2013

Heterogeneous volumetric data mapping and its medical applications

Huanhuan Xu
Louisiana State University and Agricultural and Mechanical College

HETEROGENEOUS VOLUMETRIC DATA MAPPING AND ITS MEDICAL APPLICATIONS

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The School of Electrical Engineering and Computer Science

by
Huanhuan Xu
Double B.S., Central China Normal University, 2006
M.S., University of Science and Technology of China, 2009
M.S., Louisiana State University, 2012
August 2013

Acknowledgments

First of all, I would like to express my gratitude to my supervisor, Dr. Xin Li, for his guidance, support and encouragement throughout my research projects. I am much honored to have the opportunity to work with Dr. Li. I appreciate his dedication and patience.

I also want to express my special thanks to my dissertation committee, Professor Hongchao Zhang, Professor J.Ramanujam, Professor Bahadir Gunturk, and Professor Muhammad Wahab for patiently attending my general and final exams and providing me with their thoughtful and constructive comments. I also thank Professor Hongchao Zhang and Professor Shawn Walker for their valuable discussions and suggestions in mathematical models and algorithm design during this dissertation research.

Thirdly, I would like to acknowledge my colleagues, Shenghua Wan, Wuyi Yu, Kang Zhang, and Peizhi Chen for their warm friendship and their support in a variety of ways. Also I appreciate all my instructors in the School of Electrical Engineering and Computer Sciences and in the Department of Mathematics. Their classes had great impact on my knowledge.

I would like to thank my family for their love, support, and encouragement during this journey. They taught me to be self-motivated, have a strong work ethic and take pride in my work. Without their motivation and trust, this dissertation would not have been possible.

It was my honored to receive the Mark and Carolyn Guidry doctoral fellowship from the Electrical and Computer Engineering Department, provided by Mr. Guidry. I would also like to thank Amit Sawant from University of Texas Southwestern Medical Center for providing the medical image data.

Table of Contents

List of Tables

List of Figures

Abstract

With the advance of data acquisition techniques, massive solid geometries are being collected routinely in scientific tasks, these complex and unstructured data need to be effectively correlated for various processing and analysis. Volumetric mapping solves bijective low-distortion correspondence between/among 3D geometric data, and can serve as an important preprocessing step in many tasks in compute-aided design and analysis, industrial manufacturing, medical image analysis, to name a few. This dissertation studied two important volumetric mapping problems: the mapping of heterogeneous volumes (with nonuniform inner structures/layers) and the mapping of sequential dynamic volumes.

To effectively handle heterogeneous volumes, first, we studied the feature-aligned harmonic volumetric mapping. Compared to previous harmonic mapping, it supports the point, curve, and iso-surface alignment, which are important low-dimensional structures in heterogeneous volumetric data. Second, we proposed a biharmonic model for volumetric mapping. Unlike the conventional harmonic volumetric mapping that only supports positional continuity on the boundary, this new model allows us to have higher order continuity C^1 along the boundary surface. This suggests a potential model to solve the volumetric mapping of complex and big geometries through divide-and-conquer.

We also studied the medical applications of our volumetric mapping in lung tumor respiratory motion modeling. We were building an effective digital platform for lung tumor radiotherapy based on effective volumetric CT/MRI image matching and analysis. We developed and integrated in this platform a set of geometric/image processing techniques including advanced image segmentation, finite element meshing, volumetric registration and interpolation. The lung organ/tumor

and surrounding tissues are treated as a heterogeneous region and a dynamic 4D registration framework is developed for lung tumor motion modeling and tracking. Compared to the previous 3D pairwise registration, our new 4D parameterization model leads to a significantly improved registration accuracy. The constructed deforming model can hence approximate the deformation of the tissues and tumor.

Chapter 1
Introduction

1.1 Heterogeneous Volumetric Data Mapping

During the past decades, proliferation of 3D digital photographic/scanning devices and shape modeling techniques boosts the number of available high quality 3D geometric digital data. Massive volumetric models with many multiple attributes and complex geometry are collected and need to be processed.

Computing lowly distorted volumetric mapping is a powerful tool for processing the volumetric data because it provides one-to-one correspondence between two given objects. It serves as an important preprocessing step in many tasks in broad areas of computer-aided design and analysis, industrial manufacturing, medical image analysis, and etc. Therefore it becomes an important geometric problem in computer graphics and visualization.

Compared to the boundary surface data, solid volumetric data have richer contents which include material, intensity, or any other structural information. Effective volumetric parameterization is critical to process such data that have both boundary geometry and interior structure. But due to the much more complicated topological and geometric structures of 3-dimensional manifolds, existing volumetric mapping techniques are still inadequate.

In this work, we first propose an adaptive method to compute the feature-aligned harmonic volumetric mapping between two given volumetric data. Compared to the previous harmonic mapping, it supports three different type feature alignments: feature point, feature curves and iso-surface. Thus it can handle heterogeneous volumetric data mapping.

Since the complexity and the size of the volumetric data increasing rapidly, we like to solve the model by a divide-and-conquer way, which partitions the problem into solvable sub-domains. Then the continuity along the segmentation boundary interfaces becomes an issue since harmonic mapping only allows C^0 continuity. We further propose a biharmonic model for cross-object volumetric mapping which can provide C^1 continuity along the boundary surface. Therefore compared to the harmonic volumetric mapping, it is more suitable for the geometric models with complex geometry or heterogeneous interior structures.

1.2 Volumetric Data Matching for Medical Image Analysis and Radiotherapy Optimization

Lung cancer treatment is an important and challenging issue in the medical radiation therapy. Plans of radiotherapy treatments are developed based on static computed tomography (CT) images, while treatment is performed in moving organs due to respiration. This leads to a lack of precise knowledge of the actual position of the tumor and internal organs during treatment makes the calculation of actual dose absorbed by the lungs and surrounding tissues unknown. This will potentially cause some undesired damages to the around important organs.

The ideal radiotherapy guidance requires complete spatiotemporal knowledge of the movement and deformation of the volume–the region that includes the solid tumor and surrounding tissues and organs (lung) –to be treated. In this work, we model the lung and the inside tumor as heterogeneous volumetric data and take several lung/tumor objects in different time. Then we compute the volumetric mapping between these heterogeneous lung/tumor data.

More specifically, given a time sequence of 3D CT lung tumor images, we propose a computational framework for modeling of respiratory motion of the lung

tumor and its surrounding volumes. We build up a 4D parametric representation for the deforming volumetric region, so that their movement can be tracked, analyzed, modeled, and then predicted. Therefore the computational infrastructure can provide an accurate guidance for lung tumor radiotherapy treatment planning and delivery.

1.3 Overview and Contributions

Figure 1.1 illustrates the conceptual hierarchy of my research. First we develop some volumetric mapping algorithms suitable for the heterogeneous volumetric data which is the most common data used in our surrounding physical world. These algorithms can be applied into various applications: texture mapping, hex-remeshing, etc. However, in this dissertation, we will focus on the most challenging application: lung cancer treatment in the radiotherapy management and treatment.

First, we present an efficient adaptive method to compute the harmonic volumetric mapping, which establishes a smooth correspondence between two given solid objects of the same topology. We solve a sequence of charge systems based on the harmonic function theory and the method of fundamental solutions (MFS) for designing the map with boundary and feature constraints. Compared to the previous harmonic volumetric mapping computation using MFS, this new scheme is more efficient and accurate, and can support feature alignment and adaptive refinement. Our harmonic volumetric mapping paradigm is therefore more effective for practical shape modeling applications and can handle heterogeneous volumetric data. We demonstrate the efficacy of this new framework on handling volumetric data with heterogeneous structure and nontrivial topological types.

Second, we propose a biharmonic model for cross-object volumetric mapping. This new computational model aims to facilitate the mapping of solid models with

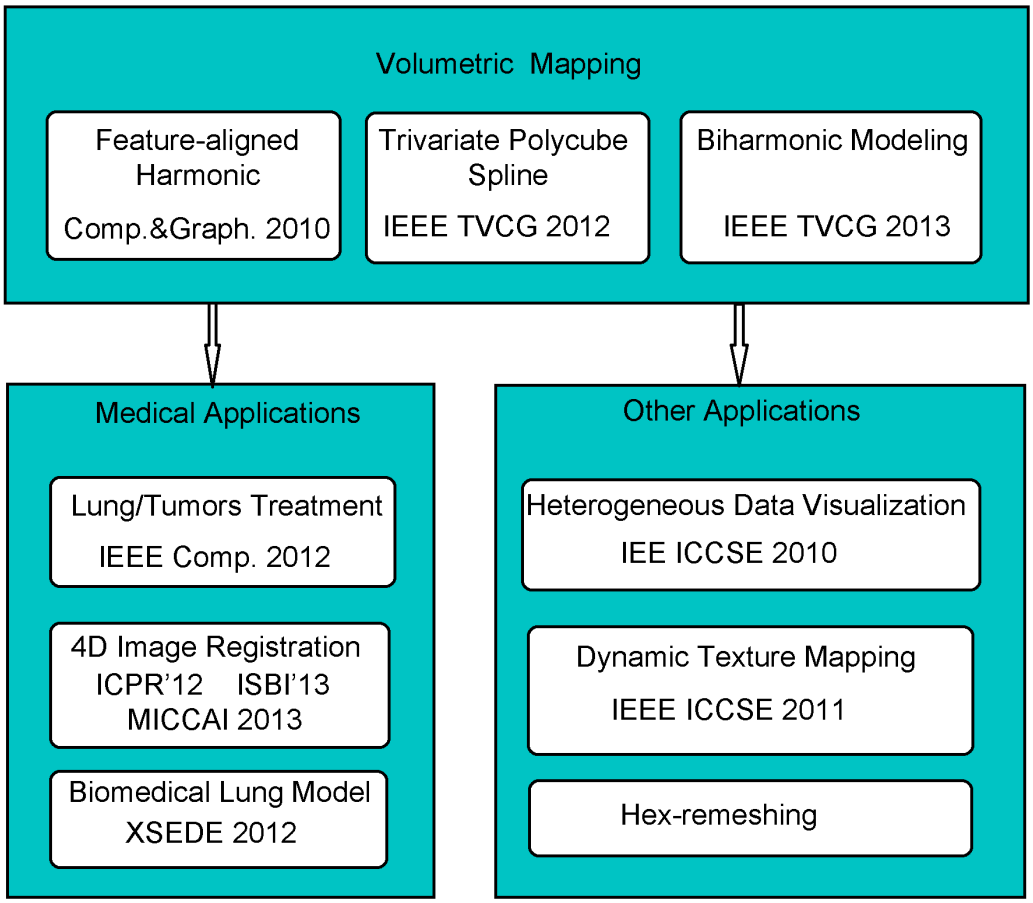


FIGURE 1.1. Hierarchy of this dissertation research: volumetric mapping research (upper row), and their applications (bottom row).

complicated geometry or heterogeneous inner structures. In order to solve cross-shape mapping through divide-and-conquer, solid models can be decomposed into sub-parts upon which mapping is computed individually. The biharmonic volumetric mapping can be performed in each sub-region separately. Unlike the widely-used harmonic mapping which only allows C^0 continuity along the segmentation boundary interfaces, this biharmonic model can provide C^1 smoothness. We demonstrate the efficacy of our mapping framework on various geometric models with complex geometry (which are decomposed into subparts with simpler and solvable geom-

etry) or heterogeneous interior structures (whose different material layers can be segmented and processed separately).

Third, we propose a computational framework for modeling the respiratory motion of lung tumors based on the biharmonic volumetric mapping. It provides a 4D parametric representation that tracks, analyzes, and models movement to provide more accurate guidance in the planning and delivery of lung tumor radiotherapy.

Last, we present an 4D image registration algorithm for lung tumor volume images. The registration will construct a deforming 3D model with continuous trajectory and smooth spatial deformation, and the model interpolates the interested region in the 4D (3D+T) CT images. The resultant non-rigid transformation is represented using two 4D B-spline functions, indicating a forward and an inverse 4D parameterization respectively. The registration process solves these two functions by minimizing an objective function that penalizes intensity matching error, feature alignment error, spatial and temporal non-smoothness, and inverse inconsistency. We apply this algorithm for respiratory motion estimation in clinic lung CT data. The experimental results demonstrate the efficacy of the algorithm.

1.4 Organization

This dissertation is organized as follows. Chapter 2 reviews the related work to this report. Chapter 3 introduces the feature-aligned harmonic volumetric mapping between two general 3D objects. Chapter 4 describes the biharmonioc volumetric mapping based on the decomposition models. Chapter 5 presents the medical imaging registration application. Chapter 6 proposes a 4D image registration framework for lung tumor motion tracking. Chapter 7 gives the conclusion of this dissertation.

Chapter 2
Related Work

2.1 Harmonic Maps and Surface Parameterization.

Surface mapping computes a one-to-one continuous map between a 2-manifold and a target domain with low distortions. It plays a critical role in various applications of graphics, CAGD, visualization, vision, medical imaging, and physical simulation. Having been extensively studied in the literature of surface parameterization, harmonic maps are usually addressed from the point of view of minimizing Dirichlet Energy. Its discrete version was first proposed by Pinkall and Polthier [5] and later introduced to computer graphics field in work of Eck et al. [6]. By discretizing the energy defined in [5], Desbrun et al. [7] constructed free-boundary harmonic maps. Surface maps that minimize harmonic energy or other stretch-distortion energy are directly used for shape blending [8] and in later shape morphing applications [9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19].

A lot of effective surface manipulation techniques and parameterization paradigms might be generalized onto 3-manifolds. A thorough survey on surface parameterization techniques is beyond the scope of this work, and we refer readers to nice survey reports of [20], [21], and [22] for details.

2.2 Volumetric Mapping

In recent years, volumetric mapping have gained great interest due to its rich applications in many fields such as computer-aided manufacturing [23], meshing [24, 25], shape registration [26, 27, 28], and trivariate spline construction [29, 30, 31]. Wang et al. [27] discretize the volumetric harmonic energy on the tetrahedral

mesh using the finite element method, parameterized volumetric shapes over solid spheres by a variational algorithm. Xia et al.[25] and Han et al.[24] use this discrete harmonic volumetric map in polycube parameterization. Most closely related to this work, in [28], we compute the harmonic volumetric mapping between two solid objects using fundamental solution methods. Later, we incorporate feature alignment in this volumetric mapping framework [3].

Volumetric interpolation is a powerful tool for shape deformation. Ju et al. [32] generalize the mean value coordinates [33] from surfaces to volumes to get a smooth volumetric interpolation for cage based deformation. Joshi et al. [34] present harmonic coordinates with non-negative weights for volumetric interpolation and deformation in concave regions. Martin et al. [29] parameterize volumetric model with trivial topology to a cylinder using the finite element method, and later generalize the algorithm [30] to more complicated models with medial surfaces. Lipman et al [35] develop Green's coordinates for volumetric deformation. Patanè[36] uses Radial Basis Function to approximate volumetric function along the volume data.

2.3 Shape Modeling using Biharmonic Functions

Biharmonic equation is a fourth-order partial differential equation used to minimize the thin-plate energy [37]. In recent years, it has been used to generate high-quality surface patch with the given positional and derivative boundary conditions [37, 38]. Relaxing each vertex of a discrete triangle mesh with harmonic weights or biharmonic weights can get a smooth surface [39]. Biharmonic equation is also used to do real-time mesh deformation [40, 41]. In [40], Helenbrook defines the boundary condition based on an explicit solution and shows the results on the 2D planar mesh. So it is mainly applied in fluid flows. Jacobson et al. [41] set the solution of biharmonic equation as the linear bending weights for 2D/3D shape deformation

and the implementation is based on the mixed finite element method. Lipman et al. [42] define the metric on 2D manifolds based on the biharmonic equation with the normal derivative equals to zeros along the boundary. Tankelevich et al. [43] compare surface reconstruction from points sets based on the solution of harmonic and biharmonic equations; they develop a special boundary condition for biharmonic equation such that it can be solved faster.

2.4 Boundary Method and MFS.

We construct the mapping through a meshless procedure by using a boundary method called *method of fundamental solution* (MFS). Notable work among boundary methods for solving elliptic partial differential equations (PDEs) includes the classical boundary integral equation and boundary element method (BIE/BEM), which has been widely used in many engineering applications [44], and was introduced into computer graphics for the simulation of deformable objects in [45]. One of the major advantages of the BIE/BEM over the traditional finite element method (FEM) and finite difference method (FDM) is that only boundary discretization is required rather than the entire domain discretization needed for solving the PDEs numerically. Compared with the BIE/BEM approach, the MFS uses only the fundamental solution in the construction of the solution of a problem, without using any integrals over boundary elements. Furthermore, the MFS is a meshless method, since only boundary nodes are necessary for all the computation. "Meshless" has the advantage of simplicity that neither domain nor mesh connectivity is required in storage and computation; so it becomes very attractive in scientific computing and modeling [46, 47]. A comprehensive review of the MFS and kernel functions for solving many elliptic PDE problems was documented in [48].

2.5 Registration in Lung Radiation Therapy

Recent advances in Image-Guided Radiation Therapy emphasize the capability of fast treatment-time tumor localization while using low imaging radiation dose. Long et al. and Brock et al. used cubic B-splines as the respiratory motion model in the 3D/2D registration framework and showed accurate estimations of the treatment-time Deformation Vector Fields (DVFs) by using limited-angle radiographs [49] and a small number of radiographs [50] respectively. Li et al. [51] extended this framework and built the respiratory motion model by doing Principal Component Analysis (PCA) on the 4D Respiratory-Correlated CTs acquired at treatment simulation. They used a GPU-accelerated gradient-descent optimization scheme that showed accurate tumor localization with a single radiograph.

There also some global methods are used to model the respiratory process[52, 53]. They estimate the lung motion by establishing temporal correspondences, e.g., between the maximum inhale phase and all other phases. [54] propose a unified approach to estimate the consistent respiratory lung motion by using iterative steps.

However, traditional gradient descent optimization approaches suffer from complexity in computing the image Jacobian, and they also require a well-defined convex metric to guarantee that the scheme reaches the global optimum.

Moreover, all previous method treat the inside tumor as a homogeneous material as lung part to simplify the problem. In practical, the material of tumor is not the same with the lung part. The elastic properties of these two part are different. Treat them differently will improve the accuracy of tumor motion tracking.

In our work, we are using 3D computer graphics algorithms to model the motion of tumor and neighboring environment. We treat the tumor as an iso-surface struc-

ture in the lung part and we compute their registration separately. It can achieve better result, both in accuracy and efficiency.

2.6 4D Image Registration

Compared with the conventional image registration techniques, 4D spatiotemporal registration can avoid the bias caused by a predetermined reference frame, and can enforce both spatial and temporal smoothness of the transformations, which indicates physically nature deformations [55]. However, the study of 4D registration methods has just started and is still far from adequate. First, existing 4D registration methods [52] are mainly guided by image intensity. The computation therefore reduces to minimizing a non-linear problem having many local minima, which also usually has high computational cost and, more importantly, requires a good initial guess to reach a desirable matching. Geometry-guided approaches such as using feature alignment constraints can effectively guide this optimization to avoid many undesirable local minima. In this work, we develop a symmetric framework for 4D image registration. With feature-alignment constraints and new optimization scheme, we can get more accurate and effective registration.

Chapter 3
Feature-aligned Harmonic Mapping

3.1 Introduction

The rapid advancement of 3D scanning techniques makes it easier to acquire massive 3D data nowadays. When data sets can be acquired in an explosive rate, computational techniques only evolve modestly. As a result, 3D data matching, analyzing, and searching become bottleneck for their efficient processing. Compared with 2D images, 3D shapes have many distinctions including larger sets of degrees of freedom and spatial variations in terms of geometry, topology, feature, and material. A viable approach for the effective shape matching and analyzing is to establish the correspondence between objects of interest, which can be computed by either solving a non-rigid bijective registration between given objects or composing two parameterizations from both objects onto one common domain. The key is to compute a mapping from one domain to another. When it is enough to purely consider boundary surfaces of the 3D data, one can focus on mapping 2d-manifolds (surfaces). Surface mapping seeks a bijection between two 2-manifolds with similar topology, aiming for least distortion (using length-, angle-, or area- preserving as the criterion) which dictates its effects in applications. Surface parameterization and inter-surface mapping have been extensively studied, playing important roles in computer graphics, and serving as ubiquitous tools for many valuable applications. For example, in computer graphics, it has been used for texture mapping, texture transfer, and morphing animation. In geometric modeling, it has been used for detail transfer, surface editing, mesh simplification. In CAGD, it has been used to construct the parametric domain for continuous representations such as splines.

In visualization, complicated geometric structures may be better visualized and analyzed by mapping surfaces and their properties to a simpler domain. In vision and medical imaging, it has been used for surface matching, data completion, and so on. Surveys of surface mapping and their applications are given in [20] and [21].

Solid volumetric data have richer contents than those of the boundary surface. When the data processing or analysis are related to material, intensity, or any other structural information defined over the whole 3D region of the object (instead of on just its boundary shell), we need to consider the shape as a 3-manifold and study the volumetric mapping. Therefore, volumetric mapping can also benefit aforementioned applications. Because of its importance, volumetric mapping and parameterization has gained greater interest in recent years, and a few related research work has been conducted towards various applications such as shape registration ([27, 28, 2]), volumetric deformation ([32, 34, 35, 56]), and trivariate spline construction ([29]), and so on. Although many valuable concepts and demos have been presented, all indicating the importance of this technique, its study has just started and is far from adequate. Several key limitations of existing algorithms prevent them from being applied into real applications with complex scenarios.

Generality. It is desirable that the mapping is general and can handle 3D shapes with variant topological types. Volumetric data from real scenarios usually have nontrivial topology, and most existing parameterization techniques ([27, 29]) focus on topological solid-sphere shapes. [28, 2] used the fundamental solution methods to compute harmonic volumetric mapping between 3D objects with general topology.

Efficiency. Solving the discretized vector field over a 3D voxelized domain or over a tetrahedral mesh usually is much slower than the surface mapping computation. The fundamental solution method of [28] is a boundary method. It reduces

the volumetric mapping computation from the whole 3D domain to the degree of freedom with the boundary size, to be solved by a linear system of equations. However, it is still very time consuming to solve because the coefficient matrix is dense and ill-conditioned.

Heterogeneity. Most existing methods consider the volumetric mapping from homogeneous viewpoints and only compute the mapping purely based on geometry, without taking into account the interior structure and features. It is desirable to develop the capability of the mapping algorithm that can accommodate heterogeneous structures and integrate domain expertise in geometric modeling and processing.

In order to tackle these aforementioned limitations, this work improves the algorithm of fundamental solution methods in mapping computation ([2]), and seeks a general and effective mapping computation algorithm with better efficiency, accuracy, and heterogeneity. We compute harmonic volumetric mapping by improving the fundamental solution methods of [2], and the side-by-side comparison shows that our new approach is more efficient and accurate. Furthermore, it supports feature alignment, which is important for many practical volumetric data processing tasks.

The main contributions of this work include:

- We use multiple fundamental solution systems and an adaptive refinement scheme for the computation of harmonic volumetric mapping. Compared to [2], this computation efficiency is greatly improved, so that large and complex data can be parameterized in the new framework. In the mean time, with an adaptive sampling scheme, the new computation also converges to a better boundary fitting result in salient manners.

- Our feature alignment scheme supports the computation of volumetric mapping composed by constrained harmonic functions that allow the alignment between various types of features including 0-manifolds (feature points), 1-manifolds (feature lines, such as skeletons), 2-manifolds (iso-surfaces).

The remainder of this chapter is organized as follows. We introduce the theory and algorithms of our methods in Section 3.2, and address important implementation issues in Section 3.3. In Section 3.4, we demonstrate some experimental results, discuss and compare our algorithms with existing volumetric mapping methods, especially [2], and show the large efficiency/accuracy improvement over the current method. We also show a direct application on hex meshing. Finally, we conclude this chapter in Section 4.7.

3.2 Theory and Algorithm

A volumetric map $\vec{f}$ between two 3-manifolds embedding in $\mathcal{R}^3$ is a bijective mapping $\vec{f} : M_1 \to M_2, M_1 \subset \mathcal{R}^3, M_2 \subset \mathcal{R}^3$. The boundary constraint is a surface mapping $\vec{f'}$ from the boundary surface of the first solid object M_1, denoted as ∂M_1, to the boundary surface of M_2, denoted as ∂M_2. The mapping $\vec{f}(\mathbf{p}) = \mathbf{q}$ ($\mathbf{p} \in M_1$, $\mathbf{q} \in M_2$) is composed by three real functions in three axis directions, i.e., $\vec{f} = (f^1, f^2, f^3)$. Each real function $f^i, (i = 1, 2, 3)$ maps the point $\mathbf{p}$ to $\mathbf{q}(q_1, q_2, q_3)$'s corresponding component q_i. This problem is then reduced to the computation of real functions f^i, $(i = 1, 2, 3)$, with the given boundary surface mapping constraints $\vec{f'} = (f'^1, f'^2, f'^3)$. We want the volumetric mapping to follow the boundary constraints and minimize a specific metric distortion. In this work, our object is to minimize the harmonic energy under the Dirichlet boundary condition discussed above, defined by the boundary surface mapping.

3.2.1 Harmonic Volumetric Mapping

Harmonicity of a mapping characterizes the smoothness of the transformation, which is a natural phenomenon that depicts the minimized physical energy that arises from the difference between two shapes. In the surface case, a harmonic map (with boundary loop mapping predetermined) finds the functions with the vanishing Laplacian everywhere, and it minimizes the Dirichlet energy and leads to a minimal surface ([5, 6]). Intuitively speaking, finding a harmonic map between two surfaces with fixed boundary correspondence is like computing the physical deformation of a rubber membrane. The membrane has the source surface as its relaxed shape configuration, and is arbitrarily wrapped onto the target shape with the boundary constraint enforced; then we let go the membrane so that it freely flows over the target shape, its final status indicates a harmonic map.

Similarly, for harmonic volumetric mapping, if we fix the boundary map which is now a surface mapping between shells of the two given solid objects, we are computing the smooth mapping of the interior region by enforcing the vanishing $3D$ Laplacian. This is equivalent to computing the final stable configuration of a solid rubber subject to its boundary surface constraint.

In formulation, given two volumetric regions $M_1 \subset \mathcal{R}^3$ and $M_2 \subset \mathcal{R}^3$ and a one-to-one mapping $\vec{f}'$ between their boundary surfaces ∂M_1 and ∂M_2: $\vec{f}'(\mathbf{p}) = \mathbf{q}$, $\mathbf{p} \in \partial M_1, \mathbf{q} \in \partial M_2$, we seek a mapping $\vec{f} : M_1 \to M_2$ such that

$$\begin{cases} \Delta \vec{f}(\mathbf{p}) = 0 & \mathbf{p} \in M_1, \\ \vec{f}(\mathbf{p}) = \vec{f}'(\mathbf{p}) & \mathbf{p} \in \partial M_1. \end{cases}$$

where Δ is the Laplace operator, defined for real function f in $\mathcal{R}^3$ as

$$\Delta f = \nabla \cdot \nabla f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2},$$

and $\Delta \vec{f} = 0$ for $\vec{f} = (f^1, f^2, f^3)$ is equivalent to $\Delta f^i = 0$ for all $i = 1, 2, 3$.

Variational approaches have been proposed ([27]) for solving the harmonic volumetric map over tetrahedral meshes. However, like the surface mapping, the quality of numerical solution in this type of methods heavily depends on the mesh quality. In the surface case, it is well known that the discrete harmonic map [6] could lead to non-bijective mapping locally when skinny triangles exist and cotangent weights become negative. Similarly, harmonic weights ([27]) defined over edges of a tetrahedral mesh, which are derived via finite element analysis has this same problem. It has been proven that if the mesh satisfies the Delaunay criterion, then even it contains obtuse triangles, the parameterization obtained using the cotangent weights will be bijective. However, to our best knowledge, there is no similar result on the discrete weight over tetrahedral meshes. A mesh-free procedure is more attractive for irregular geometries specifically with nontrivial topology/structures due to its flexibility and simplicity.

On the other hand, the linear nature of Laplacian equations indicates that the boundary-based methods are most suitable since the interior is now determined in an exact manner. In other words, according to the maximum principle of harmonic functions, the value of a harmonic function never reaches maximal or minimal values in the interior region of the domain, and values in these interior regions are fully determined by the boundary condition. The *method of fundamental solution* (MFS), based on the Green's theory is a natural boundary mesh-free method to solve this problem. MFS can be viewed as a modified Trefftz method, and the basic idea is to approximate the solution by a linear combination of fundamental solutions with sources located outside the problem domain. [28, 2] applied MFS in the computation of harmonic volumetric mapping, where three linear systems with one single coefficient matrix are solved to get the harmonic volumetric mapping between two 3D objects.

Compared to mesh-based variational methods such as [27], (1) the MFS method is a meshless boundary method, which is more efficient than this conventional mesh based FEM method, and with both time complexity and storage complexity greatly reduced; (2) MFS is more general and can flexibly handle volumetric data sets with complicated topologies, including topological noise; (3) the new MFS framework can also handle heterogeneous materials instead of just homogeneous shapes.

3.2.2 Method of Fundamental Solutions

We briefly review the idea of MFS in solving harmonic volumetric maps and define the notations that are used in our algorithms.

MFS in Harmonic Volumetric Mapping. We seek three harmonic functions $(f^1, f^2, f^3) : M_1 \to M_2$, with $\Delta f^i = 0$. Since Δ is a linear self-adjoint differential operator and M_1 is a bounded domain in $\mathcal{R}^3$, we can compute its Green function. A fundamental solution of this differential equation is a function $K(\mathbf{x}, \mathbf{x}')$ such that

$$\Delta K(\mathbf{x}, \mathbf{x}') = \delta(\mathbf{x}, \mathbf{x}'), \mathbf{x}, \mathbf{x}' \in \mathcal{R}^3,$$

where $\delta(\mathbf{x}, \mathbf{x}')$ is the Dirac delta function, the kernel K is defined everywhere except the singularity point at $\mathbf{x} = \mathbf{x}'$.

Then we have

$$f^i(\mathbf{x}) = \int K(\mathbf{x}, \mathbf{x}') g^i(\mathbf{x}') d\mathbf{x}'.$$

Such a kernel function K is known as the Green's function associated with the 3D Laplacian operator Δ, and has the formula: $K(\mathbf{x}, \mathbf{x}') = \frac{1}{4\pi} \frac{1}{|\mathbf{x}-\mathbf{x}'|}$, where $|\mathbf{x} - \mathbf{x}'|$ denotes the distance between the points $\mathbf{x}$ and $\mathbf{x}'$.

Following this scheme, solving the aforementioned harmonic mapping $\vec{f}$ is like designing electric fields. For each harmonic function f^i (for each axis direction),

we compute a particle system. The outcome electric potential field of any particle system is always a real harmonic function (guaranteed by the Kernel function), and we only need to find the particle system that fits the boundary condition, indicating the boundary surface mapping, coupled with three axis components. This process of solving the best particle system simulates the computation of f^i.

Suppose we have a particle system, and consider an electronic particle $\mathbf{Q_s}$ (called a singularity point, or a *source point*) outside the domain M_1, the corresponding fundamental solution for 3D Laplacian equation (i.e. its potential) on a point $\mathbf{p}$ can be formulated as

$$K(\mathbf{p}, \mathbf{Q_s}) = \frac{1}{4\pi} \frac{1}{|\mathbf{p} - \mathbf{Q_s}|}, \tag{3.1}$$

where $|\mathbf{p} - \mathbf{Q_s}|$ denotes the distance between the point $\mathbf{p}$ and this particle $\mathbf{Q_s}$.

Therefore, considering the entire particle system $\{\mathbf{Q_s}\}$ with a set of source points, the MFS equation to evaluate f^i on an interior or boundary point $\mathbf{p}$ is

$$f^i(\vec{w^i}, \vec{\mathbf{Q}}; \mathbf{p}) = \sum_{n=1}^{n_s} w_n^i \cdot K(\mathbf{p}, \mathbf{Q_n}), \mathbf{p} \in M_1, \tag{3.2}$$

where suppose we have n_s source points in the exterior of M_1, $\vec{\mathbf{Q}}$ is the $3n_s$-dimensional vector concatenating positions of all n_s $3D$ source points, and $\vec{w^i} = (w_1^i, w_2^i, \cdots, w_{n_s}^i)^T$ is the n_s-dimensional vector to be determined, which indicates charge amount distribution on these source points.

Boundary Fitting. When every source point $\mathbf{Q_n} \in \mathcal{R}^3, n = 1, \cdots, n_s$ is outside of M_1, any charge distribution guarantees the vanishing Laplacian $\Delta f^i(\mathbf{p}) = 0, \forall \mathbf{p} \in M_1$, only that f^i might violate the boundary conditions. Source points $\{\mathbf{Q_n}\}$ should lie outside of M_1, namely, locate on the boundary surface $\partial\widetilde{M_1}$ of a region $\widetilde{M_1}$ that contains M_1 (i.e. $M_1 \subset \widetilde{M_1} \subset \mathcal{R}^3$). In [2], an offset surface $\partial\widetilde{M_1}$ of M_1 is created (by first computing implicit distance field $d(\partial M_1)$ with respect to ∂M_1 ([57]) and then generating the polygonization ([58]) on the implicit sur-

face $d(\partial M_1) + \delta = 0$), and a set of source points are uniformly sampled on this $\partial\widetilde{M_1}$. Then it solves the charge amount on each particle such that the potential field approximates the boundary condition. The boundary condition is the surface mapping, whose fitting is conducted over a set of evaluation points (also called the *collocation points* or *constraint points*) $\{\mathbf{p_i} \in M_1\}$. Three particle systems with their charge distribution solved in this fitting process compose the volumetric map $\vec{f} = (f^1, f^2, f^3)$.

Limitations of the aforementioned routine. The above algorithm of [2] has following two key limitations:

- **Computation Efficiency and Fitting Accuracy.** Three dense linear systems need to be solved. Suppose we have n_c collocation (evaluation) points and n_s source points, we have a $Aw^i = b^i$ system where the dimension of the coefficient matrix A is $n_c \times n_s$. A is dense since every source point contributes to every constraint point. Furthermore, A is ill-conditioned. As suggested in [59], [28], [2], *Singular Value Decomposition* (SVD), due to its stableness against the ill-conditioned system, is chosen to solve this system. However, SVD decomposition is slow for large matrices. For example, the solver of [2] needs more than one day when both n_c and n_s exceed $20K$ vertices. Therefore, when handling complex volumetric data, we have to restrict n_c and n_s. This causes the salient decrease on the boundary fitting accuracy. Because now we either (1) lack enough particles for designing fine potential fields to well fit the boundary condition (when n_s is picked to be small), or (2) lack enough evaluation points to sample the shape variance on the boundary (when n_c is picked to be small).

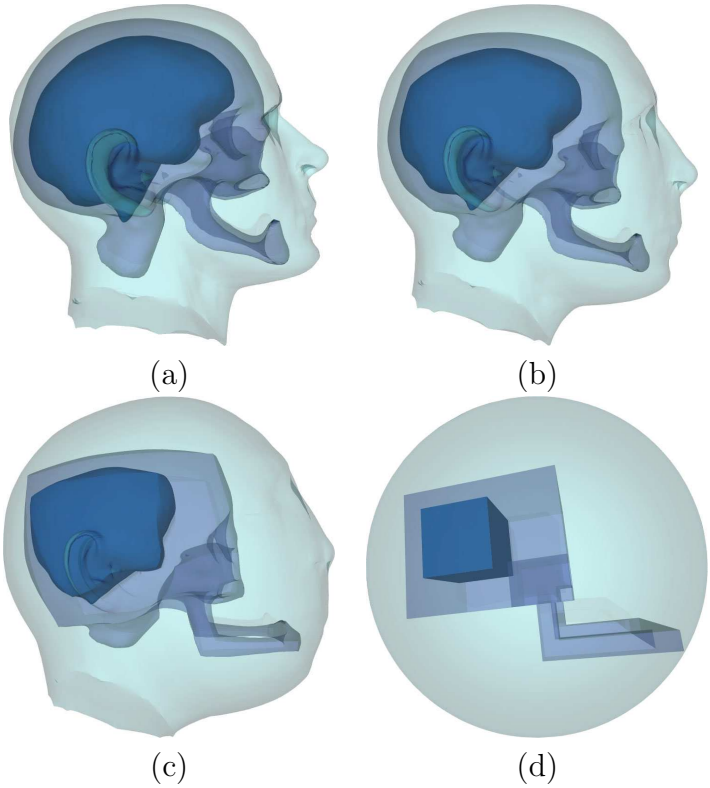


FIGURE 3.1. Heterogeneous Volumetric Mapping between Head-Skull-Brain and Polycube-Sphere. (a) The extracted and cleaned volumetric shape has three salient iso-surfaces: head, skull, and brain. A target domain (d) is generated to test the efficacy of our mapping with iso-surface constraints. (d) has a sphere, a polycube skull, and a cube inside, corresponding to three iso-layers in (a). (b) and (c) show the 30% and 60% morphing from (a) to (d), generated by linear interpolation.

- **Feature Alignment.** Like other volumetric mapping methods ([27, 32]), [2] focuses on homogeneous volumetric regions where boundary surface map is the only constraint in the mapping computation. In real scenarios, volumetric data usually contain different materials and densities, or have salient structure inside its interior region (See Fig. 3.1(a) for example). These information or structures are usually meaningful and should be considered. Therefore, a scheme that can properly handle heterogeneous structure is worthwhile, so that we will be able to align or match similar material/intensity when necessary.

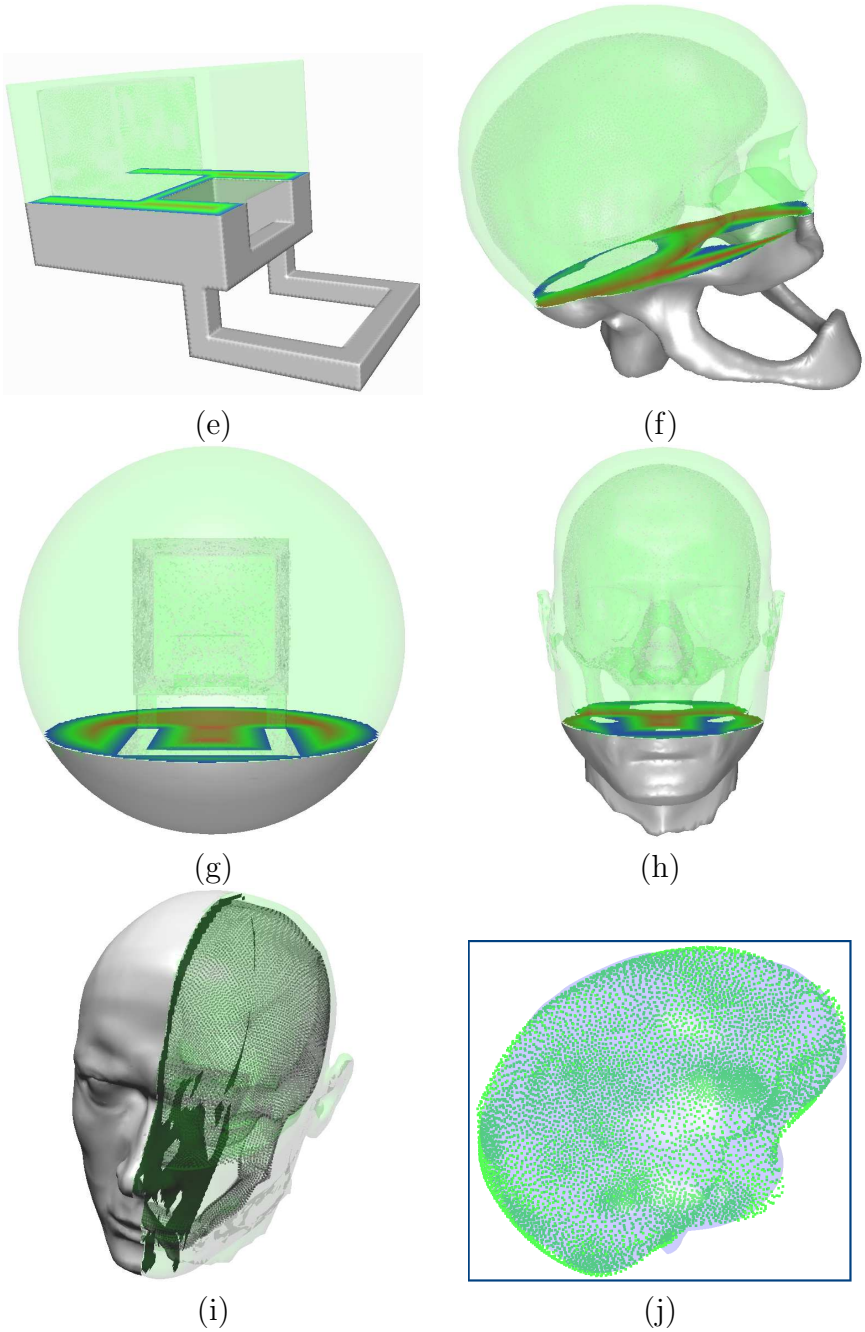


FIGURE 3.2. Heterogeneous Volumetric Mapping between Head-Skull-Brain and Polycube-Sphere (cont.). (e,g) show cross-sections on the polycube-sphere domain, (f,h,i) show corresponding cross-sections on the head-skull-brain model. The point clouds in (e)-(i) are the sampled feature points on these iso-surfaces (e,g) and their images (f,h,i) of the volumetric mapping. The color-encoding in (f,h) visualizes the mapping via the transferred distance field of (e,g). In (j), we zoom in the brain iso-surface (the grey transparent surface) and its fitting, green points are images of sampled points on the interior cube in (e). The RMSE here is 0.57%.

New Computation Scheme. To tackle these two limitations, we improve [2] as follows.

- Instead of one function $\vec{f} : M_1 \to M_2$, we compute a set of harmonic functions $\vec{f_i}$ such that their sum approximates $\vec{f}$. The computation for each $\vec{f_i}$ is more efficient and numerically more stable. Each $\vec{f_i}$ is harmonic, and therefore their sum $\sum_i \vec{f_i}$ is also harmonic. Each subsequent $\vec{f_i}$ aims to refine the existing map $\sum_{k=0}^{k=i-1} \vec{f_k}$ towards the exact boundary condition $\vec{f'}$ (which could need a too big system [28] to solve within only one shot). Now we use less constraint points and source points to compute each $\vec{f_i}$, and therefore the solving is much faster, the boundary condition for each $\vec{f_i}$ is $\delta f_i = \vec{f'} - \sum_{k=0}^{k=i-1} \vec{f_k}$. This greatly improves the **speed** of the mapping computation, and makes the MFS practical for large volumetric data.

- As we will demonstrate in our experiments, with only a few $\vec{f_i}$, the fitting accuracy can usually beat the algorithm of [28], and we can keep refining it using more $\vec{f_i}$ when necessary. Also, unlike the [2] that conducts uniform sampling over the offset surface ∂M_1, we conduct the sampling adaptively following the geometry of the given shapes. Together with the multi-MFS scheme, our scheme intuitively allows a flexible placement of source points. It is well known that in MFS, the location of source points constitutes a key issue and it has large impact on numerical stability of the MFS computation. The locations of source points are either preassigned or determined along with the coefficients of the linear combination. Most papers place source points on a surface outside M_1 uniformly and solve least square linear systems, but this not always guarantees that the computed solution converges to the exact solution as the number of source points increases. The MFS with moving

source points has been considered by several authors (e.g. [60], [48]). This leads to a slow nonlinear optimization, and still, it has been reported that the initial placement of the sources is usually very important in the convergence of these algorithms, as they converge to the first local minima encountered.

In this work, we also follow the preassign-approach that leads to linear systems. Unlike the placement scheme in [2], we allow the removal and adaptive adding-in of new source points according to the result in the previous round, and this mimics the adjustment of the source points during the MFS solving. We will show that our new scheme also improves the **accuracy** of the mapping computation using MFS.

- **Feature Alignment.** We allow the setting of **constraint points** during the mapping. In real applications, three types of constraints are very useful: 0-manifolds (feature points), 1-manifolds (feature curves or skeletons), and 2-manifolds (iso-surfaces). These constraints are treated as a part of boundary fitting. We apply an adaptive scheme to balance the feature constraint and boundary constraint.

3.2.3 Algorithm Pipeline

Our algorithm pipeline is as follows. The input is two given solid objects M_1, M_2 and their boundary surface mapping

$$\vec{f'} = (f'^1, f'^2, f'^3) : \partial M_1 \to \partial M_2.$$

The output is a harmonic volumetric mapping composed by a set of harmonic functions:

$$\vec{f} : M_1 \to M_2 = \sum_{i=1}^{i=n_f} \vec{f_i} = \sum_{i=1}^{n_f} (f_i^1, f_i^2, f_i^3),$$

such that on the boundary surface $\mathbf{p} \in \partial M_1$, $\vec{f}(\mathbf{p}) = \vec{f'}(\mathbf{p})$ and in the interior region: $\nabla^2 \vec{f} = 0$.

Each harmonic real function $f_i^j, i = 1, \ldots, n_f, j = 1, 2, 3)$ is solved by one linear system $\mathbf{A}_i^j \mathbf{w}_\mathbf{i}^\mathbf{j} = \mathbf{b}_i^j$. In the following algorithm, we omit the indices i and j (e.g. using the notation $\mathbf{A}$ instead of $\mathbf{A}_i^j$) for simplicity, assuming this will not cause any ambiguity:

1. Place source points and collocation points. (Section 4.1)

2. Compute the coefficient matrix $\mathbf{A}$, whose $(u, v)th$ element $\mathbf{A}_{uv} = K(\mathbf{P_u}, \mathbf{Q_v})$ (Equation (3.1)) for the collocation point $\mathbf{P_u}$ and source point $\mathbf{Q_v}$.

3. Decompose $\mathbf{A}$ using Singular Value Decomposition $\mathbf{A} = \mathbf{U\Sigma V}^*$. The decomposed results $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^*$ are used to solve the fitting system.

4. Set the boundary condition $\mathbf{b}$ at the right hand side of $\mathbf{A}_{uv}\mathbf{w} = \mathbf{b}$, and $\mathbf{b} = \{b_k\}$, where b_k is the boundary constraint evaluated on each collocation point.

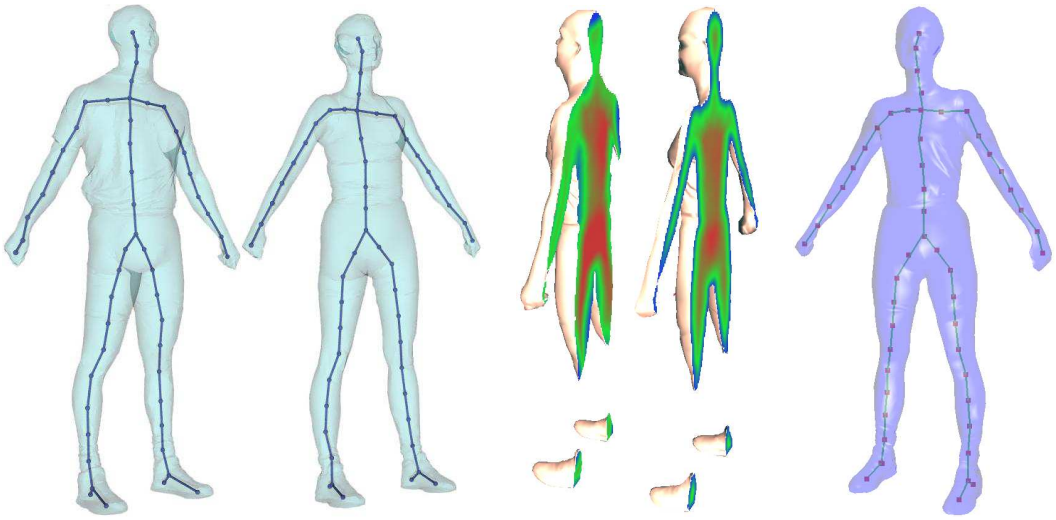
For each $\vec{f}_i$, this algorithm solves $\mathbf{A}_i^j \mathbf{w}_\mathbf{i}^\mathbf{j} = \mathbf{b}_i^j$. When $i = 1$, the boundary condition is set to be a low-resolution surface mapping from M_1 to M_2. For $i > 1$, we use a higher-resolution surface mapping, and also apply the refined boundary fitting $\delta\vec{f}' = \vec{f}' - \sum_{i=1}^{i=n_f-1} \vec{f}_i'$.

Note that our algorithm takes the boundary surface mapping $\vec{f}'$ as an input. We briefly discuss how to obtain such a surface mapping In Section 4.3.

3.3 Implementation and Discussion

3.3.1 Source Points and Collocation Points Placement

In order to set up the coefficient matrix for boundary fitting, first we need to place source points and collocation points. The n_s source points $\tilde{\mathbf{Q}} = \{\mathbf{Q_1}, \mathbf{Q_2}, \ldots, \mathbf{Q_{n_s}}\}$ are particles in the exterior of M_1 and n_c collocation points $\tilde{\mathbf{P}} = \{\mathbf{P_1}, \mathbf{P_2}, \ldots, \mathbf{P_{n_c}}\}$ are evaluation points on the boundary ∂M_1. We solve the weights (charge amount)



(a) (b) (c)

FIGURE 3.3. Volumetric Mapping from the solid Cyberware Male model to Female model. Their skeletons are illustrated in (a). The mapping with skeleton matched are illustrated in (b) via one cross section: the distance field color-encoding on the female (right) model is transferred to its corresponding point on the male (left) model. (c) shows the skeleton fitting: the green curve shows the target skeleton (i.e. the skeleton of the female), images of sampled points on the male skeleton are visualized as red points. The skeleton RMSE fitting error is 0.54%.

distribution $\mathbf{w_i}, i = 1, \ldots, n_s$ on all source points $\tilde{\mathbf{Q}}$ so that $\vec{f}(\mathbf{P_i})$ satisfies the boundary condition approximately.

The distribution of source and collocation points greatly affects the numerical stability and therefore the mapping efficiency and quality. The boundary error is sensitive to the collocation and source points, so appropriately sampling $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{P}}$ is critical. In 2D cases, theoretic studies have been conducted for analytical and simply connected domains, for example, when $M_1 \subset \mathcal{R}^2$ is a planar disk [60], uniformly sampling both collocation and source points is ideal and leads to exponential decreasing on boundary fitting errors. When $M_1 \subset \mathcal{R}^2$ is analytic, there is also discussion on the existence of optimal placement [61], and one suggestion is to take a conformal mapping Ψ from the unit disk D to M_1, and place $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{P}}$ on Ψ's images of the evenly sampled points on the parametric circle. However, for more complicated domain shapes in $2D$, or in our 3D case that $M_1 \subset \mathcal{R}^3$,

TABLE 3.1. Source and collocation points placement using geometry-adaptive (GA) sampling and uniform (UN) sampling. The experiment is conducted on mapping vase-lion model with $40k$ vertices to a solid sphere, $cRatio = 0.05, sRatio = 0.05$, and the offset surface is 0.15 times object size distant. The boundary fitting error indicates the boundary mapping quality. Using geometry-adaptive sampling leads to less boundary error.

S-Pts	C-Pts	Boundary Fitting Error	Collocation Error
UN	UN	0.0563971760	0.0499722402
GA	UN	0.0546669675	0.0270761958
UN	GA	0.0544260284	0.0469717138
GA	GA	0.0484185840	0.0449218356

the optimal placement is still unknown. [2] shows that placing source points on a nearby offset surface produces more accurate mapping result. We also adopt the offset surfaces scheme but add in adaptivity, following both the geometry and sequential fitting errors.

3.3.2 Sampling Collocation Points

In 2D analytical boundary scenarios ([48]), uniformly sampled $\tilde{\mathbf{P}}$ usually leads to a stable system and good approximation, therefore is suggested as the strategy for preassigning collocation points. [2] follows this strategy, and uses the uniform sampling scheme of [62] to generate evenly distributed collocation points on ∂M_1. The total number of collocation points (source points) is controlled by an aspect ratio $cRatio = n_c/n(\partial M_1)$ ($sRatio = n_s/n(\partial M_1)$), where $n(\partial M_1)$ is the total vertex number of ∂M_1.

However, our multi-level MFS solving shows that for most 3D piecewise-linear domains, adaptively sampling these collocation points following local geometry could lead to better convergence on boundary fitting errors. Intuitively, more evaluation points shall be placed on highly-detailed regions for better sampling the boundary variance. Our geometry adaptive sampling algorithm is as follows:

1. Tessellate the boundary surface where we need to do the sampling: it is the domain boundary ∂M_1 for collocation points, and offset surface $\partial\widetilde{M_1}$ for source points.

2. Refine M_1 (for collocation points sampling) or $\partial\widetilde{M_1}$ (for source points sampling) by subdivision and get a dense mesh $\partial M'$.

3. Conduct surface simplification on $\partial M'$ using the quadric error metric [63], which efficiently produces a good-quality approximated simplified mesh $\partial M*$. The vertex number of $\partial M*$ is determined by our sampling budget.

4. Vertices of mesh $\partial M*$ are used as sampling points.

Table 3.1 illustrates our experiments conducted on the spherical mapping of a vase-lion model with $40k$ vertices, and the mapping is for $cRatio = 0.05, sRatio = 0.05$, and the offset distance is 0.15 times object size. It clearly shows the advantage of geometry-adaptive sampling over uniform sampling in placement of both collocation points and source points.

3.3.3 Sampling Source Points

We place source points following three aspects.

- **Geometry-adaptive Sampling.** In the coarsest level (f_0 computation), we conduct geometry-adaptive sampling on source points to determine their locations.

- **Even Partitioning.** In finer levels, we partition sampled source points $\{\mathbf{Q_v}\}$ into several subsets evenly. Each time we only use a subset of charge points for efficient boundary fitting.

- **Adaptive Deletion/Insertion.** Meanwhile, in each step, we remove redundant source points by analyzing the diagonal matrix from the SVD decomposition (see Section 4.2 for details), and adaptively add in extra source points near the regions with large fitting errors by projecting badly fitted boundary points onto the offset surface.

3.3.4 Solving MFS Linear Systems by SVD

The boundary fitting is reduced to solving linear systems $\mathbf{A}\vec{w} = \vec{b}$. $\mathbf{A}$ can be dense and ill-conditioned [59], so regular linear system solvers such as Gaussian elimination, LU, and QR decompositions usually fail to produce a stable solution. As suggested in [59] and [2], we use Singular Value Decomposition (SVD) to decompose $\mathbf{A}$. There are three reasons.

1. It generates accurate and stable results when the coefficient matrix is highly ill-conditioned.

2. It flexibly gets to the least-square solution for over constrained boundary conditions (which always happen in our multi-round MFS solving).

3. Furthermore, in our approach, we also use the diagonal matrix $\mathbf{\Sigma}$ to adaptively remove redundant singularity points. When the singular value is small (in all our experiment, we set the threshold to $1e^{-5}$), the corresponding source point does not contribute much to the potential field, and therefore we remove them in the source point set from subsequent linear system solving and MFS evaluations.

3.3.5 Surface Mapping as Boundary Condition

The boundary condition of our harmonic volumetric mapping is a surface mapping between ∂M_1 and ∂M_2. Existing inter-surface mapping techniques [8, 9, 10, 11,

12, 13, 64] can be used for creating the boundary surface mapping. Although surface mapping is not the focus on this work, as discussed in [2], it is desirable to have a low distorted surface mapping, and [2] illustrates an example that larger angular distortion oftentimes leads to worse volumetric mapping result. However, it is still unknown that how exactly quality of boundary mapping and interior mapping relate. Intuitively, area preserving is also important (salient shrinkage on boundary mapping shall lead to large volume distortion). [2] uses [15] for boundary surface mapping computation which leads to least angular distortion, in our work, we use [1] to generate boundary surface mapping, which could (1) better balance area stretch and angle distortion, and (2) allow surface feature points and curves alignment which better fits our current framework.

3.3.6 Feature Alignment

Feature and structure constraints are important issues in processing many real volumetric data. Specifically, three types of features are commonly considered: 0-manifolds (feature points), 1-manifolds (feature lines), and 2-manifolds (iso-surfaces).

Feature point alignment. Since we are using a meshless paradigm, all the feature alignment shall naturally be handled in terms of points. We can simply add the feature constraint defined on each point into the linear system as a new boundary condition. So each new pair of feature points for alignment corresponds to an additional row in the coefficient matrix $\mathbf{A}$ and the boundary condition vector $\mathbf{b}$.

Feature line alignment. For feature lines (for instance, skeletons) matching, we similarly sample and put in point-by-point constraints to $\mathbf{A}$. Feature curves such as skeletons are usually represented as a piecewise graph. Fig. 3.3(a) shows an example, in which one wants the volumetric mapping follows skeleton structure

and is guided by the movement or deformation of the skeleton. Existing skeleton extraction algorithms ([65, 66, 67, 68]) usually do not guarantee that the extracted skeletons from different objects are isomorphic even when these shapes are very similar. However, most skeletonization algorithms can preserve the homotopic structure of the shape and therefore their skeletal graphs are topologically equivalent. Detail discussion on feature lines extraction, their topology, and their matching is far beyond the scope of this work, and we refer readers to the survey paper [69]. In our experiment (Fig. 3.3), we use [65] to extract the skeleton and apply 1-manifold restamping algorithm of [70] to get dense point-by-point correspondence between skeletons of the cyberware Male and Female models. Fig. 3.3 (b) visualizes the volumetric map result using the color-encoded distance field. The distance of each interior point to the boundary surface is computed and color-encoded, such a color is transferred to its corresponding point under the mapping. This visualizes the mapping behavior. (c) illustrates the skeleton fitting: in the interior of the target Female model, the green curve is its skeleton Sk_2. Sample points from the skeleton of the source Male model should match Sk_2, and the red points shows their images under the mapping. The rooted mean square fitting error is 0.5%.

Feature surface alignment. With two feature surfaces to match, we need to compute inter-surface maps between them. The inter-surface parameterization methods, which we used to generate boundary surface correspondence, can be applied to get such a map. Then we simply include corresponding point pairs as the boundary conditions. Fig. 3.1 and Fig. 3.2 show an example of a volumetric mapping over the heterogeneous data Head-Skull-Brain model, which has three salient iso-surfaces: the outer boundary is a genus-0 (head) surface, and the interior skull iso-surface is genus-2, within which there is a genus-0 brain surface. We generate

a parametric domain (d) to test the efficacy of our mapping on heterogeneous 3D data with iso-surface constraints. The outer head boundary surface is mapped onto a sphere boundary, the skull iso-surface is constrained on the polycube skull, while the brain iso-surface is mapped to a small cube inside. (b,c) show the 30% and 60% morphing from (a) to (d), generated by linear interpolation. (e,g) show two cross-sections on the polycube-sphere domain, and (f,h) show their corresponding cross-sections on the head-skull-brain model. The point clouds in (e)-(i) show sample points on the iso-surface (e, g), and their images after the volumetric mapping (f-i). Locations of these feature points in (g-i) demonstrate that the iso-surface constraints are precisely fitted, and the volumetric mapping align the feature surface very well.

Feature alignment as soft constraints and weights. As discussed above, features are aligned in a least square sense together with the boundary fitting process, so they are treated as soft constraints. Compared with the massive point number on the boundary, if feature points (or samplings on feature lines) are considered as ordinary collocation points, they might be overshadowed by boundary collocation points during the fitting. We balance this by assigning each sample feature point an extra weight w. This is equivalent to enforcing this feature w times. In all our experiments, we take $w = 20$ for feature points. This effectively leads to more precise feature alignment.

Unlike the traditional FEM-based methods that simply fixes feature vertices to enforce the constraints, in this section we discuss our method that blends several harmonic functions to get the feature-aligned map. In each iterative refinement, we use a harmonic function, so the resultant map, i.e. the summation of these functions is still globally harmonic, and there is no obvious flip-over or discontinuity around the feature regions. In the mean time, while the feature constraints is precisely

enforced, the boundary fitting accuracy could decrease a little bit (i.e. the RMSE increases slightly on the boundary).

3.4 Experimental Results and Applications

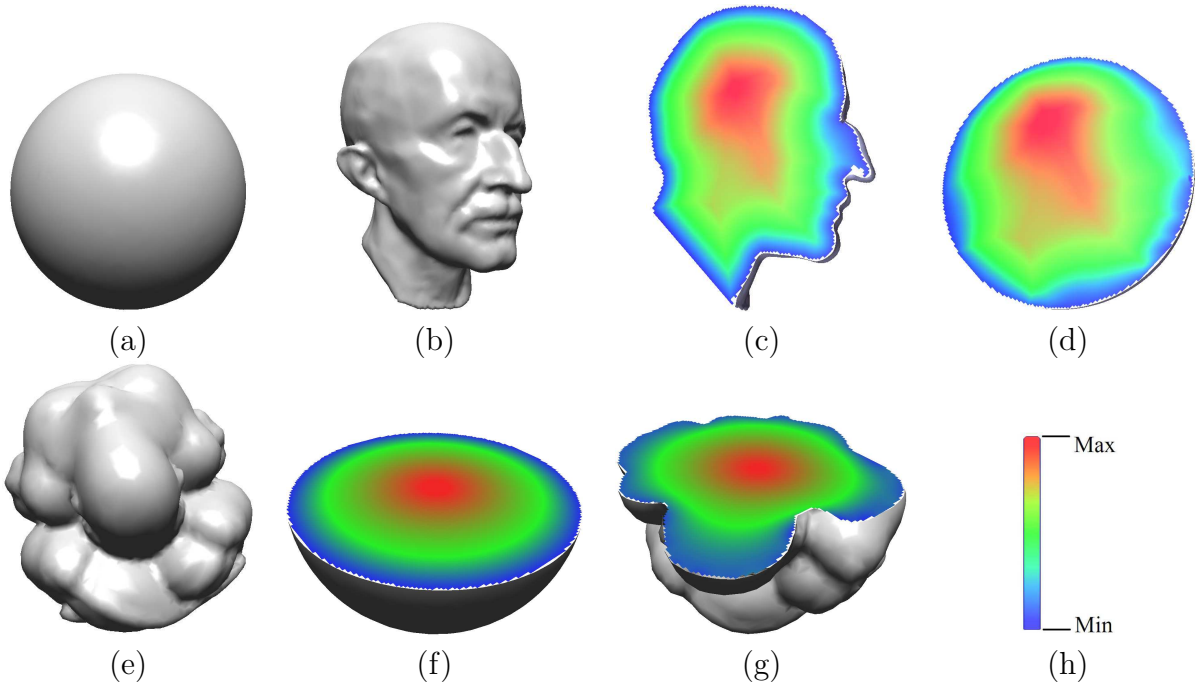


FIGURE 3.4. Spherical Volumetric Mapping. A solid sphere (a) is mapped to a solid Max-Planck model (b), the color-coded distance field on the Max-Planck model (c) is transferred onto corresponding points on the sphere (d) for visualizing the map. An Omotondo model (e) is mapped onto the solid sphere (a), and we transfer the color-encoded distance field of sphere (f) onto the Omotondo region (g). The color-coding scheme is illustrated in (h), red indicates maximum values while blue indicates minimum values.

We conduct a few volumetric mapping experiments over various volumetric data, with different sizes, topology and geometry complexities. We illustrate some of these mapping results in Fig. 3.4 and Fig. 3.5. We use the color-encoded distance field to visualize the mapping result. When a map $\vec{f} : M_1 \rightarrow M_2$ is computed, the color-encoded (red indicates the maximum while blue indicates the minimum, see Fig. 3.5(h)) distance field defined on one region can be transferred to another region, by plotting the color of a point $\mathbf{P} \in M_1$ on its corresponding image $\vec{f}(\mathbf{P}) \in$

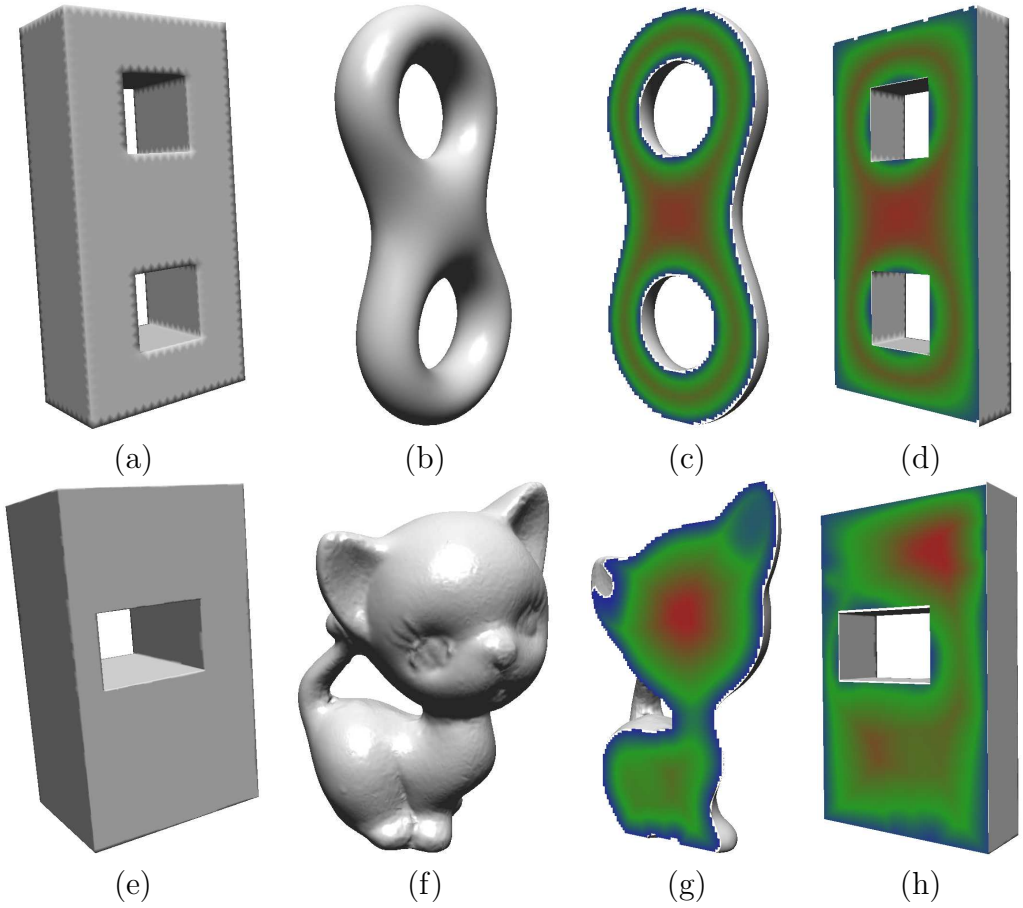


FIGURE 3.5. Mapping Between Solid Objects and Polycubes. Polycubes (a,e) are mapped to two-torus (b) and kitten (f), respectively. Color-encoded distance field of (c,g) are transferred under the mapping to (d,h).

M_2 (or inversely, plotting the color of $\mathbf{P} \in M_2$ on $\vec{f}(\mathbf{P}) \in M_1$). This visualization shows the effect of the map. For example, when we transfer the distance field defined on the Max-Planck model (Fig. 3.4(c) to the sphere, we can see a color-encoded head-shaped level-set in (d), while the original distance field of a sphere is concentric as shown in (f).

We also conduct thorough comparison between our method and the algorithm of [2]. Table 3.2 illustrates the side-by-side statistics. Using MFS [2], the *cRatio* (and *sRatio*), indicating the ratio of the number of collocation points (and source points) over the number of boundary points are listed in the CR (and SR) column; the computation times (in seconds) and rooted mean square errors (object size

TABLE 3.2. Efficiency and Accuracy Comparison. A lot of 3D solid models have been tested, from data with small vertex size to large size. Using MFS [2]: CR and SR ($cRatio$ and $sRatio$ of the mapping), computation time (in second), and the $RMSE$ (rooted mean square error of the boundary fitting) are listed; using the new computation framework, in the same row we list the statistics of the corresponding CR*, SR*, n_f (number of harmonic maps we solved), computation time, and $RMSE*$.

Models (Vertex #)	CR	SR	Time(S)	RMSE	CR*	SR*	n_f	Time*(S)	RMSE*
Omotondo/Sphere (3K)	0.4	0.8	220.41	0.01649	0.4	0.8	6	42.41	0.001514
PCube/2-Torus (6.6K)	0.4	0.8	2393.10	0.00490	0.4	0.8	6	426.23	0.00475
Male/Female (6.3K)	–	–	–	–	0.2	2	10	1000.68	0.00485
PCube/Skull (29K)	–	–	–	–	0.03	0.03	7	14.42	0.01764
Vaselion/Sphere (40K)	–	–	–	–	0.1	0.2	6	1440.39	0.0230
PCube/Kitten (80K)	–	–	–	–	0.05	0.05	6	666.48	0.0139
PCube/Horse (100K)	–	–	–	–	0.01	0.02	6	23.82	0.01449

normalized to a unit box) are also given. Using multiple MFS computation proposed in this work, we list the number of harmonic functions we computed in the column of n_f, and show corresponding ratios, time, and errors in columns of CR*, SR*, Time*, RMSE*, respectively. As we addressed previously, due to the efficiency issue, MFS[2] is not able to handle large volumetric data or feature alignment, therefore statistics on corresponding cells are blank. The statistics show that our algorithm improves the computation of [2] in both efficiency and accuracy.

Fig. 3.6 plots the reducing of boundary error during each iterative step i of computing mapping functions $\vec{f}_i$.

3.4.1 Hex-Remeshing

A direct application for volumetric mapping is hex-mesh generation. Regular mesh structure is highly desirable for finite element analysis and physically-based deformations/simulations, because regular meshes provide great efficiency for geometry processing and physically-based computation [71]. Given a 3D solid data M, we first generate a solid polycube model P, then we compute the surface mapping $\vec{f}' : \partial M \to \partial P$ and volumetric mapping $\vec{f} : M \to P$. With $\vec{f}$ we can transfer the regular structure on P to M. On a solid polycube, a regular hexahedral structure can be easily generated, and since [72] introduced the concept of surface polycube map, several techniques ([73, 74, 75, 76]) have been proposed to (automatically)

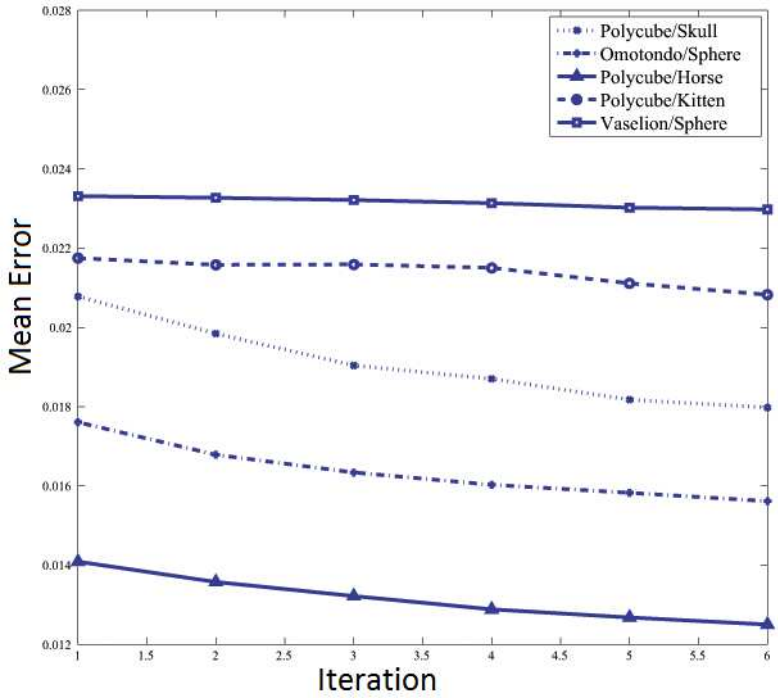
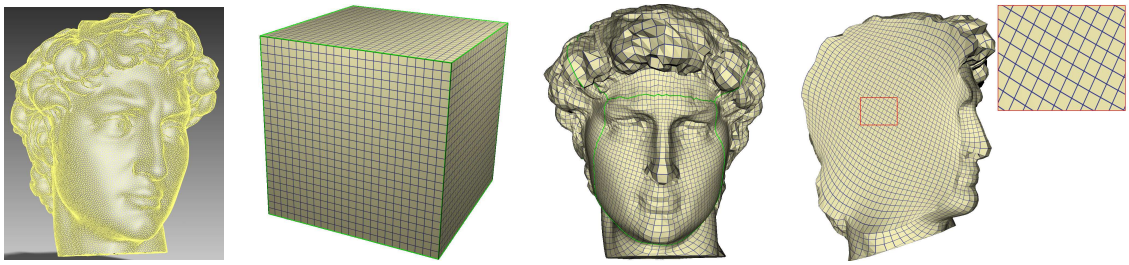


FIGURE 3.6. Boundary Rooted Mean Square Error of Volumetric Mappings.

construct polycube, and the surface-polycube mapping. In all of our experiments, we construct our polycubes using the algorithm of [73]. Fig. 3.7 illustrates an example of using a unit solid cube to remesh the solid David head. The original model M_2 is shown in (a), and the hex mesh of the parametric cube M_1 is shown in (b). We compute the volumetric map $\vec{f} : M_1 \to M_2$ from the cube to David head. Then $\vec{f}(M_1)$ is a solid with the hex connectivity of M_1 and the head shape of M_2, and it is the remeshed David head, as illustrate in (c) and (d). Fig. 3.8 shows a few more examples. A hex-remeshed two-hole torus is shown in (a). The hex-mesh structure of the polycube (b) is used to remesh the kitten, shown in (c,d). Polycube (e) is used to remesh the Chinese horse model (f-h). (f, g) visualize the result hex-mesh in its interior regions from two different cross-sections.

3.5 Summary

We present a feature-aligned volumetric harmonic mapping computation algorithm using methods of fundamental solutions. The map $\vec{f}$ is composed by a set



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

FIGURE 3.7. Hex Remeshing of the Solid David Head. (a) The original mesh structure of the David Head. (b) A simple cube domain that the hexahedral mesh is generated upon. (c) The remeshed David head and (d) a cross-section to show the interior structure.

of harmonic functions $\{\vec{f}_i\}$ which can be efficiently solved. Also, our adaptive source/collocation points placement improves the numerical issue of MFS solving. Therefore, our algorithm largely improves the existing harmonic volumetric mapping computation algorithm using MFS [2]. The new algorithm has better efficiency and accuracy, and it supports feature points, curves, or surfaces alignment, which is important for integrating/matching heterogeneous volumetric data that have intrinsic interior structure. We demonstrate that harmonic volumetric mapping can be conducted on large data, heterogeneous data, and data with feature to match, which can not be handled properly in [2].

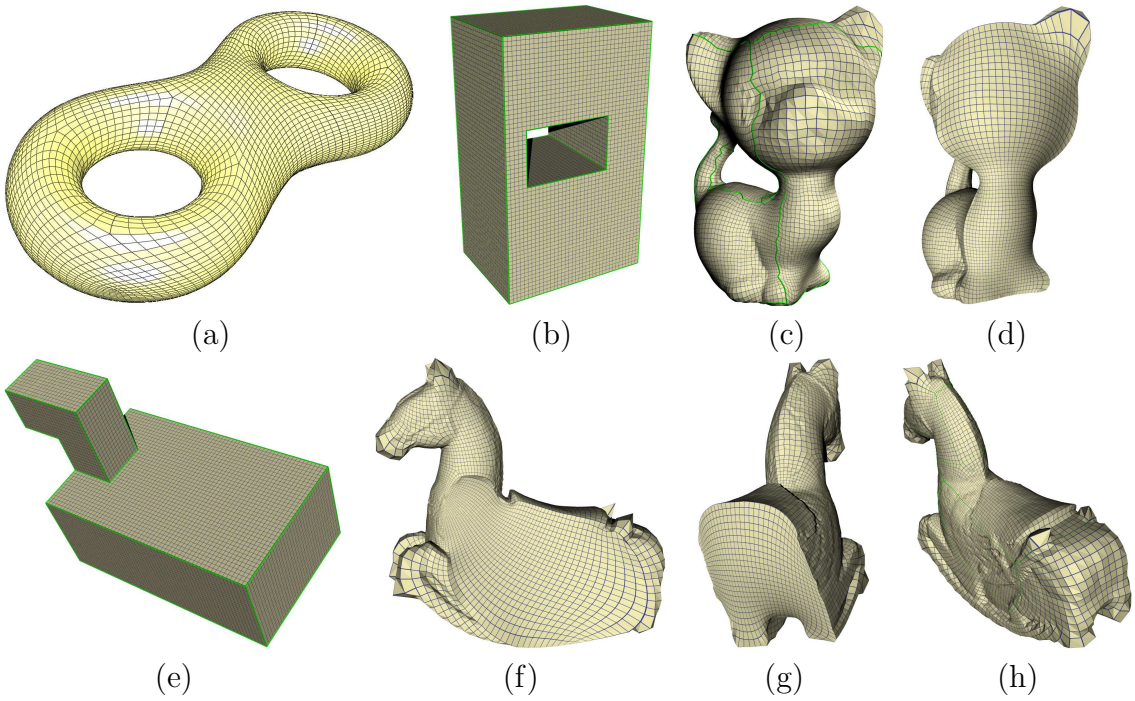


FIGURE 3.8. Hex Remeshing. (a) illustrates a hex-remeshed solid two-torus using the the polycube of Fig. 3.5(a). The hex mesh on the polycube for remeshing solid kitten is shown in (b). The remeshed kitten is illustrated in (c, d). (e)-(h) show the hex-remeshing for a solid Chinese horse model.

Chapter 4
Biharmonic Volumetric Mapping

In previous chapter, we studied the feature-aligned harmonic volumetric mapping and solved it in an effective multi-level strategy. However, the bijectiveness of harmonic volumetric mapping only has been proved to exist in several special types of shape domains. In some highly complex shape domains the bijectiveness of the harmonic volumetric mapping can not be guaranteed. In this case, we need explore an effective mapping computation framework through domain decomposition methods. In this chapter, we introduce a new volumetric mapping method called biharmonic volumetric mapping which is suitable for high complex shape domains and heterogeneous objects.

4.1 Introduction

Computing lowly distorted volumetric mapping between two given objects is an important geometric problem in computer graphics and visualization. It serves as an important preprocessing step in many tasks in broad areas of compute-aided design and analysis, industrial manufacturing, medical image analysis, and etc. With the advance of data acquisition techniques, massive volumetric models with many multiple attributes and complex geometry are collected and need to be processed. Surface mapping has been extensively studied and many effective algorithms have been developed to handle shells of 3D objects. However, many real-world objects are volumetric and have interior contents. Effective volumetric parameterization is critical to process such data that have not only boundary geometry but also interior material/structures. Due to the much more complicated topological and geometric structures of 3-dimensional manifolds, existing volumetric mapping techniques are

still inadequate. A desirable volumetric mapping model should usually have the following properties:

1) *Generality*: many real-world volumetric models have complicated geometry and interior structure (e.g. multiple material layers), therefore, the mapping algorithm needs to be general and can handle data with nontrivial topological types.

2) *Efficiency*: volumetric models usually have much bigger sizes, compared with surface data, while computational efficiency is important in many engineering applications.

An effective computation strategy to handle huge-size and geometrically complex volumetric models is through divide-and-conquer based on model decomposition. We can partition the complex model into solvable sub-domains for individual processing. Such a partitioning is often desirable and sometime necessary. First, unlike surface parameterization, bijective (i.e. no degeneracy or flip-over) volumetric map may not exist globally between two solid models with complex geometry. Second, volumetric models are often so big that the direct computation is prohibitive and efficient computation conducted on smaller sub-domains is preferred.

3) *Smoothness*: a mapping indicates a transformation between two solid objects; it should be smooth, i.e., minimizing the stretch of the transformation and thus physically natural. Individually computed maps on subparts should compose a global map with smooth transition across the cutting interfaces.

4) *Feature-preserving*: volumetric models usually have different materials, rich interior structures and features. The mapping should be feature-aware. For example, corresponding features points/curves or layers of materials should be aligned correctly.

5) *Linear Precision*: if the mapping (deformation) of the boundary surface follows a linear transformation, an induced volumetric mapping in the interior region that reproduces this transformation (see Section 4.4) is natural and therefore desirable.

Our goal is to develop a volumetric mapping model with these desirable properties.

The harmonic function is widely used in shape mapping and deformation, because it indicates the vanishing Laplacian inside the domain. A volumetric map is a harmonic map if all its three components are harmonic. It minimizes the stretching energy and therefore indicates a physically natural transformation between two solid domains. A discrete harmonic mapping can be computed efficiently by solving a linear system. However, harmonic volumetric mapping has its limitation in a divide-and-conquer computation framework. Individually computed harmonic maps could have undesirable discontinuity across the partitioning boundary interface. Typically, one is only able to enforce C^0 continuity across the mapping boundary. In other words, we can have only the positional constraints on the boundary points, but not their derivatives at the same time. Only C^0 continuity along the boundary interface introduces undesirable artifacts in parameterization and subsequent applications such as meshing, texture mapping, deformation, and physical simulations. Specifically, when volumetric parameterization is used for iso-geometric analysis, higher continuity is often required.

Therefore, we propose to use a biharmonic model to construct the mapping on decomposed sub-domains with C^1 continuity. The boundary condition in both positions and normal derivatives can be controlled, hence we can obtain better smoothness across the cutting interface in a divide-and-conquer computation. Heterogeneous volumetric models with multiple materials can also be segmented and parameterized in a similar manner.

The main contributions of this chapter include:

1. We propose a biharmonic model to solve volumetric mappings for 3D heterogenous data, following the surface mapping and boundary normal constraints. Compared with the harmonic map, it provides better boundary continuity control.

2. We develop a biharmonic volumetric mapping computation framework based on the method of fundamental solutions. Complex models can be decomposed and then parameterized effectively.

3. We demonstrate the effectiveness of our method on various models and show its applications in hex-remeshing and temporal data analysis.

We organize the remainder of this chapter as follows. In Section 4.2, we formulate the problem and give an overview of our method. Algorithm details on the boundary positional constraint and normal derivative constraint configurations will be discussed in Section 4.3.1 and Section 4.3.2, respectively. We discuss the properties of biharmonic volumetric mapping and some implementation details in Section 4.4, and show experiment results in Section 4.5. We apply our biharmonic mapping in hex-remeshing and temporal data parameterization in Section 4.6.

4.2 Algorithm Overview

4.2.1 Problem Definition

We consider the computation of the mapping $\Phi : \Omega \rightarrow M$, where volumetric domains $\Omega, M \subset \mathbb{C}R^3$ are embedded in 3-dimensional space.

$$\begin{cases} \nabla^4 \Phi = 0, & \text{in } \Omega, \\ \Phi = f, & \text{on } \partial\Omega, \\ \frac{\partial \Phi}{\partial n} = g, & \text{on } \partial\Omega, \end{cases} \tag{4.1}$$

where $\partial\Omega$ and ∂M denote the boundary surface of Ω and M, n is the surface normal on the domain boundary, $\partial\Phi/\partial n$ then denotes the outward normal derivative, f and g are prescribed functions: f is decided by a given surface mapping $f : \partial\Omega \to \partial M$, and $\frac{\partial\Phi}{\partial n}$ indicates the derivative along the normal direction on each boundary point. The mapping (vector function Φ) can be decomposed into three components ϕ^1, ϕ^2, ϕ^3 for x, y, z axis directions, each of which should be biharmonic.

Under reasonable (enough smoothness) assumption of f and g, the biharmonic model in eq (4.1) can be uniquely solved given a pair of boundary conditions f and g. A proof is sketched in the appendix. In our experiments, we found that for most real-world volumetric data, our computed surface mapping f and derivative constraint g satisfy this smoothness assumption.

4.2.2 Solving Biharmonic Mapping using MFS

Using the method of fundamental solutions (MFS) [77], the solution (for simplicity, we use ϕ to denote each component ϕ^i) to equation (4.1) can be approximated by a linear combination of fundamental solutions of both the harmonic and biharmonic equations:

$$\phi(\mathbf{h}, \mathbf{b}, \mathbf{Q}, \mathbf{x}) = \sum_{j=1}^{N_s} h_j H(\mathbf{q}_j, \mathbf{x}) + \sum_{j=1}^{N_s} b_j B(\mathbf{q}_j, \mathbf{x}) \tag{4.2}$$

where

- kernels: $H(\mathbf{q}_j, \mathbf{x}) = 1/(4\pi|\mathbf{q}_j - \mathbf{x}|)$ is the fundamental solution of the harmonic term, and $B(\mathbf{q}_j, \mathbf{x}) = |\mathbf{q}_j - \mathbf{x}|/(8\pi)$ is the fundamental solution of the biharmonic term;
- $\mathbf{Q} = \{\mathbf{q}_1, \ldots, \mathbf{q}_{N_s}\}$ is a $3N_s$-dimensional vector, representing the set of N_s singularity points, each $\mathbf{q}_j = [q_{3j-2}, q_{3j-1}, q_{3j}]^T \in \mathbb{C}R^3$ denotes the position of a singularity point,

- $\mathbf{h} = [h_1, h_2, \ldots, h_{Ns}]^T$ and $\mathbf{b} = [b_1, b_2, \ldots, b_{Ns}]^T$ are vectors of harmonic and biharmonic coefficients associated with these N_s singularity points, i.e. the degree of the freedom to control the boundary fitting.

The vanishing bilaplacian operator on ϕ is enforced by the fundamental solutions H and B, we only need to ensure the function satisfy the boundary condition. This boundary fitting is performed on a set of N_c *collocation points* defined on the domain boundary $\partial\Omega$. The kernel function $H(\mathbf{q}_j, \mathbf{x})$ with respect to $\mathbf{q}_j$ (therefore, can also be directly denoted as $H_{\mathbf{q}_j}(\mathbf{x})$) is not defined on this singularity point $\mathbf{x} = \mathbf{q}_j$. Therefore, singularity points $\{\mathbf{q_j}\}$ need to be sampled outside the function domain, i.e., on a surface $\partial\Omega'$ outside Ω, $\Omega \subset \Omega'$. It has been shown [2] that an effective MFS system can be constructed by computing an offset surface outside $\partial\Omega$ then adaptively sampling $\{\mathbf{q_j}\}$.

To perform boundary fitting on each constraint point $\mathbf{x} \in \partial\Omega$, we evaluate $\phi(\mathbf{x})$ using equation (4.2). The boundary constraints are then $\phi(\mathbf{x}) = f(\mathbf{x})$ and $\frac{\partial\phi(\mathbf{x})}{\partial n(\mathbf{x})} = g(\mathbf{x})$. Enforcing these constraints on all collocation points reduces to a linear system $A\mathbf{w} = \mathbf{t}$, where the coefficient matrix A's dimension is $2N_C * 2N_S$, $\mathbf{w}$ is the unknown $2N_S$-dimensional coefficient vector $[\mathbf{h}, \mathbf{b}]^T$, and $\mathbf{t}$ is the $2N_C$-dimensional vector indicating the boundary condition evaluated on collocation points. When $N_C > N_S$, this system is an over-determined system. With this condition we can solve the system directly. Usually, the coefficient matrix $\mathbf{A}$ is dense and ill-conditioned, we can use truncated Singular Value Decomposition (SVD) to improve its numerical stability [2, 78]. In our work, we set $N_C < N_S$, and use a regularization term to avoid over fitting (see Section 4.4.5). We found this approach gives better numerical efficiency and stability in our mapping computation.

The biharmonic equation (4.1) can be solved using this above collocation method through a 3-step pipeline: (1) place singularity and collocation points, (2) on col-

location points, formulate boundary constraints evaluated by coefficients (to be solved) associated with these singularity points, (3) solve a linear system to get all the coefficients, enforcing the boundary constraints.

4.2.3 Mapping through Model Decomposition

When the volumetric regions Ω and M are simple, we can directly compute their mapping $\phi : \Omega \to M$ using the above method. However, when Ω and M have complex topology or geometry, mapping computation through a divide-and-conquer scheme based on model decomposition is desirable. Specifically, we consistently partition Ω and M into corresponding sets of sub-parts $\{\Omega_1, \ldots, \Omega_n\}$ and $\{M_1, \ldots, M_n\}$, such that the (1) topology of Ω_i and M_i are the same and (2) the dual graphs of their decompositions are isomorphic [1]. Then, on each corresponding pair Ω_i and M_i, we compute the volumetric mapping $\phi_i : \Omega_i \to M_i$ using the above method individually. The computation of this consistent decomposition and corresponding boundary constraints include the boundary positional constraint function f and the boundary normal derivative constraint function g is not the main focus of this work, but will be briefly discussed in Section 4.3.

4.3 Boundary Condition Setting

This section elaborates the computation of boundary constraints f and g in equation (4.1).

The boundary positional constraint function f is decided by the boundary surface mapping between $\partial\Omega$ and ∂M. In this work, we construct the surface mapping f through model decomposition (Section 4.3.1).

Function g is usually unknown. In order to have smoothness across cutting boundaries, we derive g from f by using a *local to global* affine approximation technique, namely, we use local affine transformations to approximate a global

mapping. It will ensure the unchanged normal derivative along the boundary interface (Section 4.3.2).

4.3.1 Positional Constraints by Surface Mapping

We will first discuss our positional constraints setting on a simple volumetric domain then talk about the decomposition and mapping for complex models in the divide-and-conquer scheme.

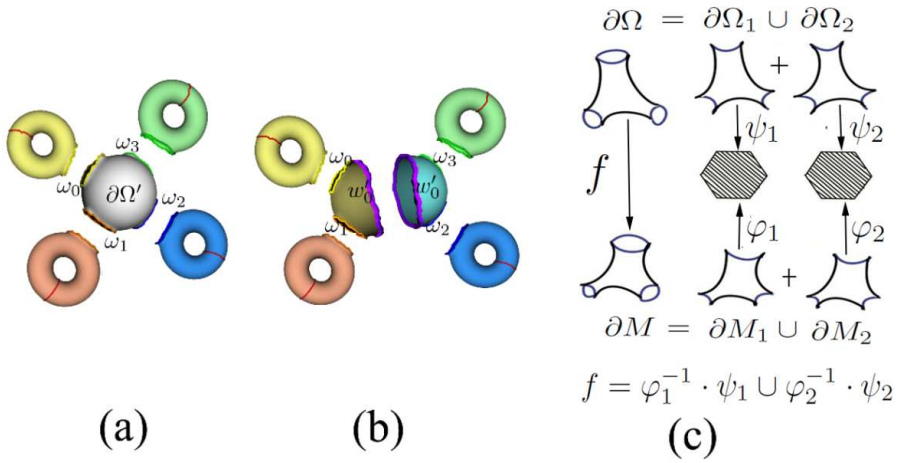


FIGURE 4.1. Decomposition and Mapping of the Boundary Surface of a 3D Model. (a-b) shows the computation of the *pants decomposition* [1]. (c) illustrated the parameterization of two corresponding pants patches, and the composed mapping.

Surface mapping on a single volumetric domain. To compute the volumetric mapping $\phi : \Omega \to M$, we need to solve the surface map f between the boundaries $\partial\Omega$ and ∂M. Cross-surface parameterization methods such as [79, 15, 1, 12, 13] can be used for computing f. In this work, we use a harmonic inter-surface map to serve as the boundary positional constraint of our biharmonic volumetric map. We briefly recap our computation algorithm, which is based on [15, 1].

Given a surface $\partial\Omega$ with G genus and B boundaries, first, we remove the *topological handles*, and get a surface (base patch) $\partial\Omega'$ with $G + B$ boundaries and some handle patch. If $G + B = 1$, $\partial\Omega'$ is a *topological disk*; if $G + B = 2$, $\partial\Omega'$

is a *topological cylinder*; if $G + B = 3$, $\partial\Omega'$ is a *topological pants patch* (a handle patch is also a topological pants patch). For $G + B > 3$, $\partial\Omega'$ can be decomposed into several topological pants patches (See [1] for detailed constructive algorithm: iteratively remove a pants patch from $\partial\Omega'$ and reduce the boundary number of $\partial\Omega'$ by 1, until $\partial\Omega'$ itself becomes a pants patch). Figure 4.1(a-b) shows an example of the pants decomposition.

Then, any $\partial\Omega$, reduced to one of the above three topological types, can be parameterized using the harmonic map on the canonical disk, cylinder, and topological hexagons. The corresponding patch ∂M is parameterized likewise. We can then compose the parameterization and get the mapping $f : \partial\Omega \to \partial M$. Figure 4.1(c) shows the process of parameterization of two corresponding pants patch over regular hexahedrons and the composed inter-patch surface map.

Divide-and-conquer scheme. Complex Ω and M can be decomposed into sub solid parts $\{\Omega_i\}$ and $\{M_i\}$. We need to have a new surface mapping scheme that set up positional constraints between $\partial\Omega_i$ and ∂M_i separately.

Many shape decomposition techniques have been developed in computer graphics literature (see [80] and [81] for thorough surveys); most of these are for partitioning a single surface following its own geometry. Here, for the mapping purpose, the partitioning of Ω and M need to be *consistent* (i.e., each sub-part Ω_i has same topology with its corresponding part M_i, and the dual graphs of these two decompositions are isomorphic). At the same time, decomposition of heterogeneous solid domains may need to follow the materials, semantics, or geometry of the objects. Volumetric decomposition is not the main focus of this work. Similar to [79, 12], we first get the consistent decomposition on the surface. Then we use minimal surfaces to fill the topological disks along the interior boundary interface.

To obtain the consistent volumetric decomposition, we first get the consistent surface decomposition. Conventional consistent surface mesh segmentation is done interactively [12, 82], via tracing shortest paths connecting manually placed markers. It can also be computed using automatic manner using such as [1, 12].

In our experiments, we first compute the consistent surface decomposition and inter-surface mapping using the algorithm of [1]. Then it can be extended into the interior volumes. It is computed through a consistent region growing upon the tetrahedral meshes [83]. We start with the compatible skeletal graphs of the two given models, and simultaneously propagate towards boundary with guaranteed visibility. Such a growing can ensure the topologically coherent adjacency relationship among neighboring sub-regions and provide a consistent volumetric decomposition (the dual graphs of decompositions of the two models are isomorphic). More implementation details can be found in [83]. Through this way we can decompose the Ω and M into n subregions that $\Omega = \{\Omega_1, \Omega_2, \ldots, \Omega_n\}$, $M = \{M_1, M_2, \ldots, M_n\}$. For each $\phi_i : \Omega_i \to M_i$, we can compute it using the above surface mapping algorithm for single volumetric domain.

4.3.2 Normal Derivative Constrains by Affine Approximation

In this section, we give an algorithm to compute the normal derivative function g by using affine approximation.

When we handle heterogeneous and decomposed volume data, we want to enforce higher continuity across the cutting interface. Since any directional derivative can be decomposed into the tangent derivative and normal derivative, so C^1 continuity consists of tangent derivative continuity and normal derivative continuity. From differential geometry, we know tangent derivative continuity can be ensured by the consistence of positional constraints. Then we need to choose a function g as the

normal derivative function such that the derivative of the mapping will not change when it goes across the sub-region boundaries.

Here we adopt an affine approximation method to set normal derivative boundary condition g for constraint points. We build an affine transformation matrix for each constraint point. Then we define normal derivative function g on this point according to this local affine function. Given the surface mapping $f : \partial\Omega \to \partial M$. For a vertex $x \in \partial\Omega$, whose one-ring neighboring vertices are $x_1, x_2, \ldots, x_n$ (Fig. 4.2), we compute a local affine function $\Phi_x(x) = A_x \times x + D_x, x \in \partial\Omega$. The global mapping is $\Phi(x) = \{\Phi_x(x)\}$ for $x \in \partial\Omega$. So the normal derivative boundary condition of the biharmonic function Φ is given as $\frac{\partial\Phi(x)}{\partial n} = A_x^T \times n$ on point x.

The A_x, D_x on each point $x \in \partial\Omega$ can be computed from the surface mapping on the one-ring region in the following linear system

$$\begin{cases} f(x_1) = A_x \times x_1 + D_x \\ f(x_2) = A_x \times x_2 + D_x \\ \qquad\qquad ... \\ f(x_n) = A_x \times x_n + D_x \end{cases} \tag{4.3}$$

where A_x is 3×3 and D_x is 3×1. We solve this system using the least square method to get A_x, D_x. If the affine transformation is degenerated, e.g., a planar local region is transformed into another planar region, the rank of the coefficient matrix of the system (4.3) reduces to 3 and the linear system becomes underdetermined. We still compute A_x, D_x that are the least squares solutions and have the smallest L_2 norm.

For a point x along the boundary interface, no matter which volumetric region it belongs to, it will be mapped to a same target point $f(x)$. According to our computation of $\Phi_x(x)$, the point from different volumetric regions will get the

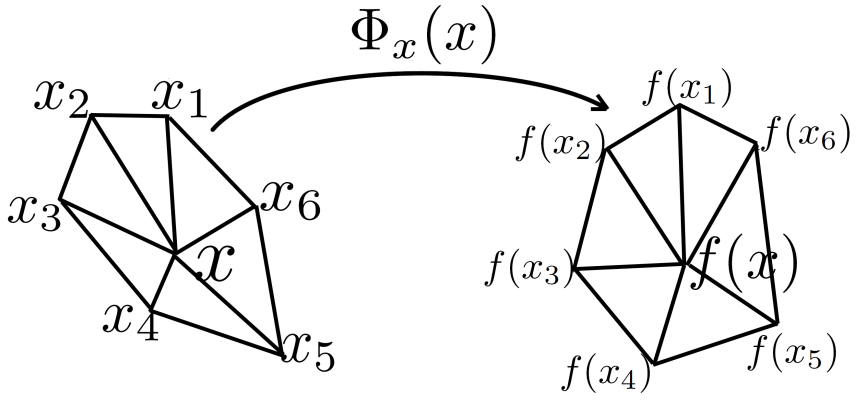


FIGURE 4.2. Local Linear Approximation. $x_1, x_2, ..., x_6$ are the one-ring of the vertex x, under the surface mapping f they becomes $f(x_1), f(x_2), ..., f(x_6)$. In this local region, f is approximated by the linear transformation $\Phi_x(x)$.

same value of A_x, D_x. That is, it will have the same normal derivative function g_x. So we can see the point along the boundary interface, it not only keeps position consistence through boundary surface mapping, but also has normal derivative consistency. We also show in Section 4.4 this biharmonic volumetric mapping keeps linear precision property.

4.4 Properties, Implementation, and Evaluations

In this section, we discuss properties, implementation details, and evaluations of our volumetric mapping.

4.4.1 Linear Precision Property

A function ϕ has the *linear precision* property if it can reproduce a linear function exactly: given a set of function values of $\phi(v_i) = r(v_i), v_i \in \partial\Omega$ for any linear function r, then $\phi(x) = r(x)$, $x \in \Omega$ [84].

Linear precision property is desirable in describing shape deformation [34]. For example, when the boundary surface transforms rigidly, it will be natural to see the interior region also transforms in the same rigid manner.

Linear precision of harmonic maps. Suppose ϕ is a harmonic function, i.e., it is a solution to Laplace's equation: $\nabla^2\phi(x) = 0, x \in \Omega$, and satisfies the boundary condition $\phi(v_i) = r(v_i), v_i \in \partial\Omega$, where $r(x)$ is a linear function. Then since ϕ and r satisfy the same boundary conditions and are both solutions to Laplace's equation (the linear function r is also a harmonic function), by the uniqueness of solutions to Laplace's equation, they must be the same function $\phi \equiv r$. So the harmonic mapping ϕ keeps the linear precision property [85].

Linear precision of our biharmonic maps: Whether biharmonic mapping has the linear precision property is determined by the setting of its boundary condition. When its boundary positional constraints are decided by a linear function, using our normal derivative setting discussed in the last section, the computed biharmonic map has the linear precision property.

Suppose the surface mapping f is a global linear transformation $\phi(x) = r(x) = Ax+D, x \in \partial\Omega$. The local linear function $\phi_x(x) = A_x \times x + D_x$ where A_x and D_x are computed from its one-ring transformation, and we have $\phi_x(x) = \phi(x) = Ax + D$ (i.e., $A_x = A, D_x = D$). The normal derivative function $\frac{\partial\phi(x)}{\partial n} = g = \{g_x | x \in \partial\Omega\} = \{A_x^T \times n | x \in \partial\Omega\} = A^T \times n$. While $\frac{\partial r(x)}{\partial n} = A^T \times n$. So, we have $\frac{\partial\phi(x)}{\partial n} \equiv \frac{\partial r(x)}{\partial n}$ and $\phi(x) \equiv r(x), x \in \partial\Omega$. Also ϕ, r are both solutions to biharmonic's equation ($\Delta^2 r(x) = 0$, since r is a linear function). According to the uniqueness solution of biharmonic's equation [86], we have $\phi \equiv r$. Therefore, this biharmonic mapping keeps the linear precision property.

4.4.2 Mapping Bijectivity

Bijectivity should usually be ensured in mapping computation. For general given shapes, finding volumetric mapping with guaranteed bijectivity is usually very difficult. To our best knowledge, mapping construction algorithms with guaranteed bijectivity are only known on simple shapes such as convex or star regions.

Our biharmonic model, like the harmonic map, can not guarantee the mapping bijectivity when the two given models are general shapes. However, our idea of processing complicated models through decomposition and divide-and-conquer is one potential direction to avoid the degeneracy/flipover caused by the model's complex topology/geometry. The shape will be decomposed into sub-parts with simple geometry, whose parameterization is more often bijective.

In practice, we can check the bijectivity of a parameterization by computing the Jacobian value on each tetrahedron or hexahedron (after remeshing). If all Jacobian values are larger than 0, then this piecewise linear mapping function is bijective. For regions with negative Jacobian, it is also possible to develop heuristic adaptive decomposition to further partition these regions. In our experiments, we have performed this Jacobian evaluation, and observed that following our proposed boundary condition, the resultant biharmonic mapping remains bijective. Figure 4.8(d)(h) lists our computed Jacobian statistics.

4.4.3 Other Boundary Condition Setting Strategies

A unique biharmonic mapping can be specified by the positional constraint function f and normal derivative function g. The position constraints are decided by the surface mapping f between boundaries of the source Ω and the boundaries of target model M.

A harmonic inter-surface map is a simple choice for f. We can analyze its continuity across the boundary of adjacent sub-parts. Any directional derivative can be expressed by the combination of the normal derivative $\frac{\partial}{\partial n}$ and two tangential derivatives ($\frac{\partial}{\partial \tau_1}$, $\frac{\partial}{\partial \tau_2}$) with the absolute value of coefficients less than one.

$$\frac{\partial}{\partial d} = cos\alpha\frac{\partial}{\partial n} + cos\beta\frac{\partial}{\partial \tau_1} + cos\gamma\frac{\partial}{\partial \tau_2}; \tag{4.4}$$

where the direction cosine of d is $(cos\alpha, cos\beta, cos\gamma)$ along the direction n, τ_1, τ_2 and $cos\alpha^2 + cos\beta^2 + cos\gamma^2 = 1$.

Using harmonic surface mapping, the only discontinuity issue could appear on the segmentation curves on the boundary surface. On each such curve c shared by two adjacent regions, the harmonic surface mapping can guarantee C^1 continuity on the normal direction n and the tangential direction τ_1 along c. Therefore the difference of the any directional derivative from the adjacent domains is bounded by the difference of the tangential derivatives $\frac{\partial}{\partial \tau_2}$ resulting from the the separately computed surface harmonic maps. This term is usually quite small. In most of our experiments, $|\frac{\partial f_1}{\partial \tau_2} - \frac{\partial f_2}{\partial \tau_2}| < 0.02$.

Using biharmonic surface mapping with a carefully developed boundary condition will provide first order smoothness along the cutting curve on the surface boundary. However, solving a 4th order biharmonic equations on the surface is much more expensive. Harmonic mapping is also biharmonic and provides relatively good boundary condition, so we simply use it for boundary positional constraint.

After fixed the surface mapping f, then each given derivative function g will indicate a unique specific biharmonic mapping. Besides using our linear approximation method to decide function g, here we also explain two other ways to construct g.

Harmonic-based boundary condition. This boundary condition first needs to compute the harmonic volumetric mapping in each region. Then it computes each biharmonic volumetric mapping with g equals to the average value of the normal derivative of its neighboring harmonic mappings.

We can see in this case if there is only one region, the resultant biharmonic mapping will be exactly the harmonic mapping. If there are more than one region, the intuition behind it is to both capture the harmonic mapping's good properties and improve the continuity along the boundary interface. We can verify that this

boundary condition has C^1 continuity and also keeps linear precision property (See Appendix for details).

The main disadvantages of this boundary condition setting is its increased computation complexity, since the computation of the g is based on the harmonic results. The total computational time doubles.

Clamped boundary condition. Another natural yet simple setting is to have $g = 0$. Setting the normal derivative function to zeros gives an implicit tangent boundary condition. This is different from the harmonic solution which has tangent discontinuity. We can see this boundary condition also satisfies $\frac{\partial \Phi_1(x)}{\partial n_1(x)} = \frac{\partial \Phi_2(x)}{\partial n_1(x)} = 0$, when $x \in \partial\Omega_{12}$. So it has C^1 continuity along the boundary interface. But it does not keep linear precision property, because $\frac{\partial \Phi_1(x)}{\partial n_1(x)} = 0$ which is usually inequivalent to $\frac{\partial r(x)}{\partial n} = A^T \times n$.

4.4.4 Measuring Mapping Distortion

Our computation algorithm does not depend on a tessellation of the volumetric region Ω and it has closed form. In order to measure the mapping distortion, we implement a metric on the tesselated tetrahedral mesh of Ω (without ambiguity, we also denote this tet-mesh as Ω). We evaluate the mapping $\phi(v_i)$ for each vertex $v_i \in \Omega$, while linearly interpolate the mapping inside each tetrahedron. Considering the Jacobian of the transformation defined on each tetrahedron, we can measure its condition number. As suggested in [87], such a condition number is an indicator of the Jacobian and is invariant to scale and rotation.

Suppose a tetrahedron T consists of four vertices $v_n, n = 0, 1, 2, 3$ with coordinates $X_n \in \mathcal{R}^3$. Define edge vectors $e_{k,n} = X_k - X_n$ with $k \neq n$ and $k = 0, 1, 2, 3$. Vertex v_n has three incident edge vectors, $e_{n+1,n}, e_{n+2,n}$ and $e_{n+3,n}$, where all the indices are modulo four. The Jacobian matrix at node n, denoted by $M_{T,n}$, is

composed of the columns of these edge vectors, namely,

$$M_{T,n} = (-1)^n(e_{n+1,n}, e_{n+2,n}, e_{n+3,n}).$$

Suppose under the mapping, the original tetrahedron deforms to a new tetrahedron T', whose corresponding matrix is $\mathbf{M_{T',n}}$. Consider the matrix $\mathbf{S_n}$ that transforms $\mathbf{M_{T,n}}$ to $\mathbf{M_{T',n}}$. Then $\mathbf{S_n} = \mathbf{M_{T',n}} \cdot \mathbf{M_{T,n}^{-1}}$. We can verify that $\mathbf{S_n}$ is independent of n. Therefore we write $\mathbf{S} = \mathbf{M_{T',n}} \cdot \mathbf{M_{T,n}^{-1}}$. Then the condition number of a tetrahedron is:

$$k(\mathbf{S}) = |\mathbf{S}||\mathbf{S^{-1}}|,$$

where $\mathbf{M_{T,n}}$ is invertible when T has the positive volume. $k(\mathbf{S})$ measures the condition number of the transformation between the original and mapped tetrahedron. We adopt the Euclidean norm of $\mathbf{S}$: $|\mathbf{S}| = [tr(\mathbf{S^T S})]^{1/2}$, then $k(\mathbf{S}) \geq 3$ where in the optimal case $k(\mathbf{I}) = |\mathbf{I}||\mathbf{I^{-1}}| = 3$. We use $k(\mathbf{S})$ to evaluate the volumetric mapping computed on tetrahedral meshes.

Fig. 4.3 shows a comparison on the polycube parameterization of Max-Planck model (also see Fig. 4.8(a,b)), using harmonic and biharmonic mappings. From the distributions of the tetrahedral condition numbers shown in Fig. 4.3 (a,b), we can see that the biharmonic mapping and harmonic mapping have similar stretching distortion. But for the boundary elements, the biharmonic model introduces less stretching in their transformations. This is visualized in Fig. 4.3 (c,d), where the boundary tetrahedra deformed from polycube domain under biharmonic mapping are less stretched than the deformation guided by the harmonic map.

4.4.5 Improving Computation Efficiency

In Section 4.2.2, the boundary fitting for ϕ is formulated as solving a linear system $Aw = t$ and it is solved by using SVD. However, SVD decomposition is slow for large matrices. To handle complex volumetric data, we have to restrict the size

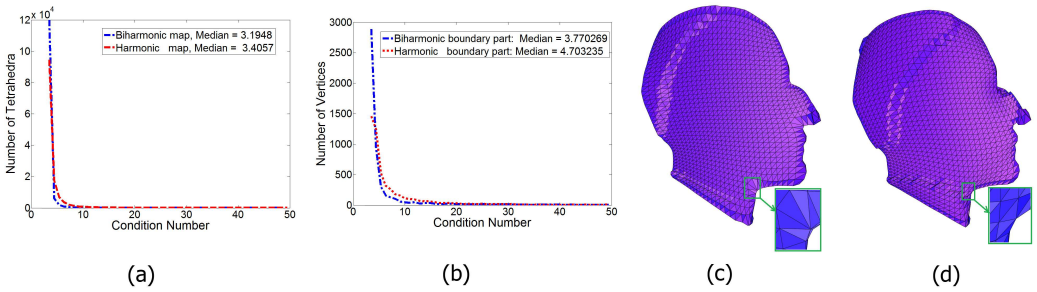


FIGURE 4.3. (a) The tetrahedral condition number distributions of the harmonic and biharmonic maps; (b) the tetrahedral condition number on the boundary elements; (c,d) Polycube parameterization of the Max-Planck model using harmonic and biharmonic mappings, (c) shows the deformed tetrahedra mesh under the harmonic map, while (d) shows the deformed tetrahedra mesh under the biharmonic map.

of constraints points and singularity points N_C and N_S. This may significantly reduces the boundary fitting accuracy: either we lack enough particles (when N_S is small) for designing fine fields to fit the boundary condition, or we lack enough constraint points (when N_C is small) to sample and reflect the shape variance on the boundary. In [3], the truncated SVD is used to avoid overfitting and improve the numerical stability and efficiency. The computation of SVD is still very expensive for big solid models.

To improve the stability (overcome the singularity of the linear system and avoid over-fitting) and improve the efficiency of the boundary fitting, we apply an additional regularization term to impose penalty on the norm of coefficients of singularity points, which is also known as ridge regression [88],

$$E_3 = \sum_{j=1}^{N_S} h_j^2 + \sum_{j=1}^{N_S} b_j^2.$$

The new boundary fitting is formulated as minimizing

$$E = E_1 + E_2 + \alpha E_3, \tag{4.5}$$

where

$$E_1 = \sum_{i=1}^{N_C} \left| \sum_{j=1}^{N_S} (h_j H(\mathbf{q}_j, \mathbf{p}_i) + b_j B(\mathbf{q}_j, \mathbf{p}_i)) - f(\mathbf{p}_i) \right|^2,$$

$$E_2 = \sum_{i=1}^{N_C} \left| \sum_{j=1}^{N_S} (h_j \nabla H \cdot n(\mathbf{p}_i) + b_j \nabla B \cdot n(\mathbf{p}_i)) - g(\mathbf{p}_i) \right|^2 ,$$

and $\{\mathbf{p}_1, \ldots, \mathbf{p}_{N_C}\}$ are a set of constraint points sampled on the domain boundary $\mathbf{p}_i \in \partial\Omega$. $\alpha > 0$ is a parameter which controls the weight balance of boundary fitting and regularization. Note that (4.5) is still quadratic, which could be denoted as $E = \frac{1}{2}\mathbf{x}\mathbf{t}\mathbf{r}A\mathbf{x} + \mathbf{b}\mathbf{t}\mathbf{r}\mathbf{x} + c$, where the positive definite matrix A can be pre-assembled. With an appropriate α, the minimization of this quadratic energy E can be efficiently solved using LU-decomposition. Compared with SVD, this will significantly improve the computational efficiency and numerical stability. Therefore, this framework can now handle big volumetric data more efficiently.

TABLE 4.1. Statistics Comparison between the SVD solver [3] and the new solver. The constraint point ratio N_c/N_Ω and source point ratio N_s/N_Ω are 0.4 and 0.8 respectively, where N_Ω is the number of vertices on the source boundary surface $\partial\Omega$. The computation time is measured in seconds.

Model		LU		SVD	
Model Name	N_Ω	Time	Boundary Fitting Error	Time	Boundary Fitting Error
Omotondo/Sphere	3002	54.66	0.4308165e-3	555.31	0.13626636e-2
PCube/2-Torus	6622	388.32	0.2532e-3	3626.03	0.1e-2
Sphere/Max-Planck	9002	946.08	0.111371e-4	8568.29	0.1890862e-3
Sphere/Igea	15002	469	0.46195e-5	1180	0.8689456e-3

To pick a suitable parameter α, we first show that α is related to the condition number of the coefficient matrix A. The numerical stability of the system depends on the condition number of the coefficient matrix. Smaller condition number indicates better numerical stability. Suppose we set a threshold value K for the condition number of A, namely, to make $cond(A) \leq K$. Then from Eqn. (4.5), $E = E_1 + E_2 + \alpha E_3$, the coefficient matrix A also consists of 3 terms $A = A_1 + A_2 + \alpha A_3$ where the semi-definite matrix A_1 comes from E_1, semi-definite matrix A_2 comes from E_2, and diagonal matrix A_3 comes from E_3.

We can compute the singular value of $A_1 + A_2$. Let σ_{max} be the maximum singular value of $A_1 + A_2$ and its minimum singular value is 0 (since A_1 and A_2 are under-determined). Also the singular value of A_3 is 1. So the condition number of the matrix A is $cond(A) = \frac{\alpha+\sigma_{max}}{\alpha}$. From $cond(A) \leq K$, it has $\frac{\alpha+\sigma_{max}}{\alpha} \leq K$ and $\frac{\sigma_{max}}{K-1} \leq \alpha$.

Second, according to [89], if we solve the linear system $Ax = b$ by LU decomposition, and the elements of A and b are accurate up to s decimal places to the left of the decimal point ($s \approx 10^{-13}$ based on IEEE 754 float type) and $cond(A) \approx 10^t$, where $t \leq s$, then the computed solution is accurate to about $s - t$ decimal places to the left of the decimal point.

So, we set the threshold value of condition number K based on the desired accuracy of the solution, then calculate α from such K. In our experiments, we take $K \approx O(10^6)$, then the computed solution is with $O(10^{-7})$ accuracy. For the spherical mapping of the Omotondo model, we have $\sigma_{max} = 36825.09$ and set $K = 2.5 * 10^6$, then $\alpha \geq \frac{36825.09}{2.5*10^6-1} \approx 0.015$. Similar computation can be applied in other models.

We test this on several models and show the side-by-side comparison on their running time and boundary fitting error, using same numbers/positions of source and constraint points. And the *boundary fitting error* is the average squared distances $|f(\mathbf{p}_i) - \Phi(\mathbf{p}_i)|^2, \mathbf{p}_i \in \partial\Omega$ between the target boundary points and the images of boundary points under the mapping. The results are shown in table 4.1, which shows the improvement on both the computation efficiency and fitting accuracy.

4.5 Experimental Results

We implement our mapping computation in C++ and perform experiments on a 3GHz Pentium-IV PC with 4G RAM. Our experimental data include the heterogeneous data and decomposed volume data.

4.5.1 Mapping Heterogeneous Volume Data

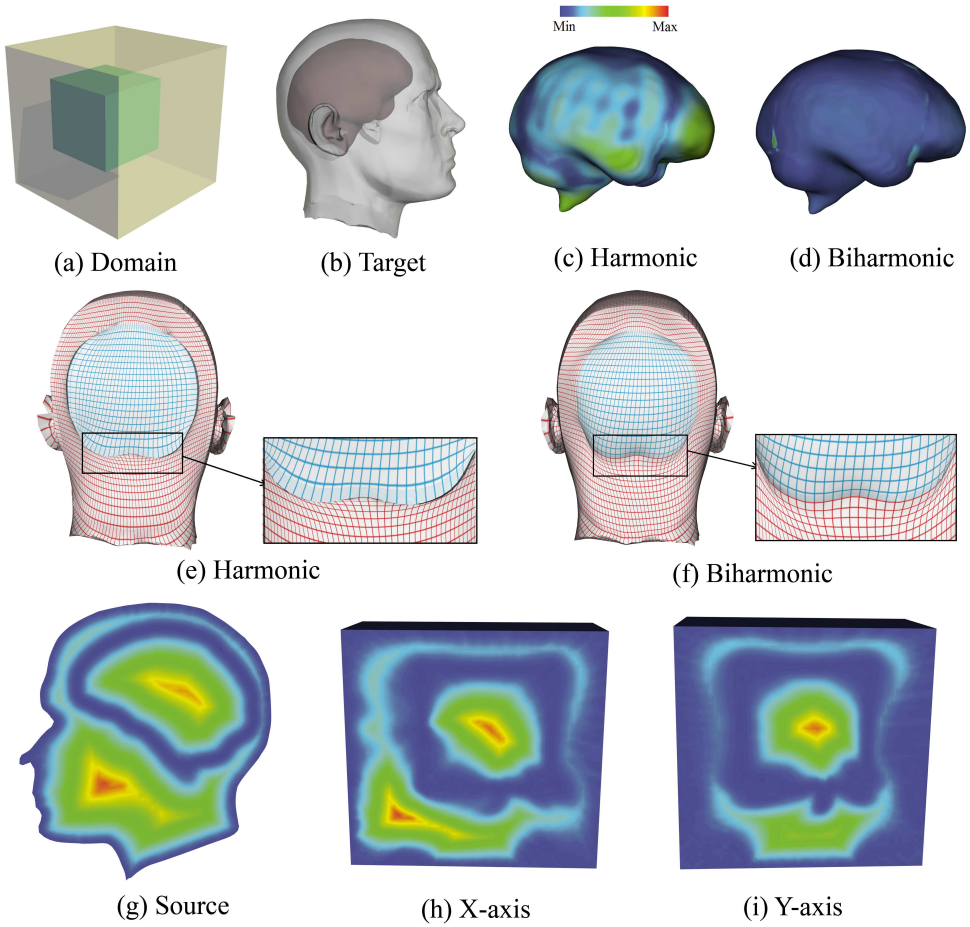


FIGURE 4.4. Parameterization of an heterogeneous Head Model onto a Cube Domain(surface vertex number is 20k). (a)The cube domain, (b) the head model, with the brain region to be mapped onto the interior cube in (a). The derivative discontinuity ($\delta_n = |\frac{\partial \Phi_1}{\partial n} - \frac{\partial \Phi_2}{\partial n}|, x \in \partial\Omega_1$) of the harmonic mapping (c) versus biharmonic mapping (d). Mapping distortion and boundary smoothness are also illustrated through parametric coordinates in (e) and (f), shown from a cross-section along Y-axis. (g-i) visualize the biharmonic mapping distribution using the transferred distance field: (g) the original distance field; (h,i) the transferred fields in x and y directions.

Figure 4.4 illustrates an example of multiple-layered volumetric model. We try to parameterize a head model (b) onto a cube domain Ω (a), with the brain region and the other layers being handled separately: we want the brain region $M_1 \subset M$ inside the head to be mapped to the smaller cube region Ω_1 inside Ω, and we denote the region outside the brain as $M_2 = M \backslash M_1$, whose corresponding parametric domain is $\Omega_2 = \Omega \backslash \Omega_1$. If we consider the volumetric mapping on both M_1 and M_2 : $\Phi_1 : \Omega_1 \rightarrow M_1$ and $\Phi_2 : \Omega_2 \rightarrow M_2$, naturally, along the brain cortex iso-surface $\partial\Omega_1$, we want the mappings Φ_1 and Φ_2 to be smooth and obtain not only the positional continuity but also derivative consistency. If the harmonic volumetric mapping is used to handle both regions separately, then the derivative transition along the isosurface is not smooth, as indicated by the parametric coordinates in (e) . When we use the biharmonic volumetric mapping, we can guarantee the nice derivative smoothness, as shown in (f). The derivative discontinuity $\delta_n = |\frac{\partial \Phi_1}{\partial n} - \frac{\partial \Phi_2}{\partial n}|, x \in \partial\Omega_1$ is computed and color-encoded in (c) and (d) to show numerically the mapping smoothness along the cutting boundary. Through the side-by-side comparison, biharmonic mapping demonstrates much better smoothness. We also use the color-encoded distance field to visualize the mapping result. When a map $\Phi : \Omega \rightarrow M$ is computed, the color-encoded (red indicate the maximum while blue indicates the minimum) distance field defined on one object can be transferred to the other, by plotting the color of a point $\mathbf{P} \in \Omega$ on its corresponding image $\phi(\mathbf{P}) \in M$ (or inversely, plotting the color of $\mathbf{P} \in M$ on $\phi(\mathbf{P}) \in \Omega$). We visualize the biharmonic volumetric mapping result in (g-i) by the transferred distance field from head/brain to the cube. Figure (g) shows the distance field defined on the head/brain domain while the transferred distance field are shown in (h) and (i) from cross sections in x- and y- directions.

4.5.2 Mapping Decomposed Volume Data

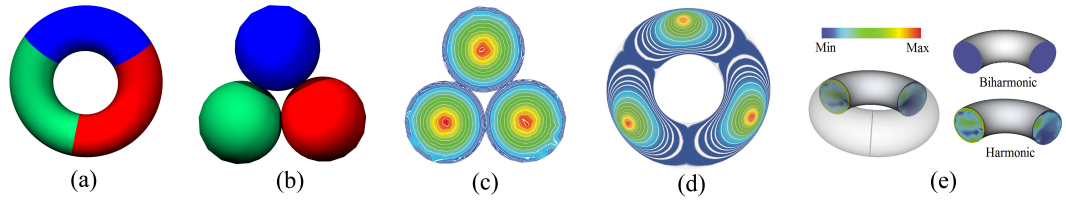


FIGURE 4.5. Biharmonic Mapping on Decomposed Models. (a) decomposition of a solid torus model; (b) a corresponding parametric domain; (c) color-encoded distance field on the parametric domain; (d) transferred distance field under the mapping; (e) boundary derivative errors δ_n under both harmonic and biharmonic mapping indicates the biharmonic mapping leads to smooth derivative transition.

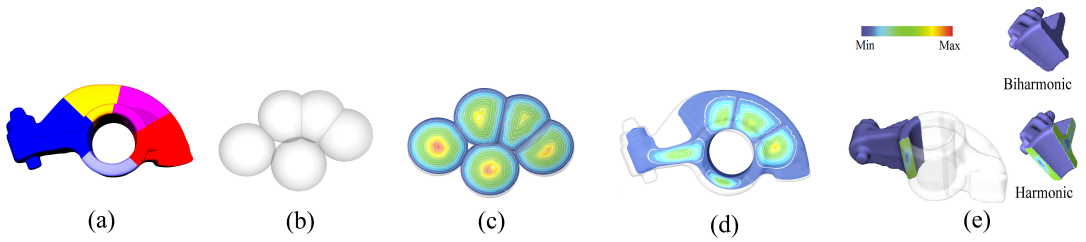


FIGURE 4.6. Biharmonic Mapping on Decomposed Models. (a) decomposition of a solid rocker-arm model; (b) a corresponding parametric domain; (c) color-encoded distance field on the parametric domain; (d) transferred distance field under the mapping; (e) boundary derivative errors δ_n under both harmonic and biharmonic mapping indicates the biharmonic mapping leads to smooth derivative transition.

Figure 4.5 and 4.6 show our biharmonic mapping computation applied on decomposed Torus and Rocker-arm models. The decomposition of the torus and Rocker-arm are illustrated using colors in (a). The corresponding target shape is shown in (b). We parameterize each sub-region of torus onto a convex cell (as indicated by the corresponding color). Adjacent sub-regions share a surface interface and we enforce the positional consistency as well as the derivative constraint g with local linear approximation setting across the boundary. The color-encoded distance field computed on each convex shape (c) is transferred into the rocker-arm (d) (i.e. the color of each point $\mathbf{x}$ on Ω is rendered on $\Phi(\mathbf{x}) \in M$) to visualize the mapping

result. Figure 4.5(e) and Figure 4.6(e) illustrate the normal derivative discontinuity δ_n along the cutting boundary. Compared with harmonic mapping, our biharmonic mapping with derivative boundary condition enforced along the cutting boundary brings a parameterization with smooth derivative transition across the boundary.

4.6 Applications

In this section, we apply our model in the hex-remeshing and 3D dynamic temporal data registration.

4.6.1 Hex-remeshing

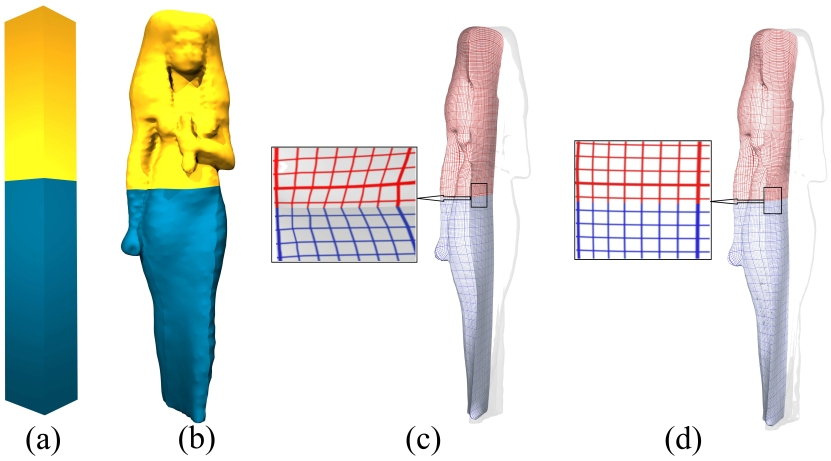
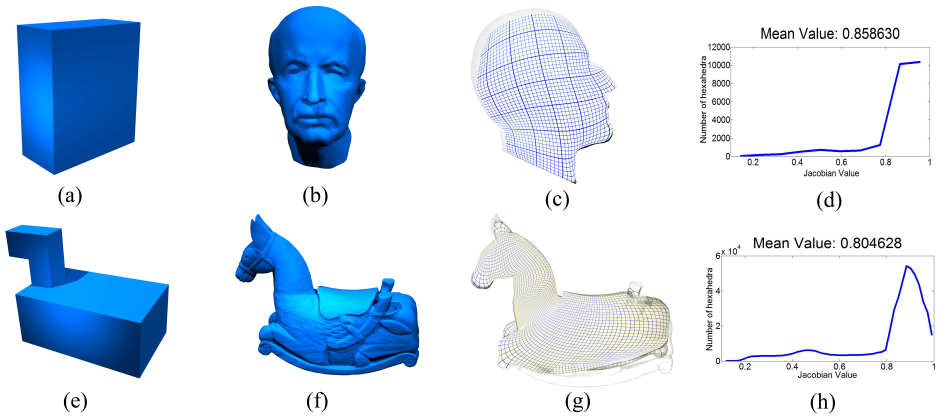


FIGURE 4.7. Biharmonic Mapping on Isis Decomposed Model(surface vertex number is 20k). (a) The polycube domain; (b) the corresponding Isis parametric domain; The regular hex-structure on the cubes transferred onto the Isis model using (c) harmonic mapping and (d) biharmonic mapping.

The hex-remeshing converts the tessellation of a volumetric model from a tetrahedral mesh to a hexahedral mesh. An effective hexahedral remeshing is desirable in many scientific and engineering tasks such as finite element simulation and iso-geometric analysis, because a high-quality hexahedral mesh can significantly facilitate the computation and analysis in many of these tasks.



(a) (b) (c) (d) (e) (f) (g) (h)

FIGURE 4.8. Biharmonic Polycube Mapping for Hex-remeshing. A cube model (surface vertex number: 12.5k) (a) is used to parameterize the Max Planck's head model (b), and the resultant hexahedral meshing of the Max Planck model is shown in (c). A poly-cube model (surface vertex number: 100k) (e) is used to parameterize the Chinese horse model (f), and the resultant hexahedral mesh is shown in (g). (d) and (h) show the distributions of Jacobian value of the deformed hexahedra.

We apply our biharmonic volumetric mapping to hex-remeshing through the polycube domain. A polycube domain is a shape glued by a lot of small regular cubes [90]. The basic idea is to compute the mapping between a solid model M and a solid polycube domain Ω. Then on Ω we can sample points regularly and obtain a perfect hexahedral mesh, then with the mapping this hexahedral mesh can be transformed into the geometry of M. If the mapping ϕ has very small distortion, the generated hexahedral mesh for M has very good quality.

In our experiment, we construct the polycube domain and obtain the surface mapping between the model and the polycube boundary using the method introduced in [82], then compute the volumetric mapping use the algorithm we proposed. We also compare our result with that generated through harmonic volumetric mapping suggested in [2].

Figure 4.4(e,f) show an example of hex-remeshing of heterogeneous volumetric models. The side-by-side comparison shows that compared with the harmonic

mapping, the biharmonic mapping generates a more smooth and therefore more desirable hexahedral mesh.

Figure 4.7 shows the parameterization of the Isis model M (b) to the polycube domain Ω (a). M is decomposed into two sub-parts: $M = \{M_1, M_2\}$, where M_1 is mapped to the upper cube and M_2 is mapped to the lower cube. The transferred hex-meshes from the polycube to Isis model are shown in (c) and (d), based on the harmonic and biharmonic volumetric mappings, respectively. We can see the biharmonic mapping provides significantly better smoothness along the cutting boundary.

We show some more biharmonic polycube hex-remeshing results in Figure 4.8. We parameterize a Max planck's head model (b) onto a unit cube (a), then the remeshed model is shown in (c). The hex-remeshing of a Chinese-horse model is shown in (e-g). We use the scaled Jacobian metric [91] which has a range [-1,1] with 1 being optimal to measure the quality of the generated mesh. The distribution of the Jacobian value in (d,h). And our results are comparable to the paper [92].

4.6.2 Consistent Parameterization of Temporal Data

Another application of our biharmonic volumetric mapping is registration. Registration has been a ubiquitous technique, which is widely used in many applications in computer vision, computational medicine, and etc. An accurate registration indicates the natural differences between models in a quantitative way, and benefit the subsequent analysis tasks. Here we use a medical example to demonstrate the usage of biharmonic volumetric mappings in 3D dynamic temporal registration (See next chapter for more detail).

Our experiment is conducted on the temporally scanned lung data collected during multiple respiratory cycles of some patient having the lung tumor. The goal is to accurately register the deforming volumetric region, so that the tumor's

deformation (and its related motion) can be described, analyzed, and used to guide the subsequent radiation (in which the the beam should correctly target the tumor without touching surrounding normal organs or tissues) [93].

Due to the natural heterogeneous property of the data we need to handle, the model decomposition is desirable. Also it is necessary to preserve the C^1 continuity across the boundary of lung and tumor domains. For example, for vessels that are surrounding organs and tissues, as shown in Figure 4.9, when we segment the organ from the surrounding environment, these vessels are also cut apart across the partitioning boundary. Transformations (mappings) inside and outside the partitioning boundary are computed separately but their transitions shall naturally be smooth.

FIGURE 4.9. Vessels Near the Tumor.

Figure 4.10 shows some registration/matching results computed using biharmonic mapping. In (a), the volumetric models are extracted from three different time frames during one respiratory cycle. We parameterize all these data onto a common sphere domain for the consistent parameterization. To analyze the motion and deformation of the tumor region (red), it shall be mapped onto the red small sphere, while the left region of its surrounding lung tissue shall be mapped to the left outside region. Then we compute our biharmonic model on these two sub-parts

separately while preserving C^1 continuity along the iso-surface. The mapping results are visualized using the transferred distance field from the canonical domain (b) to each model shown in (c). With the consistent parameterization over the canonical sphere, the registration between any pair of the models can be computed immediately, as shown in (c). The motion and deformation of the entire lung region can then be depicted on the sphere domain, and used to predict the tumor's trajectory.

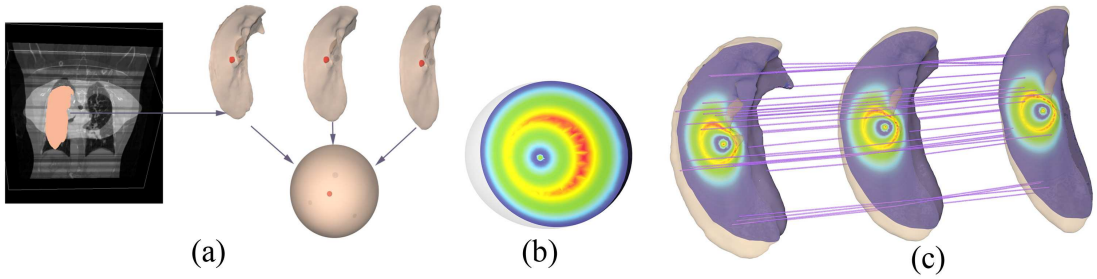


FIGURE 4.10. Dynamic registration of the heterogeneous lung/tumor models (total surface vertex number is 20k). (a) The sphere domain and deformed lung/tumor models; (b) The distance field of the sphere domain; (c) The transferred distance field and registration results of lung/tumor models.

4.7 Summary

We propose a biharmoinc volumetric mapping computation framework using the fundamental solution method. Compared with harmonic mapping, the biharmonic mapping allows better boundary control. In a divide-and-conquer computation framework for mapping huge, complex, or heterogeneous volumetric models, this biharmonic model is desirable because it can provide nice smoothness across the cutting boundary. Our proposed boundary derivative setting algorithm can ensure the linear precision property of our biharmonic transformation. Compared with harmonic mapping based on fundamental solutions [2, 3], our new solving scheme is also more accurate and efficient.

One limitation of our current volumetric mapping framework is that this mapping result depends on the boundary surface mapping. Intuitively, the volumetric mapping and its boundary surface mapping are closely related to each other. Smoother boundary surface mapping will give us better volumetric mapping. However, instead of computing surface mapping first, directly solving the volumetric mapping could be more natural in some scenarios. We will explore along this direction in the near future.

We will also explore the computation of biharmonic inter-surface mapping, and the numerical improvement of the computational efficiency and accuracy. Furthermore, we will explore other applications of biharmonic volumetric mapping.

Chapter 5
Medical Application

In previous chapter, we introduced two works in volumetric mapping for heterogeneous volumetric data. And biharmonic volumetric mapping can ensure higher continuity along the boundary.

In this chapter, we will build a computational framework for modeling the respiratory motion of lung tumors based on the biharmonic volumetric mapping. This framework provides a 4D parametric representation that tracks, analyzes, and models movement to provide more accurate guidance in the planning and delivery of lung tumor radiotherapy.

5.1 Introduction

Lung cancer is the most common cause of cancer related deaths in the US, with only 10 to 15 percent of lung cancer patients surviving five years after diagnosis. more than half of all solid tumors receive external beam ionizing radiation as part of treatment that combines radiotherapy with chemotherapy or with surgery and chemotherapy. The ultimate goal of radiation treatment, or radiotherapy, is to treat the disease while avoiding damage to the normal tissue and critical organs that surround the tumor.

Much research is directed to lung can radiotherapy, yet there is room for significant improvement. Conventional radiotherapy involves administering a prescribed tumor-killing dose, typically around 50 to 70 Gray (Gy: a unit of absorbed radiation dose) over anywhere from 25 to 35 sessions, or treatment fractions.

Technological advances and a deeper understanding of radiobiology–the study of how human tissue responds to high doses of x-ray radiation therapy–have enabled

two alternative. The first is to deliver the same total dose in significantly fewer fractions, say one to five, and accompany delivery with improved image guidance using offline and online projection x-ray imaging and computed tomography(CT). The second is to increase the total radiation dose using the historical dose per fraction(2Gy) with the same image guidance. Literature on lung tumor irradiation have frequently cited the benefits to local tumor control of escalating the dose per fraction.

The spatial relationship of lung tumors with important normal tissue structures, such as the spinal cord, esophagus, heart, brachial plexus, normal lung tissue, bronchial tree, and trachea, make total dose escalation infeasible in many cases. Moreover, the motion of tumor during respiration complicates radiation treatment planning for lung cancer. The respiratory cycles also involves movement of normal tissue structure. This movement heavily influences tumor motion, which in turn affects the tumor's surrounding organs. For these reasons, treatment planning for intrathoracic radiation (radiation within the chest cavity) requires tools that can provide the highest delivery precision and accuracy.

To meet that need, we propose a lung tumor modeling and computational framework that facilitates the tracking and prediction of respiratory movement and the deformation of organs surrounding the tumor. Preliminary results of our framework's application show that, relative to existing methods, it is more accurate and computationally efficient in the radiotherapy treatment of lung cancer. It is also flexible enough to generalize to the radiotherapy treatment of other pathologies.

5.2 Radiotherapy's Challenges

Radiotherapy treatment typically begins with the patient entering the radiation treatment room and lying supine on the treatment table, as in Figure 5.1. For lung

radiotherapy treatment, the patient is generally in a customized immobilization device to limit natural motion during treatment. In-room lasers ensure that the patient is in the right 3D position with respect to the radiation treatment machine.

FIGURE 5.1. A patient being treated with external beam radiotherapy using a Varian linear accelerator. Accuracy is essential in lung cancer treatment because respiratory movements can cause the tumor and its surrounding tissue to move and change shape.

Outside the treatment room, physicians and staff take mega- and kilovolt images of the patient to make sure that the subsequent radiation will adequately treat the tumor. Finally, the radiotherapist initiates the radiation treatment, carefully monitoring the patient during the process.

Studies have amassed considerable scientific evidence on both the benefits of dose escalation and the perils of normal tissue toxicity, and there have been tremendous gains in radiotherapy planning and delivery precision. These developments have made it critical for radiotherapy treatment to accurately capture the geometry

of the temporarily deforming organ, particularly in lung cancer radiotherapy, in which respiratory motion causes thoracic anatomy to change continuously in all four dimensions–3d space and time.

The ideal radiotherapy guidance requires complete spatiotemporal knowledge of the movement and deformation of the volume–the region that includes the solid tumor and surrounding tissues and organs–to be treated. However, pretreatment imaging remains one of the weakest aspects of current radiotherapy guidance. Typically, radiotherapists use 4DCT to acquire raw CT images or projections over several respiratory cycles. An external motion monitor aids in phase or amplitude sorting, placing projections into bins according to respiratory phase or displacement, respectively. Pretreatment imaging ends with the generation of a time series of 3D volumes, which describes the volume's motion over a single representative cycle.

Typically, radiotherapists use a maximum- or average-intensity projection (MIP or AIP) from all phases to define a motion-inclusive internal target volume. The MIP represents the superposition; the AIMP is the average. The internal target volume servers as the basis for a treatment plan and becomes the ground truth for the subsequent radiotherapy stages.

Although researchers have proposed various strategies to improve 4DCT-based planning and delivery paradigms, several fundamentally challenging issues remain to be tackled:

One issue is cycle-to-cycle complexities. As Figure 5.2 shows, respiratory motion is more complex than a single cycle can characterize. MIF and AIP images do not account for these complexities, which can lead to errors. Another challenges stems from forcing CT projection data from several cycles into a few respiratory phase bins, which can lead to severe artifacts. Figure 5.3 shows some examples of these

severities. Indeed, one study found that 45 of 50 patients had at least one artifact, ranging in size from 4.4 to 56.0 mm (mean magnitude of 11.6mm).

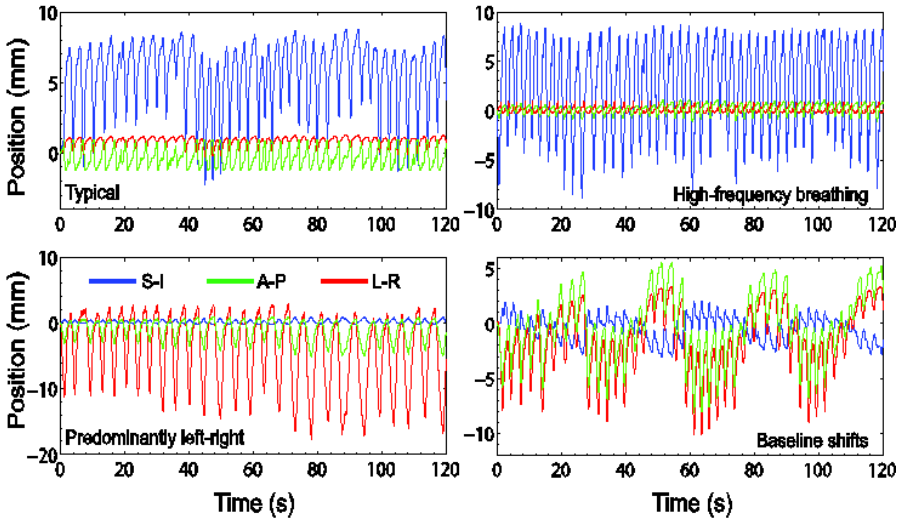


FIGURE 5.2. Representative lung tumor motion traces recorded from four patients using the Synchrony system. The traces are indicative of the wide variety of respiratory patterns that are observed clinically. (Image from Y.Suh et al., "An Analysis of Thoracic and Abdominal Tumor Motion for Stereotacic Body Radiotherapy Patients," Physics in Medicine and Biology, July 2008, pp. 3634-3640.)

Finally, at 29 to 40 milliSieverts (mSv: Sievert is the unit of any of the quantities expressed as dose equivalent), the equivalent dose for 4DCT is about four times higher than that for 3DCT (3 to 10 mSv). Such a high imaging dose discourages more frequent imaging and long-term monitoring.

These challenges make it highly desirable to have a computational radiotherapy-guidance strategy that users a 4D motion model developed from 4D magnetic resonance imaging (MRI) and a planning 3DCT acquired at a reference phase. The idea is to update the model with real-time position information and then deliver the corresponding updated radiation fluence map (a 2D map of the x-ray intensity distribution from the medical linear accelerator).

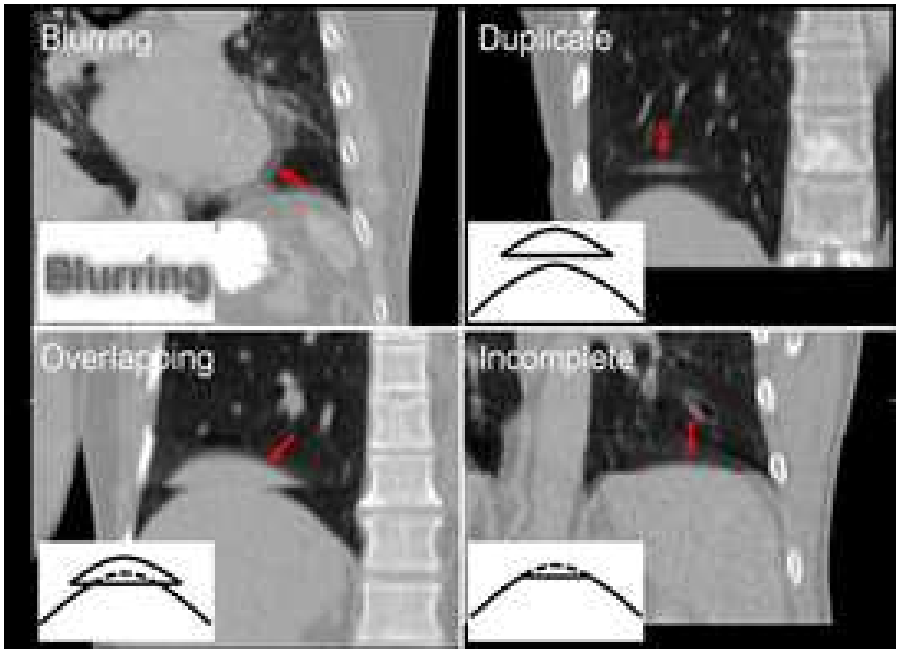


FIGURE 5.3. Examples of motion-induced artifacts observed in lung 4DCT. Clockwise from top left: blurring, duplicate, incomplete, and overlapping artifacts. (Image from T.Yamamoto et al., "Retrospective Analysis of Artifacts in Four-dimensional CT Images of 50 Abdominal and Thoracic Radiotherapy Patients," Int'l J. Radiation Oncology, Biology, and Physiology, vol.72,no.4,2008,pp.1250-1258.)

5.3 Computational Framework

To accurately model the tumor and surrounding sensitive structures, we developed a 4DCT geometric modeling framework that tackles several important tasks in analyzing and processing 3DCT volumes and sequential fluoroscopy images (projected 2D images). Our current framework uses temporally dense MR images (sliced 2D images) to refine our integrated 3DCT volumes. Registering MR images with 3D volumes is usually simpler and can be more robust.

A 4D model parameterizes irradiation volume temporally. From this deforming 4D parametric model, it is possible to extract a tight planning margin to spare normal tissues from dose radiation during delivery.

As Figure 5.4 shows, our framework has two phases: off-line modeling and planning, and online prediction and delivery. As their names imply, the first phase

focuses on modeling tumor motion and planning radiotherapy, while the second phase helps guide treatment delivery.

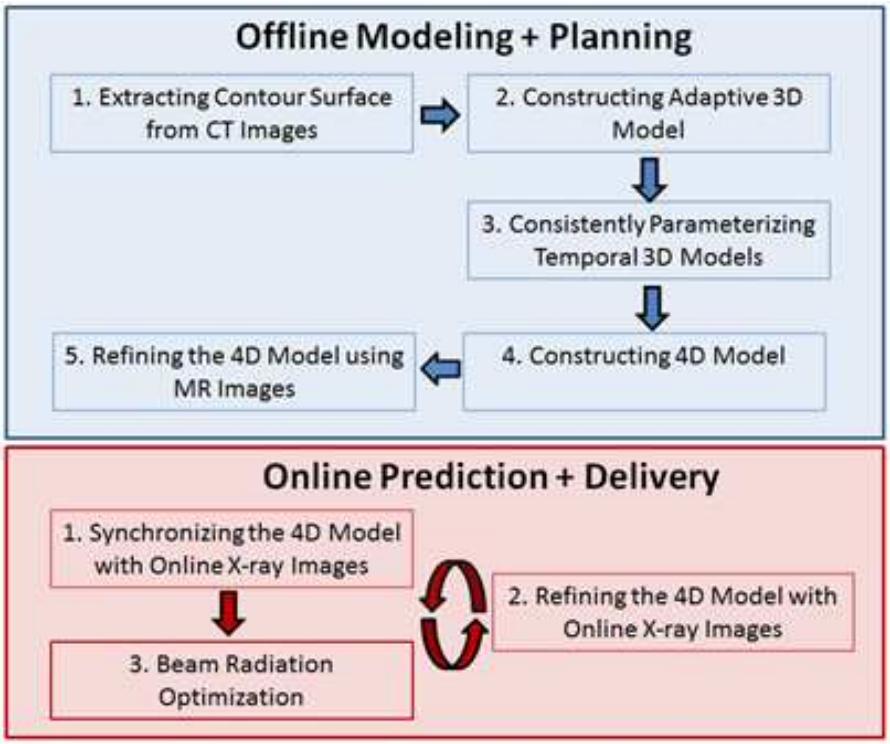


FIGURE 5.4. Two phases in the computational framework. The first phase, offline modeling and planning, emphasizes radiotherapy planning and is based on CT and magnetic resonance (MR) scans. The second phase focuses on online prediction and treatment delivery. Real-time scanned 2D images aid in synchronizing and refining the 4D model, which the system uses to predict the tumor's trajectory and geometry and guide treatment delivery.

5.4 Offline Modeling and Planning

The offline phase is concerned with modeling the tumor's motion and deformation. To better predict tumor movement, which could be affected by neighboring organs and tissues during respiration, the model covers the entire neighboring volume region.

Contour Segmentation. The first step in this phase is to clearly segment the tumor and surrounding structures within the potential irradiated volume from 3D-

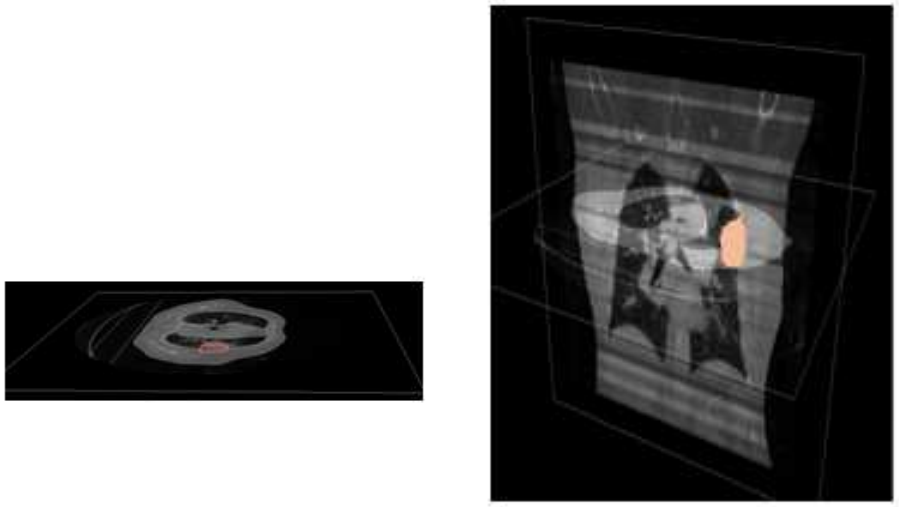
CT or MR images. Because reliable 3D image segmentation against noise is critical at this stage, we developed the template-guided graph cut (TGGC) algorithm.

To perform the 3D graph cut, the algorithm uses a novel metric that combines image intensity (noise filtered) and a predesigned implicit scalar field that captures the template shape and serves as a reference. TGGC reaches globally optimal segmentation; simple user interactions can iteratively and adaptively refine the extracted contour. A postprocess uses morphological operations (performing open then close operations with radius-3 disks) to smooth the extracted contour.

Figure 5.5 and 5.6 show our preliminary results. Figures 5.6a through 5.6d show that TGGC is superior to existing segmentation methods in extracting the object of interest from the image background. Figure 5.6a through 5.6c show the results of using the level set, watershed, and original graph cut methods, respectively. Compared with the level set method, a popular method of segmenting medical image data, TGGC takes about a third of the computation time to segment the entire 3DCT volume. The segmentation also more tightly bounds the tumor's contour. Although faster than TGGC, the watershed method leads to significant over-segmentation, an outcome that TGGC avoids.

From Volume Image to a Tetrahedral Mesh. After extracting contours of both the tumor and surrounding structures, our framework models both the geometry and material of the entire volume instead of modeling only boundary shells. It then adaptively tetrahedralizes (converts a volume image into a tetrahedral mesh) the irradiated volume.

Figure 5.7 illustrates an example of a tetrahedral representation of a tumor and surrounding tissue. Such a finite element representation is much sparser than the grid-based image representation, and any local region can be coarsened or refined adaptively when necessary. This effectively represents the region's inherent struc-



(a) (b)

FIGURE 5.5. Tumor segmentation from CT images. (a) Segmentation performed in 3D; the red solid is the tumor; (b) segmentation visualized in a cross-section.

ture, while conforming to important features and materials. It is then possible to use a linear interpolation or a spline function to represent the deforming irradiated volume.

One critical issue is how to compute the optimal sampling points for tetrahedralization. Given the number of sample points, the goal is to minimize the mean square error (mean of the square difference) between the linear interpolation and the corresponding original intensity values. Intuitively, selected points should balance the uniformity and the sampling of sharp features. It is then possible to use Delaunay tetrahedralization to tessellate the model on these sampled points.

Volumetric Mapping and Interpolation. After representing volumetric regions of interest using tetrahedral meshes, our algorithm computes bijective volumetric mapping to consistently parameterize 3D volumes and then interpolate the 4D temporal model.

Bijective volumetric mapping involves computing a lowly distorted mapping (small angle and area distortion, which is physically natural) between two consecu-

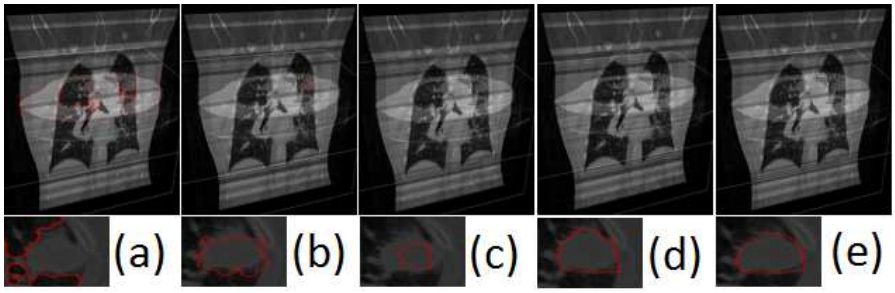
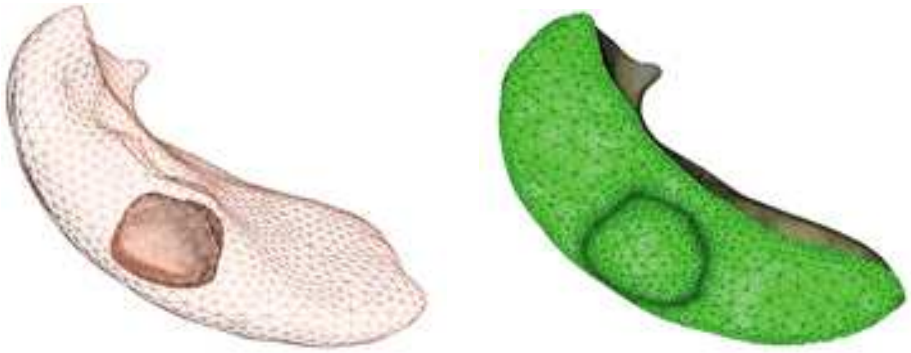


FIGURE 5.6. Comparison of segmentation results. Tumor contour segmentation based on (a) level set method, (b) watershed method, (c) original graph cut method, and our TGGC algorithm (d) without de-noise and (e) with de-noise. The final segmentation (e) is suitable for subsequent tumor modeling and tracking tasks.



(a) (b)

FIGURE 5.7. Tumor and surrounding lobe:(a) contour surfaces and (b) adaptive conversion to a tetrahedral mesh.

tive volumes through a coarse-to-fine framework. Initially, the algorithm extracts corresponding features and matches them in 3D. Then taking these features as soft constraints, it computes surface and volumetric mappings between corresponding contours and volumes. The result is a consistent parameterization of all temporally sequential volumes onto one common domain $D(u, v, w)$.

With this one-to-one correspondence in hand, the algorithm can crete a continously deforming 4D model–$M(u, v, w, t)$–by computing the physically natural shape interpolation between two consequent models. Given the (u, v, w) parameter coordinate in the domain, it can trace a point's trajectory under different time t;

similarly, given any t, it can obtain the 3D volume's location and geometry at that moment.

4D Model Refinement. The first three steps inn this phase rely on CT images, which can have very high resolution and thus very good spatial accuracy. However, CT imaging requires a high dose, and a frequent and long CT imaging sequence is impossible. For this reason, CT-sampled volumes tend to be temporally sparse.

To compensate for this disadvantage, our algorithm refines the 4D model computed in the previous mapping and interpolation step using a sequence of 3D MR images and cross-model volumetric parameterization to register the volume from the MRI and the interpolated volume. It can then correct the consistently deforming 4D model according to the matching results. The refined continuous parametric 4D model is ready for use in determining the trajectory and geometry of the volume of interest.

5.5 Refining the 4D Model and Predicting Motion

In the online phase, our framework uses real-time scanned 2D images (orthogonally mounted x-ray) to synchronize and refine the 4D model. From the refined 4D model, the framework can then predict the tumor's trajectory and geometry. This prediction makes it easier to optimize the treatment beam to target the most desirable radiation positions.

Model Synchronization and Refinement. During radiation delivery, it is possible to obtain a 2D time series of x-ray projection images and register them with the moving 3D model. Our framework then uses the results of the matching to correct the 4D model. The optimal mapping is searchable within a conservative time range, starting from the last synchronized point.

Beam Radiation Optimization. With the deforming 3D volume, it is possible to optimize the beam's radiation direction. As Figure 5.8 shows, ideally, the beam should be planned so that it can see the tumor clearly without being visually blocked by other organs. Otherwise, the radiation will hit those organs before it reaches the tumor. To solve this problem, we propose using an efficient hierarchical integer linear program (HILP). Our recent work in autonomous robotic environment inspection has demonstrated that HILP scheme can be very efficient in solving this challenging 3D region-inspection problem.

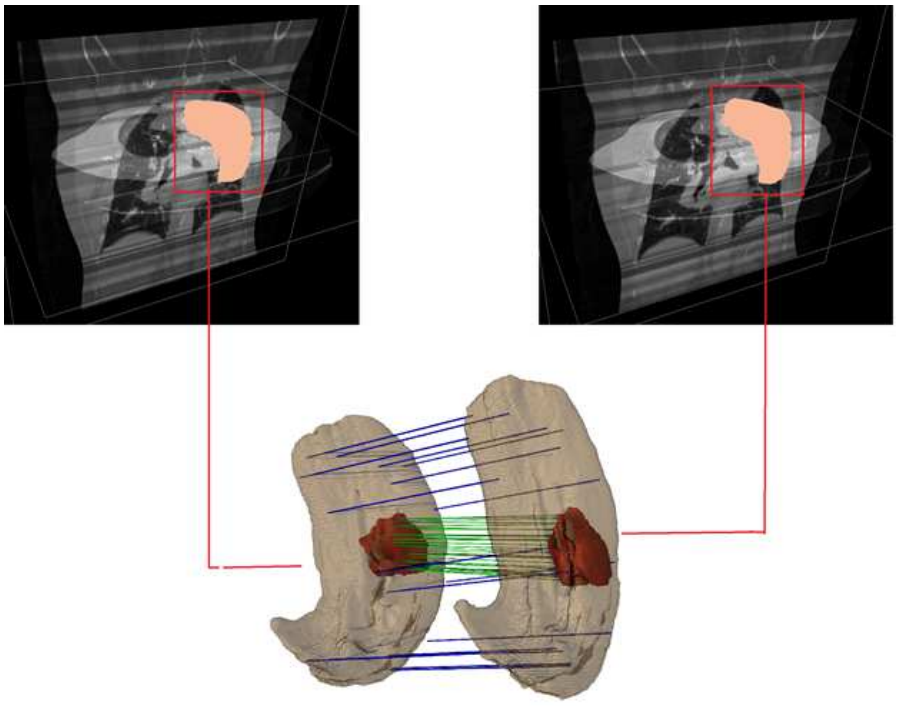


FIGURE 5.8. Tracking temporally deforming tumor and surrounding lobe. The red area represents the tumor under two times sequences. Green and blue lines indicate the correspondence between the solid regions in the two time frames (the 3D tumor and the lobe at the bottom row are rotated 90 degrees in the y-axis to better visulize the matching).

5.6 Summary

Our computation framework and platform for lung tumor modeling and tracking can greatly enhance radiotherapy planning and delivery, in large part because it effectively integrate reliable 3D image segmentation; volumetric modeling, analysis, and parameterization; physical and geometric interpolation; and tracking techniques. Generalizing our computation paradigm would allow other medical planning and treatment regimens to benefit from this integration.

Our framework already cuts segmentation preprocessing time by roughly two-thirds, and we expect advances in parallelism to decrease that time even further. Segmentation preprocessing takes $O(nlogn)$ time and $O(n)$ space, where n is the pixel number of each volume image. We can solve volumetric mapping computation within $O(m^3)$ time and $O(m^2)$ space, where m is the vertex number of interested objects.

All these geometric computation algorithms are local and can be effectively parallelized. Using GPUs can improve the entire pipeline's efficiency. We plan to explore a GPU implementation and expect to achieve significant efficiency improvement in both offline data analysis and planning, and greater optimization of real-time treatment.

Chapter 6
4D Registration for Motion Modeling

In previous chapter, given a sequence of volume image, we built the motion estimation model to describe the movement and deformation of lung and tumor based on the pairwise 3D mapping or registration. The do the interpolation to get a deforming 4D motion model. This 3D pairwise registration can ensure accurate inter-image matching and good spatial smoothness of the deformation. However, in the temporal dimension, such interpolated motion can be non-smooth and thus not physically natural. In this chapter, we present a 4D image registration based on a 4D (3D spatial + 1D temporal) free-form B-spline deformation model, which ensures interpolated motions with both spatial and temporal smoothness [94].

6.1 Introduction

Image registration is important in medical image analysis and image-guided radiotherapy management. For example, in lung cancer radiotherapy, it can establish the correspondences among the 4D (sequential volume) CT images. This correspondence can then be used to build a motion estimation model that describes the movement and deformation of organs during respiratory cycles.

Pairwise 3D Registration. Given a sequence of volume images, the conventionally popular registration approach is through the *pairwise 3D registration*, which computes a set of mapping functions f_{ij} between image i and image j. These registrations $\{f_{ij}\}$ can be interpolated to obtain a deforming volumetric model. 3D registration algorithms often approximate natural deformation between two shapes through minimizing certain physical deformation energies [95, 96, 97] or geometric smoothness [52, 98]. Pairwise 3D registrations have two general limitations. One

is its sensitivity to the selection of the reference frame, especially when describing a motion sequence undergoing large deformations.

More importantly, the second limitation of 3D piecewise registration is its lack of control on the smoothness of the resultant motion trajectory. High quality 3D image registration may provide accurate inter-image matching, but in the temporal dimension, the motion/deformation composed from the individually computed 3D matching is often not smooth and thus less physically natural [53].

4D Registration. Registration across sequential images can be solved in a 4D space directly. This can avoid the bias caused by the selection of a predetermined reference frame, and can directly enforce both spatial and temporal smoothness of the transformation to indicate more physically natural deformations. This is called the *4D Registration* [53, 52, 54, 99], which attracts a lot of attentions recently. Metz et al. [52] construct a common domain, and solve a 4D registration by reducing intensity matching errors. The computation is resolved to minimizing a non-linear and non-smooth optimization problem with many local minima, which requires a good initial guess to get a desirable matching. Geometric information such as feature correspondence can guide the optimization to avoid many undesirable local minima. Wu et al. [54] suggest a 4D registration framework utilizing both image intensity and feature guidance, and solve the registration on a refined implicit domain for lung image data. They also propose a groupwise registration scheme [100] by iteratively resolving feature correspondence and thin-plate spline deformation, which demonstrates high matching accuracy for brain image registration. This method, however, is relatively expansive. Xu et al. [101] also integrate feature guidance in 4D-image registration to improve the matching performance. To model the respiratory motion which is nearly periodic, the algorithms of [52, 101] use a geometric constraint that enforces the average deformation to be

identity. This constraint assumes the inhale and exhale phases are symmetric and the temporal samplings on these two phases are uniform, which are often not the case.

Main Contributions. In this chapter, we propose a symmetric registration computational model that uniformly incorporates a set of geometric, intensity, and motion characteristics. The transformation is represented using two 4D spline functions, which can be used to easily construct a continuously deforming parametric geometry that tracks the respiratory motion of the lung tumors/organs from the sequential CT scans. The experimental results demonstrate significant performance improvement from existing registration/tracking models in matching accuracy, trajectory smoothness, and transition inverse consistency.

6.2 Algorithm

6.2.1 Feature Extraction and Matching

To handle the registration of volumetric images, Scovanner et al. [102] proposed a 3D SIFT descriptor and applied it in action recognition. Cheung and Hamarneh extended SIFT to N-Dimension SIFT [103] (N-SIFT) and showed its effectiveness on volumetric images. However, neither descriptor is scale or rotation invariant. In order to adequately describe medical images of deforming organs, we try to improve the existing 3D SIFT descriptor in this work.

The procedure of N-SIFT includes scale space extrema detection, orientation assignment, descriptor construction and matching [103]. For an input volume image, we first extend method [104] to locate its keypoints with sub-pixel accuracy.

One limitation of N-SIFT is that it performs poorly when some local regions rotate. To reduce its sensitivity to rotation, multiple directions (rather than just one dominant direction used in [102]) can be assigned to a keypoint region. We calculate an orientation histogram of a region around the keypoint with width $6 * \sigma$

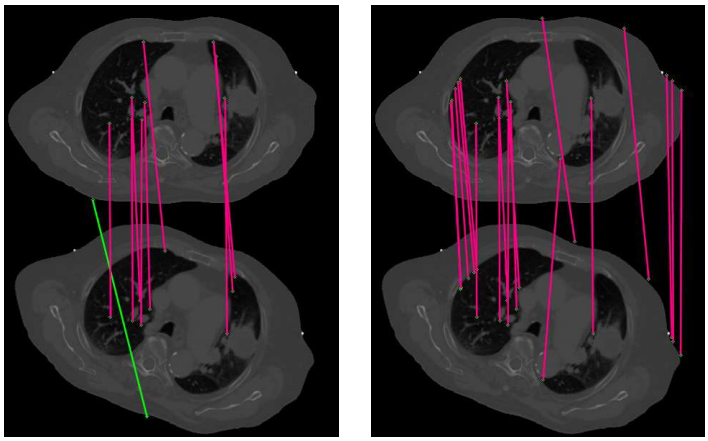
where σ is the scale of the keypoint. This orientation histogram has 36×36 bins covering 360° of the orientations. The highest peak of the histogram corresponds to the dominant direction. Here, we consider local peaks within 80% of the highest peak also to be the directions of the keypoint region. Region that is chosen in the construction of the descriptors can be reoriented according to its directions by multiplying its rotation matrixes [102]. Descriptors are constructed on the reoriented regions. Multiple directions make our 3D SIFT more robust to the image rotation.

N-SIFT is also not scale-invariant, since it computes the descriptor based the original image and the size of the region around the keypoint is fixed. Here we propose a scale selection method to deal with scale change. We construct the descriptors on the corresponding Gaussian smooth image. The region around the keypoint is defined and divided into 4×4×4 patches. We set its patch size to be $3 * \sigma$ which is related to its scale. In this way, our descriptor perform much more scale invariant.

For the matching process, since N-SIFT matches descriptors directly, a point may be matched to more than one point. Some of the matchings are wrong. Hence, we further conduct a RANSAC algorithm to deal with this one-to-many correspondence issue and remove the outliers. In our work, before doing 4D registration we first perform feature extraction and matching between every two consecutive volume images, then choose those consistent correspondences that appear in all time frames.

A simple example is given in Fig. 6.1 to demonstrate the rotation invariance of the new descriptor. A lung CT volume image (dimension 465×300×20) is used as the reference; its subsequent image has rotated by 20° along Z axis (this happens when the patient rotates). We compare the correspondences found using N-SIFT and our improved 3DSIFT. N-SIFT method extracts fewer matching pairs and has

some error matchings while our algorithm works correctly and find more matched features. Note that this matching is done on volume images although we only illustrate a 2D cross section.



(a) N-SIFT (b) Improved 3DSIFT

FIGURE 6.1. Feature Extraction and Matching.

6.2.2 4D Free-form B-spline Deformation

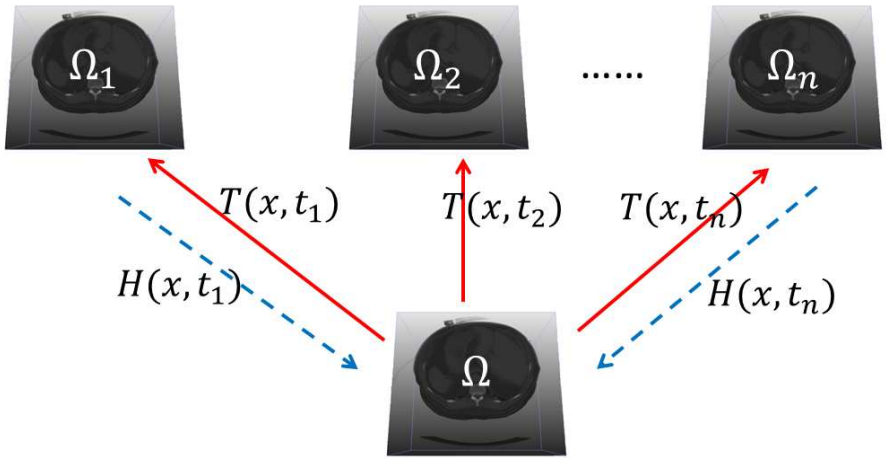


FIGURE 6.2. Model illustration.

Given sequential volume images $I_1, I_2, \ldots, I_\Gamma$, where each image $I_i(\mathbf{x}) : \Omega_i \rightarrow \mathcal{R}, \mathbf{x} \in \Omega_i \subset \mathcal{R}^3$ is a 3D intensity function[1], we want to compute a temporally deforming 3D model $T(\mathbf{x}, t) : \Omega \times \mathcal{R} \rightarrow \mathcal{R}^3, \Omega \subset \mathcal{R}^3$ that correlates all the input

[1]For sequential CT scans, their parametric domains Ω_i simply overlay in $\mathcal{R}^3$

images, as illustrated in Figure 6.2. A point $\mathbf{x} \in \Omega_i$ in I_i is correlated with a point $\mathbf{x}'$ in I_j by $\mathbf{x}' = T(T^{-1}(\mathbf{x}, t_i), t_j)$. Then, a continuous 4D deforming image $I(\mathbf{x}, t)$ can be constructed using the intensity function defined in the first image I_1, namely, $I(\mathbf{x}, t) = I_1(T(x, t_1))$.

To obtain this deforming parametric geometry and the deforming image, we need to explicitly compute two 4D functions: (1) a forward 4D parameterization T, spatially defined on a common parametric domain, $T : \Omega \times \mathcal{R} \to \mathcal{R}^3$, and (2) its inverse mapping $H = T^{-1} : \mathcal{R}^3 \times \mathcal{R} \to \Omega$ which maps coordinate space of the deforming images $\Omega_i (\subset \mathcal{R}^3) \times \mathcal{R}$ to the common domain.

To model the nonrigid freeform deformations of human organs during respiratory cycles, we use 4D B-spline functions to approximate these two transformations T and H, through which both the spatial and temporal smoothness can be formulated easily. The B-spline approximation for T can be formulated as:

$$T(\mathbf{y}) = \mathbf{x} + \sum_{\mathbf{y}_k \in N_{\mathbf{y}}} p_k \beta^r(\mathbf{y} - \mathbf{y}_k), \tag{6.1}$$

where $\mathbf{y} = (\mathbf{x}, t)$, $\mathbf{y}_k$ is a knot on the parametric domain $\Omega \times \mathcal{R}$; $\beta^r(\cdot)$ is the r-th order multidimensional B-spline polynomial (here we take $r = 3$); $\mathbf{p}_k \in \mathcal{R}^3$ are B-spline control points to be solved, and $N_{\mathbf{y}}$ denotes $\mathbf{y}$'s neighboring local support regions where the basis functions are nonzero. The knots $\mathbf{y}_k$ are defined on a 4D regular grid, uniformly overlaid the 4D image.

Because the inverse of B-spline transformation cannot be derived in close-form and the B-spline-approximated T may not be injective, we explicitly approximate this inverse mapping using another B-spline transformation H using a same formulation to eq (6.1). Then with T and H, a transformation F^{ij} from any frames i to j can be composed as

$$F^{ij}(\mathbf{x}) = T(H(\mathbf{x}, t_i), t_j), \mathbf{x} \in \Omega_i. \tag{6.2}$$

The entire 4D registration problem is formulated as an optimization on T and H that minimizes an objective function:

$$E = E_I + \alpha E_F + \lambda E_S + \rho E_C, \tag{6.3}$$

where E_I measures the intensity matching error, E_F measures the feature alignment, E_S measures the spatial and temporal smoothness of the deformation, E_C measures the inverse consistency, and α, λ, ρ are weighting factors.

Intensity Matching Error. With the assumption that the corresponded points have the same intensity, the registration should minimize the intensity differences of corresponded points. We can derive the intensity difference between corresponded points in any pair of images I_i and I_j taken in time t_i and t_j. For any point $\mathbf{x} \in \Omega_i$ in time t_i, its corresponding location in time t_j can be composed by H and T. The accumulated difference between $I_i(\mathbf{x})$ and the intensity of its corresponding coordinate in t_j can be formulated as:

$$\tilde{E}_I = \frac{1}{|S||\Gamma|^2} \sum_{t_i \in \Gamma} \sum_{t_j \in \Gamma} \sum_{\mathbf{x} \in S_i} (I_j(T(H(\mathbf{x}, t_i), t_j)) - I_i(\mathbf{x}))^2, \tag{6.4}$$

where S_i is the sets of spatial voxel coordinates in each Ω_i and for $\forall i, |S| = |S_i|$. Simultaneously solving both T and H is expensive. We first solve a forward parameterization T, then iteratively, fix the parameterization in one direction and optimize the other (see Section 6.2.3 for the complete algorithm).

To solve the initial forward parameterization T without knowing H, we formulate the reduction of intensity error by minimizing the intensity variance:

$$T_I = \frac{1}{|S||\Gamma|} \sum_{\mathbf{x} \in S} \sum_{t \in \Gamma} (I_t(T(\mathbf{x}, t)) - \bar{I}(\mathbf{x}))^2, \tag{6.5}$$

where $\bar{I}(\mathbf{x})$ is the average intensity value follows the forward parameterization: $\bar{I}(\mathbf{x}) = \frac{1}{|\Gamma|} \sum_{t \in \Gamma} I_t(T(\mathbf{x}, t))$. $S \subset \Omega$ are the spatial voxel coordinates (e.g. coordinates of all the pixels) and $\Gamma \subset \mathcal{R}$ contains the temporal coordinates indexing temporal sample images. After obtaining the initial T, we iteratively optimize H and T by minimizing:

$$E_I = T_I + \tilde{E}_I. \tag{6.6}$$

Feature Alignment Error. The optimization only guided by intensity has many local minima, and geometric features can help effectively avoid many undesirable solutions.

Using the algorithm of [105], we extract feature points using a slightly modified 3D SIFT algorithm, then compute a set of consistently corresponded feature points $\{p_{ij}\}$ across the entire sequence of images, where p_{ij} indicates the i-th feature point on time t_j, where $i = 1, \ldots, N, j = 1, \ldots, |\Gamma|$.

Each consistently corresponded feature point has a parametric coordinate $m_i, i = 1, \ldots, N$ in Ω, which is mapped to the feature p_{it} in image I_t at time t. The feature correspondence in the forward parameterization should penalize the deviation of $T(m_i, t)$ from p_{it}:

$$T_F = \frac{1}{N|\Gamma|} \sum_{t \in \Gamma} \sum_{i=1}^{N} ||p_{it} - T(m_i, t)||^2, \tag{6.7}$$

For the inverse parameterization H, the variance of $H(p_{ij}, j)$ should be minimized:

$$H_F = \frac{1}{N|\Gamma|} \sum_{i=1}^{N} \sum_{t \in \Gamma} ||H(p_{it}, t) - \bar{H}(p_{i*})||, \tag{6.8}$$

where $\bar{H}(p_{i*}) = \frac{1}{|\Gamma|}\sum_{t\in\Gamma} H(p_{i,t}, t)$ is the average coordinates of the i-th feature p_{i*}. Finally, the entire feature alignment error is:

$$E_F = T_F + H_F. \tag{6.9}$$

Deformation and Motion Smoothness. The transformation (hence both parameterizations T and H) should be spatially and temporally smooth. The 2nd-order derivatives of the B-spline transformation functions can be derived as the smoothness energy to minimize:

$$\begin{aligned} E_S &= T_S + H_S; \\ T_S &= \tfrac{1}{|S||\Gamma|}\textstyle\sum_{\mathbf{x}\in S}\sum_{t\in\Gamma}(||\frac{\partial^2 T}{\partial \mathbf{x}^2}||_F^2 + ||\frac{\partial^2 T}{\partial t^2}||^2 + 2||\frac{\partial^2 T}{\partial \mathbf{x}\partial t}||_F); \\ H_S &= \tfrac{1}{|S||\Gamma|}\textstyle\sum_{\mathbf{x}\in S_i}\sum_{t\in\Gamma}(||\frac{\partial^2 H}{\partial \mathbf{x}^2}||_F^2 + ||\frac{\partial^2 H}{\partial t^2}||^2 + 2||\frac{\partial^2 H}{\partial \mathbf{x}\partial t}||_F). \end{aligned} \tag{6.10}$$

Inverse Consistency. The registration problem between two time frames I_i and I_j should be symmetric, i.e. the correspondences established between I_i and I_j do not depend on the order we choose to deform [106]. Since the transformation between I_i and I_j is composed from T and H, this can be reached by making the composition of T and H to be an identity transformation as much as possible. The inverse consistency can be measured by:

$$E_C = \frac{1}{|S||\Gamma|}\sum_{\mathbf{x}\in S_i}\sum_{t\in\Gamma}||T(H(\mathbf{x},t),t) - \mathbf{x}||^2 + \frac{1}{|S||\Gamma|}\sum_{\mathbf{x}\in S}\sum_{t\in\Gamma}||H(T(\mathbf{x},t),t) - \mathbf{x}||^2, \tag{6.11}$$

6.2.3 Solving the Optimization

Simultaneously solving T and H reduces to a very expensive optimization problem. We develop an iterative algorithm to seek for the optimal solution. During each iteration, T (or H) is solved using a gradient-based optimization method algorithm proposed in [52], which uses a stochastic sampling strategy to reduce the

computational cost. With the B-spline representation we derive the derivatives of E_F, E_S, E_C explicitly, and we use the finite difference approximation to get the derivatives of E_I.

We first solve a forward parameterization T by minimizing $E = T_I + \alpha T_F + \rho T_S$ from equations (6.5,6.7,6.10), then with T fixed, we solve its inverse parameterization H by minimizing the entire objective function E in equation 6.3. Then iteratively, we fix one parameterization and revise its inverse parameterization, until the energy reduction is smaller than a threshold. This optimization algorithm is formulated as follows.

1) Compute an initial forward parameterization T by minimizing $T_I + \alpha T_F + \rho T_S$;

2) Fix T, and solve H by minimizing E;

3) Fix H, and solve T by minimizing E;

4) If E converges, STOP; otherwise GOTO 2).

6.3 Experimental Results

We implement our registration model via a multi-resolution strategy and use linear interpolation in the spatial domain for the derivation of intensity values for any point not on a grid. Our algorithm was implemented in C++ using an Intel Xeon X5570 @2.93 GHz, 8GB RAM. In our experiments, we set the weight factors as $\alpha = 0.1, \lambda = 0.5, \rho = 0.5$.

6.3.1 Experiments using Public Datasets

We perform 4D registration using our algorithm on two public benchmark datasets: POPI [4] and DIR-lab [107]. The dataset from POPI has one 4D CT series including ten 3D volume images ($482 \times 360 \times 141$ pixels) representing ten different phases of one breathing cycle. We also select five datasets from the DIR-lab dataset (Case-1 to Case-5) where lardmarks are available. Each dataset contains

TABLE 6.1. The landmark predication error D_i and its standard deviation σ_i (in mm) of i^{th} time frame on the POPI-data [4]. $\bar{D}$ is the average MTRE.

	$D_1(\sigma_1)$	$D_2(\sigma_2)$	$D_3(\sigma_3)$	$D_4(\sigma_4)$	$D_5(\sigma_5)$	$D_6(\sigma_6)$	$D_7(\sigma_7)$	$D_8(\sigma_8)$	$D_9(\sigma_9)$	$D_0(\sigma_0)$	$\bar{D}$
3D Reg.[52]	3.6(2.3)	2.3(1.8)	2.1(1.7)	2.2(2.0)	2.4(2.3)	2.9(2.4)	2.8(2.3)	2.1(1.7)	2.1(1.5)	2.7(2.1)	2.5
4D Reg.[52]	3.8(2.3)	2.6(2.0)	2.2(1.8)	2.2(2.0)	2.5(2.2)	2.9(2.3)	2.8(2.3)	2.2(1.8)	2.2(1.5)	2.8(2.2)	2.6
4D Reg.[101]	2.1(1.6)	1.8(1.5)	1.6(1.3)	1.6(1.2)	2.1(1.4)	2.4(1.7)	2.1(1.6)	1.7(1.0)	1.6(1.2)	1.9(1.6)	1.9
Our 1^{th} iter.	1.9(1.4)	1.6(1.2)	1.6(1.3)	1.8(1.5)	2.0(1.7)	2.0(1.7)	2.0(1.6)	1.6(1.2)	1.7(1.1)	2.2(1.6)	1.9
Our 2^{th} iter.	1.1(0.8)	1.2(0.9)	1.3(0.9)	1.2(0.8)	1.5(0.9)	1.5(1.0)	1.4(1.1)	1.1(0.6)	1.2(0.8)	1.2(0.8)	1.3
Our 3^{th} iter.	1.1(0.8)	1.2(0.9)	1.3(0.9)	1.2(0.8)	1.5(0.9)	1.5(1.0)	1.4(1.1)	1.1(0.6)	1.2(0.8)	1.2(0.8)	1.3

TABLE 6.2. The landmark predication error and its standard deviation $D_i(\sigma_i)$ (in mm) for the registration of DIR-LAB 4D dataset: $i = 1$ to 5 for Case-1 to Case-5.

	$D_1(\sigma_1)$	$D_2(\sigma_2)$	$D_3(\sigma_3)$	$D_4(\sigma_4)$	$D_5(\sigma_5)$
3D Reg.[52]	2.03(1.09)	0.72(0.44)	0.99(0.71)	1.14(0.81)	1.64(1.70)
4D Reg.[52]	2.12(1.09)	0.92(0.61)	1.39(0.93)	1.44(0.96)	1.85(1.69)
4D Reg.[101]	1.58(0.99)	0.70(0.57)	0.79(0.55)	0.91(0.75)	1.41(1.36)
Our Reg.	1.28(0.76)	0.56(0.34)	0.59(0.43)	0.69(0.49)	1.10(0.94)

6 sequential volume images. This CT pixel unit can be converted to real physical space units millimeter by multiplying a scaling factor which can be extracted from the image header file. Consistent landmarks are also available in the benchmark to measure how accurately their transformations are predicted. Denote the landmarks on frame-t as $Q_t = \{q_{t,1}, q_{t,2}, \ldots, q_{t,n}\}$, we evaluate the prediction accuracy of the registration using a *Mean Target Registration Error* (MTRE) on the landmarks on frame r:

$$D_r = \frac{1}{n|\Gamma|} \sum_{t \in \Gamma} \sum_{q_{r,i} \in Q_r} ||F^{rt}(q_{r,i}) - q_{t,i}||, \tag{6.12}$$

where F^{rt} is the transformation between frames r and t, composed by the forward and inverse parameterizations following equation (6.2).

Unlike existing 4D parameterization methods that solve mappings in two directions separately, our model uses a symmetric objective function that can be

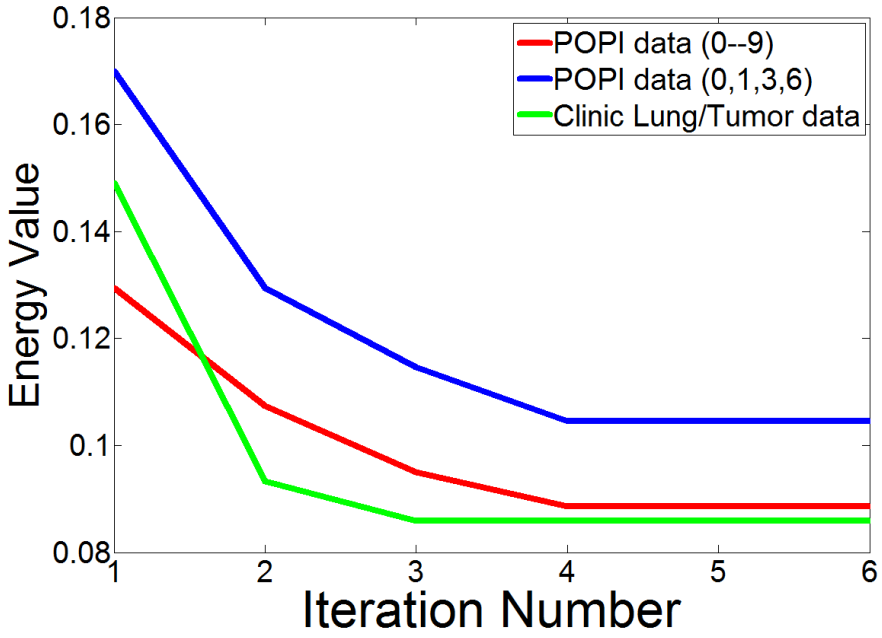


FIGURE 6.3. Convergence of Energy E.

optimized with guaranteed convergence. Fig. 6.3 shows the convergence of the energy E (eq-(6.3)) when parameterizing the POPI dataset. Our algorithm converges in 4 iterations.

We compared our registration results with existing 3D pairwise registration [52] and 4D registration [52, 101] algorithms using the benchmark data from POPI and DIR-lab. The results are documented in Table 6.1 and Table 6.2. On the POPI dataset, we evaluated the registration using the matching error of the consistent 40 landmarks. On each of DIR-lab datasets, both maximum inhale and exhale phases possess 300 landmarks, whose matching errors were used to evaluate the registration accuracy.

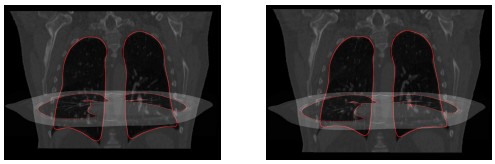
Our algorithm results in significantly smaller MTRE errors, i.e., better accuracy of landmarks prediction. Furthermore, in practice, the breathing cycles are often sampled in a non-perfectly uniform manner: the exhale is longer than the inhale. In this case, the geometric constraints adopted in [52] and [101], which enforces

the average movement of each point during the entire cycle to be identity, will not be correct. The geometric smoothness cost (eqn 6.10) suggested in our framework is a more robust description against this non-uniformity.

For lung motion tracking purpose, we first segment the lung contours using a template-guided 3D graph-cut algorithm improved from [93], and construct a finite element mesh model, then we temporally deform this 3D model and get an explicit tracking of them during respiratory cycles. Figure 6.4 illustrates a few snapshots of this tracking. In (a), a surface contour is segmented from image I_1, (b) shows the tracking contour in the 6-th time frame. (d) and (e) color-encode displacement fields computed in $1st$ and $6th$ time frames (Blue to Red: smaller to bigger displacement). One can also measure the registration accuracy by performing an extra segmentation on some image I_i, then compare the deviation from this segmentation to the predicted surface geometry. In (c), we compute the Hausdorff distance between two surfaces and color-encode this deviation. This Hausdorff illustrated matching, between the maximum inhalation (I_6) and maximum exhalation status (I_1) which undergoes a largest deformation, infers the maximum matching errors during the respiratory cycles.

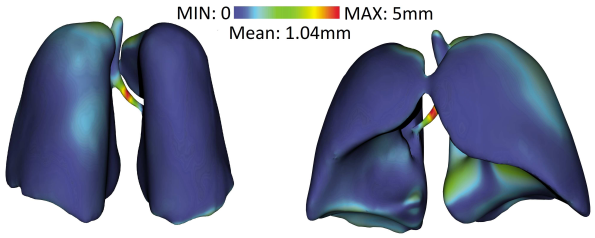
6.3.2 Motion Modeling of Our Clinical Lung Tumor Scans

Similarly, we use our 4D registration algorithm to build a deforming 3D finite element mesh model to track the motions of the lung and tumor captured in our clinic CT scans. The second row of Fig. 6.5 shows the segmented lung/tumor contour surfaces. The tracking and motion estimation of the lung and tumor during the entire respiratory cycle can be computed, as illustrated in the last row. The colors on the 2nd and 3rd surfaces encodes the Hausdorff difference between the deformed and target contour surfaces.

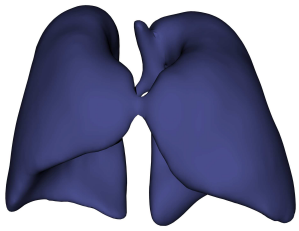


(a)S_1 from I_1

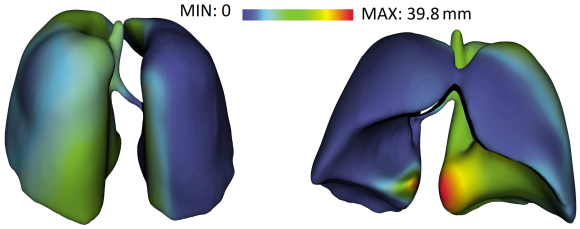
(b) $F^{16}(S_1)$ and I_6



(c) Registration Error on I_6



(d) Displacement field in $1st$ time frame



(e) Displacement field in $6th$ time frame

FIGURE 6.4. Lung/Tumor Tracking via a Deforming Surface Geometry. (a,b) illustrate the 3D iso-surfaces with the volume images (through the iso-x and iso-z cross sections). Red contours are the intersection between the cross section planes and the deforming lung surface. (a) shows the segmented iso-surface S_1 on I_1 while (b) shows the tracking result, which is the intersection of $F^{16}(S_1)$ and I_6. (d,e) color-encode on the tracked geometry $F^{16}(S_1)$ its displacement field from S_1. (c) color-encodes the registration error, which is the Hausdorff distance from $F^{16}(S_1)$ to S_6 (the iso-surface directly extracted from I_6).

Given two images I_i and I_j, we compute a difference image between the deformed I_i and I_j to evaluate the registration accuracy. Specifically, this difference image is $I_i(\mathbf{x}) - I_j(T^{ij}(\mathbf{x}))$ and can be normalized:

$$\delta_{ij} I(\mathbf{x}) = \begin{cases} \frac{|I_i(x) - I_j(T^{ij}(x))|}{I_i(x) + I_j(T^{ij}(x))}, & I_i(x) + I_j(T^{ij}(x)) \neq 0 \\ 0, & otherwise \end{cases} \tag{6.13}$$

Small δI indicates accurate registration. The first row of Fig 6.5 visualizes the 2D projection of the difference images between I_1, I_4, and I_4, I_8. We can see the difference values are small with the mean difference value of $0.48 * 10^{-3}$ and $1.2 * 10^{-3}$. These indicate that our registration introduces small matching errors. To further refine the matching near the regions with bigger registration error, in the near future, we will develop hierarchical trivariate T-spline function to support

adaptive refinement. More knots/control points can be inserted adaptively into these regions to reduce the registration error.

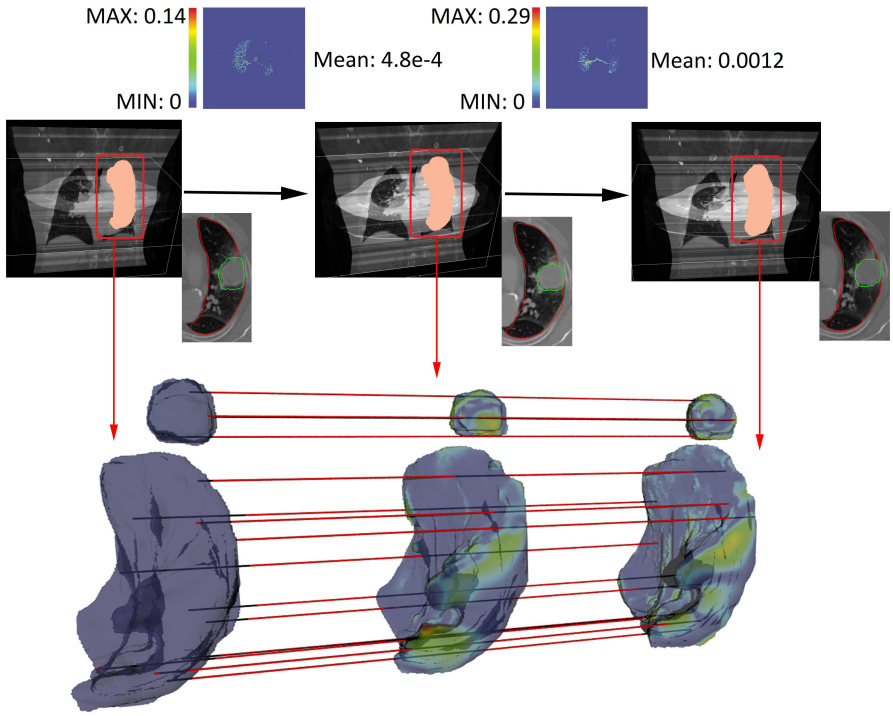


FIGURE 6.5. Dynamic contour and surface tracking and registration.

6.4 Summary

We propose an effective 4D registration algorithm for dynamic volume images. The 4D parameterization is represented using two coupled B-spline functions and solved by minimizing an objective function E measuring intensity matching, feature alignment, spatial and temporal smoothness, and transitive inverse-consistency. Compared with existing 3D and 4D registration models, this algorithm has unique advantages in matching dynamic volume image sequences that undergo relatively big nonrigid deformation and/or are non-uniform in the temporal dimension. To minimize E, we alternatively optimize the forward and inverse parameterizations T and H, which iteratively refines each other in a symmetric manner. Our exper-

imental results demonstrate that this computational model greatly improves the registration accuracy of existing methods.

Chapter 7
Conclusion

In this dissertation, we have studied on heterogeneous and dynamic volumetric data mapping for graphics and medical imaging applications. We first proposed an efficient way to compute feature-aligned harmonic volumetric mapping using the fundamental solution methods. It allows the alignment between three types of feature including feature points, feature lines and iso-surfaces. Then we improved the harmonic volumetric model to a biharmonic volumetric model which is a fourth-order partial different equation. Compared to the previous harmonic model, this can ensure higher continuity along the boundary. Thus it is more suitable for high complex volumetric data and heterogeneous data. Then we developed a computational framework for modeling the respiratory motion of lung tumors. We treated the lung with a tumor inside as a heterogeneous data and modeled its motion by computing biharmonic volumetric mapping between two 3D volumetric data. Last, we developed a 4D image registration algorithm to parameterize temporal CT lung volume images and used it for lung tumor motion analysis and tracking.

References

[1] X. Li, X. Gu, and H. Qin. Surface mapping using consistent pants decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 15(4):558–571, 2009.

[2] X. Li, X. Guo, H. Wang, Y. He, X. Gu, and H. Qin. Meshless harmonic volumetric mapping using fundamental solution methods. *IEEE Trans. on Automation Science and Engineering*, 6, 2009.

[3] X. Li, H. Xu, S. Wan, Z. Yin, and W. Yu. Feature-aligned harmonic volumetric mapping using mfs. *Computers and Graphics (SMI10)*, 34(3):242 – 251, 2010.

[4] J. Vandemeulebroucke, D. Sarrut, and P. Clarysse. Point-validated pixel-based breathing thorax model. In *Intern. Conf. on the Use of Computers in Radiation Therapy, 2007*.

[5] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. In *Experimental Mathematics*, volume 2, pages 15–36, 1993.

[6] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *SIGGRAPH*, pages 173–182, 1995.

[7] M. Desbrun, M. Meyer, and P. Alliez. Intrinsic parameterizations of surface meshes. *Comput. Graph. Forum*, 21(3):209–218, 2002.

[8] T. Kanai, H. Suzuki, and F. Kimura. Three-dimensional geometric metamorphosis based on harmonic maps. *The Visual Comput.*, 14(4):166–176, 1998.

[9] A.W.F. Lee, D. Dobkin, W. Sweldens, and P. Schröder. Multiresolution mesh morphing. In *Proc. SIGGRAPH*, pages 343–350, 1999.

[10] T. Michikawa, T. Kanai, M. Fujita, and H. Chiyokura. Multiresolution interpolation meshes. In *Proc. Pacific Graphics*, pages 60–69, 2001.

[11] E. Praun, W. Sweldens, and P. Schröder. Consistent mesh parameterizations. In *Proc. SIGGRAPH*, pages 179–184, 2001.

[12] V. Kraevoy and A. Sheffer. Cross-parameterization and compatible remeshing of 3D models. *ACM Trans. Graph.*, 23(3):861–869, 2004.

[13] J. Schreiner, A. Asirvatham, E. Praun, and H. Hoppe. Inter-surface mapping. *SIGGRAPH.*, 23(3):870–877, 2004.

[14] T. Kwok, Y. Zhang, and C.C.L. Wang. Efficient optimization of common base domains for cross-parameterization. *IEEE Transactions on Visualization and Computer Graphics, accepted*, 2011.

[15] X. Li, Y. Bao, X. Guo, M. Jin, X. Gu, and H. Qin. Globally optimal surface mapping for surfaces with arbitrary topology. *IEEE Trans. on Visualization and Computer Graphics*, 14(4):805–819, 2008.

[16] X. Li, X. Gu, and H. Qin. Surface matching using consistent pants decomposition. In *SPM '08: Proc. ACM Symp. on Solid and Physical Modeling*, pages 125–136, 2008.

[17] X. Li, X. Gu, and H. Qin. Surface mapping using consistent pants decomposition. *invited to IEEE Trans. on Visualization and Computer Graphics*, 2008.

[18] J. Bennett, V. Pascucci, and K.Joy. Genus oblivious cross parameterization: Robust topological management of inter-surface maps. In *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 238–247, 2007.

[19] K. Zhang and X. Li. Optimizing geometry-aware pants decomposition. In *Proc. Pacific Conference on Computer Graphics and Applications*, pages 11–16, 2012.

[20] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*, Mathematics and Visualization, pages 157–186. Springer, 2005.

[21] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Found. Trends. Comput. Graph. Vis.*, 2(2):105–171, 2006.

[22] K. Hormann, B. Lévy, and A. Sheffer. Mesh parameterization: Theory and practice. In *ACM Siggraph 2007 Course*, 2007.

[23] C. C. L. Wang, K.-C. Hui, and K.-M. Tong. Volume parameterization for design automation of customized free-form products. *Aut. Sci. and Eng., IEEE Trans. on*, 4(1):11 –21, jan. 2007.

[24] S. Han, J. Xia, and Y. He. Hexahedral shell mesh construction via volumetric polycube map. In *Proc. ACM Symposium on Solid and Physical Modeling*, pages 127–136, 2010.

[25] J. Xia, Y. He, X. Yin, S. Han, and X. Gu. Direct-product volumetric parameterization of handlebodies via harmonic fields. *Shape Modeling and Applications, International Conference on*, pages 3–12, 2010.

[26] A.A. Joshi, D.W. Shattuck, P.M. Thompson, and R.M. Leahy. Surface-constrained volumetric brain registration using harmonic mappings. *Medical Imaging, IEEE Transactions on*, 26(12):1657 –1669, dec. 2007.

[27] Y. Wang, X. Gu, T. F. Chan, P. M. Thompson, and S. T. Yau. Volumetric harmonic brain mapping. In *IEEE International Symp. on Biomedical Imaging: Macro to Nano.*, pages 1275–1278, 2004.

[28] X. Li, X. Guo, H. Wang, Y. He, X. Gu, and H. Qin. Harmonic volumetric mapping for solid modeling applications. In *Proc. ACM symp. on Solid and physical modeling*, pages 109–120, 2007.

[29] T. Martin, E. Cohen, and R.M. Kirby. Volumetric parameterization and trivariate b-spline fitting using harmonic functions. In *Proc. ACM Solid and Physical Modeling*, pages 269–280, 2008.

[30] T. Martin and E. Cohen. Volumetric parameterization of complex objects by respecting multiple materials. *Comput. Graph.*, 34(3):187–197, 2010.

[31] B. Li, X. Li, K. Wang, and H. Qin. Generalized polycube trivariate splines. *Shape Modeling and Applications, International Conference on*, 0:261–265, 2010.

[32] T. Ju, S. Schaefer, and J. D. Warren. Mean value coordinates for closed triangular meshes. *SIGGRAPH*, 24(3):561–566, 2005.

[33] M. S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, 20(1):19–27, 2003.

[34] P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki. Harmonic coordinates for character articulation. In *SIGGRAPH '07*, pages 71–81, 2007.

[35] Y. Lipman, D. Levin, and D. Cohen-Or. Green coordinates. *ACM Trans. Graph.*, 27(3):1–10, 2008.

[36] G. Patanè, M. Spagnuolo, and B. Falcidieno. Topology- and error-driven extension of scalar functions from surfaces to volumes. *ACM Trans. Graph.*, 29:4:1–4:20, 2009.

[37] J. Monterde and H. Ugail. A general 4th-order pde method to generate bézier surfaces from the boundary. *Comput. Aided Geom. Des.*, 23(2):208–225, 2006.

[38] A. Jacobson, E. Tosun, O. Sorkine, and D. Zorin. Mixed finite elements for variational surface modeling. *Comp. Graph. Forum (proc. of Euro./ACM SIGGRAPH Symp. on Geom. Proc.)*, 29(5):1565–1574, 2010.

[39] L. Kobbelt, S. Campagna, J. Vorsatz, and H. Seidel. Interactive multi-resolution modeling on arbitrary meshes. In *SIG.'98: Proc. of the 25th conference on Comp. graph. and interactive techn.*, pages 105–114, New York, USA, 1998.

[40] B. T. Helenbrook. Mesh deformation using the biharmonic operator. *Int. J. for Numerical. Methods in Engineering.*, 56(7):1–11, 2003.

[41] Alec J., Ilya B., Jovan P., and Olga S. Bounded biharmonic weights for real-time deformation. *ACM Trans. on Grap.*, 30(4):to appear, 2011.

[42] Y. Lipman, R. Rustamov, and T. Funkhouser. Biharmonic distance. *ACM Trans. Graph.*, 29(3):1–11, 2010.

[43] R. Tankelevich, G. Fairweather, and A. Karageorghis. Three-dimensional image reconstruction using the pf/mfs technique. *Engineering Analysis with Boundary Elements*, 33(12):1403 – 1410, 2009.

[44] P.K. Banerjee. *The Boundary Element Methods in Engineering.* McGraw-Hill, New York, 1994.

[45] D. L. James and D. K. Pai. Artdefo: accurate real time deformable objects. In *Proc. SIGGRAPH '99*, pages 65–72, 1999.

[46] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139:3–47, 1996.

[47] X. Guo, X. Li, Y. Bao, X. Gu, and H. Qin. Meshless thin-shell simulation based on global conformal parameterization. *IEEE Transactions on Visualization and Computer Graphics*, 12(3):375–385, 2006.

[48] G. Fairweather and A. Karageorghis. The method of fundamental solution for elliptic boundary value problems. *Advances in Computational Mathematics*, 9(1-2):69–95, September 1998.

[49] Y. Long, J. A. Fessler, and J. M. Balter. Accuracy estimation for projection-to-volume targeting during rotational therapy: A feasibility study. *Medical Physics*, 37(6):2480–2490, 2010.

[50] R. S. Brock, A. Docef, and M. J. Murphy. Reconstruction of a cone-beam ct image via forward iterative projection matching. *Medical Physics*, 37(12):6212–6220, 2010.

[51] R. Li, X. Jia, J.H. Lewis, X. Gu, M. Folkerts, C. Men, and S.B. Jiang. Real-time volumetric image reconstruction and 3d tumor localization based on a single x-ray projection image for lung cancer radiotherapy. *Medical Physics*, 37(6):2822–2826, 2010.

[52] C.T. Metz, S. Klein, M. Schaap, T. van Walsum, and W.J. Niessen. Nonrigid registration of dynamic medical imaging data using nd + t b-splines and a groupwise optimization approach. *Med.Img.Analy.*, 15:238 – 249, 2011.

[53] K.K. Bhatia, J.V. Hajnal, B.K. Puri, A.D. Edwards, and D. Rueckert. Consistent groupwise non-rigid registration for atlas construction. In *Intl. Symp. on Biomedical Imaging.*, volume 1, pages 908 – 911, 2004.

[54] G. Wu, Q. Wang, J. Lian, and D. Shen. Estimating the 4d respiratory lung motion by spatiotemporal registration and building super-resolution image. MICCAI'11.

[55] J.-M. Peyrat, H. Delingette, M. Sermesant, Chenyang Xu, and N. Ayache. Registration of 4d cardiac ct sequences under trajectory constraints with multichannel diffeomorphic demons. *IEEE Trans.Med.Img.*, 29:1351–1368, 2010.

[56] M. Ben-Chen, O. Weber, and C. Gotsman. Variational harmonic maps for space deformation. *ACM Trans. Graph.*, 28(3):1–11, 2009.

[57] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha. Fast distance queries with rectangular swept sphere volumes. In *Proc. IEEE International Conference on Robotics and Automation*, pages 3719–3726, 2000.

[58] J. Bloomenthal. An implicit surface polygonizer. *Graphics Gems IV*, pages 324–349, 1994.

[59] P. A. Ramachandran. Method of fundamental solutions: singular value decomposition analysis. *Comunications in Numerical Methods in Engineering*, 18(11):789–801, 2002.

[60] R. Mathon and R. L. Johnston. The approximate solution of elliptic boundary-value problems by fundamental solutions. *SIAM J. Numer. Anal.*, 14(4):638–650, 1977.

[61] M. Katsurada and H. Okamoto. The collocation points of the fundamental solution method for the potential problem. *Computers and Mathematics with Applications*, 31:123–137(15), January 1996.

[62] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Matching 3D models with shape distributions. In *SMI '01: Proc. International Conference on Shape Modeling & Applications*, pages 154–166, 2001.

[63] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[64] R. Zayer, C. Rössl, Z. Karni, and H.-P. Seidel. Harmonic guidance for surface deformation. *Computer Graphics Forum*, 24(3):601–609, 2005.

[65] T. Dey and J. Sun. Defining and computing curve-skeletons with medial geodesic function. In *Proc. Eurographics Symp. on Geometry Processing (SGP)*, pages 143–152, 2006.

[66] V. Pascucci, G. Scorzelli, P.-T. Bremer, and A. Mascarenhas. Robust online computation of reeb graphs: simplicity and speed. *ACM Trans. Graph.*, 26(3):58, 2007.

[67] O. Au, C.-L. Tai, H.-K. Chu, D. Cohen-Or, and T.-Y. Lee. Skeleton extraction by mesh contraction. *ACM Transactions on Graphics*, 27(3), 2008.

[68] Y. He, X. Xiao, and H.-S. Seah. Harmonic 1-form based skeleton extraction from examples. *Graph. Models*, 71(2):49–62, 2009.

[69] N. D. Cornea, D. Silver, and P. Min. Curve-skeleton properties, applications, and algorithms. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):530–548, 2007.

[70] P. Alliez, É. C. deVerdière, O. Devillers, and M. Isenburg. Isotropic surface remeshing. In *Proceedings of Shape Modeling International*, pages 49–58, 2003.

[71] J. Teran, N. Molino, R. Fedkiw, and R. Bridson. Adaptive physics based tetrahedral mesh generation using level sets. *Engineering with Computers*, 21(1):2–18, 2005.

[72] M. Tarini, K. Hormann, P. Cignoni, and C. Montani. Polycube-maps. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 853–860, New York, NY, USA, 2004. ACM.

[73] H. Wang, Y. He, X. Li, X. Gu, and H. Qin. Polycube splines. *Computer Aided Design*, 40(6):721–733, 2008.

[74] H. Wang, M. Jin, Y. He, X. Gu, and H. Qin. User-controllable polycube map for manifold spline construction. In *Proc. ACM Symp. on Solid and Physical Modeling*, pages 397–404, 2008.

[75] J. Lin, X. Jin, Z. Fan, and C. C. L. Wang. Automatic polycube-maps. In *GMP*, pages 3–16, 2008.

[76] Y. He, H. Wang, C.-W. Fu, and H. Qin. A divide-and-conquer approach for automatic polycube map construction. *Comput. Graph.*, 33(3):369–380, 2009.

[77] A. Karageorghis and G. Fairweather. The method of fundamental solutions for the numerical solution of the biharmonic equation. *Journal of Comp. Physics*, 69(2):434 – 459, 1987.

[78] B. Jin. A meshless method for the laplace and biharmonic equations subjected to noisy boundary data. *Computer Modeling in Engineering and Sciences*, 6(3):253 – 262, 2004.

[79] T. KwoK, Y. Zhang, and C. Wang. Efficient optimization of common base domains for cross-parameterization. *Visualization and Computer Graphics, IEEE Transactions on*, PP(99):1, 2011.

[80] A. Agathos, I. Pratikakis, S. Perantonis, N. Sapidis, and P. Azariadis. 3D mesh segmentation methodologies for CAD applications. *Computer-Aided Design and Applications*, 4(1-6):827–841, 2007.

[81] A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008.

[82] H. Wang, Y. He, X. Li, X. Gu, and H. Qin. Polycube splines. In *SPM '07: Proc. of the 2007 ACM symposium on Solid and physical modeling*, pages 241–251, 2007.

[83] W. Yu and X. Li. Computing 3d shape guarding and star decomposition. *Computer Graphics Forum*, 30(7):2087–2096, 2011.

[84] L. Garcia-puente and F. Sottile. Linear precision for parameteric patches. *Adv. Comput. Math*, 33:191–214, 2010.

[85] P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki. Harmonic coordinates for character articulation. *ACM Trans. Graph.*, 26, July 2007.

[86] L.C. Evans. *Partial differential equations*. Providence, Rl: American Mathematical Society, 1998.

[87] L. A. Freitag and P. M. Knupp. Tetrahedral element shape optimization via the jacobian determinant and condition number. In *Proc. 8th International Meshing Roundtable*, pages 247–258, 1999.

[88] J. Wang, F. Wang, C. Zhang, H.C. Shen, and L. Quan. Linear neighborhood propagation and its applications. *Pattern Analy. and Mach. Intel., IEEE Trans. on*, 31(9):1600 –1615, 2009.

[89] D. Watkins. *Fundamentals of Matrix Computations*. Wiley Interscience, 2002.

[90] M. Tarini, K. Hormann, P. Cignoni, and C. Montani. Polycube-maps. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH, pages 853–860, 2004.

[91] P. Pebay, D. Thompson, J. Shepherd, P. Knupp, C. Lisle, V. Magnotta, and N. Grosland. New applications of the verdict library for standardized mesh verification pre, post, and end-to-end processing. *International Meshing Roundtable*, pages 535 – 552, 10 2007.

[92] J. Gregson, A. Sheffer, and E. Zhang. All-hex mesh generation via volumetric polycube deformation. *Computer Graphics Forum*, 30(5):1407–1416, 2011.

[93] S. S. Iyengar, X. Li, H. Xu, S. Mukhopadhyay, N. Balakrishnan, A. Sawant, and P. Iyengar. Toward more precise radiotherapy treatment of lung tumors. *Computer*, 45:59–65, 2012.

[94] H. Xu and X. Li. A symmetric 4d registration algorithm for respiratory motion modeling. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2013.

[95] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/kanade meets horn/schunck: combining local and global optic flow methods. *Int. J. Comput. Vis.*, 61(3):211–231, Feb. 2005.

[96] B.C. Vemuri, J. Ye, Y. Chen, and C.M. Leonard. A level-set based approach to image registration. In *Mathematical Methods in Biomedical Image Analysis, Proceedings. IEEE Workshop on*, pages 86 –93, 2000.

[97] V. Gorbunova, J. Sporring, P. Lo, M. Loeve, H. Tiddens, M. Nielsen, A. Dirksen, and M. Bruijne. Mass preserving image registration for lung ct. *Medical Image Analysis*, 16(4):786 – 795, 2012.

[98] M. Heinrich, M. Jenkinson, S. Brady, and J. Schnabel. Mrf-based deformable registration and ventilation estimation of lung ct. *Med. Imag., IEEE Trans. on*, 2013.

[99] Z. Xue, K. Wong, and S. Wong. Joint registration and segmentation of serial lung ct images for image-guided lung cancer diagnosis and therapy. *Computerized Medical Imaging and Graphics*, 34(1):55 – 60, 2010.

[100] G. Wu, Q. Wang, H. Jia, and D. Shen. Feature-based groupwise registration by hierarchical anatomical correspondence detection. *Human Brain Mapping*, 33(2):253–271, 2012.

[101] H. Xu and X. Li. Consistent feature-aligned 4d image registration for respiratory motion modeling. In *International Symposium on Biomedical Imaging*, pages 580–583, 2013.

[102] P. Scov., S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. MM'07.

[103] W. Cheung and G. Hamarneh. N-sift: N-dimensional scale invariant feature transform for matching medical images. In *ISBI 2007*, pages 720 –723, 2007.

[104] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.

[105] H. Xu, P. Chen, W. Yu, A. Sawasnt, S. Iyengar, and X. Li. Feature-aligned 4D Spatiotemporal Image Registration. In *Intern. Conf. on Pattern Recognition*, pages 2639–2642, 2012.

[106] G. Christensen and H. Johnson. Consistent image registration. *Medical Imaging, IEEE Transaction on*, 20(7):568 –582, 2001.

[107] R. Castillo, E. Castillo, R. Guerra, V. E. Johnson, T. McPhail, A.K. Garg, and T.Guerrero. A framework for evaluation of deformable image registration spatial accuracy using large landmark point sets. *Phys Med Biol*, 54:1849–1870, 2009.

[108] P. Grisvard. *Elliptic Problems in Nonsmooth Domains*. Pitman, Boston, 1985.

[109] G. Geymonat. Trace theorems for Sobolev spaces on Lipschitz domains. necessary conditions. *Ann. Math. Blaise Pascal*, 14(2):187–197, 2007.

Appendix A: Existence of Biharmonic Functions on a Solid Domain

This appendix is to show that the biharmonic problem in eq (1) is well-posed in the framework of Sobolev spaces (cf. [108] and references therein).

Let $\Omega \subset \mathbb{R}^3$ be a bounded smooth domain or a polyhedron , $f \in H^{7/2}(\partial\Omega)$ and $g \in H^{5/2}(\partial\Omega)$ where $H^{7/2}(\partial\Omega)$ and $H^{5/2}(\partial\Omega)$ are fractional Sobolev spaces as defined in [108]. In practice, we only require a finite sampling of the functions f and g. Consider the following problem

$$\nabla^4 \Phi = 0 \quad \text{in } \Omega, \tag{7.1a}$$

$$\Phi = f \quad \text{on } \partial\Omega, \tag{7.1b}$$

$$\frac{\partial \Phi}{\partial n} = g \quad \text{on } \partial\Omega, \tag{7.1c}$$

where n denotes the outer normal of Ω and the boundary conditions (7.1b) and (7.1c) are understood in trace sense. By trace theorem[1], there exist $\zeta \in H^4(\Omega)$ such that $\zeta = f$ on $\partial\Omega$ and $\partial\zeta/\partial n = g$ on $\partial\Omega$. Moreover,

$$\|\zeta\|_{H4(\Omega)} \leq C(\|f\|_{H^{7/2}(\partial\Omega)} + \|g\|_{H^{5/2}(\partial\Omega)}) \tag{7.2}$$

where C is a constant depending on Ω. Then Φ solves (7.1a)-(7.1c) if and only if $\psi = \Phi - \zeta \in H^2(\Omega)$ solves the following biharmonic equation with homogeneous Dirichlet boundary condition

$$\nabla^4 \psi = -\nabla^4 \zeta \quad \text{in } \Omega \tag{7.3a}$$

$$\psi = 0 \quad \text{on } \partial\Omega, \tag{7.3b}$$

$$\frac{\partial \psi}{\partial n} = 0 \quad \text{on } \partial\Omega, \tag{7.3c}$$

Following the classical variational approach (cf. Chapter 1 & Chapter 7 in [108]), the problem (7.3a)-(7.3c) is uniquely solvable in $H^2(\Omega)$. Moreover,

$$\|\psi\|_{H2(\Omega)} \leq C(\|\nabla^4 \zeta\|_{L_2(\Omega)}.) \tag{7.4}$$

where C is a constant depending only on Ω.

Therefore, the problem (7.1a)-(7.1c) has a unique solution in $H^2(\Omega)$. Moreover, it follows from (7.2) and (7.4) that the solution depends continuously on the data f and g, i.e.,

$$\|\Phi\|_{H^2(\Omega)} \leq C(\|f\|_{H^{7/2}(\partial\Omega)} + \|g\|_{H^{5/2}(\partial\Omega)})$$

[1] Trace theorem is standard for smooth domains but becomes complicated for nonsmooth domains (Cf.[109]).

Appendix B: Harmonic-based boundary condition

This appendix is to illustrate the way to set g according to the harmonic volumetric mapping results.

Assume we have two adjacent regions Ω_1 and Ω_2, and their interface boundary is $\partial\Omega_{12}$ (see Fig.7.1). First, we compute two harmonic volumetric mapping Φ_1^h, Φ_2^h for each region according to their surface mapping. Then we compute their biharmonic volumetric mapping Φ_1, Φ_2. We set g based on the harmonic results:

$$\frac{\partial\Phi_1(x)}{\partial n_1(x)} = \begin{cases} \frac{\partial\Phi_1^h(x)}{\partial n_1(x)}, & \text{if } x \in \partial\Omega_1 \ \ \& \ \ x \notin \partial\Omega_{12}, \\ \frac{1}{2}(\frac{\partial\Phi_1^h(x)}{\partial n_1(x)} + \frac{\partial\Phi_2^h(x)}{\partial n_1(x)}), & \text{if } x \in \partial\Omega_{12} \end{cases}$$

$$\frac{\partial\Phi_2(x)}{\partial n_2(x)} = \begin{cases} \frac{\partial\Phi_2^h(x)}{\partial n_2(x)}, & \text{if } x \in \partial\Omega_2 \ \ \& \ \ x \notin \partial\Omega_{12}, \\ -\frac{1}{2}(\frac{\partial\Phi_1^h(x)}{\partial n_1(x)} + \frac{\partial\Phi_2^h(x)}{\partial n_1(x)}), & \text{if } x \in \partial\Omega_{12} \end{cases}$$

If we compute the biharmonic map on just one region Ω, the resultant biharmonic mapping will be exactly the harmonic mapping. If a model decomposed into multiple regions $\{\Omega_i\}$, for the point along the boundary interface $x \in \partial\Omega_{12}$, we set g to be the average value of the normal derivative of that two harmonic mappings. Since $n_1 = -n_2$, then we have $\frac{\partial\Phi_1(x)}{\partial n_1(x)} = \frac{\partial\Phi_2(x)}{\partial n_1(x)}, \forall x \in \partial\Omega_{12}$. So this boundary condition has C^1 continuity along the boundary interface. It is easy to verify that it also keeps linear precision property: Φ_1^h, Φ_2^h satisfy linear precision property, then their linear combination will also satisfies linear precision property.

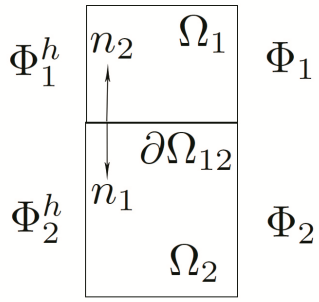


FIGURE 7.1. The decomposed domain illustrating the harmonic-based boundary condition.

Vita

Huanhuan Xu was born in Jiangxi, China, in 1984. She graduated from Centra China Normal University, in 2006 with double bachelors' degree in the Computer Science Department and the Mathematics Department with Honors. She earned a master degree in the Automation Department from University of Science and Technology of China in 2009. She has been enrolled in the doctoral program in the Department of Electrical and Computer Engineering since 2009 with a second master degree in Mathematics.

She was offered a graduate research assistantship from Center for Computation and Technology and a Mark and Carolyn Guidry doctoral fellowship from Department of Electrical and Computer Engineering in 2009-2013. Her research interests include Medical Imaging Processing and Analysis, Computer Graphics and Geometric Modeling.