Graphical Models 79 (2015) 1–11

Contents lists available at ScienceDirect

Graphical Models

journal homepage: www.elsevier.com/locate/gmod

An efficient volumetric method for non-rigid registration ☆

Ran Zhang [a], Xuejin Chen [a,*], Takaaki Shiratori [b], Xin Tong [b], Ligang Liu [a]

[a] University of Science and Technology of China, No. 96, Jinzhai Road, Hefei, Anhui, PR China
[b] Microsoft Research Asia, Building 2, No. 5 Dan Ling Street, Haidian District, Beijing, PR China

ARTICLE INFO

Article history:
Received 19 March 2014
Received in revised form 3 December 2014
Accepted 6 January 2015
Available online 11 March 2015

Keywords:
Non-rigid registration
Distance field
Deformation model

ABSTRACT

We propose a novel and efficient volumetric method for registering 3D shapes with non-rigid deformations. Our method uses a signed distance field to represent the 3D input shapes and registers them by minimizing the difference between their distance fields. With the assumptions that the sampling points in each cell of the object volume follow the same rigid transformation, and the transformations of the sampling cells vary smoothly inside the object volume, a two-step method is used for the non-rigid registration. The first step is the locally rigid registration, which minimizes the difference between the source and target distance fields of the sampling cells. The second step is the globally non-rigid registration, which minimizes the difference between the transformations of adjacent cells. In just a few iterations, our method rapidly converges for the registration. We tested our method on several datasets, and the experimental results demonstrate the robustness and efficiency of our method.

1. Introduction

Non-rigid shape registration is a fundamental problem in geometric modeling and model acquisition. The development of real-time 3D scanners makes non-rigid registration increasingly important. There are various types of non-rigid shapes in our daily life, like the human body, animals, etc.

The goal of non-rigid registration is to find a global deformation for the source shape to match the target shape. Semantic correspondence of two shapes is very difficult to compute if the shapes are non-rigid. A vast amount of work has been devoted to this problem. The general framework has been to preserve the local geometry and regularize the globally non-rigid deformation. Volumetric methods have attracted more and more attention recently due to their improved registration compared to using surface-based methods. However, registration of large deformations remains very challenging for the existing volumetric methods.

In this paper, we propose a fast and robust volumetric method for non-rigid registration, which can regularize the global deformation while preserving the local geometry. We do not require an initial template, or other shape priors, or explicit correspondences. Our approach works well on several datasets, including the synthetic data generated by morphing the source surface, and the mesh pairs from a scanning system. The local geometry is preserved, while a natural global deformation is achieved with our volumetric scheme.

2. Related work

Shape registration is a classic problem in computer graphics and computer vision. A substantial amount of research has been devoted to static shape registration.

☆ This paper has been recommended for acceptance by Pierre Alliez.

* Corresponding author at: CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System.
E-mail addresses: cuminflea@gmail.com (R. Zhang), xjchen99@ustc.edu.cn (X. Chen), takaakis@microsoft.com (T. Shiratori), xtong@microsoft.com (X. Tong), lgliu@ustc.edu.cn (L. Liu).

http://dx.doi.org/10.1016/j.gmod.2015.01.003
1524-0703/

The representative method is Iterative Closest Point (ICP) [1], which uses the closest points as correspondences, and then calculates a rigid transformation by minimizing the distance between the correspondences. Recently, the KinectFusion algorithm [2,3] was proposed, which uses a fusion method to reconstruct a consistent volumetric representation [4] and achieves realtime performance.

Non-rigid registration has become an active area of research due to the development of high-performance 3D scanners. Most of the work focuses on measuring the similarity between shapes and regularizing the global deformation. Ref. [5] introduces a method that looks for the optimal transformation for each corresponding element, and regularizes the globally non-rigid transformation using the dynamic sampling graph [6]. It is used in [7] for capturing a room-sized dynamic scene. Their later work uses a template as a prior to prevent the erroneous topologies, so that the method can handle large deformations [8]. Chang and Zwicker [9] present a registration method for articulated objects using feature points and a reduced deformation model. Ref. [10] introduces an articulated global registration algorithm using the dynamic sample graph to align multiple range scans of an articulated model. Another method uses a subspace deformation model for registration of non-rigid shapes with fixed topology [11]. All of the above methods above perform surface-based registration.

Recently, volumetric methods have been proposed for non-rigid registration [12,13]. Ref. [12] uses a vector field to register local geometry and Incremental Free-Form Deformation (IFFD) to achieve global deformation. Ref. [13] introduces a method that uses a signed distance field [14] for geometry representation, and then uses Free-Form Deformation (FFD) as the deformation model to regularize the globally non-rigid deformation. Our method utilizes a similar scheme, but applies a rigid cell deformation model [15] to achieve a more robust registration.

All these methods treat the deformation model as an important component for non-rigid registration. Deformation models can be roughly classified into surface-based models, and volumetric or space models. Embedded deformation [6] and as-rigid-as-possible surface manipulation [16] are typical surface-based deformation models. Embedded deformation builds a graph on the surface of the shape, and finds the optimal affine transformations by solving a nonlinear minimization problem. The as-rigid-as-possible method solves for a global deformation, while preserving the local rigidity. Surface-based deformation models are efficient and easy to implement. However, the volumetric model is demonstrated to be more robust when applied to large deformations [15,17].

Volumetric deformation models use a volumetric graph, instead of a surface-based graph, to solve the global deformation of the volumetric graph. In the Free-Form Deformation (FFD) [18] method, the shape is embedded in space grids consisting of control points and deformed by moving the control points of the FFD grids. The Volumetric Graph Laplacian method [17] introduces a volumetric graph that consists of boundary edges and interior edges. Then, the shape morphing problem can be solved using a method like Laplacian surface manipulation [19]. To achieve the global deformation, the difference of adjacent space grids is minimized to achieve a global deformation [15,20].

3. Overview

A volumetric scheme is used in our registration of the deformed shapes. The entire pipeline is shown in Fig. 1. With the signed distance field (SDF) representation, as shown in Fig. 1(a), the registration problem of these two shapes becomes the registration of the two signed distance fields. Our system iteratively runs a two-step process to obtain both the *locally rigid registration* and the *globally non-rigid registration*. The first step is performed by minimizing the difference between the source and target signed distance fields for each sampling cell. In the second step, a non-linear deformation model is employed to generate a natural globally non-rigid deformation, given the locally rigid registration results, as Fig. 1(b) shows. The final optimal non-rigid registration, as Fig. 1(c) shows, is achieved after several iterations.

4. Signed distance field representation

In our system, we employ the signed distance function, also called Signed Distance Field (SDF) [14], as the volumetric representation for 3D shapes. For simplicity, we

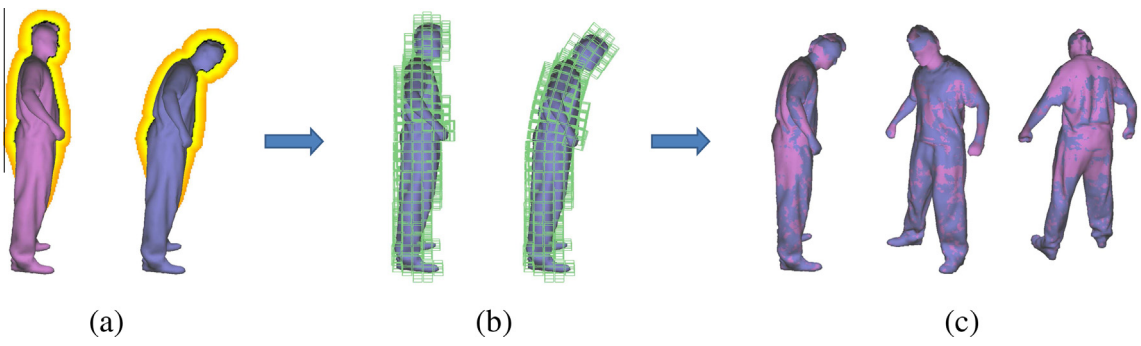

Fig. 1. System overview: (a) signed distance fields of the source and target shapes, (b) locally rigid registration, and (c) globally non-rigid registration.

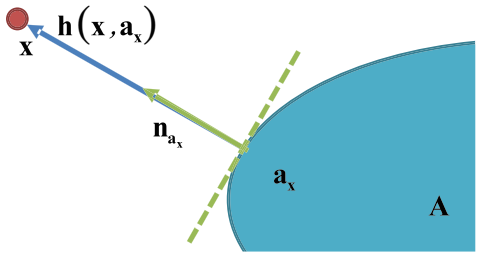

Fig. 2. The signed distance is calculated by finding the closest element and the angle weighted pseudonormal of the element.

explain it in a 2D case, which can be easily extended to 3D cases.

Given a 2D shape, denoted as A, for an arbitrary point, denoted as $\mathbf{x}$, in the 2D space, we find the closest element (point, face or edge) on the shape, denoted as $\mathbf{a_x}$, as shown in Fig. 2. The signed distance of $\mathbf{x}$ is defined as

$$\phi_A(\mathbf{x}) = sgn(\mathbf{n}_{\mathbf{a_x}} \cdot \mathbf{h}(\mathbf{x}, \mathbf{a_x}))|\mathbf{h}(\mathbf{x}, \mathbf{a_x})|, \quad (1)$$

where $\mathbf{h}(\mathbf{x}, \mathbf{a_x})$ is a vector from the closest point on the element to the point $\mathbf{x}$, $\mathbf{n}_{\mathbf{a_x}}$ is the angle weighted pseudonormal of the element $\mathbf{a_x}$, and $sgn()$ is the sign function. An important advantage of this volumetric representation is its resistance to surface noise; however, the disadvantage is its large memory cost. By using a distance field computation method called Parallel Banding Algorithm (PBA) [21], the computation time can be greatly reduced. Some work using the implicit representation omitted the field far from the surface in order to reduce the computational cost (i.e., narrow-band representation). However, the field that is far from the surface but still inside the surface carries useful information for robustness in the system. Therefore, we retain all parts of the distance field. We sample the space by uniformly spreading the sampling points, like a uniform lattice. By calculating the signed distance at each sample point, we generate the SDF Φ_A of the shape A. Fig. 3 shows the computed signed distance field for a human body model.

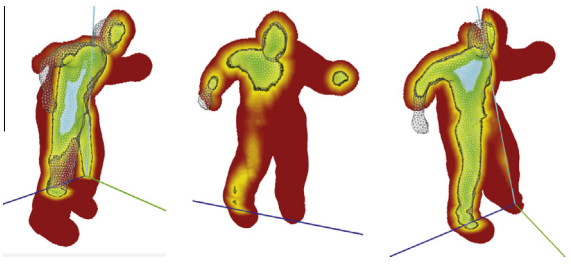

Fig. 3. Generated signed distance field for a human body model. The yellow and red areas denote the points with positive signed distances, which means that these points are outside the surface. The green and blue areas represent the points inside the surface. We cut the SDF by different planes to demonstrate the generated SDF. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5. Locally rigid, globally non-rigid registration

Given the signed distance field representations for the source and target shapes, we register the non-rigid object using a volumetric registration framework assuming that the shape is locally rigid. In order to maintain the local rigidity, we divide the space into regular cube cells. Each cell contains the same number of sampling points. All sample points in a cell follow the same transformation in the deformation process.

Our non-rigid registration algorithm consists of two steps. In the locally rigid registration step, we look for the rigid transformation for each cell individually by minimizing the difference between the SDFs at the sample points in the source and the target shapes. In the globally non-rigid registration step, we minimize the difference between the transformations of adjacent cell pairs. The details of the algorithm are described in the following sections.

5.1. Locally rigid registration

Assuming that each cell preserves its rigidity during deformation, we compute the rigid transformation $\mathbf{T_i}$ for the cell C_i from the source distance field to the target distance field. We extend Fujiwara's approach [13] to a 3D application for the locally rigid registration. While all the sampling points in a cell follow the same transformation, each cell is registered by minimizing the differences between the SDFs at the points in the source shape and the transformed points in the target shape. We use a quaternion $\mathbf{q}$ to represent the 3D rotation. Therefore, a rigid transformation $\mathbf{T}$ can be determined by its rotation and translation parameters $\mathbf{w} = \{\mathbf{q}, \mathbf{t}\}$.

The error function of locally rigid registration is:

$$E_{local} = d(\Phi_A, \Phi_B) = \sum_{C_i \in A} E_{l_i} = \sum_{C_i \in A} \sum_{\mathbf{p}_j \in C_i} \left(\phi_B(\mathbf{T}(\mathbf{p}_j; \mathbf{w}_i)) - \phi_A(\mathbf{p}_j)\right)^2, \quad (2)$$

where Φ_A and Φ_B are the signed distance fields for the source shape A and target shape B respectively. $\mathbf{p}_j$ is a sample point in the cell C_i in the source shape A. $\mathbf{w}_i$ represents the parameters of the rigid transformation for the cell C_i. $\mathbf{T}(\mathbf{p}_j, \mathbf{w}_i)$ is a function that transforms the point $\mathbf{p}_j$ with the transformation parameters $\mathbf{w}_i$.

Minimizing the expression in Eq. (2) is a non-linear optimization task. This is because the transformation function $\mathbf{T}$ is non-linear in $\mathbf{w}$, and the signed distance function $\phi(\mathbf{p})$ is generally non-linear in $\mathbf{p}$. In fact, the signed distance $\phi(\mathbf{p})$ is essentially unrelated to the point coordinates $\mathbf{p}$. We use an iterative registration method proposed by Lucas and Kanade [22] to solve the minimization problem. The parameter $\mathbf{w}_i$ is updated iteratively with the incremental values $\Delta\mathbf{w}_i$ in each iteration as $\mathbf{w}_i \leftarrow \mathbf{w}_i + \Delta\mathbf{w}_i$.

Each cell is registered independently. We separately minimize the local error function E_{l_i} for each cell with respect to $\Delta\mathbf{w}_i$ in each iteration as

$$E_{l_i} = \sum_{\mathbf{p}_j \in C_i} \left(\phi_B(\mathbf{T}(\mathbf{p}_j; \mathbf{w}_i + \Delta\mathbf{w}_i)) - \phi_A(\mathbf{p}_j)\right)^2. \quad (3)$$

The Lucas–Kanade algorithm can be regarded as a Gauss–Newton gradient descent non-linear optimization algorithm. A first-order Taylor expansion is used to linearize the target SDF ϕ_B respect to $\mathbf{w}_i$. Then the local registration error is written as

$$E_{l_i} = \sum_{\mathbf{p}_j \in C_i} \left(\phi_B(\mathbf{T}(\mathbf{p}_j; \mathbf{w}_i)) + \nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \Delta \mathbf{w}_i - \phi_A(\mathbf{p}_i) \right)^2, \quad (4)$$

where $\nabla \phi_B$ is the gradient of the target distance field. It is computed using the Sobel operator.

Minimizing the expression in Eq. (4) is a least squares problem and has a closed-form solution, which can be derived by setting the partial derivative of the error function E_{l_i} with respect to $\Delta \mathbf{w}_i$ to zero:

$$\frac{\partial E_{l_i}}{\partial \Delta \mathbf{w}_i} = 2 \sum_{\mathbf{p}_j} \left[\nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \right]^T \left[\phi_B(\mathbf{T}(\mathbf{p}_j; \mathbf{w}_i)) + \nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \Delta \mathbf{w}_i - \phi_A(s) \right] = \mathbf{0}. \quad (5)$$

This yields the value of parameter $\Delta \mathbf{w}_i$:

$$\Delta \mathbf{w}_i = \mathbf{H}_i^{-1} \sum_{\mathbf{p}_j} \left[\nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \right]^T \left[\phi_A(\mathbf{pj}) - \phi_B(\mathbf{T}(\mathbf{p}_j; \mathbf{w}_i)) \right], \quad (6)$$

where $\mathbf{H}_i$ is the Gauss–Newton approximation to the Hessian matrix written as:

$$\mathbf{H}_i = \sum_{\mathbf{p}_j} \left[\nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \right]^T \left[\nabla \phi_B \frac{\partial \mathbf{T}}{\partial \mathbf{w}_i} \right]. \quad (7)$$

We update the parameters by adding the incremental values $\Delta \mathbf{w}_i$ to the initial parameters $\mathbf{w}_i$. This process is iteratively computed until it converges (see Fig. 4).

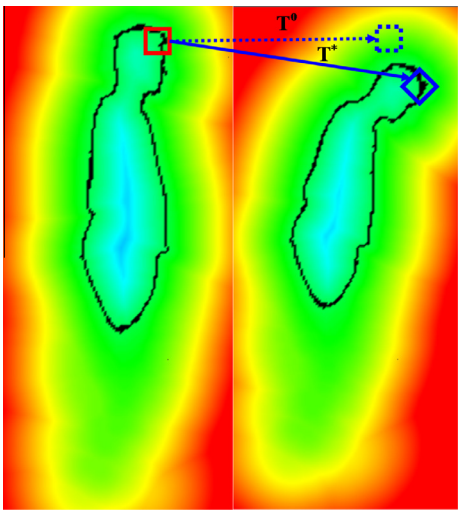

Fig. 4. Look for the locally rigid transformation to minimize the difference between the two SDFs in an iterative way. $\mathbf{T}^0$ is the initial transformation. $\mathbf{T}^*$ is the optimal transformation to transform the source cell (red) into the target model (blue cell). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2. Globally non-rigid registration

From the locally rigid registration, we get the transformation for each cell that aligns the source and target shapes. However, the computed local transformation for each cell is a local optimal. It may be semantically incorrect due to the non-rigid deformation. Consequently, additional global constraints are needed to regularize the transformations. Starting from the locally rigid transformation, we use the deformation model proposed in [15] that produces the global deformation for a volumetric representation.

5.2.1. Deformation model

Given the rigid transformation $\mathbf{T}_i$ of an cell C_i, the deformation model generates a rigid transformation $\mathbf{T}_j$ for its adjacent cell C_j by minimizing the difference of $\mathbf{T}_i$ and $\mathbf{T}_j$. This leads to a global smoothing of the cell transformations.

The difference of the two transformations is defined by the squared distances of the transformed shapes of the two adjacent cells with both transformations [23]:

$$E_{ij} = \int_{C_i \bigcup C_j} \|\mathbf{T}_i \mathbf{p} - \mathbf{T}_j \mathbf{p}\|^2 d\mathbf{p}, \quad (8)$$

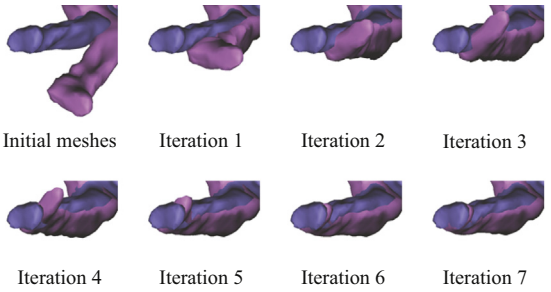

Fig. 5. Close-up views of the registration progress.

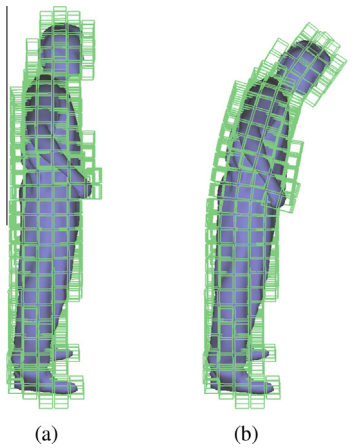

Fig. 6. (a) Space discretization of the input human body model. The green cubes are the sampling cells. (b) Deformed cells after the global, non-rigid registration. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where C_i and C_j are two adjacent cells. For simple cell shapes, such as tetrahedra or hexahedra, the above integral can be evaluated analytically. In our system, all cells are uniform cubes, so the integral can be analytically calculated.

The smooth energy of the deformation model is defined as a sum of all pairs of adjacent cells C_i and C_j:

$$E_{smooth} = \sum_{i,j} E_{ij}(\mathbf{T}_i, \mathbf{T}_j). \quad (9)$$

5.2.2. Weighted cell transformations

Minimizing Eq. (9) assumes that each transformation $\mathbf{T}_i$ has the same effect as other transformations. However, the initial transformations $\mathbf{T}_i^{loc}$ generated in the locally rigid registration step lead to different registration errors for different cells. The transformations that accurately register the corresponding cells should be preserved, while the poorly registered transformations require significantly adjustments. We introduce the weighted fitting energy E_{fit} written as

$$E_{fit} = \sum_i \mu_i \int_{C_i} \left(\mathbf{T}_i\mathbf{p} - \mathbf{T}_i^{loc}\mathbf{p}\right) d\mathbf{p}, \quad (10)$$

where $\mathbf{T}_i$ is the rigid transformation to be optimized for cell C_i. $\mathbf{T}_i^{loc}$ is the transformation generated in the locally rigid registration step for cell C_i, and μ_i is an adaptive weight for the cell C_i determined by its fitting energy:

$$\mu_i = \frac{e_{max} - e_i^{loc}}{e_{max}} + \varepsilon, \quad (11)$$

Table 1
Performance statistics. N_{cell}: Cell Number. N_{iter}: Iteration Number. E_{reg}: Registration Error (relative to the bounding cube diagonal). T: Time.

Dataset	N_{cell}	N_{iter}	E_{reg} ours (10^{-4})	E_{reg} [12] (10^{-4})	T ours (s)	T [12] (s)
Human	809	14	5.6	5.1	37	67
Samba	589	15	7.3	9.8	20	32
Dance	807	10	10.6	11.5	34	75
Swing	580	16	10.4	6.27	21	33
Plant	490	23	9.57	26.1	14	18

where e_i^{loc} is the local registration error for cell C_i after the locally rigid registration step. e_{max} is the maximum error of all the cells. ε is added so that the transformations with large registration errors are not neglected. After an experimental analysis (see Fig. 14), we decided to use $\varepsilon = 0.001$ in our experiments.

5.2.3. Optimization of globally non-rigid registration

Combining E_{smooth} in Eq. (9) and E_{fit} in Eq. (10), we define the globally non-rigid registration energy function as:

$$E_{global} = E_{smooth} + \alpha_{fit} E_{fit}, \quad (12)$$

where α_{fit} is introduced to balance the smoothing and fitting energies during the iterative optimization.

Minimizing E_{global} can be regarded as a shape matching problem, which can be solved using a Newton-type solver. We adopt the nonlinear minimization algorithm proposed in [20]. In each iteration, the rigid transformation $\mathbf{T}_i$ can be

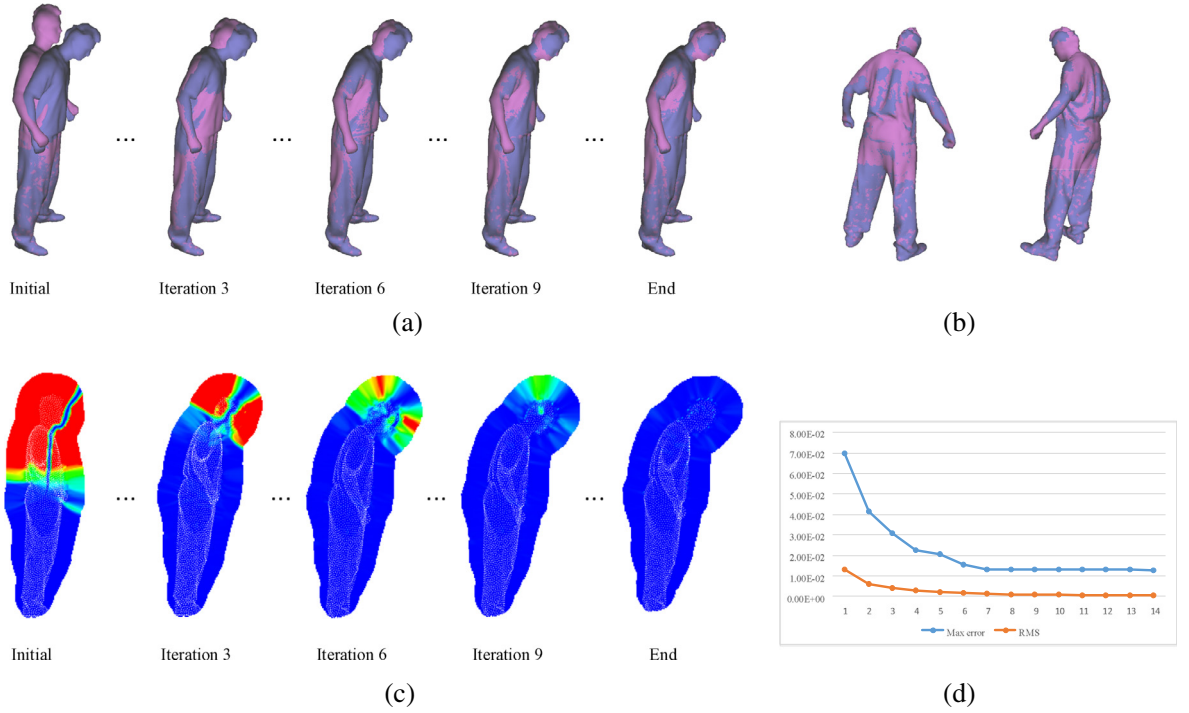

Fig. 7. Synthetic data registration: (a) Deformed surfaces of selected iterations. Initially, the pink model is the source shape and the blue one is the target shape. (b) Registration results from two different views, (c) Error maps of selected iterations, (d) Maximum error and RMS error curves (relative to the bounding cube diagonal). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

linearized to an affine transformation represented by linear and angular velocities $\mathbf{v}_i,\ \boldsymbol{\omega}_i \in \mathbb{R}^3$:

$$\mathbf{T}_i\mathbf{p} \approx \mathbf{A}_i\mathbf{p} := \mathbf{p} + (\boldsymbol{\omega}_i \times \mathbf{p}) + \mathbf{v}_i. \quad (13)$$

Substituting the affine approximations $\mathbf{A}_i$ for the rigid transformations $\mathbf{T}_i$ in Eq. (12) leads to an energy quadratic in the linear and angular velocities $\mathbf{v}_i,\ \boldsymbol{\omega}_i$. Eq. (12) can be minimized by solving a sparse linear system. This process results in the affine transformation for each cell. Finally, the affine transformations are projected onto rigid transformations, and then solved by using SVD decomposition of the covariance matrix.

5.3. Iterative locally rigid, globally non-rigid registration

The locally rigid, globally non-rigid registration of the source and target shapes can finally be obtained by iteratively performing the locally rigid registration and globally non-rigid registration. Each iteration consists of two steps:

- Minimizing Eq. (2) to look for an optimal transformation for each cell, as described in Section 5.1.
- Minimizing Eq. (12) to compute the globally optimal transformations for the cube cells, as described in Section 5.2.

Initially, $\alpha_{fit}^0 = 1$. During the iteration, we increase α_{fit}^{k+1} to $2\alpha_{fit}^k$ at the $k+1$ iteration until $\alpha_{fit} \geqslant 100$. This is because that at each iteration, the locally rigid registration error for each cell is minimized. To begin with, each cell is registered at a local neighborhood in a Gaussian–Newton manner. It is difficult to find a good rigid transformation for the cells with large motion. Therefore, globally smoothness is employed to correct the locally rigid transformations with large fitting error. This allows the rigid fitting error to decrease with each successive iteration. Consequently, the optimization energy will be less and less affected by the global smoothness. These two steps are iteratively computed until convergence. Finally, a robust locally rigid globally non-rigid registration is obtained. Fig. 5 shows the registration progress of a local part of the input meshes.

5.3.1. Space discretization

As mentioned, for simple shapes, E_{smooth} can be evaluated analytically. We choose a space discretization scheme using uniform cubes, i.e., each cell is a uniform cube. The cells located outside the shape do not preserve distance field rigidity during deformation. Therefore, the cells that do not contain the sample points inside or on the surface are not taken into account. The space discretization result for a human body model is shown in Fig. 6.

5.3.2. Surface warping

Once we obtain the transformation $\mathbf{T}_i$ for each cell that maps the cell from source to target shape, we construct a space deformation for the distance field using trilinear interpolation, i.e., interpolate a transformation for each mesh vertex using trilinear interpolation of its eight neighboring cells.

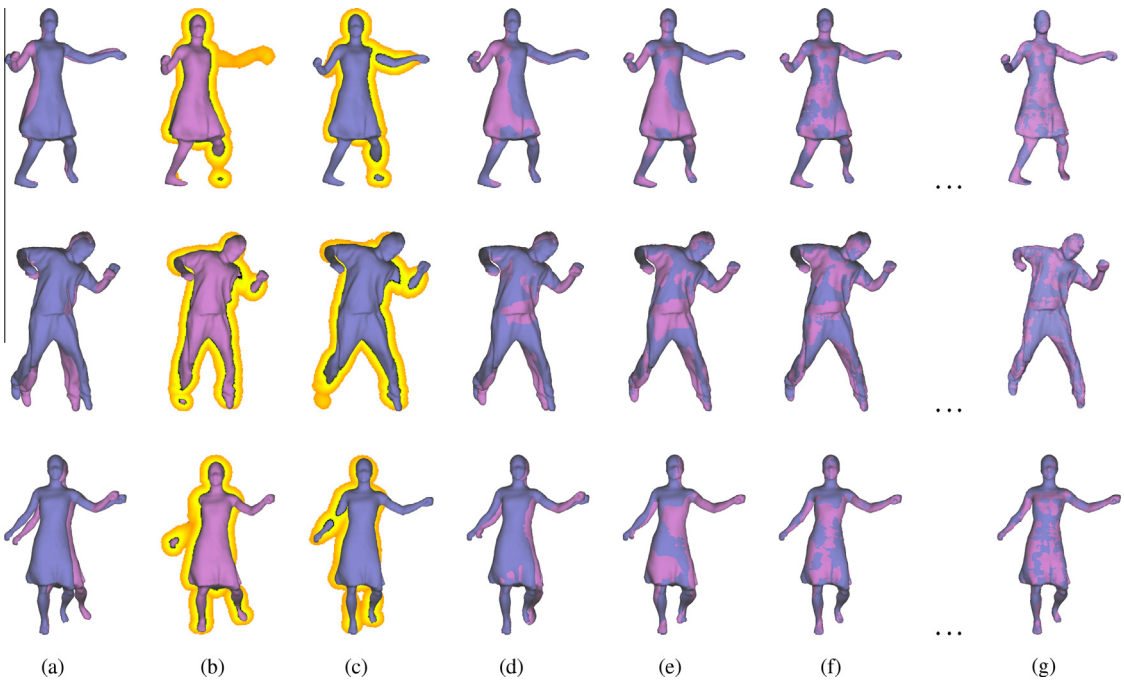

Fig. 8. Real data registration. (a) Source shape (pink) and target shape (blue). (b and c) The SDFs of the source and target shape, respectively. (d) Initial registration. (e and f) Iterations. (g) Final registration. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6. Experiments

6.1. Accuracy

The primary mechanism for evaluating a registration algorithm is its registration accuracy. Several methods have been used to measure the registration error [5,12]. We use a volumetric measurement that is the average distance between the distance fields around the target shape and the deformed source shape.

$$E_{measure} = \sum_{\phi_{A'}(\mathbf{p}) < d_{off}} (\phi_{A'}(\mathbf{p}) - \phi_B(\mathbf{p}))^2 \quad (14)$$

where $\phi_{A'}$ is the distance field of the deformed source shape, and ϕ_B is the distance field of the target shape. $\mathbf{p}$ is the sample points in the space, and d_{off} is an offset value. Only the sample points that satisfy $\phi_{A'}(\mathbf{p}) < d_{off}$ are taken into account. In our implementation, $d_{off} = 20$.

Our non-rigid registration method has been implemented on a platform using core i7 with 3.4 GHz CPU and GTX570 GPU. The proposed method has been tested on both synthetic data and real data. In all experiments, the distance field resolution is $256 \times 256 \times 256$ for the bounding cube of the shape. Each cell is an $8 \times 8 \times 8$ cube.

Fig. 9. Comparison between the results of our method (the first and third rows) and [12] (the second and fourth rows). We produce similar registration results with [12]. However, the computational time is longer than ours.

6.2. Results

6.2.1. Synthetic data

We use a human model from [24] as the source shape, and then we use our deformation model to deform the source shape as a synthetic target shape. Our method registers the two shapes in 14 iterations. The deformed mesh and the registration error are shown in Fig. 7. As demonstrated, our method has the capacity to rapidly converge after several iterations. The final registration error for the source and target distance fields is listed in Table 1.

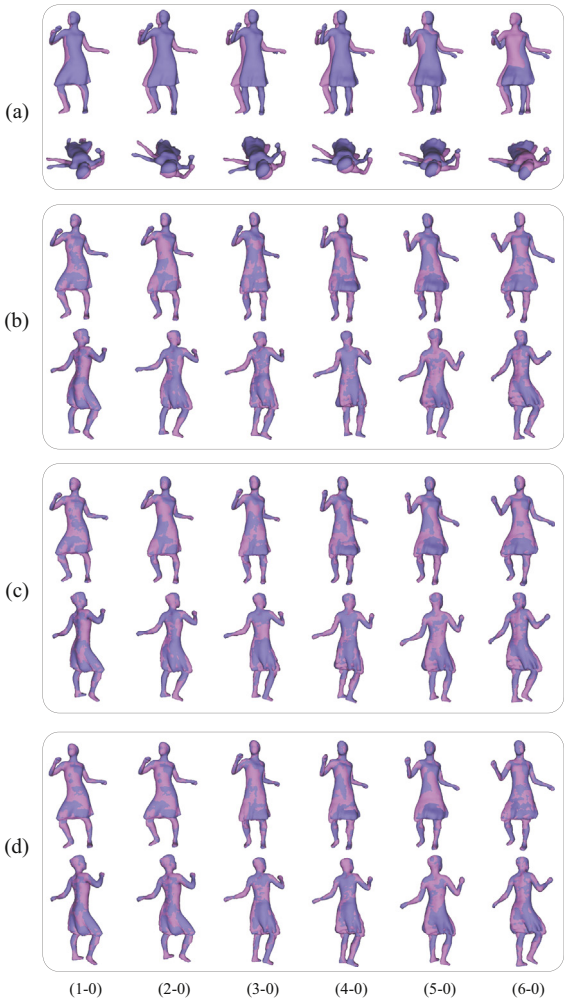

Fig. 10. Registration results of six pairs of meshes with different deformations and different ways to assign α_{fit}. Left to right: mesh1–mesh0, mesh2–mesh0, mesh3–mesh0, mesh4–mesh0, mesh5–mesh0, mesh6–mesh0. We use the same target mesh (blue) in these six pairs to demonstrate the robustness of our algorithm on different deformations. Top to bottom: (a) Input meshes from two points of view. (b) Registration results from two viewpoints by assigning $\alpha_{fit}^{k+1} = 2\alpha_{fit}^{k}$. (c) Registration results from two viewpoints by assigning $\alpha_{fit}^{k+1} = 1.2\alpha_{fit}^{k}$. (d) Registration results from two viewpoints by assigning $\alpha_{fit}^{k+1} = \alpha_{fit}^{k} + 1$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6.2.2. Real data

We tested our method on several datasets (street dance, samba, swing) from [24] as well. A series of examples is shown in Fig. 8. Our method registers the source and target shapes accurately in just over 15 iterations. Fig. 9 includes a comparison to [12] and depicts various views of the deformed mesh from different viewpoints. The results demonstrate that our method is capable of registering a wide range of data accurately.

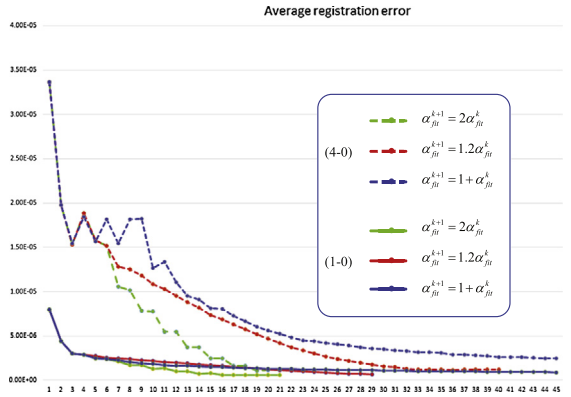

Fig. 11. The convergences with different methods of assigning α_{fit}. For both pairs of meshes with small deformation (mesh1–mesh0) and large deformation (mesh4–mesh0), our algorithm converges fast with $\alpha_{fit}^{k+1} = 2\alpha_{fit}^{k}$.

6.2.3. Statistics

For the tested datasets, the statistics are listed in Table 1. We can see that our algorithm converges in less than 25 iterations for various data sets. Ref. [12] produces similar registration results but the computational time is longer than ours.

6.3. Large deformation

Cases with large deformations often present significant hurdles for conducting non-rigid registration. By using the locally rigid and globally non-rigid registration framework, our method can successfully register two shapes with large deformations and no significant topology change. Three sets of results are shown in Figs. 10, 12, 13.

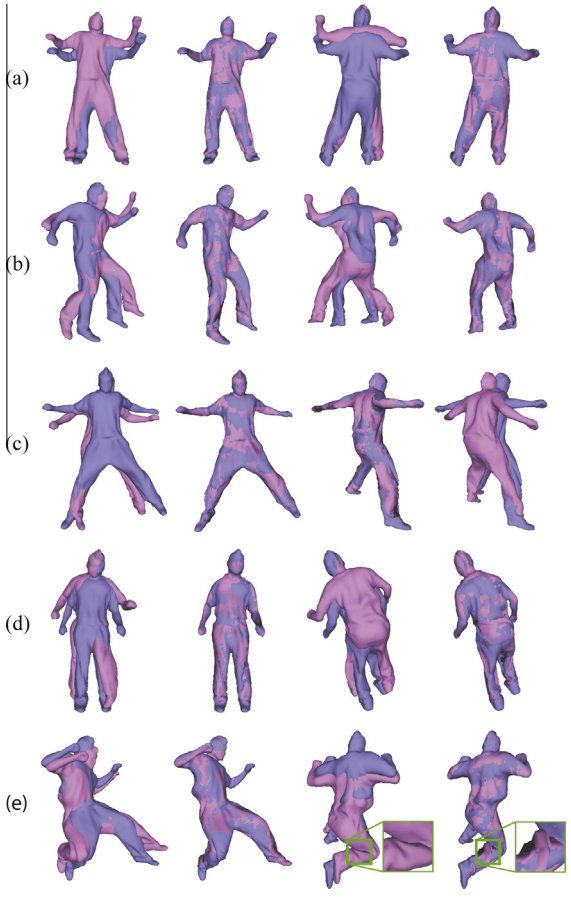

Fig. 12. Registration results of five pairs of meshes in the bouncing sequence [24]. Each row shows the two input meshes and the registrations results from two different points of view. The last row shows a limitation of our method. The left two columns are input meshes and registered result from a point of view. The right two columns are input meshes and registered result from another point of view.

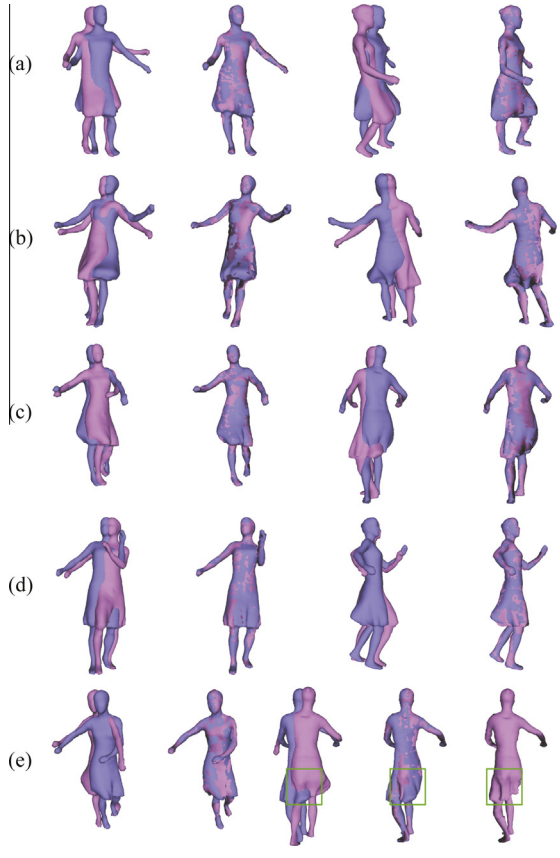

Fig. 13. Registration results of five pairs of meshes in the samba sequence [24]. Each row shows the two input meshes and the registrations results from two different points of view. The last row shows a limitation of our method. The left two columns are input meshes and registered result from a point of view. The right two columns are input meshes and registered result from another point of view.

6.3.1. Assigning α_{fit}

Our algorithm is robust for large deformations with different parameters. We have tested our algorithm in different ways to assign the α_{fit} in Eq. (12) to six pairs of meshes from the swing sequence [24]. We use the same target mesh in these six pairs to demonstrate the robustness of our algorithm on different deformations.

The registration results are shown in Fig. 10. Our algorithm generates convincing results of registering meshes with large ranges of deformations. The first two columns are meshes with smaller deformations, and the other four columns exhibit much larger deformations between the source and target meshes, especially at the arms, skirts, and legs.

The iterative progress of our algorithm as applied to different pairs of meshes is shown in Fig. 11. We can see that our algorithm converges at a very small registration error with different parameters. In comparison, assigning $\alpha_{fit}^{k+1} = 2\alpha_{fit}^{k}$ leads to the fastest convergence, while the minimum energy remains similar.

6.3.2. ε in Eq. (11)

In Eq. (11), each cell is assigned a weight in the global optimization step to suppress the propagation of the transformations with large fitting errors. ε is added so that the transformations with large registration errors are not neglected. We show four registration results of the same meshes using different ε in Fig. 14, and the convergence in Fig. 15. The results show that $\varepsilon = 0$, $\varepsilon = 0.001$, and $\varepsilon = 0.01$ generate similar results, while a larger ε leads to very slow convergence. Our algorithm also works well without ε.

6.3.3. SDF resolution

Generally speaking, a larger SDF resolution will increase the registration accuracy. However, the computational complexity is $O(n^3)$ of the SDF resolution. We compare the registration results of two meshes with SDF resolution $256 \times 256 \times 256$ and $128 \times 128 \times 128$ in Fig. 16. The comparison shows how larger SDF resolutions cause the registered mesh to cover additional spaces on the target mesh. However, the computational time of $256 \times 256 \times 256$ SDF is nearly eight times of that of $128 \times 128 \times 128$ SDF.

6.4. Discussion

Our non-rigid registration algorithm uses a locally rigid, globally non-rigid registration method. The locally rigid registration step provides a rigid transformation for each cell, which consists of several voxels. Our cell-based method is able to generate much more stable results than voxel-based or point-based methods. Meanwhile, the

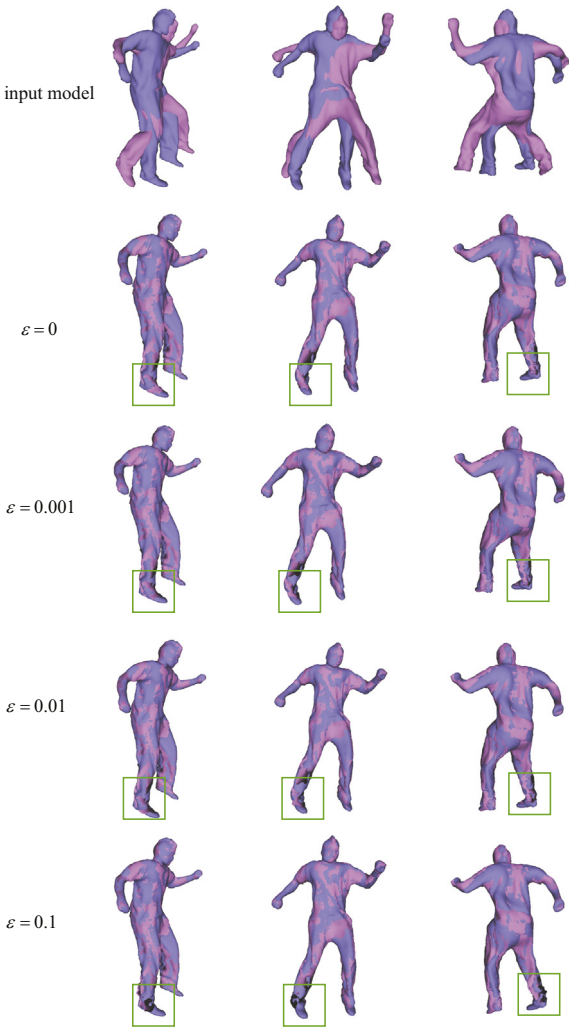

Fig. 14. Registration results of the same meshes using $\varepsilon = 0$, $\varepsilon = 0.001$, $\varepsilon = 0.01$, and $\varepsilon = 0.1$, from top to bottom, respectively.

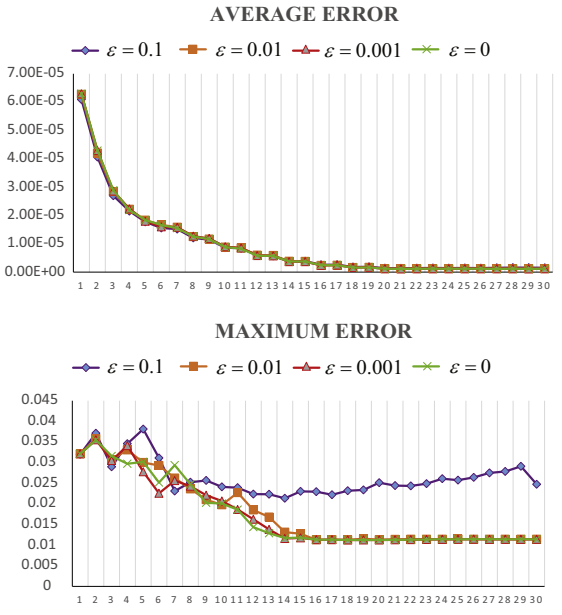

Fig. 15. Comparison of registration progress with different ε used in Eq. (11).

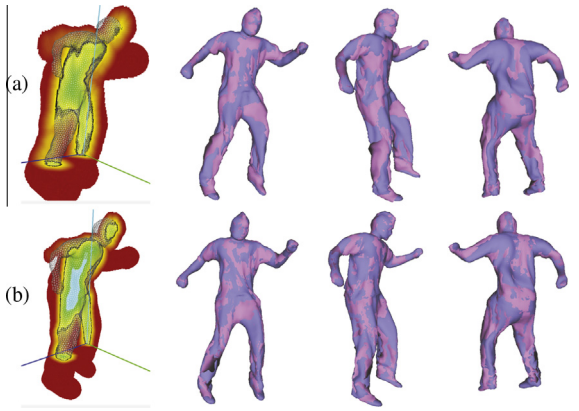

Fig. 16. Comparison between difference SDF resolutions. (a) $128 \times 128 \times 128$. (b) $256 \times 256 \times 256$. Left to right: SDF of the target mesh; registration results from three points of view.

cell-based locally rigid registration method provides a confidence to each cell. With the confidence of each cell, weighted transformations can be used in globally non-rigid registration, thereby reducing the influence of mismatched cells. We show a series of results to demonstrate the robustness of our algorithm to large deformations. Even though mismatches exist early on, the globally non-rigid registration with weighted cell transformations is able to regularize the global deformation after several iterations.

The proposed method is very efficient due to the volumetric deformation model's reduction of the computational cost. This is accomplished by downsizing the scale of the volumetric graph in the deformation model of the optimization. With the weighted cell transformation, the computational time of the globally non-rigid registration optimization is also reduced. The results in Table 1 reveal the efficiency of our method.

6.5. Limitations and future work

Our non-rigid registration method relies on the assumption of locally rigid, globally non-rigid shapes. As a result, our approach has difficulty handling cases where the local geometry changes dramatically, especially if the topology of the source mesh is very different from the target mesh. This can be seen in Fig. 12(e), where our algorithm was unable to compute a reasonable registration around the knee of the input model. However, this local error does not affect the other parts of the mesh. User interaction could be involved in the future to help improve the registration results.

For complex shapes with large changes in convexity, like the fine details on the skirt in Fig. 13(e), our method fails to align the mesh accurately. The distribution of the sampling points in a cell like this region is too complex to lead a good descent direction for the Gaussian–Newton method. One option is to leverage the computational complexity and registration accuracy by using a hierarchical scheme, which refines the registration on different levels of details.

In cases with extremely large deformations, the region-matching result may be entirely incorrect, or the deformation model may not regularize the global deformation accurately. Good initial alignment and explicit correspondences can be used to rectify this problem. In the future, we would like to address these issues and optimize the performance of the system. A hierarchical resolution for the distance field and a parallelized algorithm will provide significant increases in speed.

7. Conclusion

We introduce a robust and efficient volumetric registration algorithm for non-rigid shapes. Through matching the source and target signed distance fields, our method preserves the local geometry by keeping the local rigidity of the distance field, and it achieves a natural global deformation for the volumetric model. The experimental results on different datasets have demonstrated that our method gives an accurate registration for non-rigid shapes.

Acknowledgments

We would like to thank the anonymous reviewers for their generous reviews that shaped the direction of the paper. This work was supported by the National Natural Science Foundation of China (NSFC) under Nos. 61472377 and 61331017, and the Fundamental Research Funds for the Central Universities under No. WK2100060011. They are National Natural Science Foundation of China (61222206) and One Hundred Talent Project of the Chinese Academy of Sciences.

References

[1] P.J. Besl, N.D. McKay, A method for registration of 3-d shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1992) 239–256.
[2] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, A. Fitzgibbon, Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera, in: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST '11, ACM, New York, NY, USA, 2011, pp. 559–568, http://dx.doi.org/10.1145/2047196.2047270.
[3] R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohli, J. Shotton, S. Hodges, A. Fitzgibbon, Kinectfusion: real-time dense surface mapping and tracking, in: Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 127–136, http://dx.doi.org/10.1109/ISMAR.2011.6092378.
[4] B. Curless, M. Levoy, A volumetric method for building complex models from range images, in: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96, ACM, New York, NY, USA, 1996, pp. 303–312, http://dx.doi.org/10.1145/237170.237269.
[5] H. Li, R.W. Sumner, M. Pauly, Global correspondence optimization for non-rigid registration of depth scans, in: Proceedings of the Symposium on Geometry Processing, SGP '08, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2008, pp. 1421–1430.
[6] R.W. Sumner, J. Schmid, M. Pauly, Embedded deformation for shape manipulation, ACM Trans. Graph. 26 (2007).
[7] M. Dou, H. Fuchs, Temporally enhanced 3D capture of room-sized dynamic scene with commodity depth cameras, IEEE Virtual Real. (2014).
[8] H. Li, B. Adams, L.J. Guibas, M. Pauly, Robust single-view geometry and motion reconstruction, ACM Trans. Graph. 28 (2009) 175:1–175:10.
[9] W. Chang, M. Zwicker, Range scan registration using reduced deformable models, Comput. Graph. Forum 28 (2009) 447–456.

[10] W. Chang, M. Zwicker, Global registration of dynamic range scans for articulated model reconstruction, ACM Trans. Graph. 30 (2011) 26:1–26:15.
[11] M. Wand, B. Adams, M. Ovsjanikov, A. Berner, M. Bokeloh, P. Jenke, L. Guibas, H.-P. Seidel, A. Schilling, Efficient reconstruction of nonrigid shape and motion from real-time 3d scanner data, ACM Trans. Graph. 28 (2009) 15:1–15:15.
[12] Z.-Q. Cheng, W. Jiang, G. Dang, R.R. Martin, J. Li, H. Li, Y. Chen, Y. Wang, B. Li, K. Xu, S. Jin, Non-rigid registration in 3d implicit vector space, in: Proceedings of the 2010 Shape Modeling International Conference, SMI '10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 37–46, http://dx.doi.org/10.1109/SMI.2010.21.
[13] K. Fujiwara, K. Nishino, J. Takamatsu, B. Zheng, K. Ikeuchi, Locally rigid globally non-rigid surface registration, in: Proceedings of the 2011 International Conference on Computer Vision, ICCV '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 1527–1534, http://dx.doi.org/10.1109/ICCV.2011.6126411.
[14] M.W. Jones, J.A. Baerentzen, M. Sramek, 3D distance fields: a survey of techniques and applications, IEEE Trans. Visual. Comput. Graph. 12 (2006) 581–599.
[15] M. Botsch, M. Pauly, M. Wicke, M. Gross, Adaptive space deformations based on rigid cells, Comput. Graph. Forum 26 (2007) 339–347.
[16] O. Sorkine, M. Alexa, As-rigid-as-possible surface modeling, in: Proceedings of the Fifth Eurographics Symposium on Geometry Processing, SGP '07, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2007, pp. 109–116.
[17] K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo, H.-Y. Shum, Large mesh deformation using the volumetric graph Laplacian, ACM Trans. Graph. 24 (2005) 496–503.
[18] T.W. Sederberg, S.R. Parry, Free-form deformation of solid geometric models, SIGGRAPH Comput. Graph. 20 (1986) 151–160.
[19] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, H.-P. Seidel, Laplacian surface editing, in: Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '04, ACM, New York, NY, USA, 2004, pp. 175–184, http://dx.doi.org/10.1145/1057432.1057456.
[20] M. Botsch, M. Pauly, M. Gross, L. Kobbelt, Primo: coupled prisms for intuitive surface modeling, in: Proceedings of the Fourth Eurographics Symposium on Geometry Processing, SGP '06, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2006, pp. 11–20.
[21] T.-T. Cao, K. Tang, A. Mohamed, T.-S. Tan, Parallel banding algorithm to compute exact distance transform with the GPU, in: Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D '10, ACM, New York, NY, USA, 2010, pp. 83–90, http://dx.doi.org/10.1145/1730804.1730818.
[22] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Proceedings of the 7th International Joint Conference on Artificial Intelligence, IJCAI'81, 1981, pp. 674–679.
[23] H. Pottmann, Q.-X. Huang, Y.-L. Yang, S.-M. Hu, Geometry and convergence analysis of algorithms for registration of 3d shapes, Int. J. Comput. Vision 67 (2006) 277–296.
[24] D. Vlasic, I. Baran, W. Matusik, J. Popović, Articulated mesh animation from multi-view silhouettes, ACM Trans. Graph. 27 (2008) 97:1–97:9.