2012

Non-rigid registration of 2-D/3-D dynamic data with feature alignment

Huanhuan Xu
Louisiana State University and Agricultural and Mechanical College, hxu4@lsu.edu

NON-RIGID REGISTRATION OF 2-D/3-D DYNAMIC DATA WITH FEATURE ALIGNMENT

A Thesis

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
Master of Science in Electrical Engineering

in

The Division of Electrical & Computer Engineering

by
Huanhuan Xu
Double B.S. in Central China Normal University, 2006
M.S., University of Science and Technology of China, 2009
December, 2012

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, and have learned a lot from him. I appreciate his dedication and patients.

Secondly, I sincerely thank Professor Hongchao Zhang, Professor Lu Peng for being a part of my committee.

Thirdly, much help from my labmates, Shenghua Wan, Wuyi Yu, Kang Zhang, Qing Huang is also appreciated. And thanks to Dr. Shiyuan Gu who just graduated in the Mathematics Department of LSU for the discussion and suggestions.

Finally, I would like to thank my family for their love, support, and encouragement during this journey.

This work was in part sponsored by the Mark and Carolyn Guidry doctoral fellowship from the Division of Electrical and Computer Engineering in the School of Electrical Engineering and Computer Science of LSU.

Table of Contents

List of Figures

Abstract

In this work, we are computing the matching between 2D manifolds and 3D manifolds with temporal constraints. That is, we are computing the matching among a time sequence of 2D/3D manifolds. It is solved by mapping all the manifolds to a common domain, and then builds their matching by composing the forward mapping and the inverse mapping.

At first, we solve the matching problem between 2D manifolds with temporal constraints by using mesh-based registration method. We propose a surface parameterization method to compute the mapping between the 2D manifold and the common 2D planar domain. We can compute the matching among the time sequence of deforming geometry data through this common domain. Compared with previous work, our method is independent of the quality of mesh elements and more efficient for the time sequence data.

Then we develop a global intensity-based registration method to solve the matching problem between 3D manifolds with temporal constraints. Our method is based on a 4D (3D+T) free-from B-spline deformation model which has both spatial and temporal smoothness. Compared with previous 4D image registration techniques, our method avoids some local minimum. Thus it can be solved faster and achieve better accuracy of landmark points prediction.

We demonstrate the efficiency of these works on the real applications. The first one is applied to the dynamic face registering and texture mapping. The second one is applied to lung tumor motion tracking in the medical image analysis.

In our future work, we are developing more efficient mesh-based 4D registration method. It can be applied to tumor motion estimation and tracking, which can be used to calculate the read dose delivered to the lung and surrounding tissues. Thus this can support the online treatment of lung cancer radiotherapy.

Chapter 1
Introduction

1.1 Background

A classical problem in computer vision and computer graphics is aligning two geometry data, taken from different time or different viewpoints. This problem is known as registration, and its objective is to recover the spatial alignment between these two geometry data. Dynamic registration is considering the registration among a time sequence of the data. Time series of data are acquired for various reasons, such as monitoring of bone growth in children (long time interval), monitoring of tumor growth(medium interval), or post-operative monitoring of healing(short interval).

Registration plays an important in data processing ([1]), matching and recognition. Typically, registration is required in remote sensing (multi-spectral classification, environmental monitoring, change detection, image mosaicing, weather forecasting, creating super-resolution images, integrating information into geographic information systems (GIS)), in medicine (combining computer tomography(CT) and MR data to obtain more complete information about the patient, monitoring tumor growth, treatment verification, comparison of the patient's data with anatomical atlases), in cartography (map updating), and in computer vision (target localization, automatic quality control), to name a few.

The literature treating registration methods is very extensive (e.g., [2] for a survey). On one side are mesh-based algorithms, which compute the registration through geometrical surface mapping approaches [3]. The main applications of these methods are facial animation and expression analysis.

On the other side are intensity-based algorithms, which use most of the intensity information in the data set [4]. The main application based on this technique is medical image analysis. For example, in the lung cancer treatment, the intensity-based registration algorithms are used to predict the growth of the tumor and its surrounding organs which can improve the precision of the radiotherapy treatment.

1.2 Main Contributions

First, we formulate the face registration problem as a feature-aligned surface mapping problem. As we known, feature aligned surface mapping can improve the effect of texture mapping by enforcing feature correspondence between 3D meshes and image. We present a novel constrained texture mapping computation framework based on the method of fundamental solution (MFS). We first flatten the surface onto planar domain, then compose a 2D harmonic map with user specified feature aligned with corresponding texels in the texture image. We demonstrate that our framework can efficiently conduct feature-aligned texture mapping with dynamic deforming geometry data (Chapter 3).

Then, we also develop a feature-aware 4D spatiotemporal image registration method. Combined with segmentation method, this method can be used for 3D manifold registration with temporal constraint. Our proposed model is based on a 4D(3D+t) free-form B-spline deformation model which has both spatial and temporal smoothness. We first introduce an automatic 3D feature extraction and matching method based on an improved 3D SIFT descriptor, which is scale- and rotation- invariant. Then we use the results of feature correspondence to guide an intensity-based deformable image registration. Experimental results show that our method can lead to smooth temporal registration with good matching accuracy; therefore this registration model is potentially suitable for dynamic tumor tracking. Thus it can provide more accurate guidance in the planning and delivery of lung tumor radiotherapy treatment (Chapter 4).

1.3 Organization

There are five chapters in this thesis. Chapter 1 is a brief introduction of registration method and its applications. Chapter 2 mainly gives the related work of this thesis. Chapter 3 presents harmonic surface mapping algorithm and its application in the dynamic facial registration. Chapter 4 gives an algorithm of the feature-aligned 4D image registration. Chapter 5 is the summary of the whole thesis.

Chapter 2
Related Work

2.1 Harmonic Maps and Surface Parameterization

Surface mapping computes a one-to-one continuous map between a $2D$-manifold and a target domain with low distortions. It has been playing a critical role in various applications ranging from graphics, CAGD, visualization, vision, medical imaging, to 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]. Most parameterization techniques handle surfaces with disk topology. When surfaces have non-trivial topology, local approaches or global approaches can both be used to generated texture maps. Generally, local methods segment input surfaces into a set of topological disks then apply disk paramterization, while global methods directly compute holomorphics forms ([20],[21]) or flattened metrics ([22], [23]) over the global atlas. Texture mappings using local methods need to carefully handle distortion along cutting boundary while global methods should take care of artifacts on singularity points.

A thorough survey on surface parameterization techniques is beyond the scope of this work, and we refer readers to reports of [24], [25], and [26] 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 [27], meshing [28, 29], shape registration [30, 31, 32], and trivariate spline construction [33, 34, 35]. Wang et al. [31] 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.[29] and Han et al.[28] use this discrete harmonic volumetric map in polycube parameterization. Most closely related to this work, in [32], we compute the harmonic volumetric mapping between two solid objects using fundamental solution methods. Later, we incorporate feature alignment in this volumetric mapping framework [36].

Volumetric interpolation is a powerful tool for shape deformation. Ju et al. [37] generalize the mean value coordinates [38] from surfaces to volumes to get a smooth volumetric interpolation for cage based deformation. Joshi et al. [39] present harmonic coordinates with non-negative weights for volumetric interpolation and deformation in concave regions. Martin et al. [33] parameterize volumetric model with trivial topology to a cylinder using the finite element method, and later generalize the algorithm [34] to more complicated models with medial surfaces. Lipman et al [40] develop Green's coordinates for volumetric deformation. Patanè[41] uses Radial Basis Function to approximate volumetric function along the volume data.

2.3 Dynamic Data Registration

According to the type of processing data, registration method can be classified to mesh-based registration and image-based registration. When the input data consists of a time sequence of the data, the registration can be computed through pairwise optimization or global optimization.

Compared with the image-based registration, mesh-based registration has less computational cost and can be solved faster. We can build more physical meaningful model based on the mesh structure to simulation the real world situation. [42] used conformal maps to analyze similarities of 2D shapes and [43, 44] also computed the 3D face meshes and brain meshes registration by using conformal maps. [45] developed a non-rigid registration algorithm of 3D dynamic facial data by using least-squares conformal maps with additional feature correspondences. However, all previous mesh-based registrations are solved by finite element method which is highly depends on the quality of the mesh elements. It will become very ill-condition when the mesh quality is very bad. In our work, we developed a meshless method to solve the mesh-based registration based on the harmonic mapping, which follows the physical institution and independents on the quality of the mesh element.

On the other hand, image-based method is applied in more general images and uses the full image information to compute the registration. So usually image-based method can get more accurate results than mesh-based method. Currently, spatio-temporal schemes become popular, which consist in a global formulation of the motion estimation problem for temporal image sequences. Rather than computing frame-to-frame matching individually [46, 47, 48], the entire sequence is considered simultaneously, allowing to enforce the temporal coherence of the deformation across the sequence. By making assumptions such as smoothness about the temporal variations of the transformation, these approaches often enable a more compact and restrictive description of the full motion estimation problem. Spatio-temporal deformable registration has received considerable attention in literature, mostly in cardiac image analysis and respiratory-correlated image analysis [49, 50, 51].

However, most of the current spatiotemporal dynamic images are fully guided by the image's intensity which does not consider any feature guided. As we known, feature constraints can guide the optimization and avoid some local minimal. In our work, we developed a feature-aligned 4D spatiotmeporal image registration algorithm to match deforming volume images. It can achieve better result, both in accuracy and efficiency.

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 [52], and was introduced into computer graphics for the simulation of deformable objects in [53]. 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 true meshless method, since only boundary nodes are necessary for all the computation. "Meshless" has the advantage of simpilicity that neither domain nor mesh connectivity is required in storage and computation; so it becomes very attractive in scientific computing and modeling [54],[55]. A comprehensive review of the MFS and kernel functions for solving many elliptic PDE problems was documented in [56].

2.5 Feature-aligned Texture Mapping

Compared with conventional surface parameterization methods, constrained texture mapping considers the additional feature correspondence between 3D model and texture images. Levy [57] proposed a method to satisfy the user-specified constraints in the least-squares sense. And it works well for a small number of constraints but can lead to invalid parameterization while dealing with a large set of constraints ([58]). Desbrun et al.[7] used Lagrange multipliers to incorporate positional constraints into the formulation of parameterization, and Eckstein et al.[59] used Steiner vertices to satisfy the constraints. Matchmaker [60] automatically partitions a mesh into genus-0 pathches, and satisfied the user-specified correspondences between the patches and one or two texture image(s). Cross-parameterization [61]and inter-surface mapping [62] proposed a similar approach for the design of mapping between two surfaces instead of between a surface and texture space, and can also handle feature correspondences. Further Zhou et al. [63] suggested a similar approach between a surface and multiple texture images. Tai et.al [64] presented an approach to reduce texture distortion via texture synthesis to better fit the 3D geometry. FlexiStickers [65] combined the photography effects with the constrained parameterization approach for texture mapping using casual images. [66] used the parameterization of [57] to get the first texture mapping for 3D object, then found 2D-to-2D matching between video images to get a video sequence texture mapping. [67, 68] developed texture mapping methods for supporting point sets of 3D objects.

Chapter 3

Dynamic Harmonic Texture Mapping using Methods of Fundamental Solutions

3.1 Introduction

In real-time computer graphics, texture mapping is an important technique that enhances the visual effects of simple geometric shapes by wrapping image textures to 3D meshes. Many complicated scenes with detailed textures and materials can therefore be rendered efficiently. A key ingredient that dictates its quality is a lowly distorted surface mapping between the surface and the image domain (assigning (u, v) texel coordinates to the mesh vertices). Surface parameterization ([24], [25]) has been studied to generate angle-preserving or area-preserving texture mappings. Surface parameterization methods that do not consider the feature matching (i.e. enforcing the registration of specific feature points or lines) could lead to less desirable results in texture mapping applications. Feature-aligned texture mapping techniques have been developed by employing constrained parameterization if user-defined constraints are given [57],[58],[69], [70], [64], [65], [60]. In these techniques, a distortion metric is usually minimized while feature constraints are being enforced. For example, in the Figure 3.1(c), through a constrained texture mapping, we could get a better visual result with more precise alignment of the eyes, mouth, etc.

When we have the dynamic texture image (e.g. a video) or geometric shape (e.g. a sequence of tracked motion), to get a sequence of texture-mapped meshes, a lot of mappings need to be computed efficiently. For example, [66] used the method of [57] to get the first texture mapping for 3D object, then found 2D-to-2D matching between video images to get a video sequence mapping effect. Most existing work

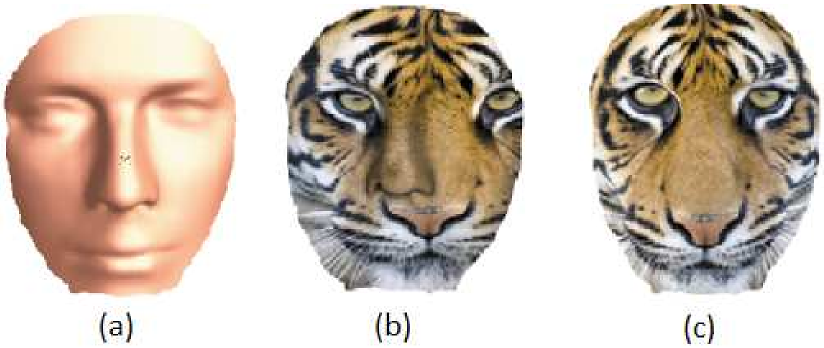

FIGURE 3.1. Texture mapping a male face.

on texture mapping are mesh-based methods, [67], [68] developed texture mapping algorithms on point sets of 3D objects.

Methods of fundamental solutions (MFS) have been used as an effective routine for solving elliptic partial differential equations (PDEs). Unlike the classical boundary integral equation and boundary element method (BIE/BEM), which has been widely used in many engineering applications [52] and computer graphics tasks such as deformable objects simulations [53], the MFS uses only the fundamental solution kernel functions in the construction of the solution of a problem, with no need of integrals over boundary elements. MFS is a meshless method, since only boundary nodes are necessary for all the computation. Dynamic motion data captured by motion scanners usually output the acquired shapes as range data (by low quality triangle meshes) or point clouds. A meshless approach to handle their textures will be desirable.

Contribution and Overview. The main contributions of this paper include

- **An efficient meshless texture mapping framework** has been presented using methods of fundamental solutions. The 2D harmonic mapping with feature alignment has been studied in this framework. The distortion is low and the features are aligned with small fitting errors.

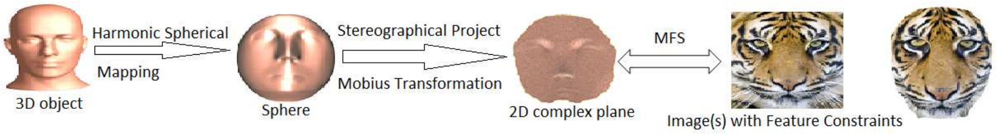

FIGURE 3.2. Processing pipeline of our algorithm

- We use this framework to efficiently compute **dynamic texture mapping**, which can handle the texture mapping from a dynamic sequence of images like a video to a model, and the texture mapping from a static texture to a dynamic morphing meshes. The computation is very efficient, the texture mapping for each frame only take a matrix-vector multiplication to compute.

The rest of this paper is organized as follows. We describe the basic idea of our approach in Section 3.2 and introduce the algorithm and implementation in Section 3.3 and Section 3.4. The experimental results are shown in Section 3.5. Finally, we discuss the limitation and future work of our framework, and conclude the paper in Section 3.6.

3.2 Overview

The input of our algorithm consists of a surface M, a texture source image I_s, and two sets of corresponding user-defined features (constraints) $\{r_i, t_i\}, i = 1, \ldots, n$ where $r_i \in M$ and $t_i \in I_s$. We are seeking a harmonic map $h : M \to T \subset I_s$, s.t. $\Delta h = 0$ and $h(r_i) = t_i$.

Our computational pipeline has two steps, where in the first step, the 3D surface M is parameterized onto the planar domain $D \subset \mathbb{E}^2$ by a harmonic map $g : M \to D$ without enforcing feature alignment, where many robust discrete algorithm for computing harmonic maps have been studied; in the second step, the planar surface D is mapped to a region on the texture space by another harmonic map $f : D \to T \subset I_s$. The final texture mapping h is the composition $h : M \to T = f \circ g$:

Figure 3.2 shows our algorithm pipeline, which will be elaborated in the following sections.

1. $g : M \to D$: Without feature points, parameterize 3D surfaces onto the 2D domain (Section 3.3).
2. $f : D \to T$: With feature alignment, compute the 2D harmonic map using MFS (Section 3.4).

The computation of g is the well studied surface parameterization problem. Without the feature restrictions, parameterization of genus-0 surfaces M can be efficiently solved by linear systems or variational optimizations. However, enforcing feature alignments directly in this step is not straightforward, pinning specific vertices could lead to serious local flip-over. Therefore, we leave the feature alignment to the second step, which is a planar harmonic mapping problem, where we can efficiently compute f using a set of harmonic functions using methods of fundamental solutions, meanwhile aligning feature points.

3.3 Conformal Surface Flattening

In the first step of our computation, we flatten surfaces onto the planar domain. As addressed in Chapter 2, many paramterization techniques have studied the conformal parameterization of topological disks ([71], [38], [72]), topological spheres ([73], [74]), and high-genus surfaces ([20], [21]) represented by triangular meshes. [55] studied the global parameterization of point cloud surfaces. Most of these work focus on reducing the angular distortion of the flattening, so that the geometric shape is best preserved on the domain (infinitely small circles on tangent plane are mapped to infinitely small planar circles instead of ellipses). Parameterizations using other metrics such as distance-preserving or area-preserving are also studied.

For surface texture mapping applications, although has larger angular distortion, such a map prevents large area distortion and uneven sampling of the texture space. However, in the first step of our pipeline, we choose to use the conformal parameterization. We try to preserve the shape as much as possible in the first step, and balance the area-distortion using planar harmonic maps, by following user specified feature-alignment.

In this work we focus on texture mapping genus-0 surfaces, i.e. topological disks and topological spheres, while our proposed framework can be easily extended to general surfaces by applying global conformal parameterization in the first step. We first discuss our algorithm in handling closed genus-zero surfaces, then we explain how it works for open genus-zero surfaces.

We compute the spherical harmonic map g_1 from a closed genus-zero surface M onto a unit sphere, then we use the stereographic projection g_2 to project it onto the complex plane $\mathbb{C}$, with a planar Möbius transformation g_3, we can get the conformal transformation of the interested (for texture mapping) region on the surface to a bounded planar region: $g : S \rightarrow D = g_3 \circ g_2 \circ g_1$.

First, we use the spherical parameterization [75], [76] to map the input surface M onto the unit sphere S^2 harmonicly. This harmonic map $g_1 : M \rightarrow S^2$ is conformal ([75]) and preserves local geometric shapes. The conformal mapping is not unique but forms a Möbius group differ by the Möbius transformation.

Then we project S^2 to the complex plane $\overline{\mathbb{C}}$. Formally, the stereographic projection g_2 from the north pole to the complex plane (virtually locates on the south pole) is defined by

$$g_2(P) = (\frac{x}{1-z}, \frac{y}{1-z}), P = (x, y, z) \in S^2. \tag{3.1}$$

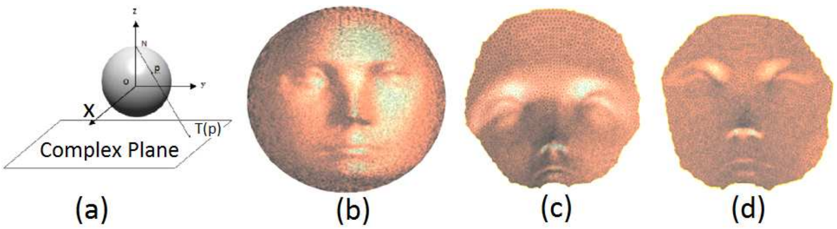

FIGURE 3.3. Stereographic Projection.

In Figure 3.3, (a) illustrates such a stereographic projection, and the result of the stereographic projection from unit sphere (b) to the plane is shown in (c).

Definition 1 (Möbius Transformation)**.** *The mapping* $g_2 : \mathbb{C} \to \mathbb{C}$ *is a Möbius transformation if*

$$g_3(z) = \frac{az+b}{cz+d}, z, a, b, c, d \in \mathbb{C}, ad - bc = 1.0 \tag{3.2}$$

A Möbius transformation has six degrees of freedom, thus we sample a triplet of three random points from the 3D mesh, which uniquely defines one Möbius transformation. In practice we use the following three constrains on the complex plane: 0, 1, and ∞, and requiring three sample vertices on M to be mapped to them correspondingly. Figure 3.3(d) shows the mesh after the Möbius transformation.

It is well know that the stereographic projection g_2 and the Möbius transformation g_3 are also conformal mappings from sphere S^2 to $\overline{\mathbb{C}}$, and $\overline{\mathbb{C}}$ to $\overline{\mathbb{C}}$, respectively. Therefore, together with the conformal spherical map g_1, we get $g = g_3 \circ g_2 \circ g_1$ is a conformal (and harmonic) mapping.

3.4 Harmonic Planer Mapping using MFS

3.4.1 Planar Harmonic Mapping

We want to compute a harmonic vector function $f : D(\subset \mathbb{C}) \to T(\subset I_s)$. This is equivalent to computing two harmonic scalar fields f^0, f^1. In each direction, $f^i(z)$ maps the 2D point z to a corresponding row or column in the texture space I_s. If we extract the boundaries of D and T which are two simple closed loops, denote them as ∂D and ∂T, and assign a boundary loop mapping $f' : \partial D \to \partial T$; then the computation of f is reduced to

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

where the operator Δ is the $2D$-Laplacian operator

$$\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2},$$

and $\Delta f = 0$ is equivalent to both harmonic scalar functions $\Delta f^0 = 0$ and $\Delta f^1 = 0$.

3.4.2 Planar Harmonic Mapping using MFS

We denote the fundamental solutions of the operator Δ:

$$K(p, q) = -\frac{1}{2\pi} \log |p - q|, \tag{3.3}$$

where p and q denote points in the plane $\mathbb{R}^2$.

The method of fundamental solutions is a meshless algorithm that approximates the boundary data. It is described as follows:

A.1 Place N_s points $q_1, q_2, ..., q_{N_s}$ in the exterior of D;

A.2 The approximate solution is sought by:

$$f_{N_s}(\mathbf{w}, \mathbf{Q}; p) = \sum_{n=1}^{N_s} \omega_n K(p, Q_n), p \in D, \tag{3.4}$$

where $\{\omega_n\}$ are constants to be determined;

A.3 Determine $\{\omega_n\}$ by requiring that f_{N_s} satisfies the boundary condition approximately.

f_{N_s} satisfies Laplace's equation exactly. Therefore, here for both f^0 and f^1, we compute the corresponding $\{\omega_n^0\}$ and $\{\omega_n^1\}$ in order to enforce the boundary condition of $f_{N_s}^0$ and $f_{N_s}^1$.

The boundary fitting has two steps:

B.1 Select N_c points $p_1, p_2, \ldots, p_{N_c}$ from boundary ∂D;

B.2 Solve $\{w_n^i\}$ for $f^i, (i = 0, 1)$ by the following N_c equations:

$$f_{N_s}^i(\omega_n^i, \mathbf{Q}; p_m) = f'^i(p_m), m = 1, 2, \ldots, N_c, \tag{3.5}$$

or equivalently (following Eq.(3.3))

$$\sum_{n=1}^{N_s} \omega_n^i log|p_m - q_n| = -2\pi f'^i(p_m), m = 1, ..., N_c, \tag{3.6}$$

We explain more details about their implementation in the following subsections.

3.4.3 Boundary Fitting

In our framework, the user-defined constraint points $R = \{r_i\}$ include several boundary points $R_B = \{r_j\}$ and some interior feature constrains $R_I = \{r_k\}$. After the flattening, these points in R are mapped onto the plane by conformal mapping g. Then the boundary fitting processing is performed as follows. We use the Dijkstra algorithm to trace the shortest boundary paths connecting these planar boundary points in $g(R_B)$, the entire path forms a closed loop $\partial D \subset \mathbb{R}^2$. We linearly interpolate the mapping of each point p on ∂D (e.g. if p is on the path between the interval from $g(r_i)$ to $g(r_j)$, the image of p is the linear interpolation of t_i and t_j using the arc-length ratio of p on the path $[g(t_i), g(t_j)]$). All these boundary points plus the feature constraints are called constraint points or collocation points.

The source points $q_n \in \mathbb{R}^2, n = 1, \ldots, n_s$ should lie outside of D. In other words, they locate on the boundary $\partial\widetilde{D}$ of a region $\widetilde{D}$ containing D (i.e.$D \subset \widetilde{D} \subset \mathbb{E}^2$).

Following [77], we choose a big disk as $\widetilde{D}$, and sample source points on its boundary circle $\partial\widetilde{D}$.

Following Eq.3.5, we can assemble the coefficient matrix $\mathbf{A}$ and reduce the boundary fitting process to a linear system: $\mathbf{Aw^i} = \mathbf{t^i}$. We solve two linear systems, and compose the final harmonic map using the harmonic functions determined (Eq.3.4) by source points and the two resultant vectors ω^0, ω^1.

Solving Linear System using SVD. When we have N_c collocation points and N_s source points, then we need to solve the $A\omega^i = b^i$ system where the coefficient matrix is with the dimension of $n_c \times n_s$. This is a dense matrix since any source point will have contribution to every constraint point. Furthermore, $\mathbf{A}$ is ill-conditioned, and to solve such MFS linear system. As suggested in [78], we use Singular Value Decomposition (SVD) to decompose $\mathbf{A}$ because it (1) generates accurate and stable results when the coefficient matrix is highly ill-condition, and (2) flexibly gets to the least-square solution for over constrained boundary conditions (which always happen in our MFS solving for feature alignment). 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 approximation, and therefore we remove them in the source point set for MFS evaluations.

3.4.4 Feature Alignment

Feature points can be aligned in the least square sense together with boundary fitting process, so that they are treated as soft constraints. Since all points traced on the boundary are parts of the fitting, compared to feature points provided by the user, they have a much larger number. In order to prevent boundary points from totally dominating the boundary fitting, we shall assign each sample feature point an extra integer weight α. Then in matrix $\mathbf{A}$, we add in α rows of corresponding feature constraints. This effectively leads to precise feature alignment.

3.4.5 Source and Collocation Points Placement

The location of source and collocation points is an important issue since it dictates the numerical stability of the fitting. The coefficient matrix is determined by p_i and q_j, which can be ill-conditioned[79]. In general, the condition number of the matrix increases as the distance from $\widetilde{\partial D}$ to ∂D increases, though the accuracy of the MFS approximation increases under this situation [80].

Extensive research has been conducted from computational mathematic points of view, in order to suggest their placement. Usually, the assumptions such as that ∂D is an analytic Jordan curve, and that the given boundary condition is analytic on ∂D, etc. are used. Numerical convergence discussions are then based on these. Generally, current literatures suggests two strategies for source point locations. (1) To place source points on a circle within three times the diameter of ∂D, and (2) To sample them on an offset surface of ∂D. As shown in [10], source point locations desirable for analytic cases might work perfectly for discrete surfaces. We conducted experiments by evaluating fitting error under different shapes of $\widetilde{D}$ and different distances from ∂D. The boundary constrain error is defined as: $\sum_p \|f'(p) - f_{N_s}(p)\|^2$ for all collocation points p. Figure 3.4illustrates different source point placement in our experiments. We scale each D to a unit box to get a normalized boundary constrain error. In Figure 3.4(a), the source points are placed on a circle with the radius equals to $1.02 * R$ where R is the radius of D; in this case, the boundary constraint error is 1.145823. In (b) and (c), we place the source points in an offset boundary $\widetilde{\partial D}$. d means the distance from $\widetilde{\partial D}$ to ∂D. (b) shows the source points are sampled on an offset surface while $d = 0.02$. (c) shows source

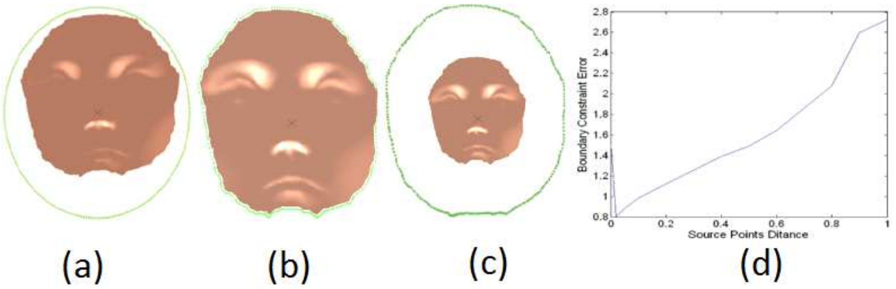

FIGURE 3.4. Source Points Placement.

are sampled on a bounded circle while $d = 0.8$. (d) plots the boundary constraint error with respect to d when source points are placed on offset boundaries. The x-axis indicates the d value while the y-axis show the corresponding constraint error.

Our experimental results indicates that: (1) f gets smaller constrain error when source points are closer to ∂D; (2) source points in an offset boundary leads to better fitting than a circle; (3) when $d < 0.02$, the approximation for the fundamental solution becomes unstable, and the boundary constrain error increases drastically. In all our experiments, we take $d = 0.02$.

3.4.6 Algorithm Pipeline

Algorithm discussed in this section can be formulated as follows. Suppose r_i and t_i are the user specified feature constraints. After the conformal flattening, we use the $g(r_i)$ to denote the feature r_i on the plane.

- *Input:* $D, I_s, \{g(r_i) \in D\}$ and $\{t_i \in I_s\} i = 1, ..., n$.

- *Output:* f^i for $i = 0, 1$.

1 Use Dijkstra algorithm to trace the boundary on M, and interpolate the boundary mapping of h from the loop on M to a loop in I_s. With g, this induces the boundary condition $f'(\partial D)$.

2 Place the source points and set the collocation points (including boundary constraints and feature constraints).

3 Compute the coefficient matrix $\mathbf{A}$. Its element $\mathbf{A}_{uv}$ takes the value of the kernel function K on the collocation point p_u and the source point q_v.

4 Decompose the coefficient matrix using Singular Value Decomposition.

5 Solve this linear system of step 3, and get the harmonic map $f = (f^0, f^1)$.

3.5 Experimental Results

Based on algorithm described above, we have implemented an interactive system for dynamic harmonic texture mapping. Our experiments are conducted on a PC with Pen-tium IV 2.66GHz and 2.00GB RAM. In Figure 3.5, a face model (a) is mapped to a male image (b). The green points on the 3D mesh and the yellow dots in the image denote the feature constrains. The mapping result is illustrated in (c). (d) shows the checker board texture mapping, indicating the low distortion of our method. Figure 3.6 shows another mapping example from the Igea model (a) to a female image (b). The texture mapping of the face and checkerboard are illustrated in (c) and (d).

In Figure 3.7, our texture mapping effect from the cow model (a) to a tiger image (b) is compared with [57]. With less angular distortion under feature-alignment, our harmonic texture mapping (c) has better visual effect than [57] in (d).

Table 3.1 shows the computation time for static texture mappings. An advantage of our framework is discussed in the following section: once the mapping between meshes and the images has been computed once, it can be efficiently applied for dynamic data.

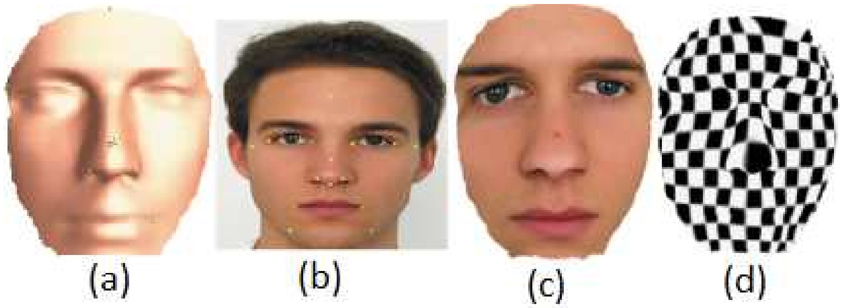

FIGURE 3.5. Texture mapping result for the male face model

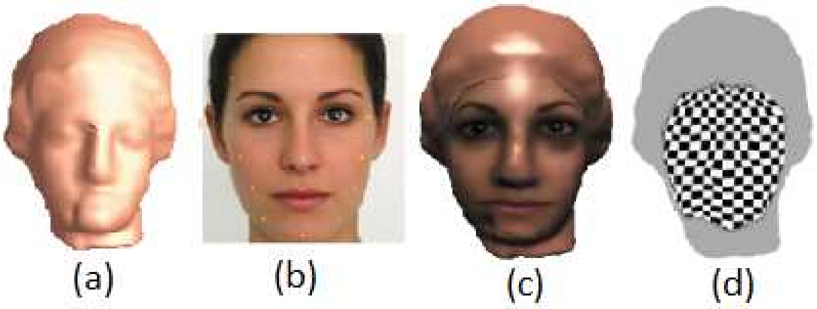

FIGURE 3.6. Texture maaping result for the Igea model

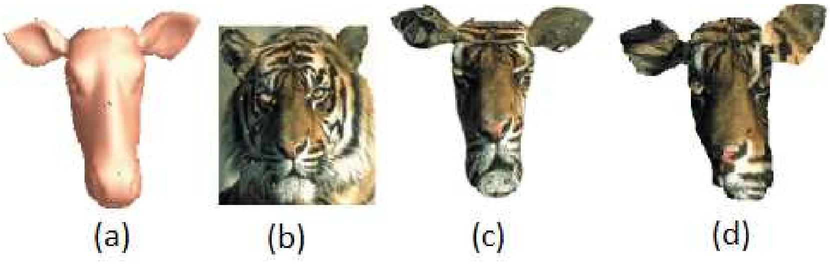

FIGURE 3.7. Texture mapping result for the cow model

3.5.1 Sequential Maps: Handling Dynamic Data

Texture mapping from a sequential images(videos) onto a 3D surface can be computed efficiently by only one SVD decomposition. The recomputation of planar harmonic maps under different boundary conditions only takes a few multiplications. Once the SVD for the coefficient matrix $\mathbf{A}$ has been computed ($\mathbf{A} = \mathbf{UWV^T}$), under a new boundary condition $\mathbf{b}'$, the MFS weights ω_i can be immediately computed by the product of $VW^{-1}U^T\mathbf{b}'$.

We apply this framework to the face expression database (JAFFE database, http://kasrl.org/jaffe.html) for demonstrating our dynamic texture mapping. Figure 3.8(b) shows the samples of the dataset. In this case, we use OPENCV2.0 to detect face in the image which shows in green rectangle. Figure 3.8(a) shows the 3D object flatten in the 2D plane. And we detect the boundary constraint points automatically which show the outside green rectangle. Then we use this two corresponding rectangle as our boundary constraint to solve our texture mapping automatically. The important point is that we use SVD to solve $\mathbf{AX} = \mathbf{B}$ problem. When we change different texture images, we just need one decomposition of $\mathbf{A}$ to get all results which is very efficient. Figure 3.9(a) shows the feature points in 3D object; (b) shows the feature points in the image by manually setting; c shows our mapping result of Jaffe Face Database.

Second, we apply our method for one image to N 3D objects. In this case, we place the source points in the image. Figure 3.10(a) shows the source points and feature points in the image. Figure 3.10(b) shows the samples of our 3D face database. Figure 3.11 shows our mapping results with one time decomposition.

If both the image and mesh are dynamic, we can still efficiently apply our framework by computing two planar maps f_1 and f_2 from a common planar domain C to

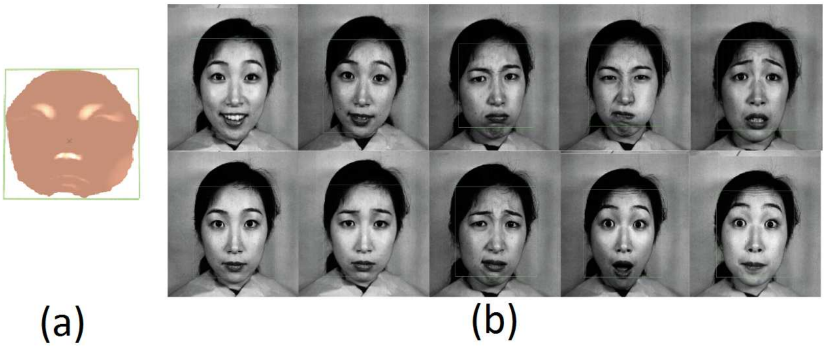

FIGURE 3.8. 2D flatten face and Jaffe Face Database

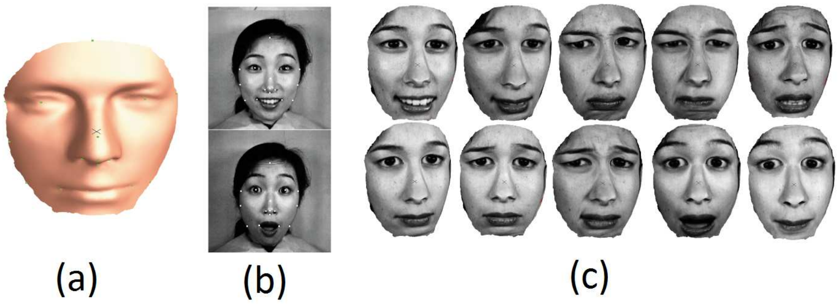

FIGURE 3.9. 3D feature points and results of Jaffe Face Database.

both the flatten surface D and the texture space T, then by linear interpolation we can still get the correspondence between surfaces and images. The sequential deviations of both the mesh and image can still be treated as the changed boundary conditions, and therefore solved using the pre-computed decomposition results.

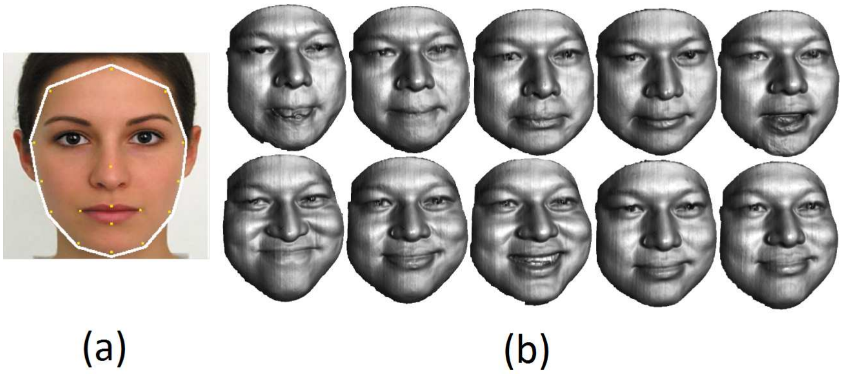

FIGURE 3.10. Texture image and samples of face dataset

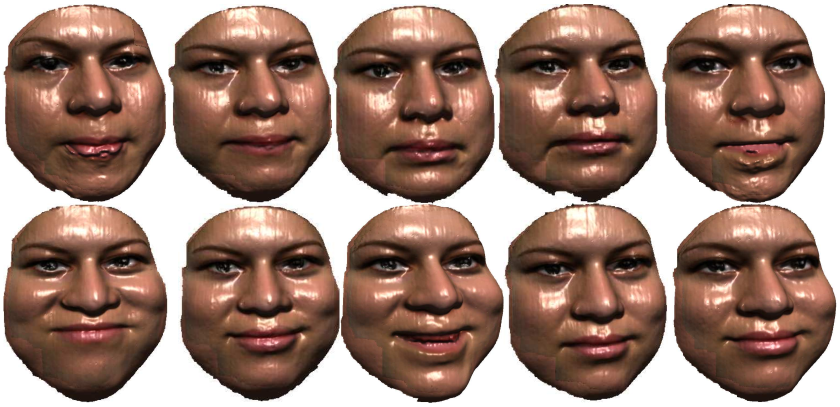

FIGURE 3.11. Mapping results for face dataset

TABLE 3.1. Performance results

Model/Texture	#Tris/Vers	Running times(s)
Cow/tiger	4315	5.81
Fandisk/checkboard	6475	6.26
Igea/female	12002	17.34
Face/male	54831	150.82

3.6 Summary

In this paper, we have presented a meshless constrained texture mapping; it provides a robust solution to the mesh structure. Experimental results show that this new method is efficient and promising in the dynamic facial registration.

Chapter 4

Feature-aligned 4D Spatiotemporal Image Registration

4.1 Introduction

Advancing modern 4-D (3D spatial + 1D temporal) CT techniques provide abundant spatial and temporal data of the patient for clinical monitoring and diagnosis. Temporally parameterizing these scan data can facilitate many clinical analysis and planning tasks. For example, in lung cancer radiation radiotherapy, 4D-CT images can be used to model the motions and deformations of the tumor and surrounding organs, to guide treatment planning [81]. Image registration plays an important role in the current motion estimation methods by establishing temporal correspondences [49, 50].

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 natural deformations [82].

However, most of the current spatiotemporal dynamic images are fully guided by the image's intensity [49, 50, 51]. The aligning computation therefore reduces to minimizing a non-linear problem having many local minima, which usually has high computational cost and, more importantly, requires a good initial guess to reach a desirable matching. Feature constraints can effectively guide the optimization from getting trapped on locally. For example, in many video tracking tasks, the SIFT descriptor has demonstrated great efficacy and been widely used due to its discriminative feature [83]. Directly generalized SIFT descriptor in 3D [84], however, could be sensitive to scalings and rotations of the deforming objects in the volume images. In this paper, we first introduce a modified 3D-SIFT descriptor that can handle these more reliably, then we develop a feature-constrained 4D dynamic registration algorithm to spatially and temporally match deforming volume images. This paper has two main contributions.

1. We propose an improved 3D feature extraction and matching algorithm based on N-SIFT method. The new method can detect more corresponding features and have less matching error.

2. We formulate a 4D spatiotemporal feature alignment metric that minimizes the position invariance over time to guide the image registration which leads to more accurate results.

4.2 Method

4.2.1 Feature Point Extraction and Matching

To handle the registration of volumetric images, Scovanner et al. [85] proposed a 3D SIFT descriptor and applied it in action recognition. Cheung and Hamarneh extended SIFT to N-Dimension SIFT [84] (N-SIFT) and showed its effectiveness on volumetric images. However, neither descriptor is scale or rotation invariant. To adequately describe images of deforming organs, we shall improve the existing 3D SIFT descriptor.

The procedure of N-SIFT includes scale space extrema detection, orientation assignment, descriptor construction and matching [84]. For an input volume image, we first extend method [83] to locate its keypoints with sub-pixel accuracy.

One limitation of N-SIFT is its sensitivity against local rotation. To more robustly handle this, we can assign multiple directions (rather than just one dominant direction used in [85]) 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 [85]. 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 on the original image and the size of the region around the keypoint is fixed. We use 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 is robust against scaling.

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. 4.1 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.

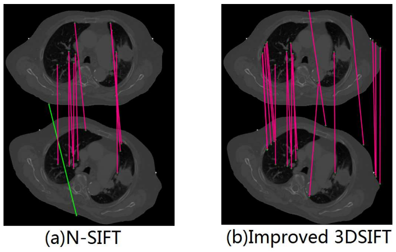

FIGURE 4.1. Feature extraction and matching.

4.2.2 4D Free-form B-spline Deformation

We present a 4D deformation model, based on a 4D free-form B-spline incorporating both the spatial and time dimensions [49]. Denote the 4D input image as $I(\mathbf{y})$, where $\mathbf{y} = (\mathbf{x}^T, t)^T \in R^3 \times R$ is a coordinate in I which consists of a spatial location $\mathbf{x} \in R^3$ and temporal location $t \in R$. The B-spline based coordinate transformation $\mathbf{T}_\mu$ is defined as follows:

$$\mathbf{T}_\mu(\mathbf{y}) = \mathbf{y} + \sum_{\mathbf{y}_k \in N_y} \mathbf{p}_k \beta^r(\mathbf{y} - \mathbf{y}_k), \tag{4.1}$$

where $\mathbf{y}_k$ is a knot on the parametric domain; $\beta^r(\cdot)$ is the r-th order multidimensional B-spline polynomial; $\mathbf{p}_k$ is the B-spline control points to be solved, and N_y denotes the neighboring region providing local support to the B-spline at $\mathbf{y}$. The knots $\mathbf{y}_k$ are defined on a 4D regular grid, uniformly overlaid on the image. The parameter vector μ consists of the collection of the first 3 elements of each $\mathbf{p}_k$. The last element of each $\mathbf{p}_k$ is fixed to zero, which ensures that only deformation in the spatial domain are allowed. In the following $T_\mu(\mathbf{y})$ is interchanged with $T_\mu(\mathbf{x}, t)$ for convenience.

In order to align all the images, we assume that after correct registration the intensity values at corresponding spatial locations over time are equal. Hence we should minimize the image intensity changes over time. An implicit reference frame is used to eliminate the need to choose a reference time point image. The dissimilarity metric, or cost function, is therefore defined as:

$$C(\mu) = \frac{1}{|S||\Gamma|} \sum_{\mathbf{x} \in S} \sum_{t \in \Gamma} (I(\mathbf{T}_\mu(\mathbf{x}, t)) - \bar{I}_\mu(\mathbf{x}))^2, \tag{4.2}$$

where $\bar{I}_\mu(\mathbf{x})$ is the average intensity value over time after applying transformation $\mathbf{T}_\mu$,

$$\bar{I}_\mu(\mathbf{x}) = \frac{1}{|\Gamma|} \sum_{t \in \Gamma} I(\mathbf{T}_\mu(\mathbf{x}, t)), \tag{4.3}$$

and S and Γ are the set of spatial and temporal voxel coordinates respectively.

As none of the images are chosen as an anatomical reference, it is necessary to add a geometric constraint to define the reference coordinate frame. Similar to [50], we define the reference frame by constraining the average deformation to be the identity transformation

$$\frac{1}{|\Gamma|} \sum_{t \in \Gamma} [\mathbf{T}_\mu(\mathbf{x}, t)]_{\mathbf{x}} = \mathbf{x}, \tag{4.4}$$

where $[\cdot]_{\mathbf{x}}$ means get the position component $\mathbf{x}$ from current 4D point $(\mathbf{x}, t)$.

The derivative of $C(\mu)$ with respect to μ is given by:

$$\frac{\partial C}{\partial \mu} = \frac{2}{|S||\Gamma|} \sum_{\mathbf{x} \in S} \sum_{t \in \Gamma} (I(\mathbf{T}_\mu(\mathbf{x}, t)) - \bar{I}_\mu(\mathbf{x})) \frac{\partial I(\mathbf{T}_\mu(\mathbf{x}, t))}{\partial \mu} \tag{4.5}$$

Then the optimal deformation field can be computed by the adaptive stochastic gradient descent optimizer (ASGD) [86].

$$\hat{\mu} = \text{ arg } \min_{\mu} C(\mu), \text{ subject to } (4.4) \tag{4.6}$$

After this registration all time point images are aligned in the implicit reference frame.

4.2.3 Feature-aligned Registration

In order to compute the transformation $\mathbf{T}_{\mu}^{ij}$ which maps coordinates from time point i to time point j, we need to compute the inverse mapping $\mathbf{T}_{\mu}^{-1}$ which maps coordinates from the input image coordinate frame to the reference frame. Since the mapping $\mathbf{T}_{\mu}$ may not be bijective, its inverse mapping $\mathbf{T}_{\mu}^{-1}$ may not actually exist. Here we define an approximate inverse mapping using a B-spline $\mathbf{T}_{\mathbf{v}}$ by minimizing

$$F_{Pos}(\mathbf{v}) = \frac{1}{|Y|} \sum_{\mathbf{y} \in Y} ||\mathbf{T}_{\mathbf{v}}(\mathbf{T}_{\hat{\mu}}(\mathbf{y})) - \mathbf{y}|| \tag{4.7}$$

where Y is the set of knots. In order to prevent foldings in the transformations we choose smaller grid spacing to yield more accurate results.

After our feature extraction and matching, we get the coherent corresponding features of all spatial images along temporal dimension. We enforce the feature matching constraints in the inverse registration.

Suppose we have N coherent features. We denote the ith feature point on time j (on jth image) as p_{ij}, where $i = 1, \ldots, N, j = 1, \ldots, \Gamma$. Intuitively, after correct registration the corresponding feature should be at the same point in the reference frame. That is, for each i, we shall also minimize the variance of $\mathbf{T}_{\mathbf{v}}(p_{ij}, j)$ in the reference image, where $(p_{ij}, j)^T$ denotes a 4D vector in the spatial-temporal space.

The cost function for feature alignment is

$$F_{Fea}(\mathbf{v}) = \frac{1}{N|\Gamma|}\sum_{i=1}^{N}\sum_{t\in\Gamma}||[T_{\mathbf{v}}(p_{it},t)]_{\mathbf{x}} - [\bar{T}_{\mathbf{v}}(p_{i}\cdot)]_{\mathbf{x}}|| \tag{4.8}$$

where

$$\bar{T}_{\mathbf{v}}(p_{i}\cdot) = \frac{1}{|\Gamma|}\sum_{t\in\Gamma}[T_{\mathbf{v}}(p_{i,t},t)]_{\mathbf{x}} \tag{4.9}$$

The final objective function for estimating the optimal deformation field is formulated as:

$$F_{\mathbf{v}} = F_{Pos} + \lambda F_{Fea} \tag{4.10}$$

where λ is the weighting factor controlling the strength of the feature constraint term. We determine those transform parameters that minimize the total metric as

$$\hat{\mathbf{v}} = \ \arg\min_{\mathbf{v}} F(\mathbf{v}). \tag{4.11}$$

We also solve Eq-(4.11) using ASGD where

$$\begin{aligned}\frac{\partial F_{Pos}}{\partial \mathbf{v}} = \frac{1}{|Y|}\sum_{y\in Y} & \frac{1}{||\mathbf{T}_{\mathbf{v}}(\mathbf{T}_{\hat{\mu}}(\mathbf{y})) - \mathbf{y}||} \\ & (\mathbf{T}_{\mathbf{v}}(\mathbf{T}_{\hat{\mu}}(\mathbf{y})) - \mathbf{y}) \\ & (\frac{\partial T_{\mathbf{v}}(T_{\hat{\mu}}(y))}{\partial \mathbf{v}})\end{aligned} \tag{4.12}$$

$$\begin{aligned}\frac{\partial F_{Fea}}{\partial \mathbf{v}} = \frac{1}{N|\Gamma|}\sum_{i=1}^{N}\sum_{t\in\Gamma} & \frac{1}{||[T_{\mathbf{v}}(p_{it},t)]_{\mathbf{x}} - [\bar{T}_{\mathbf{v}}(p_{i}\cdot)]_{\mathbf{x}}||} \\ & ([T_{\mathbf{v}}(p_{it},t)]_{\mathbf{x}} - [\bar{T}_{\mathbf{v}}(p_{i}\cdot)]_{\mathbf{x}}) \\ & (\frac{\partial T_{\mathbf{v}}(p_{it},t)}{\partial \mathbf{v}} - \frac{\partial \bar{T}_{\mathbf{v}}(p_{i}\cdot)}{\partial \mathbf{v}})\end{aligned} \tag{4.13}$$

Finally, we can get the transformation from time point i to time point j:

$$T^{ij}_{\hat{\mu},\hat{v}}(\mathbf{x}) = [T_{\mu}([T_{\mathbf{v}}(\mathbf{x},t_i)]_{\mathbf{x}},t_j)]_{\mathbf{x}}. \tag{4.14}$$

TABLE 4.1. The registration error in mm, on 40 landmarks among 0^{th},5^{th},9^{th} time frames of the POPI-data. $E_{i,j}$ is the matching error from i^{th} to j^{th} frame, $\bar{E}$ is the mean error for the whole sequence.

	$E_{0,5}$	$E_{0,9}$	$E_{5,0}$	$E_{5,9}$	$E_{9,0}$	$E_{9,5}$	$\bar{E}$
[49]	2.94	0.98	2.88	2.86	0.99	2.91	2.26
Our's	2.87	0.85	2.77	2.74	0.87	2.84	2.16

4.3 Implementations and Experiments

We implement our 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 Core E7300 @2.66 GHz, 4GB RAM. The registration on the POPI-model and our tumor data take approximately 45 mins, of which 22 mins were spent to compute the 4D forward registration and 23 mins were spent to compute the 4D inverse registration.

Experiments on POPI Dataset. Our first experiment is conducted on the POPI dataset [87]. This dataset contains one 4D CT series including ten 3D volumes representing ten different phases of one breathing cycle. In the 3D volume at time frame t, the coherent landmarks (a set of 3D points, denote as $P_t = \{p_{t,1}, p_{t,2}, ..., p_{t,|P_t|}\}$) are available and can be used to evaluate the registration. We use the time frames 0, 5, and 9 with 571 feature correspondences to do group registration. The registration results were evaluated by the *mean target registration error* (MTRE) between the set of landmark points $\{P_0, P_5, P_9\}$. Denote MTRE as $E_{r,t} = \frac{1}{|P_t|} \sum_{p_{t,i} \in P_t} ||T^{r,t}(p_{r,i}) - p_{t,i}||$, where $p_{t,i}$ is a landmark i in time t. In our experiments, we set the control weight in Eq. (4.10) as $\lambda = 0.1$. Table 4.1 shows the comparison between our method and the algorithm of [49]: our method outperforms [49] by introducing smaller MTRE errors.

Lung Tumor Registration. Our second experiment is to apply our registration model in dynamic tumor tracking (Fig. 4.2). We detect 202 feature correspondences among the image sequence. Before registration, we segment the tumor in the first frame by using 3D graph-cut segmentation [81]. Then with our registration results, we track this tumor in the following second/third time sequence (shows in the second/third column of fig. 4.2). The bottom of this figure depicts the registration of this tumor among different time sequences.

Furthermore, we compute an unbiased difference image between the deformed image and the target image to evaluate the registration accuracy. Assume the 3D source image is $I^i(\mathbf{X})$ in i-th frame, the 3D target image is $I^j(\mathbf{X})$ in j-th frame. The deformed image is $I(Tx)$ where $Tx = T^{ij}(\mathbf{x})$ and $\mathbf{x}$ from the source image. In order to avoid the influence of the gray value of original pixel, we normalize the difference frame: if $I(Tx) + I^j(Tx) \neq 0$ then $I^d(Tx) = \frac{|I(Tx)-I^j(Tx)|}{I(Tx)+I^j(Tx)}$; otherwise, $I^d(Tx) = 0$. It is easy to check that this metric is symmetric between the deformed image and target image. Smaller I^d indicates more accurate registration.

Fig 4.3 (a) shows the projection of the difference image between the second and the third frame. (b) shows the histogram of the computed difference value. We construct this histogram based on the normalized difference frame between the deformed second frame and the third frame. We count the occurrence of each difference value and divide it by the total number of the pixels to get its probability. We can see in larger than 90% pixels, the difference value is less than 0.1, and the mean difference value is 0.016. These indicates that our registration introduces very small error between deformed image and the target image. Thus our registration can be used for tumor motion tracking (see Fig.4.2).

Also, this visualization (Fig 4.3 (a)) can also help us to identify the region with large registration errors for subsequent matching refinement. We can see around the boundary part and the central of left lung part have larger difference value. In the future, we will develop hierarchically spline scheme to support adaptive refinement, so that we can insert more knots in these regions to reduce the registration error.

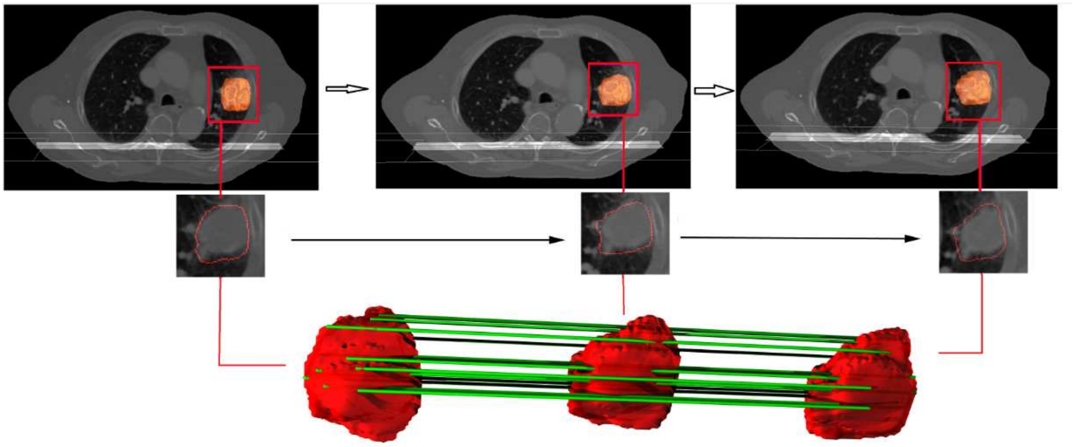

FIGURE 4.2. Tumor tracking with our registration.

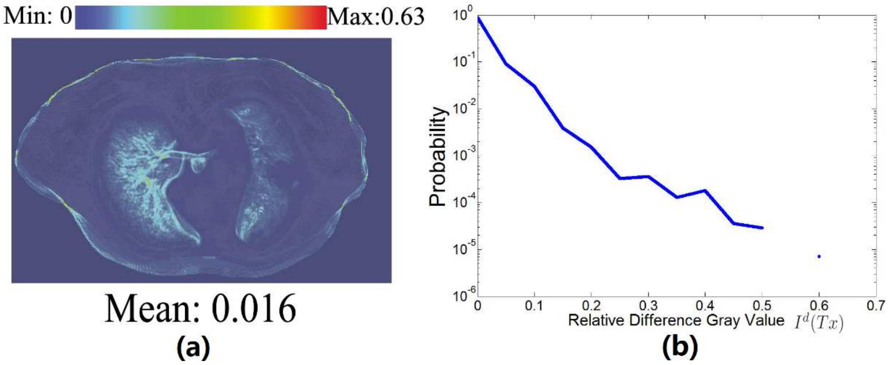

FIGURE 4.3. Difference image and its histogram.

4.4 Summary

We propose an automatic feature-guided 4D image registration framework. We develop an improved 3D-SIFT descriptor for reliable feature extraction and matching. Compared with existing 4D registration model we achieve better landmark predication accuracy. Our model also has good ability to do tumor motion estimation which can greatly facilitate lung tumor radiotherapy planning and management.

Chapter 5
Conclusion

In this thesis, we study the non-rigid matching of dynamic deforming 2D and 3D data. For the dynamic surface texturing problem, we proposed a a meshless texture mapping framework, based on conformal flattening and planar harmonic transformation computed by MFS. This framework can effectively handle feature alignment and efficiently map deforming face meshes data or 2D image data.

For the dynamic medical volume motion modeling problem, we proposed a registration method for dynamic motion estimation in medical imaging data. This registration method combines the feature extraction, global cost function based on 4D B-spline transformation and feature alignment. We also compare our model with state-of-art 4D registration method and it shows we can achieve better accuracy on the landmark predication. With the segmentation method our method can also be used to do tumor motion tracking which is important in the lung cancer treatment.

References

[1] Fausto Bernardini and Holly Rushmeier. The 3d model acquisition pipeline. *Computer Graphics Forum*, 21(2):149–172, 2002.

[2] Barbara Zitov and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977 – 1000, 2003.

[3] T. J. Hutton, B. F. Buxton, and P. Hammond. Automated registration of 3d faces using dense surface models. In *Proc. BMVC*, pages 45.1–45.10, 2003.

[4] Jianhua Yao and Russell Taylor. Assessing accuracy factors in deformable 2d/3d medical image registration using a statistical pelvis model. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV, page 1329. IEEE Computer Society, 2003.

[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] Tsz-Ho Kwok, Yunbo Zhang, and Charlie 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] Janine Bennett, Valerio Pascucci, and Kenneth 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] Kang Zhang and Xin Li. Optimizing geometry-aware pants decomposition. In *Proc. Pacific Conference on Computer Graphics and Applications*, 2012.

[20] X. Gu and S.-T. Yau. Global conformal surface parameterization. In *Proc. Symp. Geometry Processing*, pages 127–137, 2003.

[21] N. Ray, W. Li, B. Lévy, A. Sheffer, and P. Alliez. Periodic global parameterization. *ACM Trans. Graph.*, 25(4):1460–1485, 2006.

[22] Alla Sheffer and Eric de Sturler. Parameterization of faceted surfaces for meshing using angle-based flattening. *Eng. Comput. (Lond.)*, 17(3):326–337, 2001.

[23] M. Jin, F. Luo, and X. Gu. Computing surface hyperbolic structure and real projective structure. In *Proc. ACM symposium on Solid and physical modeling*, pages 105–116, 2006.

[24] 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. 2005.

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

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

[27] 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.

[28] Shuchu Han, Jiazhi Xia, and Ying He. Hexahedral shell mesh construction via volumetric polycube map. In *Proc. ACM Symposium on Solid and Physical Modeling*, pages 127–136, 2010.

[29] Jiazhi Xia, Ying He, Xiaotian Yin, Shuchu Han, and Xianfeng Gu. Direct-product volumetric parameterization of handlebodies via harmonic fields. *Shape Modeling and Applications, International Conference on*, pages 3–12, 2010.

[30] 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.

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

[32] 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.

[33] 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.

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

[35] Bo Li, Xin Li, Kexiang Wang, and Hong Qin. Generalized polycube trivariate splines. *Shape Modeling and Applications, International Conference on*, 0:261–265, 2010.

[36] 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.

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

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

[39] P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki. Harmonic coordinates for character articulation. In *SIGGRAPH*, pages 71–81, 2007.

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

[41] Giuseppe Patanè, Michela Spagnuolo, and Bianca Falcidieno. Topology- and error-driven extension of scalar functions from surfaces to volumes. *ACM Trans. Graph.*, 29:4:1–4:20, 2009.

[42] E. Sharon and D. Mumford. 2d-shape analysis using conformal mapping. *Int. J. Comput. Vision*, 70(1):55–75, October 2006.

[43] X. Gu, Y. Wang, T. Chan, P. Thompson, and S. T. Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Trans. Med. Imaging*, 23(8):949–958, 2004.

[44] Yalin Wang, Ming-Chang Chiang, and P.M. Thompson. Mutual information-based 3d surface matching with applications to face recognition and brain mapping. In *Computer Vision, Tenth IEEE International Conference on*, volume 1, pages 527 – 534 Vol. 1, Oct. 2005.

[45] Sen Wang, X.D. Gu, and Hong Qin. Automatic non-rigid registration of 3d dynamic data for facial expression synthesis and transfer. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1 –8, June 2008.

[46] Stephen Marsland, Carole J. Twining, and Chris J. Taylor. A minimum description length objective function for groupwise non-rigid image registration. *Image and Vision Computing*, 26(3):333 – 346, 2008.

[47] V. Boldea, G.C. Sharp, S.B. Jiang, and D. Sarrut. 4d-ct lung motion estimation with deformable registration: quantification of motion nonlinearity and hysteresis. *Med Phys*, 35(3):1008–18, 2008.

[48] Joseph M. Reinhardt, Kai Ding, Kunlin Cao, Gary E. Christensen, Eric A. Hoffman, and Shalmali V. Bodas. Registration-based estimates of local lung tissue expansion compared to xenon ct measures of specific ventilation. *Medical Image Analysis*, 12(6):752 – 763, 2008.

[49] 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.

[50] 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.

[51] Guorong Wu, Qian Wang, Jun Lian, and Dinggang Shen. Estimating the 4d respiratory lung motion by spatiotemporal registration and building super-resolution image. MICCAI'11.

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

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

[54] 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.

[55] 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.

[56] 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.

[57] Bruno Levy. Constrained texture mapping for polygonal meshes. In *SIGGRAPH: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM, 2001.

[58] Yanwen Guo, Jin Wang, Hanqiu Sun, Xiufen Cui, and Qunsheng Peng. A novel constrained texture mapping method based on harmonic map. *Computers and Graphics*, 29(6):972 – 979, 2005.

[59] I Eckstein, V Surazhsky, and C Gotsman. Texture mapping with hard constraints. *COMPUTER GRAPHICS FORUM*, 20(3):C95–C104, 2001.

[60] Vladislav Kraevoy, Alla Sheffer, and Craig Gotsman. Matchmaker: constructing constrained texture maps. *ACM Trans. Graph.*, 22(3):326–333, 2003.

[61] Vladislav Kraevoy and Alla Sheffer. Cross-parameterization and compatible remeshing of 3d models. *ACM Trans. Graph.*, 23(3):861–869, 2004.

[62] J Schreiner, A Asirvatham, E Praun, and H Hoppe. Inter-surface mapping. *ACM TRANSACTIONS ON GRAPHICS*, 23(3):870–877, AUG 2004.

[63] K Zhou, X Wang, YY Tong, M Desbrun, BN Guo, and HY Shum. TextureMontage: Seamless texturing of arbitrary surfaces from multiple images. *ACM TRANSACTIONS ON GRAPHICS*, 24(3):1148–1155, JUL 2005.

[64] Yu-Wing Tai, Michael S. Brown, Chi-Keung Tang, and Heung-Yeung Shum. Texture amendment: reducing texture distortion in constrained parameterization. In *ACM SIGGRAPH Asia papers*, pages 1–6. ACM, 2008.

[65] Yochay Tzur and Ayellet Tal. Flexistickers: photogrammetric texture mapping using casual images. *ACM Trans. Graph.*, 28(3):1–10, 2009.

[66] Xianwang Wang, Qing Zhang, Ruigang Yang, Brent Seales, and Melody Carswell. Feature-based texture mapping from video sequence. In *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, pages 1–1. ACM, 2008.

[67] Matthias Zwicker, Mark Pauly, Oliver Knoll, and Markus Gross. Pointshop 3d: an interactive system for point-based surface editing. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 322–329. ACM, 2002.

[68] Patrick Degener and Reinhard Klein. Texture atlas generation for inconsistent meshes and point sets. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications*, pages 156–168. IEEE Computer Society, 2007.

[69] G. Zigelman, R. Kimmel, and N. Kiryati. Texture mapping using surface flattening via multidimensional scaling. *Visualization and Computer Graphics, IEEE Transactions on*, 8(2):198–207, Apr-Jun 2002.

[70] Tong-Yee Lee, Shao-Wei Yen, and I-Cheng Yeh. Texture mapping with hard constraints using warping scheme. *Visualization and Computer Graphics, IEEE Transactions on*, 14(2):382–395, March-April 2008.

[71] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. In *SIGGRAPH*, pages 362–371, 2002.

[72] A. Sheffer, C. Gotsman, and N. Dyn. Robust spherical parameterization of triangular meshes. *Computing*, 72(1-2):185–193, 2004.

[73] C. Gotsman, X. Gu, and A. Sheffer. Fundamentals of spherical parameterization for 3D meshes. *ACM Trans. Graph.*, 22(3):358–363, 2003.

[74] X. Gu and B. Vemuri. Matching 3D shapes using 2D conformal representations. In *MICCAI (1)*, pages 771–780, 2004.

[75] X. Gu and S.T. Yau. Computing conformal structure of surfaces. *Communications in Information and Systems*, 2(2):121–146, 2002.

[76] X. Li, X. Gu, and H. Qin. Curve space: Classifying curves on surfaces. *Communications in Information and Systems*, 7(3):207–226, 2007.

[77] 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(3):409 – 422, 2009.

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

[79] T. Kitagawa. On the numerical stability of the method of fundamental solutions applied to the dirichlet problem. *Japan Journal of Applied Mathematics*, 35:507–518, 1988.

[80] M. A. Golberg and C. S. Chen. The method of fundamental solutions for potential, helmholtz and diffusion problems. *Boundary integral methods: Numerical and mathematical aspects (A99-30801 07-64)*, 1:103–176, 1999.

[81] S. S. Iyengar, Xin Li, Huanhuan Xu, Supratik Mukhopadhyay, N. Balakrishnan, Amit Sawant, and Puneeth Iyengar. Toward more precise radiotherapy treatment of lung tumors. *IEEE Computer*, 45:59–65, 2012.

[82] 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.

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

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

[85] Paul Scov., Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. MM'07.

[86] Stefan Klein, Josien P. Pluim, Marius Staring, and Max A. Viergever. Adaptive stochastic gradient descent optimisation for image registration. *Int. J. Comput. Vision*, 81(3):227–239, March 2009.

[87] J. Vandemeulebroucke, D. Sarrut, and P. Clarysse. Point-validated pixel-based breathing thorax model. In *ICCR07*.

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 and Mathematics with Honors. She earned a master degree in Automation from University of Science and Technology of China in 2009.

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 is in Computer Graphics and Geometric Modeling. She is also interested in Computer Vision and Pattern Recognition.