Non-rigid Registration in 3D Implicit Vector Space

Zhi-Quan Cheng[1], Wei Jiang[1], Gang Dang[1], Ralph R. Martin[2], Jun Li[1], Honghua Li[1]
Yin Chen[1], Yanzhen Wang[1], Bao Li[1], Kai Xu[1], Shiyao Jin[1]
[1]PDL Laboratory, National University of Defense Technology, P. R. China
[2]Schoool of Computer Science & Informatics, Cardiff University, Wales, UK

***Abstract*—We present an implicit approach for pair-wise non-rigid registration of moving and deforming objects. Shapes of interest are implicitly embedded in the 3D implicit vector space. In this implicit embedding space, registration is performed using a global-to-local framework. Firstly, a non-linear optimization functional defined on the vector distance function is used to find the global alignment between shapes. Secondly, an incremental cubic B-spline free form deformation is used to recover the non-rigid transformation parameters. Local non-rigid registration is posed in terms of minimising an energy functional, for which we give a closed-form linear system and solve it using an improved iterative Gauss-Seidel method. Our approach can consistently produce smooth and continuous registration fields, and correctly establish dense one-to-one correspondences. It can naturally deal with both open partial and closed shapes, and imperfect models with gaps and noise, through its use of the implicit vector representation. Experimental results on several datasets demonstrate the robustness of the proposed method.**

***Keywords*—Non-rigid registration; global alignment; dense correspondence; vector distance function; implicit vector space.**

1. Introduction

Registration is a fundamental problem in shape acquisition and reconstruction. However, forming a coherent model of an object from several datasets is nontrivial, and the approach to doing so depends on the shape representation, as well as the registration strategy and nature of the transformations which map corresponding points between datasets. When the object evolves non-rigidly over time, the description of the transformation is more complicated than in the rigid case. Practical algorithms must also face other challenges—for example, scanned data is noisy, and typically has gaps arising from occlusion during data acquisition.

Shape representation is an important choice when considering approaches to registration. Existing methods perform 3D non-rigid registration based on point set [1], [2], [3], [4], [5], parametric description [6], [7], and volumetric representation [8], [9], [10]. Point set is used as a prime choice for the non-rigid registration, since scanners directly generate sample points. A known limitation of point set representation is that, given two shapes to be registered each represented by a point cloud, the two point sets are not sampled at corresponding locations. This can lead to inherent inconsistencies in the two point sets, causing problems when attempting to find point correspondences between the two shapes. Parametric representation [6], [7] may also be used for non-rigid registration. However, finding explicit parameterization of one shape in 3D dimensions is hard in most situations. Thus, such methods mainly focus on open partial shapes and do not handle whole shapes. Volume representation [8], [9], [10] is another choice for registration. However, for the open partial shapes, it is a challenging task to generate appropriate volumes, and such methods also must assume that any gaps in the data are not large.

Implicit representation has already been used in 2D registration, in the level set method [11], since it has the advantage that one can perform numerical computations on shapes using a fixed Cartesian grid without parameterizing them. Implicit representations can handle both open partial and closed shapes, and can do so stably and robustly in the presence of shape perturbations and holes. However, the conventional implicit signed distance function (SDF) representation has difficulties in defining inside/outside for open shapes, particularly near the boundaries, similarly to volumetric representation [8], [9]. Implicit representation has been used for tasks such as editing [12], shape interpolation [13], [14], and analysis [15], but they do not provide correspondences between two models. Recently, a particular implicit vector distance function (VDF) representation [16], [17] has been proposed which is more flexible than the traditional SDF. In 2D, [17] solves the problem of pairwise non-rigid registration using an implicit VDF representation, for both whole and partial shapes. However, the extension from 2D to 3D is nontrivial, since transformations are more complex in 3D. For example, 3D rotations do not commute and cannot be linearized as in the 2D case. In the paper, a 3D VDF-based registration algorithm is proposed, which can automatically find dense correspondences between different poses of a dynamic object undergoing moving and deforming transformation, by warping the relevant 3D implicit vector spaces. By taking the advantage of VDF representation, our approach defines appropriate energy functions which are optimized to recover the transformation parameters in both rigid and non-rigid registration schemes.

Our approach requires the fairly mild assumption that the object does not change its genus during the registration. We do not need to assume that the motion

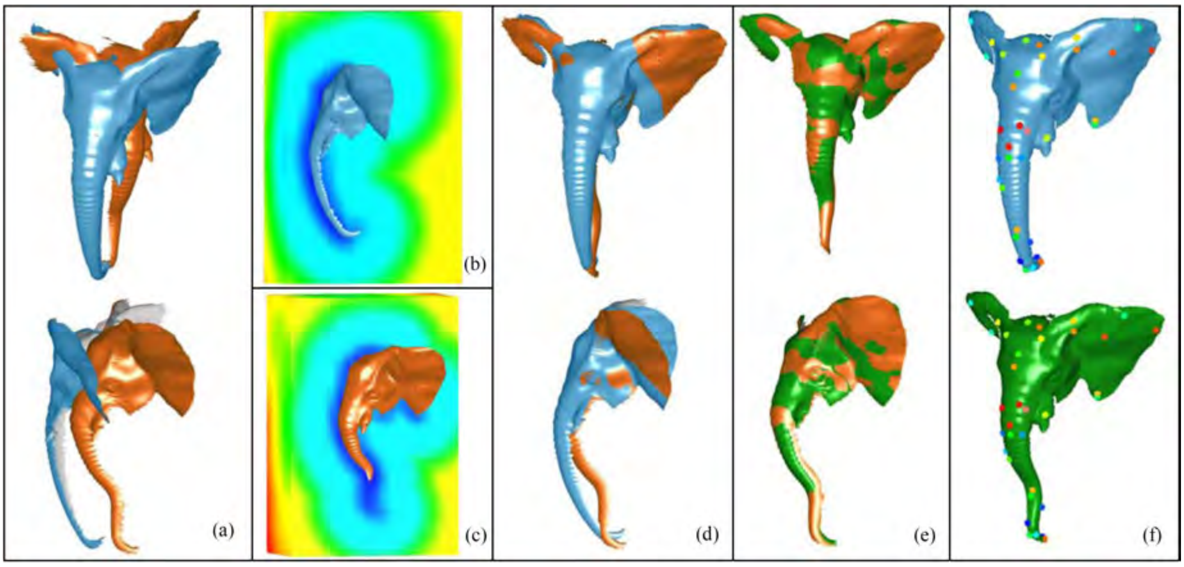

Fig. 1. Our global-to-local registration framework in the 3D implicit vector space. (a) Initial poses, front and side views (blue: source shape; brown: target shape). (b) Cross-section of the source vector space displayed as the L^2 Euclidean metric of the VDF, points on the shape overlapping the zero level set. (c) Target vector space. (d) Global alignment after nonlinear optimization. (e) Result after local non-rigid registration using IFFD; the transformed source shape (green) is overlaid on the target shape (brown). (f) Dense correspondences: 34 randomly selected corresponding points shown with matching colors.

is globally or locally rigid, nor do we have any assumptions on the topology. In contrast to many previous methods, we do not require additional information or conditions, such as marker correspondences [18], [19], manual segmentation [20], small-scale differences between shapes [1], [2], [4], [8], or prior templates [21].

We perform non-rigid registration using a global-to-local framework, illustrated in Figure 1. The shapes of interest are represented in an implicit form, embedded in the vector space (displayed as the Euclidean metric in Figures 1(b), 1(c)). Global alignment (see Figure 1(d)) is achieved by finding the rigid rotation and translation parameters which are the solution to a nonlinear sum-of-squared-differences (SSD) functional. Local non-rigid registration (see Figure 1(e)) is carried out by finding parameters of a cubic B-spline-based incremental free form deformation (IFFD) [22], by minimizing another SSD functional. Unlike many traditional non-rigid registration methods, we do not use nonlinear gradient descent method to estimate the IFFD control point positions in implicit vector spaces. Instead, we approximate the functional for non-rigid registration using a Taylor series expansion, leading to a linear system with a closed-form solution.

The main contributions of our work include:

- We give a practical implicit non-rigid registration approach for moving and deforming shapes, which establishes dense one-to-one correspondences (see Figure 1(f)) between points of source and target shapes. It is robust to the presence of gaps and noise in the data, and it can simultaneously handle both open partial and closed shapes.
- Based on the VDF representation, global alignment is defined as a nonlinear optimization problem. Global VDF alignment can accurately find the rigid motion between shapes. It can not only work well for non-deforming objects, but also find the underlying rigid transformation between salient parts of deformable objects.
- A closed-form non-rigid registration equation is derived from the VDF representation, which can readily be solved using an improved iterative Gauss-Seidel method. This has advantages over typical gradient descent optimization methods which need more careful initialization and are more time consuming.

The remainder of this paper is organized as follows. After related work is addressed in Section 2, the 3D VDF representation is introduced in Section 3. Section 4 considers global rigid alignment, followed by local non-rigid registration in Section 5. Experimental results, limitations, and further work are discussed in Section 6, and conclusions drawn in Section 7.

2. Related Work

Due to space limitations, we mainly focus on previous work on 3D non-rigid shape registration, as well as relevant 2D implicit registration.

For the non-rigid registration problem, transformations are used to match a source shape with the target shape, allowing for differences in pose. The transformation includes not only a global rigid transformation (rotation, translation) but also local transformations that deform a shape locally and non-rigidly. The most widely approach to find the transformation is to use non-rigid variants of the iterative closest point (ICP) technique [23], for example [18], [24], [25], [26], [27], [28], [29]. Non-rigid ICP algorithm is able to find a smooth deformation that aligns the shapes, but are only suitable for small-scale differences in shape. In addition to difficulties caused by the presence of holes, searching for nearest corresponding points is time-consuming for large 3D datasets.

Other as-rigid-as-possible deformation models have also been applied to non-rigid registration [2], [4], [5], [6], [30]. To find the transformation, isometry is used as a general basic criterion both for registering the input models [5], [9], [26], [31], [32], [33] and for determining the deformation [3], [10], [29], [30], [34], [35], [36], [37], [38], [39], [40], [41]. In this paper, we explicitly separate the global and local transformation, deriving the rigid and non-rigid transformation via warping VDF fields. Particularly, by using an IFFD model [22] to represent the local deformation, we are thus freed from the restrictions of the as-rigid-as-possible assumption.

Different registration strategies have been used to recover the optimal transformation parameters given a shape representation and a transformation model. Existing approaches belong to two categories. The first directly search for correspondences between shapes, and then estimate the transformation parameters using the correspondences [5], [19], [32], [34], [35], [41], [42], [43], [44], [45], [46]. Correspondences can be explicitly given [18], [19], [41], or determined from salient geometric features, such as curvature [5], slippage [32], or meshDOG [42]. They may also come from other intrinsic domains, such as Möbius space [36], (global) multidimensional scaling [29], [37], [38], or heat kernel signatures [39], [40]. The second class of approach seeks to recover the optimal transformation parameters through optimization of some global energy functional [1], [2], [4], [5], [6], [8], [10], [21], [30], [47]. As pointed out in [5], [6], global optimization methods generally could give stable results. Our approach falls into the latter category.

2D implicit registration is also relevant, where the concept of VDF was first used. VDF is a further extension of the conventional SDF, which has been widely applied to 2D image registration [48] and segmentation [49]. Typical dissimilarity measurements adopted by SDF non-rigid registration methods are mainly defined based on SSD [50], [51], mutual information [52], and statistical analysis [53]. A variational scheme based on VDF has been described for finding 2D transform parameters for both global and local registration [17]. However, using such an approach in 3D is not a direct extension of the 2D case, due to the extra complexity of the transformation calculation.

3. Vector distance function representation

This section briefly reviews the basic concepts and algebra of the VDF [16]. For a surface model $\mathcal{M}$ in 3D, the VDF is given by $f : \mathcal{R}^3 \to \mathcal{R}^3$, and defined as

$$f(\mathbf{x}) = \mathbf{x} - \mathbf{p}_{\mathcal{M}}(\mathbf{x}), \qquad \forall \mathbf{x} \in \Omega, \tag{1}$$

where Ω is the embedding space, and $\mathbf{p}_{\mathcal{M}}(\mathbf{x})$ is the point on $\mathcal{M}$ with minimum Euclidean distance to $\mathbf{x}$. In this representation, it is easy to catch that, firstly, the shape of interest is represented as the zero level set of the VDF: shape points satisfy $\|f\| = 0$, where $\| \cdot \|$ is the L^2 Euclidean metric. Secondly, for the closed model with zero level set, there is a relationship between the VDF and the traditional SDF:

$$d(\mathbf{x}) = sign(\mathbf{x}) \|f(\mathbf{x})\|, \tag{2}$$

where $d(\mathbf{x})$ is the distance field, and $sign(\mathbf{x})$ is positive outside the shape, and negative inside it.

The VDF is more flexible than the SDF when dealing with arbitrary topology [16]. The definition of SDF is undefined at the boundary for open shapes, which prevents the effective use of registration formulations minimizing an energy function of level set differences. The VDF case does not suffer from this problem, and can naturally cope with both open and closed shapes. Particularly, for open shapes, there is no issue of sign.

4. Global alignment in 3D vector space

In this section, we explain how the VDF is used to recover the global alignment (rotation and translation) for non-deforming and deformable shapes. After a theoretical problem formulation is deduced in Sections 4.1, the global alignment is solved in Section 4.2, then experimental results are presented in Section 4.3.

4.1 Global VDF alignment formulation

Given a source shape $\mathcal{S}$ and a target $\mathcal{T}$, consider a global transformation $\mathbf{A}$ which rigidly aligns $\mathcal{S}$ to $\mathcal{T}$. $\mathbf{A}$ has parameters $\mathbf{\Phi}$ comprising a rotation $\mathbf{R}$ and translation $\mathbf{T}$. For any point $\mathbf{x}$ in the space, the transformation $\mathbf{A}$ applied to $\mathbf{x}$ can be written as $\mathbf{A}(\mathbf{\Phi}; \mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{T}$. From the definition of VDF, applying the transformation to any pair of points $\mathbf{x}, \mathbf{p}_{\mathcal{S}}(\mathbf{x}) \in \Omega_{\mathcal{S}}$ results in a corresponding pair of points $\mathbf{x}', \mathbf{p}_{\mathcal{T}}(\mathbf{x}') \in \Omega_{\mathcal{T}}$. We find that:

$$\begin{aligned} f_{\mathcal{T}}(\mathbf{A}(\mathbf{\Phi}; \mathbf{x})) &= \mathbf{x}' - \mathbf{p}_{\mathcal{T}}(\mathbf{x}') \\ &= \mathbf{R}\mathbf{x} + \mathbf{T} - (\mathbf{R}\mathbf{p}_{\mathcal{S}}(\mathbf{x}) + \mathbf{T}) \\ &= \mathbf{R}(\mathbf{x} - \mathbf{p}_{\mathcal{S}}(\mathbf{x})) \\ &= \mathbf{R} f_{\mathcal{S}}(\mathbf{x}). \end{aligned} \tag{3}$$

Thus, the VDF transformation is rigid-invariant under global alignment.

Finding the optimal global alignment for two (non-deforming) surfaces can be computed by finding the transformation $\mathbf{A}$ between $\mathcal{S}$ and $\mathcal{T}$, which minimizes a dissimilarity measurement, similar to the 2D case [17]. The dissimilarity can be defined as

$$\mathbf{r}_d(\mathbf{\Phi}; \mathbf{x}) = \mathbf{R} f_{\mathcal{S}}(\mathbf{x}) - f_{\mathcal{T}}(\mathbf{A}(\mathbf{\Phi}; \mathbf{x})). \tag{4}$$

For simplicity, $\mathbf{r}_d(\mathbf{\Phi}; \mathbf{x})$ is abbreviated as $\mathbf{r}_d$ after now. By using all points of a sampling grid in the vector space, a global SSD energy functional can be defined, which should be minimized:

$$E_g(\mathbf{\Phi}) = \int_{\Omega_{\mathcal{S}}} \mathbf{r}_d^T \mathbf{r}_d \, d\mathbf{x}. \tag{5}$$

In this formulation, the point-correspondence problem is converted into a nonlinear optimization to minimize the SSD. In fact, this approach is also effective for deformable shapes. Generally, there is a dominant rigid alignment between deformable shapes which are non-rigid deformed in some local regions. This approach works well in practice because the use of VDFs downgrades the importance of local deformations. So it is insensitive to outliers. For the same reason, the global VDF alignment is robust to noise in measured data.

The vector distance fields derived from $\mathcal{S}$ and $\mathcal{T}$ are defined over all space. To deal with this technical limitation, and to decrease computational complexity, we consider regions defined by two equal-distance neighborhoods around the shapes. That is sampling is limited to a small band ε around $\mathcal{S}$ and $\mathcal{T}$, formed by inward and outward offsetting. The strategy is reasonable, since far away sample points have negligible influence. We set ε to 10 times the interval size of the sampling grid (with $70 \times 70 \times 70$ resolution) by default for all examples. Formally:

$$E_g(\mathbf{\Phi}) = \xi_\varepsilon^2(f_{\mathcal{S}}, f_{\mathcal{T}}) E_g(\mathbf{\Phi}), \tag{6}$$

where ξ_ε is the binary function given by

$$\xi_\varepsilon(f_{\mathcal{S}}, f_{\mathcal{T}}) = \begin{cases} 0 & \text{if} \min(||f_{\mathcal{S}}||, ||f_{\mathcal{T}}||) > \varepsilon \\ 1 & \text{otherwise} \end{cases}. \tag{7}$$

4.2 Nonlinear optimization solution

The parameters of the global transformation $\mathbf{A}$ comprise a rotation $\mathbf{R}$ and a translation $\mathbf{T}$ which we wish to find. In the 2D case [17], $\mathbf{R}$ can be found directly, due to linearity of 2D rotation. However, finding $\mathbf{R}$ is a nonlinear problem in 3D, and careful choice of representation is important in solving this problem. We use quaternions to estimate the rotation, following the approach in [54]. In our approach we first relax the problem to find an optimal linear transformation, and then extract the rotation part from it. Our overall framework follows Algorithm 1.

Algorithm 1 Nonlinear optimization of global VDF alignment

```
repeat
    initialize $\mathbf{\Phi}$
    while (energy decreases AND (iterations $< maxInIter$))
        use $\mathbf{J_r}$ to update estimate of $\mathbf{\Phi}$
    obtain rotation $\mathbf{R}$ from $\mathbf{q}$
    transform $\mathcal{S}$ by $(\mathbf{R}, \mathbf{T})$
until (energy is low enough OR (iterations $< maxOutIter$))
```

Our global VDF alignment method employs the Levenberg-Marquardt (LM) method [55] in an inner loop to minimize the nonlinear energy given in Equation 6; there are 7 unknown parameters. The initialization for $\mathbf{\Phi} = (\mathbf{q}, \mathbf{T})$ is $\mathbf{q} = (1, 0, 0, 0)$ and $\mathbf{T} = (0, 0, 0)$; the former is the quaternion representing the rotation, and the latter the translation. Each LM iteration solves a linearized problem to improve $\mathbf{\Phi}$, by using the Jacobian $\mathbf{J_r}$ of $\mathbf{r}_d$ calculated as follows:

$$\begin{aligned} \frac{\partial \mathbf{r}_d}{\partial \mathbf{q}} &= [\nabla_{\mathbf{q}} \mathbf{R} f_{\mathcal{S}}(\mathbf{x}) - \nabla f_{\mathcal{T}}^T(\mathbf{A}(\mathbf{\Phi}; \mathbf{x}))] \nabla_{\mathbf{q}} \mathbf{A}(\mathbf{\Phi}; \mathbf{x}) \\ \frac{\partial \mathbf{r}_d}{\partial \mathbf{T}} &= -\nabla f_{\mathcal{T}}^T(\mathbf{A}(\mathbf{\Phi}; \mathbf{x})) \nabla_{\mathbf{T}} \mathbf{A}(\mathbf{\Phi}; \mathbf{x}). \end{aligned} \tag{8}$$

The process is repeated until either a maximum number ($maxInIter$, set to 30) of iterations has been performed, or the change in the energy function $E_g(\mathbf{\Phi})$ is insignificant. In our experiments, it usually takes about 20 iterations for the change in energy to be less than $10^{-4}(1 + E_g(\mathbf{\Phi}))$.

Once the LM optimization has converged, we perform polar decomposition on the corresponding matrix of $\mathbf{q}$ to obtain the rotation $\mathbf{R}$. Finally, the rotation $\mathbf{R}$ and translation $\mathbf{T}$ are applied to update the pose (orientation and position) of $\mathcal{S}$.

Just like any nonlinear optimization, our system converges to a local optimum which may not represent globally best solution. We employ an outer iterative improvement procedure to find a better local minimum by restarting the optimizer from the updated pose of $\mathcal{S}$. After several iterations, $\mathcal{S}$ is gradually aligned with $\mathcal{T}$. The outer iteration is terminated when reaching a maximal number ($maxOutIter$, set to 10), or the final change in energy is less than $10^{-6}(1 + E_g(\mathbf{\Phi}))$.

4.3 Global Alignment Experiments

Results of performing global VDF alignment for 3D shapes are given in Figures 1(c), 2, 3, 4(b), 6(b). Different poses of a non-deforming shape are rigidly oriented and translated in Figure 2, and registration of different motions of a deformable shape are shown in Figure 3. The results are compared with an efficient ICP variant algorithm [56]. For the tetrahedron object rotated 45$^{\text{o}}$ along one given axis as shown in Figure 2, [56] outputs the input as the final result, since that pose corresponds to a local minimal for its SSD energy function. In contrast to [56], the parameters of the global transformation are correctly recovered by our approach. Figure 4 shows that [56] works poorly for deformable shapes—see examples of the person's legs and arms—since it aligns shapes by determining assumed correspondences between points in the source and target shapes based on nearest distances. Our global VDF alignment algorithm provides more reasonable results, as we find the main rigid transformation between the salient parts of objects. Our results for deformable shapes demonstrate that: our global VDF alignment approach does more than just finding the most obvious rigid transformation for non-deforming models, it finds the rigid motion between salient similar parts. Therefore, our approach could also effectively solve the salient partial similarity measurement problems for deformable objects, like [29].

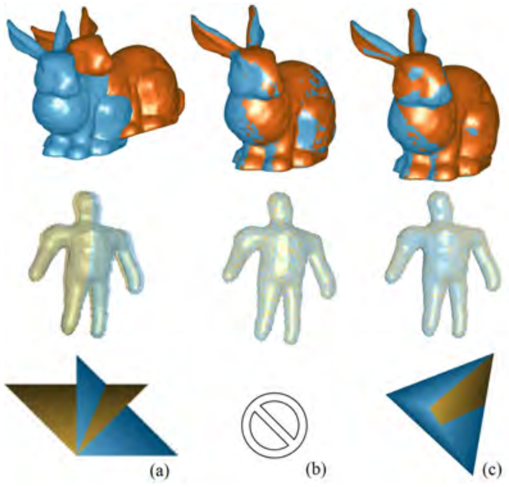

Fig. 2. Global alignment of non-deforming shapes: bunny (top row), rigid walker (center row), and tetrahedron (bottom row). The initial state is shown at the left (a), the alignment results of [56] and our approach are at the center (b) and right column (c) respectively. For the special tetrahedron case, [56] outputs the input as the result.

TABLE 1
Global alignment experiments.

	[56]		global VDF	
	SSD	Time	SSD	Time
bunny	0.0093	1.2s	0.0343	8.9s
rigid walker	0.0162	0.4s	0.0166	2.4s
tetrahedron	×	×	0.000328	5.1s
open walker	185.9	3.3s	169.2	8.6s
closed walker	40.46	1.4s	38.75	9.5s

Table 1 compares the registration results in terms of quality and time (seconds) for the shapes in Figure 2, 3 with the ICP variant [56]. Registration quality is qualitatively assessed in terms of the final SSD for all sample points. For our global VDF alignment, the sample points are located in the vector space, while for [56], they are directly sampled from the shapes. The ICP variant [56] only uses a few of the sampled points (fewer than 500) to find the rigid transformation, which gives it the fast speed. In our approach, the accuracy is determined by the resolution of the sampling grid: the higher resolution used, the more accurate alignment obtained, at the cost of longer time. We use a 70×70×70 grid by default. The ICP variant [56] produces better accuracy for the bunny and rigid walker than our approach with this default resolution setting, but when using a higher resolution (e.g., 96 × 96 × 96) grid, our registration accuracy is better. We note, however, that SSD measurement is not a particularly good metric of registration quality as it does not take correspondences into consideration. This ICP variant works well for non-deforming objects, but performs less well for deformable objects (see, e.g., the walkers in Figure 3). Our approach can handle both non-deforming and deformable objects well.

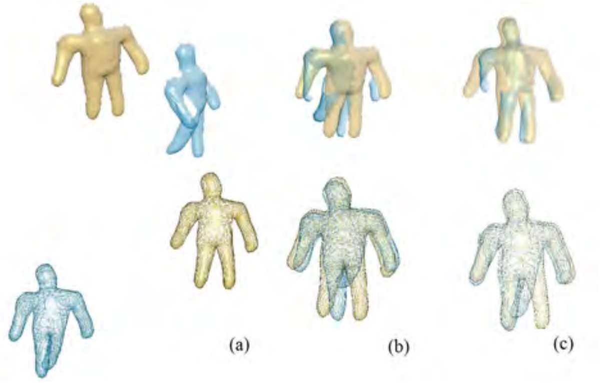

Fig. 3. Global alignment of deformable shapes: open walker (tow row) and closed walker (bottom row). For the input models shown at the left (a), the alignments of [56] and our approach are displayed at the center (b) and right (c). The reasonable rigid motion between salient parts could be estimated by our approach.

To verify the robustness of our approach, an experiment was carried out for 40 non-deforming examples of the bunny and walker models using the default grid size. In each case, $\mathcal{S}$ was fixed and $\mathcal{T}$ was generated by applying a transformation to it. The translation parameters T_x, T_y, T_z were selected randomly in $[-60, 60]$ respectively, and for the rotation parameters, q_0 was randomly selected in $[-1, 1]$, then the other parameters are randomly generated by guaranteeing a unit quaternion was obtained. These generated instances gave ground truth for each case. Our approach produced successful results in all 40 cases, the energy going down smoothly until alignment was achieved. The measurements show that the errors to the ground truth are small since the relative error for each parameter is less than 3%.

5. Local non-rigid registration

Having found a global alignment, a local transformation is determined to recover natural non-rigid transformation parameters that give a dense correspondence between the implicit vector space of the final transformed source shape $\mathcal{S}$ and that of $\mathcal{T}$. In Section 5.1, we introduce the IFFD space deformation model used to represent this transformation, then explain how to deduce its parameters in the closed-form system in Section 5.2. The solution of this ill-conditioned linear system is discussed in Section 5.3.

5.1 IFFD deformation model

We represent the non-rigid part of the transformation using an IFFD deformation model with a cubic B-spline basis function. IFFD manipulates a regular control lattice $\mathbf{P} = \{\mathbf{P}_{l,m,n}\} = \{(P^x_{l,m,n}, P^y_{l,m,n}, P^z_{l,m,n})\}$, $l = 0, \ldots, L$, $m = 0, \ldots, M$, $n = 0, \ldots, N$ overlaid on the vector space $\Omega_{\mathcal{S}}$:

$$\begin{aligned}\Omega_{\mathcal{S}} &= \{\mathbf{x}\} = \{(x, y, z)| \\ &\quad l_x \le x \le h_x, l_y \le y \le h_y, l_z \le z \le h_z\},\end{aligned} \tag{9}$$

where l_x and h_x are the lowest and highest x-axis coordinate value of the lattice, l_y, h_y, l_z and h_z are similarly defined. We denote its initial regular configuration without deformation by $\mathbf{P}^0$, and its deformed configuration by $\mathbf{P} = \mathbf{P}_0 + \delta\mathbf{P}$. Then the IFFD parameters $\mathbf{\Theta} = \delta\mathbf{P}$ are the translations of the control lattice points in each of the x, y, z directions:

$$\mathbf{\Theta} = \{\delta\mathbf{P}_{l,m,n}\} = \{(\delta P^x_{l,m,n}, \delta P^y_{l,m,n}, \delta P^z_{l,m,n})\}. \quad (10)$$

As the control lattice deforms from $\mathbf{P}^0$ to $\mathbf{P}$, the deformed position $\mathbf{x}$ in the embedding space is defined by a tensor product of cubic B-splines:

$$L(\mathbf{\Theta}; \mathbf{x}) = \sum_{a=0}^{3}\sum_{b=0}^{3}\sum_{c=0}^{3} B_a(u)B_b(v)B_c(w) \times (\mathbf{P}^0_{i+a,j+b,c+k} + \delta\mathbf{P}_{i+a,j+b,c+k}), \quad (11)$$

where

$$\begin{aligned} u &= \frac{x-l_x}{h_x-l_x}\left((L-1) - \left\lfloor \frac{x-l_x}{h_x-l_x} \times (L-1)\right\rfloor\right), \\ v &= \frac{y-l_y}{h_y-l_y}\left((M-1) - \left\lfloor \frac{y-l_y}{h_y-l_y} \times (M-1)\right\rfloor\right), \\ w &= \frac{z-l_z}{h_z-l_z}\left((N-1) - \left\lfloor \frac{z-l_z}{h_z-l_z} \times (N-1)\right\rfloor\right), \end{aligned}$$

$B_a(u), B_b(v), B_c(w)$ represent the a^{th}, b^{th} and c^{th} cubic B-spline basis function respectively, i, j, k are the indices of the nearest associated lattice point to $\mathbf{x}$:

$$\begin{aligned} i &= \left\lfloor \frac{x-l_x}{h_x-l_x} \times (L-1)\right\rfloor, \\ j &= \left\lfloor \frac{y-l_y}{h_y-l_y} \times (M-1)\right\rfloor, \\ k &= \left\lfloor \frac{z-l_z}{h_z-l_z} \times (N-1)\right\rfloor, \end{aligned}$$

and $\delta\mathbf{P}_{i+a,j+b,k+c}$, $(a, b, c) \in [0, 3] \times [0, 3] \times [0, 3]$ are the deformations of the 64 control points around $\mathbf{x}$.

Based on the linearization principle of B-splines, a B-spline through collinear control points is itself linear. Hence the initial regular configuration of the control lattice $\mathbf{P}^0$ generates the undeformed shape, i.e., for any $\mathbf{x}$ within the lattice, we have

$$\mathbf{x} = \sum_{a=0}^{3}\sum_{b=0}^{3}\sum_{c=0}^{3} B_a(u)B_b(v)B_c(w)\mathbf{P}^0_{i+a,j+b,c+k}. \quad (12)$$

Combining Equations 11 and 12, and letting $L(\mathbf{\Theta}; \mathbf{x})$, being $\sum\limits_{a=0}^{3}\sum\limits_{b=0}^{3}\sum\limits_{c=0}^{3} B_a(u)B_b(v)B_c(w)\delta\mathbf{P}_{i+a,j+b,c+k}$, denote the incremental deformation for $\mathbf{x}$, we then have:

$$L(\mathbf{\Theta}; \mathbf{x}) = \mathbf{x} + \delta L(\mathbf{\Theta}; \mathbf{x}). \quad (13)$$

5.2 Closed-form system for IFFD parameters

Non-rigid registration is achieved by incrementally evolving $\mathbf{P}$ in such a way that the deformation parameters $\mathbf{\Theta}$ minimize the non-rigid dissimilarity for $\mathbf{x}$, now measured using:

$$\mathbf{r}_n(\mathbf{\Theta}; \mathbf{x})) = f_{\mathcal{S}}(\mathbf{x}) - f_{\mathcal{T}}(L(\mathbf{\Theta}; \mathbf{x})). \quad (14)$$

From now on, for simplicity, $f_{\mathcal{T}}(L(\mathbf{\Theta}; \mathbf{x}))$ is abbreviated as $f_{\mathcal{T}}$, $\delta L(\mathbf{\Theta}; \mathbf{x})$ as δL, and $\mathbf{r}_n(\mathbf{\Theta}; \mathbf{x}))$ as $\mathbf{r}_n$. Again, we define a non-rigid SSD energy to be minimized:

$$E_n(\mathbf{\Theta}) = \int_{\Omega_{\mathcal{S}}} \mathbf{r}_n^T \mathbf{r}_n \, d\mathbf{x}. \quad (15)$$

We regularize it using a smoothness term on the local deformation field δL. This is

$$E_s(\mathbf{\Theta}) = \int_{\Omega_{\mathcal{S}}} (\|\frac{\partial \delta L}{\partial x}\|^2 + \|\frac{\partial \delta L}{\partial y}\|^2 + \|\frac{\partial \delta L}{\partial z}\|^2) \, d\mathbf{x}. \quad (16)$$

Overall, the energy functional to be minimized for local non-rigid registration is

$$E_l(\mathbf{\Theta}) = E_n(\mathbf{\Theta}) + \alpha E_s(\mathbf{\Theta}). \quad (17)$$

The first term tries to produce the desired local deformation while the second penalizes large rates of change of deformation. For the reasons similar to the former global VDF alignment, only the sample points, located in the small band to the source and target shapes, are used, and we set ε to 5 times the interval size of the sampling grid.

For the above formulation, gradient descent method is used to estimate each deformation parameter $\theta_i \in \mathbf{\Theta}$ as follows:

$$\begin{aligned} \frac{\partial E_l(\mathbf{\Theta})}{\partial \theta_i} &= -2\int_{\Omega_{\mathcal{S}}} {\mathbf{r}_n}^T (\nabla f_{\mathcal{T}})^T \frac{\partial \delta L}{\partial \theta_i} \, d\mathbf{x} \\ &\quad +2\alpha \int_{\Omega_{\mathcal{S}}} ((\frac{\partial \delta L}{\partial x})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial x}) \\ &\quad +(\frac{\partial \delta L}{\partial y})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial y}) \\ &\quad +(\frac{\partial \delta L}{\partial z})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial z})) \, d\mathbf{x} \\ &= \mathbf{0}. \end{aligned} \quad (18)$$

If incremental step of each control point $\mathbf{x}$ was not large, then we can use the following Taylor series expansion to approximate the vector representation:

$$f_{\mathcal{T}} = f_{\mathcal{T}}(\mathbf{x} + \delta L) \approx f_{\mathcal{T}}(\mathbf{x}) + (\nabla f_{\mathcal{T}}(\mathbf{x}))^T \delta L. \quad (19)$$

Setting $f(\mathbf{x}) = f_{\mathcal{S}}(\mathbf{x}) - f_{\mathcal{T}}(\mathbf{x})$, Equation 18 leads to:

$$\begin{aligned} &\int_{\Omega_{\mathcal{S}}} f^T(\mathbf{x}) \, (\nabla f_{\mathcal{T}}(\mathbf{x}))^T \frac{\partial \delta L}{\partial \theta_i} \, d\mathbf{x} = \\ &\quad \int_{\Omega_{\mathcal{S}}} ((\nabla f_{\mathcal{T}}(\mathbf{x}))^T \delta L)^T \, (\nabla f_{\mathcal{T}}(\mathbf{x}))^T \frac{\partial \delta L}{\partial \theta_i} \, d\mathbf{x} + \\ &\quad \alpha \int_{\Omega_{\mathcal{S}}} ((\frac{\partial \delta L}{\partial x})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial x}) + \\ &\quad (\frac{\partial \delta L}{\partial y})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial y}) + \\ &\quad (\frac{\partial \delta L}{\partial z})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial z})) \, d\mathbf{x}. \end{aligned} \quad (20)$$

This provides a linear system, as in the 2D case [17], to give a closed-form solution for the final $\mathbf{P} = \mathbf{P}^0 + \mathbf{\Theta}$, in the form

$$\mathbf{K} = \mathbf{Q}\,\mathbf{P}. \quad (21)$$

Here, elements of $\mathbf{K}$ and $\mathbf{Q}$ are given by

$$\begin{aligned}
K_{row} &= \int_{\Omega_{\mathcal{S}}} f^{\mathcal{T}}(\mathbf{x})\,(\nabla f_{\mathcal{T}}(\mathbf{x}))^T\,\frac{\partial \delta L}{\partial \theta_i}\,d\mathbf{x},\\
Q_{row,col} &= \int_{\Omega_{\mathcal{S}}} ((\nabla f_{\mathcal{T}}(\mathbf{x}))^T \delta L')^T\,(\nabla f_{\mathcal{T}}(x))^T\,\frac{\partial \delta L}{\partial \theta_i}\,d\mathbf{x}\\
&\quad +\alpha \int_{\Omega_{\mathcal{S}}} ((\frac{\partial \delta L'}{\partial x})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial x})\\
&\qquad +(\frac{\partial \delta L'}{\partial y})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L'}{\partial y})\\
&\qquad +(\frac{\partial \delta L'}{\partial z})^T \frac{\partial}{\partial \theta_i}(\frac{\partial \delta L}{\partial z}))\,d\mathbf{x}. \qquad (22)
\end{aligned}$$

Here, $\delta L'$ is the abbreviation for $\delta L'(\mathbf{\Theta};\mathbf{x})$, the cubic B-spline coefficient associated with each control point.

One typical advantage of the proposed closed-form solution over the gradient descent is the robustness. Not only the final results are no longer sensitive to the initialization, but also the local minimum problem during the computation has been eliminated. In addition, the execution time could be greatly shortened, since the gradient descent method always takes many iterations (greater than 2) to converge towards a local minimum.

5.3 *Improved iterative Gauss-Seidel solution*

The symmetric indefinite matrix $\mathbf{Q}$ in Equation 21 is ill-conditioned. Experiments show that the direct single value decomposition and traditional iterative techniques [57] for linear system could not obtain desirable results, since the conditioned number $\mathbf{Q}$ is generally greater than 10^{36}. We thus propose an improved iterative Gauss-Seidel solver by changing $\mathbf{Q}$ into a diagonally dominant matrix, as follows.

Given an ill-conditioned linear equation

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \qquad (23)$$

where $\mathbf{A}$ is a (non-singular) ill-conditioned $n \times n$ matrix, firstly, we decompose $\mathbf{A}$ into $\mathbf{A} = \mathbf{B} + \mathbf{H}$, where $\mathbf{B} = \mathrm{diag}(a_{11}, ..., a_{nn})$.

$$(\mathbf{B} + \mathbf{H})\mathbf{x} = \mathbf{b} \qquad (24)$$

Next, $\mathbf{D}$ is simultaneously added to both sides,

$$(\mathbf{B} + \mathbf{H} + \mathbf{D})\mathbf{x} = \mathbf{b} + \mathbf{D}\mathbf{x}, \qquad (25)$$

where $\mathbf{D}$ is a diagonal matrix satisfying $d_i = \mathrm{sign}\{\sum_{j=1}^{n} |a_{ij}|, a_{ii}\}$ and $\mathrm{sign}\{a, b\}$ returns the value of a with the sign of b. Then, we may write:

$$(\mathbf{B} + \mathbf{D})\,\mathbf{x} = \mathbf{b} + (\mathbf{D} - \mathbf{H})\,\mathbf{x}. \qquad (26)$$

It is easy to deduce that

$$\mathbf{x} = (\mathbf{B} + \mathbf{D})^{\mathbf{-1}}\,[\mathbf{b} + (\mathbf{D} - \mathbf{H})\mathbf{x}] \qquad (27)$$

leads to an iterative formula:

$$\mathbf{x^{(k+1)}} = (\mathbf{B} + \mathbf{D})^{\mathbf{-1}}[\mathbf{b} + (\mathbf{D} - \mathbf{H})\mathbf{x^{(k)}}.] \qquad (28)$$

Let $\mathbf{M} = (\mathbf{B} + \mathbf{D})^{\mathbf{-1}}(\mathbf{D} - \mathbf{H})$, $\Delta\mathbf{x} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$, then

$$\Delta\mathbf{x}^{(k)} = \mathbf{M}\Delta\mathbf{x}^{(k-1)} = \cdots = \mathbf{M}^k \Delta\mathbf{x}^{(0)}. \qquad (29)$$

This is convergent, since $(\mathbf{B} + \mathbf{H} + \mathbf{D})$ is a strictly diagonally dominant matrix satisfying the necessary convergence condition. Consequently, we obtain an improved Gauss-Seidel-like iterative formula:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)} + d_i x_i^{(k)}}{b_{ii} + d_i}. \qquad (30)$$

6. Results

Our global-to-local non-rigid registration method has been implemented using C++ under Windows XP, on a Pentium Xeon 2.8GHz dual processor platform. We tested our approach with several datasets: one real dataset of a walker generated by Pinocchio [58], three datasets of elephant, arm, and haoTorso from [6], and a real-world dancer dataset [44]. In each case a pair of shapes was chosen as $\mathcal{S}$ and $\mathcal{T}$.

To account for both large-scale and highly local non-rigid deformations, we use an efficient multi-resolution implementation of the IFFD framework. In all experiments, an initial IFFD lattice resolution of $10 \times 10 \times 10$ is used, and then the resolution of the entire lattice is repeatedly doubled until a satisfactory deformation is achieved. This may be performed as often as needed, but typically 2 levels were sufficient for all examples.

6.1 *Experiments*

Different non-rigid shape registration experiments for open and closed shapes are demonstrated in Figures 1 and 4. Our approach gives good, reasonably dense correspondences for both partial and complete shapes. The non-rigidly registered shapes do not have self-intersections or other artifacts, these demonstrate the effectiveness of our approach.

Our algorithm also works well in the presence of missing data. For the real scanned haoTorso [6] shown in Figure 5, our method produces accurate registration, with quality comparable to [6]. To test the robustness of our approach, we degraded the data by manually cutting salient holes in the noisy models: see the noisy arm shown in Figure 6. Our approach is able to provide a successful registration even in this case.

The primary quality of a registration algorithm is accuracy, which is demonstrated visually by the output of our experiments. Computational efficiency is another important issue, again demonstrated by our experiments. The average convergence time for global alignment of two 3D shapes is less than 10s, and it takes less than 20s to perform local non-rigid registration. For the elephant, dancer and arm examples, times are listed in Table 2. In Table 2, The first row gives the number of points for each input model. The next four rows give parameters for the sampling grid resolution, the band

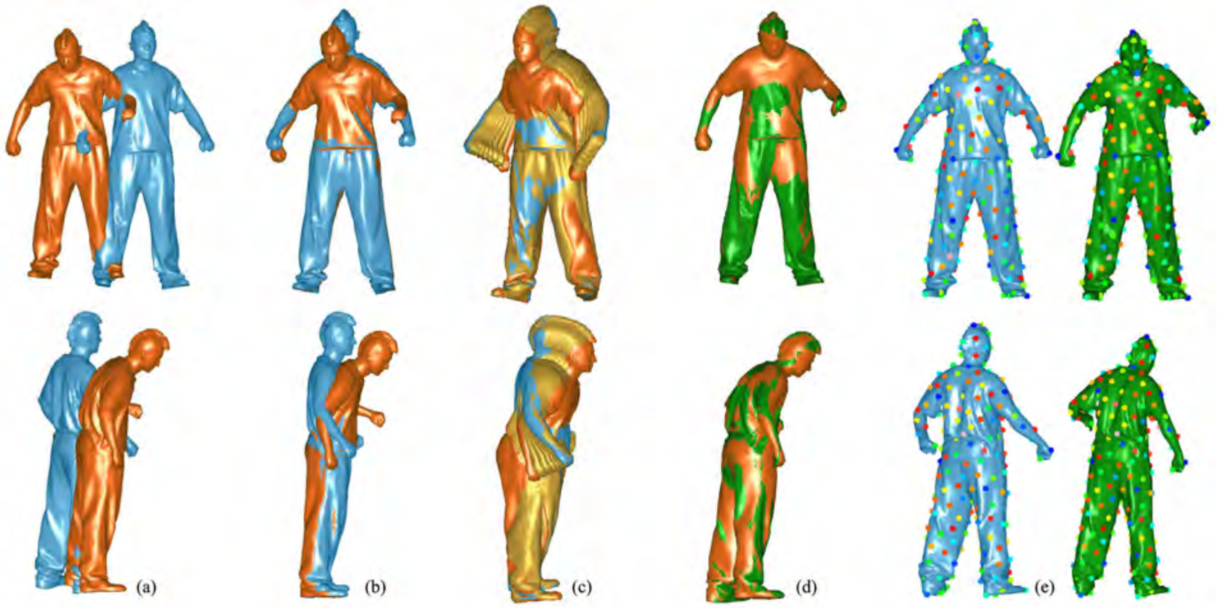

Fig. 4. Global-to-local non-rigid registration for closed dancer models in front view (top row) and side view (bottom row). (a) Initial poses (source: blue, target: brown). (b) Global alignment using SSD minimization, showing aligned shapes. (c) Iterations of local non-rigid registration. (d) Final non-rigid registration using IFFD; the transformed source shape (green) is shown overlaid on the target shape (brown). (e) Dense correspondences produced by global-to-local registration (221 corresponding points).

size relative to the interval size of the sampling grid, the initial IFFD resolution, and the number of iterations needed to solve the linear equation. The final two rows list times for global alignment and local registration respectively. For the haoTorso and elephant objects, our approach spends less than 17s to finish the non-rigid registration, while [6] takes over one minute to obtain similar registration quality.

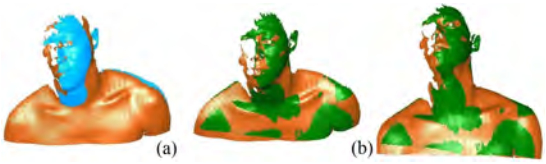

Fig. 5. Non-rigid Registration of haoTorso: (a) original position, (b) results from two viewpoints.

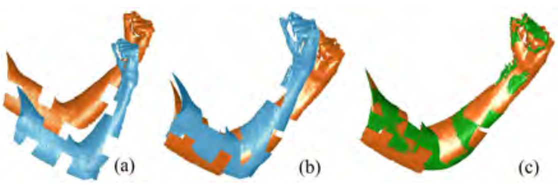

Fig. 6. Non-rigid registration of the noise arm, (a) Original position, (b) global alignment, (c) final non-rigid registration.

6.2 *Limitations and further work*

Our local non-rigid registration method relies on linearizing the non-rigid deformation energy functional using a first order Taylor expansion. As a result, our approach does not handle great large-scale deformations well. The incorporation of good feature constraints is likely to improve registration accuracy and could help to overcome this limitation. Incorporating correspondences between feature points can be done as follows. Assume the number of features is N_f. For each feature, let the pair of corresponding points be $\mathbf{x}_{\mathcal{T}_i}$ $(i = 1, \ldots, N_f)$ on the target shape $\mathcal{T}$ and $\mathbf{x}_{\mathcal{S}_i}$ on the globally transformed source shape $\mathcal{S}$. Then the following feature correspondence term is added into Equation 17.

$$E_f = \sum\nolimits_{i=1}^{N_f} \mathbf{r}_{f_i}^T \mathbf{r}_{f_i}, \quad \mathbf{r}_{f_i} = \mathbf{x}_{\mathcal{S}_i} - L(\mathbf{\Theta}; \mathbf{x}_{\mathcal{T}_i}). \quad (31)$$

We show an example in Figure 7, which uses feature point constraints to register two shapes of a walker. The two shapes significantly differ in the left arm and left leg. Such larger deformations cause our original method to stick in a local minimum (Figure 7.2.b). After incorporating a small number of feature correspondences a much better solution is obtained (Figure 7.1.b). Beside these, Figure 7 also shows that if the deformation (e.g., right arm and right leg of the walker) was not so larger, similar registration quality could be obtained without feature constraints.

TABLE 2
EXPERIMENTAL STATISTICS.

dataset	elephant	dancer	arm
# points	24k	10k	15k
# sampling resolution	70	70	70
# band size	5	5	5
# IFFD resolution	10	10	10
# iteration	8	10	6
global alignment	7.4s	9.2s	8.9s
local registration	16.4s	17.5s	13.9s

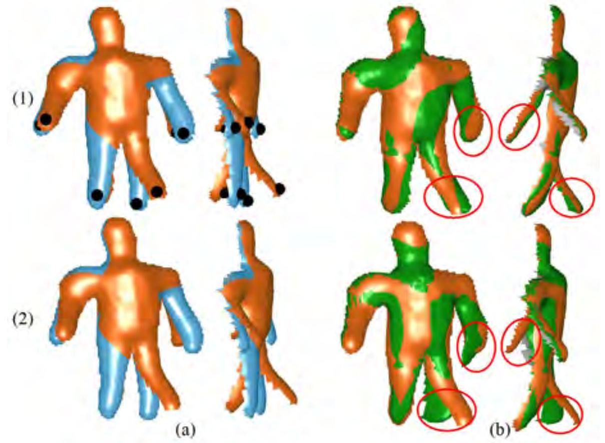

Fig. 7. The non-rigid registration of two shapes of one walker (source: blue, target red) with (1) and without (2) feature constraints marked by black spheres. (a) Original poses, (b) final results.

Since our method is an implicit approach, it shares disadvantages common to such approaches. The size of the representation is always that of the ambient space even for shapes with low detail. However, as noted in [22], this disadvantage is also a strength since it provides flexibility for highly detailed models. Choosing an appropriate resolution for the IFFD lattice is also a tricky issue.

Some potential extensions could be made to our algorithm. Firstly, instead of using IFFD with a regular control lattice, other FFD models could be used that allow the placing of control points at arbitrary locations, such as Dirichlet Free Form Deformation [59]. Secondly, it would be straightforward to integrate explicit techniques as in [12] to handle changing topology [47] during the registration. Thirdly, it would be interesting to apply our approach to object tracking, dynamic model reconstruction, and morphing, since our registration framework provides a way to handle a large amount of data and establish intraframe correspondences.

7. Conclusions

We have introduced a global-to-local registration approach for moving and deforming shapes. The advantages of our technique are that it does not require any user specified markers, or prior templates. We have demonstrated that the 3D implicit vector distance function representation yields a robust method that achieves meaningful, accurate rigid and non-rigid registrations even in the presence of large transformations. Our approach could work well both for open partial and closed objects, even for inperfect noisy data with gaps.

Ultimately, the problem of non-rigid registration of deformable shapes is ill-posed and no algorithm is applicable to all scenarios. We believe, however, that our approach pushes the limits of what can be achieved with minimal prior information, and that the initial research opens up the possibility of geometrical modeling and processing in 3D implicit vector spaces.

8. Acknowledgements

We wish to thank Hao Li, Will Chang, Jovan Popović and Daniel Vlasic for providing the data sets. We also appreciate Szymon Rusinkiewicz for making his implementation of the ICP variant algorithm publicly available. The work is supported by Natural Science Foundation of China (Project Numbers 60773020, 60970094), the National High-tech R&D Program of China (Project Number 2009AA01Z301), and the Pre-research funding of National University of Defense Technology (Project Number JC09-06-01).

References

[1] N. Mitra, S. Flory, M. Ovsjanikov, N. Gelfand, L. Guibas, and H. Pottmann, "Dynamic geometry registration," in *Proc. Eurographics Symp. on Geometry Processing*, 2007, pp. 173–182.

[2] M. Wand, P. Jenke, Q. Huang, M. Bokeloh, L. Guibas, and A. Schilling, "Reconstruction of deforming geometry from time-varying point clouds," in *Proc. Eurographics Symp. on Geometry Processing*, 2007, pp. 49–58.

[3] W. Chang and M. Zwicker, "Automatic registration for articulated shapes," in *Proc. Eurographics Symp. on Geometry Processing*, 2008, pp. 1459–1468.

[4] J. Sübamuth, M. Winter, and G. Greiner, "Reconstructing animated meshes from time-varying point clouds," in *Proc. Eurographics Symp. on Geometry Processing*, 2008, pp. 1469–1476.

[5] Q. Huang, B. Adams, M. Wicke, and M. Guibas, "Non-rigid registration under isometric deformations," in *Proc. Eurographics Symp. on Geometry Processing*, 2008, pp. 1149–1458.

[6] H. Li, R. Summer, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," in *Proc. Eurographics Symp. on Geometry Processing*, 2008, pp. 1421–1430.

[7] W. Zeng, Y. Zeng, Y. Wang, X. Yin, X. Gu, and D. Samaras, "3d non-rigid surface matching and registration based on holomorphic differentials," in *Proc. Eurographics Conf. on Computer Vision*, 2008, pp. 1–14.

[8] A. Sharf, D. Alcantara, T. Lewiner, C. Greif, A. Sheffer, N. Amenta, and D. Cohen-Or, "Space-time surface reconstruction using incompressible flow," in *Proc. SIGGRAPH AISA*, 2008, Article No.: 110.

[9] D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer, "Articulated shape matching using laplacian eigenfunctions and unsupervised point registration," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[10] W. Chang and M. Zwicker, "Range scan registration using reduced deformable models," in *Proc. Eurographics*, 2009, pp. 447–456.

[11] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.

[12] X. Guo, J. Hua, and H. Qin, "Scalar-function-driven editing on point set surfaces," *IEEE Comput. Graph. Appl.*, vol. 24, no. 4, pp. 43–52, 2004.

[13] G. Turk and J. F. O'Brien, "Shape transformation using variational implicit surfaces," in *Proc. SIGGRAPH*, 1999, pp. 335–342.

[14] C. Walder, B. Schölkopf, and O. Chapelle, "Implicit surface modelling with a globally regularised basis of compact support," in *Proc. Eurographics*, 2006, pp. 635–644.

[15] G. Unal, D. Nain, G. Slabaugh, and T. Fang, "Customized design of hearing aids using statistical shape learning," in *Proc. MICCAI*, 2008, pp. 518–526.

[16] J. Gomes and O. Faugeras, "The vector distance functions," *Int. J. Comp. Vis.*, vol. 52, pp. 161–187, 2003.

[17] H. A. E. Munim and A. A. Farag, "Shape representation and registration using vector distance functions," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[18] B. Allen, B. Curless, and J. Popović, "Articulated body deformation from range scan data," in *Proc. SIGGRAPH*, 2002, pp. 612–619.

[19] R. White, K. Crane, and D. Forsyth, “Capturing and animating occluded cloth,” in *Proc. SIGGRAPH*, 2007, Article No.: 34.

[20] Y. Pekelny and C. Gotsman, “Articulated object reconstruction and markerless motion capture from depth video,” *Computer Graphics Forum*, vol. 27, no. 2, pp. 243–253, 2008.

[21] H. Li, B. Adams, L. Guibas, and M. Pauly, “Robust single-view geometry and motion reconstruction,” in *Proc. SIGGRAPH AISA*, 2009, Article No.: 175.

[22] T. Sederberg and S. Parry, “Free-form deformation of solid geometric models,” in *Proc. SIGGRAPH*, 1986, pp. 151–160.

[23] P. Besl and N. McKay, “A method for registration of 3-d shapes,” *IEEE Trans. Pat. Ana. & Mach. Int.*, vol. 14, no. 2, pp. 239–256, 1992.

[24] B. Allen, B. Curless, and J. Popovic, “The space of human body shapes: reconstruction and parameterization from range scans,” *ACM Trans. on Graphics*, vol. 22, no. 3, pp. 587–594, 2003.

[25] D. Anguelov, P. Srinivasan, H.-C. Pang, D. Koller, S. Thrun, and J. Davis, “The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces,” in *Proc. NIPS*, 2004, pp. 33–40.

[26] M. Pauly, N. Mitra, J. Giesen, M. Gross, and L. Guibas, “Example-based 3d scan completion,” in *Proc. Eurographics Symp. on Geometry Processing*, 2005, p. No. 23.

[27] B. Amberg, S. Romdhani, and T. Vetter, “Optimal step nonrigid icp algorithms for surface registration,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[28] B. Brown and S. Rusinkiewicz, “Global non-rigid alignment of 3-d scans,” *ACM Trans. on Graphics*, vol. 26, no. 3, p. No. 21, 2007.

[29] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, “Topology-invariant similarity of nonrigid shapes,” *Int. J. Comp. Vis.*, vol. 81, no. 3, pp. 281–301, 2009.

[30] M. Wand, B. Adams, M. Ovsjanikov, A. Berner, M. Bokeloh, P. Jenke, L. Guibas, H.-P. Seidel, and A. Schilling, “Effcient reconstruction of non-rigid shape and motion from real-time 3d scanner data,” *ACM Trans. on Graphics*, vol. 28, no. 2, Article No.: 15, 2009.

[31] H. Zhang, A. Sheffer, D. Cohen-Or, Q. Zhou, O. V. Kaick, and A. Tagliasacchi, “Deformation-driven shape correspondence,” in *Proc. Eurographics Symp. on Geometry Processing*, 2008, pp. 1431–1439.

[32] A. Tevs, M. Bokeloh, M. Wand, A. Schilling, and H.-P. Seidel, “Isometric registration of ambiguous and partial data,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[33] M. Ruggeri and D. Saupe, “Isometry-invariant matching of point set surfaces,” in *Eurographics Workshop on 3D Object Retrieval*, 2008, pp. 17–24.

[34] N. Ahmed, C. Theobalt, P. Dobrev, H.-P. Seidel, and S. Thrun, “Robust fusion of dynamic shape and normal capture for high-quality reconstruction of time-varying geometry,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[35] N. Ahmed, C. Theobalt, C. Roessl, S. Thrun, and H.-P. Seidel, “Dense correspondence finding for parametrization-free animation reconstruction from video,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[36] Y. Lipman and M. Funkhouser, “Möbius voting for surface correspondence,” in *Proc. SIGGRAPH*, 2009, Article No.: 72.

[37] A. Bronstein, M. Bronstein, and R. Kimmel, “Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching,” *National Academy of Sciences*, vol. 103, no. 5, pp. 1168–1172, 2006.

[38] F. Mémoli and G. Sapiro, “A theoretical and computational framework for isometry invariant recognition of point cloud data,” *Found. Comput. Math.*, vol. 5, no. 3, pp. 313–347, 2005.

[39] J. Sun, M. Ovsjanikov, and L. J. Guibas, “A concise and provably informative multi-scale signature based on heat diffusion.” *Computer Graphics Forum*, vol. 28, no. 5, pp. 1383–1392, 2009.

[40] M. M. Bronstein and I. Kokkinos, “Scale-invariance in local heat kernel descriptors without scale selection and normalization,” 2009, INRIA Research Report 7161.

[41] M. Kilian, N. J. Mitra, and H. Pottmann, “Geometric modeling in shape space,” *ACM Trans. on Graphics*, vol. 26, no. 3, Ariticle No.: 64, 2007.

[42] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, “Surface feature detection and description with applications to mesh matching.” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009, pp. 373–380.

[43] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, “Performance capture from sparse multi-view video,” *ACM Trans. on Graphics*, vol. 27, no. 3, Article No.: 98, 2008.

[44] D. Vlasic, I. Baran, W. Matusik, and J. Popović, “Articulated mesh animation from multi-view silhouettes,” *ACM Trans. on Graphics*, vol. 27, no. 3, Article No.: 97, 2008.

[45] S. I. Park and J. K. Hodgins, “Capturing and animating skin deformation in human motion,” in *Proc. SIGGRAPH*, 2006.

[46] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong, “Modeling deformable objects from a single depth camera,” in *Proc. International Conf. on Computer Vision*, 2009.

[47] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro, “A gromov-hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching,” *Int. J. Comp. Vis.*, 2009.

[48] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, “A review of recent range image registration methods with accuracy evaluation,” *Image and Vision Computing*, vol. 25, no. 5, pp. 578–596, 2007.

[49] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, “Cortex segmentation: A fast variational geometric approach,” *IEEE Transactions on Medical Imaging*, vol. 21, pp. 1544–1551, 2002.

[50] D. Rueckert, L. Sonoda, C. Hayes, D. Hill, M. Leach, and D. Hawkes, “Nonrigid registration using free-form deformations: Application to breast mr images,” *IEEE Transactions on Medical Imaging*, vol. 8, pp. 712–721, 1999.

[51] N. Paragios, M. Rousson, , and V. Ramesh, “Non-rigid registration using distance functions,” *Computer Vision and Image Understanding*, vol. 89, pp. 142–165, 2003.

[52] X. Huang, N. Paragios, and D. Metaxas, “Shape registration in implicit spaces using information theory and free form deformations,” *IEEE Trans. Pat. Ana. & Mach. Int.*, vol. 28, no. 8, pp. 1303–1318, 2006.

[53] J. Schnabel, D. Rueckert, M. Quist, J. Blackall, A. Castellano-Smith, T. Hartkens, G. Penney, W. Hall, H. Liu, C. Truwit, F. Gerritsen, D. Hill, and D. Hawkes, “A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations,” in *Proc. of the 4th International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2001, pp. 573–581.

[54] O. D. Faugeras and M. Hebert, “A 3-d recognition and positioning algorithm using geometrical matching between primitive surfaces,” in *Proc. of the Eighth international joint conference on Artificial intelligence*, 1983, pp. 996–1002.

[55] K. Madsen, H. Nielsen, and O. Tingleff, “Methods for non-linear least squares problems,” 2004, lecture notes, available at http://www.imm.dtu.dk/courses/02611/nllsq.pdf.

[56] S. Rusinkiewicz and M. Levov, “Efficient variants of the icp algorithm,” in *Proc. 3D Digital Imaging and Modeling*, 2001, pp. 145–152.

[57] G. Strang, *Linear Algebra and Its Applications (3rd ed)*. Addison Wesley, 2005.

[58] I. Baran and J. Popović, “Automatic rigging and animation of 3d characters,” in *Proc. SIGGRAPH*, 2007, Article No.: 72.

[59] S. Ilic and P. Fua, “Using dirichlet free form deformation to fit deformable models to noisy 3d data,” in *Proc. Eurographics Conf. on Computer Vision*, 2002, pp. 704–717.