$, as shown in Figure A.1. Using geometric interpretations of the inner and cross products, $\lambda^A_{ij}$ can be simplified as follows:

$$\lambda^A_{ij} = \frac{1}{2} \frac{(v_i - v_k) \cdot (v_j - v_k)}{\| (v_i - v_k) \times (v_j - v_k) \|} = \frac{1}{2} \frac{\cos(\alpha) \|v_i - v_k\| \|v_j - v_k\|}{\sin(\alpha) \|v_i - v_k\| \|v_j - v_k\|} = \frac{\cot(\alpha)}{2}$$

Performing the same operation for $\lambda^B_{ij}$ and substituting the result into the formula for $k_{ij}$, we get the following expression:

$$k_{ij} = [\cot(\alpha) + \cot(\beta)]/2$$  \hspace{1cm} (A.6)
In [39] Gu et al. make an important point that conformal maps, \( f \), of genus-0 surfaces satisfy the Laplace equation \( \Delta_S f = 0 \) where \( \Delta_S \) is the Laplace-Beltrami operator defined as \( \Delta_S = \text{div}_S \text{grad}_S \). In fact, the derivative of \( \Phi(f, M) \) with respect to \( f \) is a (discrete) Laplacian [87], which implies that harmonic maps are also conformal.

\[
L(f, M) = \sum_{(v_i, v_j) \in E} k_{ij} [f(v_i) - f(v_j)]
\] (A.7)

This suggests that conformal maps for piecewise linear surfaces, \( M \), of genus-0 can be computed by minimizing \( \Phi(f, M) \). The method of obtaining the solution to this optimization problem is described in detail in the Methods section of Chapter 2.

## B Derivative of the Reisz-s Energy Functional

Reisz s-energy of a system of particles with unequal positive charges is defined as:

\[
R_s = \sum_{i} \sum_{j \neq i} q_i q_j \frac{s_i}{d_{ij}^s}
\] (B.1)

Let \( x_i = (x_{i1}, x_{i2}, x_{i3})^T \) be the position of the \( i \)-th particle on the surface of the sphere. The gradient of \( R_s \) with respect to \( x_i \) is therefore:

\[
\frac{\partial R_s}{\partial x_i} = 2 \sum_{j \neq i} \frac{\partial}{\partial x_i} \left( \frac{q_i q_j}{d_{ij}^s} \right) = 2 \sum_{j \neq i} \left( \frac{\partial q_i}{\partial x_i} - s_i \frac{q_i}{d_{ij}^s} \frac{\partial d_{ij}}{\partial x_i} \right)
\] (B.2)

To adaptively distribute the charged particles over the surface of the sphere we modify particle charges according to their position. Let \( q_i = q(x_i) \), where \( q : \mathbb{R}^3 \rightarrow \mathbb{R}^+ \). For an arbitrary position on the sphere, \( q(x_i) \) can be approximated using linear interpolation, so that:

---

\( ^5 \) Laplace-Beltrami operator is a generalization of the Laplace operator to manifolds [40].
\begin{equation}
q(x_i) = uq_1^i + vq_2^i + wq_3^i \tag{B.3}
\end{equation}

where \{q_k^i \mid k = 1,2,3\} are the charges defined on the mesh vertices. Let \{x_k^i \mid k = 1,2,3\} be the vertex coordinates of the spherical triangle containing \(x_i\). Then \(u, v\) and \(w\) are the spherical barycentric coordinates of \(x_i\) if they satisfy (B.4).

\begin{equation}
x_i = [x_1^i \quad x_2^i \quad x_3^i][u \quad v \quad w]^T \\
u + v + w \geq 1 \\
u, v, w \geq 0 \tag{B.4}
\end{equation}

From (B.3) it follows that:

\begin{equation}
\frac{\partial q_i}{\partial x_i} = \begin{bmatrix}
\frac{\partial u}{\partial x_i} & \frac{\partial v}{\partial x_i} & \frac{\partial w}{\partial x_i}
\end{bmatrix} [q_1^i \quad q_2^i \quad q_3^i]^T \tag{B.5}
\end{equation}

Using Cramer's rule we can evaluate \(u, v\) and \(w\) from (B.4) as:

\[
\begin{align*}
u &= \frac{\text{det}([x_1^i \quad x_2^i \quad x_3^i])}{\text{det}(A_i)} \\
w &= \frac{\text{det}([x_1^i \quad x_2^i \quad x_1^i])}{\text{det}(A_i)}
\end{align*}
\]

where \(A_i = [x_1^i \quad x_2^i \quad x_3^i]\), hence:

\begin{equation}
\frac{\partial q_i}{\partial x_i} = \frac{1}{\text{det}(A_i)} \text{Adj}(A_i) [q_1^i \quad q_2^i \quad q_3^i]^T \tag{B.6}
\end{equation}

\(\text{Adj}(A_i)\) is the adjoint matrix of \(A_i\).

In evaluating \(\frac{\partial d_{ij}}{\partial x_i}\), recall that \(d_{ij}\) is the distance between two particles at positions \(x_i\) and \(x_j\). Two possible measures of \(d_{ij}\) is a Euclidean norm and geodesic distance. Since the objective is to evenly distribute the particles on the surface, the distance metric intrinsic to the surface is the most appropriate, therefore \(d_{ij} = \cos^{-1}(x_i \cdot x_j)\). The derivative of \(d_{ij}\) can be easily evaluated and is shown in (B.7).

\begin{equation}
\frac{\partial d_{ij}}{\partial x_i} = -\left[1 - (x_i \cdot x_j)^2 \right]^{-2} (x_i \cdot x_j)x_j \tag{B.7}
\end{equation}
Finally, substituting (B.6) and (B.7) into (B.2), we obtain the expression for the gradient that can be used to adaptively distribute an ensemble of charged particles across the surface of the unit sphere:

$$\frac{\partial R_s}{\partial x_i} = 2 \sum_{j \neq i}^N q_j d_{ij}^l \left[ \text{Adj}(A_i) \begin{bmatrix} q_i^1 & q_i^2 & q_i^3 \end{bmatrix} + \frac{q_is}{d_{ij}} \frac{x_i \cdot x_j}{1 - (x_i \cdot x_j)^2} x_j \right] \quad (B.8)$$

In a system where all particles have unit charge, (B.8) simplifies to (B.9),

$$\frac{\partial R_s}{\partial x_i} = 2s \sum_{j \neq i}^N \frac{x_i \cdot x_j}{d_{ij}^{s+1}} \left[ 1 - (x_i \cdot x_j)^2 \right] x_j \quad (B.9)$$

The formulation in (B.9) obviously no longer requires interpolation of the particle charges and can be used to distribute the particles uniformly across the surface of the unit sphere.

## C Mesh Complexity

The purpose of this section is derive the relationship between the complexity of the base mesh, the number of subdivisions (i.e. hierarchy levels) and the desired complexity of the final mesh. This goal can be achieved by considering sequences describing the changes in the number of mesh elements between the successive subdivisions and the Euler-Poincaré formula. In this study, triangular quadrisection is the subdivision method of choice. The mechanism of one subdivision is illustrated in Figure 2.9, while the corresponding changes in the number of vertices ($N_V$), edges ($N_E$) and faces ($N_F$) are described by equations (C.1)-(C.4). Equation (C.4) is the Euler-Poincaré formula used to relate the number of polygonal features ($N_V, N_E, N_F$) and surfaces boundaries ($N_B$) to the genus of the surface [21].
At the present, we are concerned only with genus-0 surfaces that do not have any boundaries. Such are the surfaces homeomorphic to the sphere, so (C.4) can be simplified to (C.5).

\[ N^i_V = N^i_E - N^i_F + 2 \]  

Let \( N^0_V, N^0_E \) and \( N^0_F \) be the number of vertices, faces and edges of the base mesh, respectively. From (C.3), the sequence of the number of edges at increasing levels of complexity is the following:

\[
\begin{align*}
N^2_E &= 2N^1_E + 3N^1_F \\
N^3_E &= 2(2N^1_E + 3N^1_F) + 3N^2_F \\
N^4_E &= 2(2N^1_E + 3N^1_F) + 3N^2_F + 3N^3_F
\end{align*}
\]

Based on this growth pattern, the number of edges after \( n \) subdivisions is therefore:

\[ N^n_E = 2^{n-1}(2N^0_E + 3N^0_F) + 3 \sum_{i=1}^{n-1} 2^{n-1-i}N^i_F \]  

From (C.2), it is clear that the number of faces quadruples after every subdivision:

\[ N^n_F = 4^n N^0_F \]  

Substituting (C.7) into (C.6) and simplifying:

\[ N^n_E = 2^{n-1}(2N^0_E + 3N^0_F) + 3N^0_F 2^{n-1}(2^n - 2) \]  

Now substituting (C.7) and (C.8) into (C.5):

\[ N^n_V = 2^{n-1}(2N^0_E + 3N^0_F) + 3N^0_F 2^{n-1}(2^n - 2) - 2^{2n} N^0_F + 2 \]
The expression above provides a connection between the number of mesh vertices after \( n \) subdivisions and the number of faces and edges of the base mesh. It is more natural, however, to have a direct relationship between \( N_V^0 \) and \( N^n_V \). Such an expression can be used to determine the number of particles needed to adaptively sample the sphere \( (N_V^0) \) to obtain a mesh with desired number of vertices \( (N^n_V) \) after \( n \) subdivisions. Although it is not possible to derive an exact relationship between these variables for a general conformation of particles, it is possible to make a very good approximation by assuming that in the ground state, the majority of the particles will have a valence of six. Accordingly, the base mesh must have three times more edges than vertices:

\[
N_E^0 \approx 3N_V^0
\]  
(C.10)

Substituting (C.10) into (C.5),

\[
N_E^0 \approx N_E^0 - N_V^0 + 2 = 2(N_V^0 + 1)
\]  
(C.11)

then substituting (C.10) and (C.11) into (C.9), we get the sought expression:

\[
N^n_V \approx 4^n N_V^0 + 2^n (2^n - 3) + 2 \approx 4^n N_V^0
\]  
(C.12)

### D Signed Distance Transform

A signed distance transform (SDT) is a common implicit representation of surfaces. For any point of the domain of the function, the value of SDT corresponds to the signed Euclidean distance to the closest point on the surface \([63,46,47,88]\). By convention negative values of SDT indicate that the point lies inside the region enclosed by the surface, while the positive values indicate that the point lies outside that region. Let \( S \in \mathbb{R}^3 \) be the solid region enclosed by the surface and \( \delta S \subset S \) be the surface boundary, then the SDT of \( S \) at a point \( \mathbf{x} \) is defined as:
According to the definition above it follows that the surface is defined as the zero level-set of $F_S(x)$. One the useful mathematical properties of SDT is based on the fact that $\nabla F_S(x)$ is orthogonal to the level-set passing through $x$, providing an easy means of evaluating surface normal [63].

In this study, the interest lies in computing the SDT of a closed polygonal surface, $M = [V,E]$, where $V$ and $E$ are sets of mesh vertices and edges, respectively. An excellent review of the various methods used to address this problem provided in [46]. In the present study, the method used to compute the SDTs was mainly based on the algorithms described in [46,47,88]. The basic procedure is summarized in Algorithm 4.

**ALGORITHM 4: Estimating Continuous Distance Field of PL Surface**

1. Identify the spatial limits of the surface (i.e. bonding box of the surfaces). Offset the limits of the domain outwards in all direction by a fixed amount. Discretize the domain with a regular rectangular lattice.
2. For all nodes $\{x_i\}$ of the rectangular grid do the following:
   a. Compute the signed distance to all mesh vertices. Let $F_V(x_i)$ be the shortest of these distances.
   b. Compute the shortest signed distance to all mesh edges (see subsection D.1). Let $F_E(x_i)$ be the shortest of these distances.
   c. Compute the shortest signed distance to all mesh faces (see subsection D.2). Let $F_T(x_i)$ be the shortest of these distances.
   d. Set the value of the SDT at $x_i$ to the one of the above distances that has the smallest modulus.
3. Construct continuous representation of the distance field using tricubic interpolation with clamped boundary conditions. In this study we used 'csape' function from MATLAB’s Curve Fitting Toolbox.
The operations in the steps 2a-c are complicated by the fact that we seek a signed distance as opposed to an absolute distance. This requires a test to determine whether a point is internal or external to the surface boundary. For differentiable surfaces this test would involve finding the sign of the dot product between the normal vector at some point on the surface and the direction vector to that point. Although continuous, the polygonal mesh is not differentiable at the vertices and edges. In [47], Aanæs and Bærentzen proved that the so called angle-weighted normal can be used to reliably distinguish the points interior and exterior to the closed surface mesh. For any vertex in the triangular mesh, the angle weighted normal is computed using (D.3), where $\alpha_i$ denotes the angle of the face incident on the vertex. This result of this operation is illustrated in Figure D.1. In case of edges, the sum in (D.3) is performed only over the two faces sharing the edge and both normals are weighted by $\pi$.

$$\vec{N} = \sum \alpha_i \vec{n}_i / \left\| \sum \alpha_i \vec{n}_i \right\| \quad \text{(D.3)}$$

**Figure A.2:** Angle weighted normal.

**Figure A.3:** The seven cases of a distance to a triangle.
In addition to the above consideration it is essential to recognize the finite span of the edge and face elements. In other words, what is the shortest distance to the edge and likewise, the shortest distance to the face? In general this value is not the same as the distance to the line passing through the edge or the distance to the plane passing through the face. The means by which the signed distance to these features can be determined is described below.

**D.1 Distance to an Edge**

Let \( \mathbf{x} \) be some arbitrary point and \( (\mathbf{v}_j, \mathbf{v}_k) \) be an edge, where \( \mathbf{v}_j \) and \( \mathbf{v}_k \) are the vertices binding the edge. Let \( \mathbf{v}(t) = \mathbf{v}_j + t(\mathbf{v}_k - \mathbf{v}_j) \) be a point on the line segment connecting \( \mathbf{v}_j \) and \( \mathbf{v}_k \), where \( t \in [0,1] \). We seek a value of \( t \) that minimizes the quantity \( \| \mathbf{x} - \mathbf{v} \|^2 \).

Applying the variational approach it can be shown that the optimal value of this parameter is:

\[
t^{**} = \begin{cases} 
0, & t^* < 0 \\
\frac{t^*}{1}, & 0 \leq t^* \leq 1 \\
1, & t^* > 1 
\end{cases} \tag{D.4}
\]

\[
t^* = \frac{(\mathbf{x} - \mathbf{v}_j) \cdot (\mathbf{v}_k - \mathbf{v}_j)}{(\mathbf{v}_k - \mathbf{v}_j) \cdot (\mathbf{v}_k - \mathbf{v}_j)} \tag{D.5}
\]

Hence the shortest distance to the edge is \( \| \mathbf{x} - \mathbf{v}(t^{**}) \| \). The sign of this distance is determined from the scalar product of the direction vector \( \mathbf{x} - \mathbf{v}(t^{**}) \) and the angle-weighted edge normal.
D.2 Distance to a Face

Let \( \mathbf{x} \) be some arbitrary point and let \((\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_k)\) be the vertices of a triangular face, \( T_{ijk} \), specified in anticlockwise order so that the normal \( \mathbf{N} = (\mathbf{v}_j - \mathbf{v}_i) \times (\mathbf{v}_k - \mathbf{v}_i) \) points to the exterior of the surface. When evaluating the distance to \( T_{ijk} \), there are seven possible cases (R1-R7) shown in Figure D.2. Let \( d \) be the distance between \( \mathbf{x} \) and the plane passing through \( T_{ijk} \), such that \( d = (\mathbf{x} - \mathbf{v}_i) \cdot \mathbf{N} \) and let \( \mathbf{x}_o \) be the projection of \( \mathbf{x} \) onto this plane, \( \mathbf{x}_o = \mathbf{x} - d\mathbf{N} \). We now have to determine in which of the seven regions shown in Figure D.2 the point \( \mathbf{x}_o \) is situated relative to \( T_{ijk} \). There are number of ways this test can be performed. The approach used in the present study is based on evaluation of the signs of the three dot products listed in Table D.2. In instance where all three dot products are negative, it means \( \mathbf{x}_o \) is situated inside the triangle and \( d \) should be recorded as the distance between \( \mathbf{x} \) and \( T_{ijk} \). The distances recorded for the remaining six instances could be set to infinity because they were evaluated previously in steps 2a and 2b. Note, the vectors \( \mathbf{n}_1, \mathbf{n}_2 \) and \( \mathbf{n}_3 \) listed in Table D.2 can be pre-computed to make the test more efficient.

Table A.2: Expressions used to test if an arbitrary point, \( \mathbf{x}_o \), is inside a triangle defined by vertices \( \{\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_k\} \).

| \( (\mathbf{x}_o - \mathbf{v}_i) \cdot \mathbf{n}_1 \) | \( (\mathbf{x}_o - \mathbf{v}_j) \cdot \mathbf{n}_2 \) | \( (\mathbf{x}_o - \mathbf{v}_k) \cdot \mathbf{n}_3 \) |
| \( \mathbf{n}_1 = (\mathbf{v}_j - \mathbf{v}_i) \times \mathbf{N} \) | \( \mathbf{n}_2 = (\mathbf{v}_k - \mathbf{v}_j) \times \mathbf{N} \) | \( \mathbf{n}_3 = (\mathbf{v}_i - \mathbf{v}_k) \times \mathbf{N} \) |
Evaluating $\partial \lambda_m / \partial s_{ji}$

$\frac{\partial \lambda_m}{\partial s_{ji}}$ corresponds to the derivative of the m-th eigenvalue with respect to the position of j-th (centered) landmark on the i-th training shape. In [67] this term is evaluated analytically using singular value decomposition (SVD).

The major attribute of the SVD based approach is that it does not require explicit calculation of the covariance matrix of training shape vectors but instead infers the principal shape components and the corresponding eigenvalues from the left orthogonal and diagonal matrices of $S = (L - \bar{L})/\sqrt{N_s - 1}$, respectively, where $L$ is a $3N_p \times N_s$ matrix whose column entries are the training shape vectors and $\bar{L}$ is a matrix with all columns set to $\bar{V}$, the mean shape vector (see 3.1.2). The shape matrix $S$ can be written as a product $S = U D W^T$ where $U$ and $W$ are left and right column orthogonal $3N_p \times N_s$ and $N_s \times N_s$ matrices, respectively, and $D$ is a $N_s \times N_s$ diagonal matrix of singular values. The columns of $U$ are the eigenvectors of $LL^T$ and the corresponding eigenvalues are the diagonal entries of $D$ squared ($\lambda_m = d_{mm}^2$).

In [67] it is shown that the derivative of the m-th singular value, with respect to $s_{kn}$ entry of $S$ can be evaluated using (E.1), where the scalars $u_{km}$ and $w_{nm}$ are elements of $U$ and $W$, respectively.

$$\frac{\partial d_{mm}}{\partial s_{kn}} = u_{km} \cdot w_{nm} \quad (E.1)$$

---

$^6$ $N_s =$ number of training shapes. $N_p =$ number of landmarks.

$^7$ $d_{mm}$ is equal to the standard deviation ($\sigma_m$) of the m-th eigenmode.
Recall that the shape vector, $V_i$, of the i-th training shape is composed of an ordered list of landmark co-ordinates such that $V_i = (v_1, v_2, v_3, ..., v_{N_P})^T$ where $v_j = (x_j, y_j, z_j)$.

Accordingly, the derivative of the m-th eigenvalue with respect to the j-th (centered) landmark on the i-th training shape is

$$
\frac{\partial \lambda_m}{\partial s_{ji}} = \begin{bmatrix}
\frac{\partial \lambda_m}{\partial s_{3(j-1)+1,j}}, & \frac{\partial \lambda_m}{\partial s_{3(j-1)+2,j}}, & \frac{\partial \lambda_m}{\partial s_{3(j-1)+3,j}}
\end{bmatrix}^T
$$

(E.2)

or more specifically

$$
\frac{\partial \lambda_m}{\partial s_{ji}} = 2d_{mm}w_{im} \cdot \left[u_{3(j-1)+1,m}, u_{3(j-1)+2,m}, u_{3(j-1)+3,m}\right]^T
$$

(E.3)

## F Algorithm for Minimization of the Description Length Cost Function

**Inputs:**
- Set of pre-registered training shapes, $T = \{M_i \mid i = 1, ..., N_T\}$
- Set of normalized conformal parameterizations, $P = \{f_i \mid i = 1, ..., N_P\}$
- Number of landmarks, $N_P$
- Optimization parameter, $\lambda_{cut}$
- Description length cost function, $DL(\ )$
- Convergence threshold, $\varepsilon$$.
- Maximum number of iterations, $N_{MAX}$
- Set of initial step sizes, $\{\Delta_i \mid i = 1, ..., N_P\}$.

**Output:**
- Set of registered parameterizations, $\{F_i \mid i = 1, ..., N_P\}$

1. Select a reference shape $M_{ref} \in T$ and adaptively sample its parameter domain with $N_P$ landmarks $L = \{x_j \mid j = 1, ..., N_P\}$ using the procedure described in Chapter 2 (subsection 2.3.4). Define $M_L$ as a triangulation of $L$.

2. Compute the centroids and standard deviations for four different scales of Gaussian kernel configurations (3, 10, 18 and 28 kernels) using the MATLAB Sphere Partitioning Toolbox [83,84].

3. Initialize $F_i = f_i$.
4. For every kernel configuration, starting with the one that has the largest radius of influence do the following:

\[ E_o = DL\left(\{F_i^{-1}(x_j)\}, \lambda_{cut}\right); \]
\[ k = 0; \quad dE = \text{Inf}; \]
while \( dE > \varepsilon \) AND \( k < N_{MAX} \)

\[ k = k + 1; \]
\[ \text{idx} = \text{RP}(N_S); \quad \text{randomly permuted sequence of numbers from 1 to } N_S \]
\[ E_{k0} = DL\left(\{F_i^{-1}(x_j)\}, \lambda_{cut}\right); \]

for \( j = 1: N_S \)

\[ n = \text{idx}(j); \quad \text{index of the current training shape} \]
\[ \text{if } \Delta_n < 10^{-8} \text{ OR } n = \text{ref, continue; end} \]
- Deform \( F_n \) using (3.36)
- Evaluate the cost function:

\[ E_j = DL\left(\left\{F_n^{-1}(x_j), \{F_i^{-1}(x_j)\}_{i \neq n}\right\}, \lambda_{cut}\right) \]
\[ \text{if } E_j > E_{k0} \]
\[ \Delta_n \leftarrow \Delta_n / 2 \]
- Do NOT accept the changes made to \( F_n \)
\[ \text{else} \]
\[ E_{k0} = E_j; \]
\[ \Delta_n \leftarrow 1.2 \times \Delta_n \]
\[ \text{end} \]

if \( \text{mod}(k, 10) == 0 \) \% adjust the pose of \( M_n \)

\[ T_n^* = \arg \min_{T \in SO(3)} DL\left(\left\{T \circ F_n^{-1}(x_j), \{F_i^{-1}(x_j)\}_{i \neq n}\right\}, \lambda_{cut}\right) \]
\[ M_n \leftarrow T_n^*(M_n) \]
\[ \text{end} \]

\[ E_k = E_{k0}; \]
\[ \text{if } k > 100, \quad dE = (E_{k-1} - E_k) / 50; \text{ end} \]
\[ \text{end} \]

5. Subdivide \( M_L \) and repeat Step 4 using new \( M_L \) vertices as landmarks. Stop if \( M_L \) has already been subdivided twice.
BIBLIOGRAPHY


