Robust Non-Rigid Registration with Reweighted Position and Transformation Sparsity

Kun Li, *Member, IEEE,* Jingyu Yang, *Senior Member, IEEE,*
Yu-Kun Lai, *Member, IEEE,* and Daoliang Guo

Abstract—Non-rigid registration is challenging because it is ill-posed with high degrees of freedom and is thus sensitive to noise and outliers. We propose a robust non-rigid registration method using reweighted sparsities on position and transformation to estimate the deformations between 3-D shapes. We formulate the energy function with position and transformation sparsity on both the data term and the smoothness term, and define the smoothness constraint using local rigidity. The double sparsity based non-rigid registration model is enhanced with a reweighting scheme, and solved by transferring the model into four alternately-optimized subproblems which have exact solutions and guaranteed convergence. Experimental results on both public datasets and real scanned datasets show that our method outperforms the state-of-the-art methods and is more robust to noise and outliers than conventional non-rigid registration methods.

Index Terms—Non-rigid registration, noise and outliers, deformation, position sparsity, transformation sparsity

1 Introduction

Non-rigid registration is an active research area in computer graphics and computer vision [17], [24], [32], [37], and is a key technique for dynamic 3-D reconstruction using a depth camera. Commodity depth sensors, e.g., Microsoft Kinect, become cheaper and more widely used, but depth images and reconstructed point clouds captured by such devices contain much noise. Hence, non-rigid registration methods robust to noise and outliers are highly desirable to scan dynamic scenes with deformable objects.

Given two input 3-D shapes, one as the template shape and the other as the target shape, non-rigid registration aims to find a suitable transformation that when applied deforms the template shape to be aligned with the target shape. Non-rigid registration is often formulated as an optimization problem. Most methods formulate some energy functional with both position and transformation constraints. The position constraint measures the closeness of the transformed template shape and the target shape, and the transformation constraint measures the fitness to model, which might include the smoothness, namely the total energy of transformation differences of all the local neighbors. Most work uses the classic squared ℓ_2-norm in the position constraint and the transformation constraint [21], [3], [33]. However, the quadratic energy functional is more easily affected by noise and outliers. To address this problem, Yang $et\ al.$ [40] propose a sparse non-rigid registration (SNR) method with an ℓ_1-norm regularized model for the transformation constraint. However, their position constraint is still based on the ℓ_2-norm. In practice, e.g. for near piece-wise rigid deformation, which is common for real-world deformable objects, the positional error tends to concentrate on small regions. This cannot be modeled well using the ℓ_2-norm.

- *Kun Li is with the Tianjin Key Laboratory of Cognitive Computing and Application, School of Computer Science and Technology, Tianjin University, Tianjin 300072, China.*
- *Jingyu Yang and Daoliang Guo are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China.*
- *Yu-Kun Lai is with the School of Computer Science and Informatics, Cardiff University, Wales, UK.*
  Corresponding author: Jingyu Yang (Email: yjy@tju.edu.cn)

In this paper, we propose a non-rigid registration method with sparsity-regularized position and transformation constraints. The distribution of positional errors and transformation differences for typical non-rigid deformation can be well modeled using the Laplacian distribution, or equivalently, the ℓ_1-norm should be used to measure both the positional errors and transformation differences. To promote the sparsity, we adopt a reweighted sparse model, which is solved by the alternating direction method of multipliers (ADMM). The proposed method is evaluated on public datasets [10], [38] and real datasets captured by a RGB-D depth sensor. The results demonstrate that the proposed method obtains better results than the state-of-the-art non-rigid registration methods.

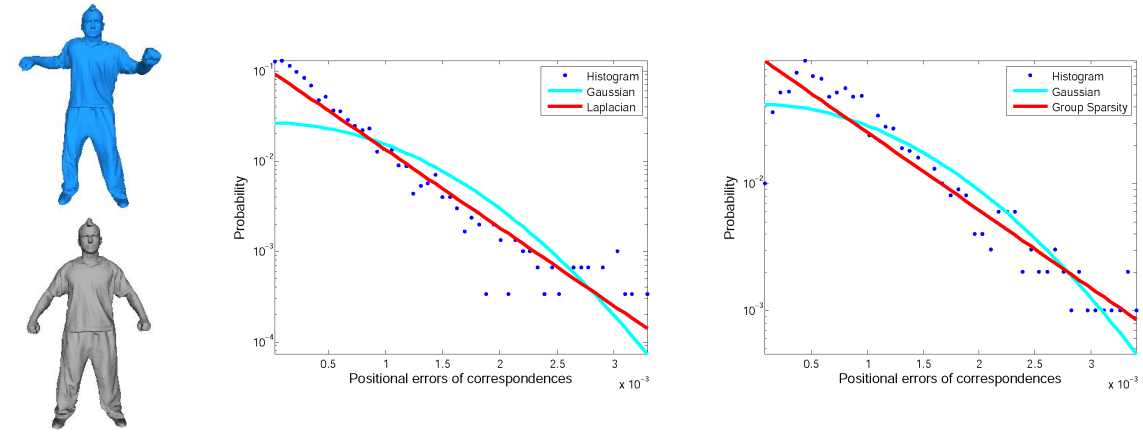
The main contributions of this work are summarized as:

- We propose a non-rigid registration method on both position and transformation sparsity. The proposed model is robust against outliers as the sparsity terms allow a small fraction of regions with larger deviations.
- We incorporate orthogonality constraints in the sparsity-inducing non-rigid registration framework to promote locally rigid transformations.
- We equip the proposed non-rigid registration model with a reweighted scheme to iteratively enhance sparsity in the series of alternating optimization subproblems.

2 Related Work

3-D shape registration consists of rigid registration and non-rigid registration. Rigid registration aims to find a *global* rigid-body transformation, while non-rigid registration needs to find a set of *local* transformations that align two shapes.

In rigid registration, the 3-D shapes are assumed to be aligned by a Euclidean transformation, including rotation and translation. Iterative Closest Point (ICP) and its variants [5] are the dominant algorithms for rigid registration. This kind of methods alternates between two steps: 1) finding closest points and 2) solving the optimal transformation. As an improved method of ICP, Chen *et*



(a) Source and target (b) ℓ_1 sparsity (c) Group sparsity

Figure 1. Normalized histograms and the associated fitted Laplacian and Gaussian distributions of positional errors measured in the ℓ_1 norm (with equal contribution from each dimension) (b) or with Euclidean distance (c) for *Bouncing* dataset (a). The graphs show proportion of correspondences (y-axis in logarithmic scale) with specific positional errors (x-axis).

al. [12] minimize the shortest distance between a point in the template and the tangent plane of the closest point on the target. Pottmann *et al.* [28] propose a registration method with quadratic convergence, which gives faster and more stable convergence than the standard ICP [27]. Bouaziz *et al.* [8] propose a new variant of the ICP algorithm, which uses sparsity-inducing norms to represent the positional constraint and achieves better results for the situation with noise and outliers. Their work focuses on rigid registration with low degrees of freedom, and hence regularization is not necessary.

When shapes have large deformations from template to target, automatic non-rigid registration is necessary. It is more challenging due to its high degrees of freedom, and an appropriate deformation model is the key for an efficient and robust algorithm.

Some methods compute global rigid transformations for bones and local non-rigid transformations near joints, which is essentially a piecewise rigid transformation model. Allen *et al.* [1] place markers on the object to help reconstruct the pose of scan and use it as a basis for modeling deformation. Pekelny *et al.* [26] use predefined bone information to find bone transformations.

Some models take more generic deformations into consideration. Chui *et al.* [14] use the thin-plate spline (TPS) as the non-rigid transformation model. Papazov *et al.* [25] allow points to move freely and use an additional uniform distribution to limit noise and outliers, and propose an ordinary differential equation (ODE) model. Local affine transformations [2] are also frequently used in non-rigid registration. Liao *et al.* [22] use differential coordinates as local affine transformations with smoothness constraints. Amberg *et al.* [3] use a stiffness term to ensure similarity of adjacent transformations. Rouhani *et al.* [29] model non-rigid deformation as an integration of locally rigid transformations. In our work, we use local affine transformations with an orthogonality constraint as it allows more flexibility to capture fine surface details while keeping local shapes.

Non-rigid registration is often formulated as an energy functional with data and regularization terms. Most of the non-rigid registration work models the data term in the ℓ_2-norm in a least-squares sense [34], [3].

Regularization terms help to preserve smoothness, making the optimization more robust to noise and outliers, and ℓ_2-norm is also widely used in regularization terms. Süßmuth *et al.* [35] use a generalized as-rigid-as-possible energy [33] to promote smoothness. Liao *et al.* [22] define a transformation model using the TPS [14], and use graduated assignment for non-rigid registration and optimization. Wand *et al.* [39] take a set of time-varying point data as input, and reconstruct a single shape and a deformation field that fit the data. To improve robustness, Li *et al.* [21] solve correspondences, confidence weights, and a deformation field within a single optimization framework using ℓ_2-norm. Their method however requires adjacent frames to be sufficiently close to work effectively. Hontani *et al.* [18] propose a statistical shape model (SSM) which is incorporated into the nonrigid ICP (NICP), and outliers can be detected based on their sparseness. Based on the observation that many deformable objects, in particular human bodies, have near articulated motions, Guo *et al.* [17] introduce ℓ_0 regularization for motions which provide more accurate and robust tracking in dynamic 3D reconstruction. However, since their method is based on a tracking pipeline, adjacent frames are required to have high similarity. Moreover, their sparse regularization is only applied to motions. Yang *et al.* [40] propose a sparse non-rigid registration (SNR) method with an ℓ_1-norm regularized model for the smoothness. However, their ℓ_2-norm position constraint cannot model the concentration of positional errors well.

Non-rigid registration is also related to and often an important component in dynamic 3D (or 4D) reconstruction. Li *et al.* [20] propose a pioneering solution to dynamic reconstruction from a sequence of depth images captured by a single depth camera. The method produces impressive results but requires to capture the coarse 3D template of the deforming object. It also assumes adjacent scans are reasonably close. More recently, the work [41] achieves real-time reconstruction with GPU acceleration. However, the method still requires the complete template model to be scanned in advance. To reconstruct dynamic 3D deforming objects without a template prior is still challenging, and state-of-the-art techniques such as [15] utilize multi-camera systems (24 cameras producing 8 depth streams are used in [15]) to achieve real-time 4D performance capture. Our work considers general non-rigid registration where scans can have substantial deformation and no template prior is required.

In this paper, based on the observation that the deformations of 3-D surfaces vary smoothly and the positional distances and transformation differences are sparse, we propose a non-rigid registration method with sparse position and transformation constraints. The model is efficiently solved by the alternating direction method under the augmented Lagrangian multiplier framework.

3 Motivation

The data terms in previous shape registration models [17], [21], [40] are quadratic, which implicitly assumes the Gaussian distribution of positional errors. However, transformations in certain common scenarios, such as articulated motion of humans, are largely piecewise smooth signals residing on 3D surfaces, resulting in larger positional errors for geometric details and joints and smaller errors for the remaining surfaces. This suggests that the positional errors are sparse, and should be modeled by a heavy-tailed distribution, rather than being dense and modeled by a rapidly vanishing Gaussian distribution. This is verified in Fig. 1(b). We uniformly pick up 10% ground truth matchings (vertices) as correspondences, and solve for the transformations using the SNR method [40] which measures the positional errors in the standard quadratic term to avoid bias towards the ℓ_1-norm. The Laplacian distribution fits the histogram of positional errors significantly better than the Gaussian distribution, suggesting the use of sparsity-promoting ℓ_1-norm in the data term.

Let $\mathbf{d}_i = (\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$ be the difference between the i^{th} transformed template vertex position and the position of its corresponding target vertex. Our ℓ_1-norm sparsity measures equally the sum of coordinate differences in each dimension for all the corresponding vertices (*i.e.*, $E_1 = \sum_i \|\mathbf{d}_i\|_1 = \sum_i (|\tilde{x}_i| + |\tilde{y}_i| + |\tilde{z}_i|)$). Another possibility is to use the sum of Euclidean distances (group sparsity) between corresponding points (*i.e.*, $E_g = \sum_i \|\mathbf{d}_i\|_2 = \sum_i \sqrt{\tilde{x}_i^2 + \tilde{y}_i^2 + \tilde{z}_i^2}$), which also well fits the distribution of positional errors as shown in Fig. 1(c). The group sparsity advocates sparsity for each Euclidean distance as a whole, while the ℓ_1-norm allows a large distance along a particular dimension. In this sense, ℓ_1-norm is more flexible to preserve large non-rigid deformation along some dimensions. Such an advantage is also observed in the anisotropic total variation (TV) [16] that applies the ℓ_1-norm to the image gradient over the isotropic TV [30] that measures TV as the sum of ℓ_2-norm (not squared). Birkholz [6] showed that anisotropic TV achieves better denoising performance in preserving the geometries of corners in images. We choose the ℓ_1-norm to measure the positional errors for its potential flexibility, and also for its easier and faster implementation with an element-wise shrinkage (cf. Table 5 for statistics of running times).

In this paper, we make the assumption that the surfaces to be registered undergo transformations which are near piecewise smooth. This covers a broad range of practical scenarios ranging from common (near) articulated deformations such as human bodies to certain non-articulated deformations such as facial expressions. In such cases, substantial changes of transformations or large registration errors occur in relatively local areas. Note that our model does not require such consistency to satisfy entirely, and can well cope with situations such as muscle bulge, change of local shapes at joints, etc. Our assumptions also fit well with inaccurate correspondences and sparse noise/outliers, as they also induce sparse distributions of errors.

4 The Proposed Method

4.1 Iterative Framework

We iteratively compute the deformation between the template shape and the target shape. Each iteration consists of two steps. In the first step, the correspondences between template and target are estimated using the registration result obtained from the last iteration. At the beginning of the iteration, we use a technique based on local geometric similarity and diffusion pruning of inconsistent correspondences [36] as it often provides reliable correspondences. Alternative correspondence techniques or manual specification of a few correspondences may instead be used (an example is shown in Fig. 5). These computed correspondences are used to initialize the correspondence mapping, referred to as f. Then, during the iterative process, we update f by using the closest points between template and target shapes to find additional correspondences similar to ICP. In the second step (Sec. 4.2), we propose an energy-minimization approach based on double sparsity representation to estimate the non-rigid transformations using the correspondences obtained from the first step.

4.2 Deformation Estimation

Let $\mathbf{v}_i \triangleq [x_i, y_i, z_i, 1]^\top$ be a 3D point in the homogenous coordinates. Denote by $\mathcal{V} \triangleq \{\mathbf{v}_1, \cdots, \mathbf{v}_N\}$ a template set of 3D points and by $\mathcal{U} \triangleq \{\mathbf{u}_1, \cdots, \mathbf{u}_M\}$ a target set of 3D points, where N and M are the numbers of points. Denote by $\mathbf{u}_{f(i)} \in \mathcal{U}$ the correspondence of $\mathbf{v}_i \in \mathcal{V}$. Define $f : \{1, \cdots, N\} \mapsto \{0, 1, \cdots, M\}$ as the index mapping from the template points to the target points, where $f(i) = 0$ means the corresponding vertex cannot be found for the i-th vertex. Denote by $\mathbf{X}_i$ the 3×4 transformation matrix for point $\mathbf{v}_i$. Define $\mathcal{X} \triangleq \{\mathbf{X}_1, \cdots, \mathbf{X}_N\}$ as the set of non-rigid transformations. For compact notation, we define $\mathbf{X} \triangleq [\mathbf{X}_1, \cdots, \mathbf{X}_N]^\top$ as a matrix containing the N transformation matrices to be solved. The proposed method is to find non-rigid transformations $\mathbf{X}$ that transform the template $\mathcal{V}$ into the target $\mathcal{U}$ as accurately as possible, given a correspondence mapping f.

The non-rigid registration is formulated as the minimization of the following energy function:

$$E(\mathbf{X}; f) = E_{data}(\mathbf{X}; f) + \alpha E_{smooth}(\mathbf{X}) + \beta E_{orth}(\mathbf{X}), \quad (1)$$

where $E_{data}(\mathbf{X})$, $E_{smooth}(\mathbf{X})$ and $E_{orth}(\mathbf{X})$ are data term, smoothness term, and orthogonality constraint, respectively. α and β adjust the importance of different terms. The data term measures the position accuracy, the smoothness term imposes a smoothness constraint so that the original ill-posed problem (defined by only the data term) is now well-posed, and the orthogonality constraint promotes locally rigid transformations, which is particularly needed for underconstrained scenarios such as partial meshes.

Data term: We measure the accuracy of deformation as the closeness of the transformed points to their corresponding target points. We assign a weight, denoted by w_i, for each point. The weight w_i is one if there is a corresponding point on the target shape for $\mathbf{v}_i$, and zero otherwise. Hence, we propose the following data term

$$E_{data}(\mathbf{X}; f) = \sum_{\mathbf{v}_i \in \mathcal{V}} w_i \left\| \mathbf{X}_i \mathbf{v}_i - \tilde{\mathbf{u}}_{f(i)} \right\|_1, \quad (2)$$

where $\tilde{\mathbf{u}}_{f(i)}$ is the Cartesian coordinate of $\mathbf{u}_{f(i)}$.

For the compact representation in algorithm derivation, we define the following matrix/vector form of the variables to reformulate data term (2):

$$\begin{aligned}\mathbf{W} &= \text{diag}\left(\sqrt{w_1}, \cdots, \sqrt{w_N}\right), \\ \mathbf{V} &= \text{diag}\left(\mathbf{v}_1^\top, \cdots, \mathbf{v}_N^\top\right), \\ \tilde{\mathbf{U}}_f &= \begin{bmatrix}\tilde{\mathbf{u}}_{f(1)} & \cdots & \tilde{\mathbf{u}}_{f(N)}\end{bmatrix}^\top,\end{aligned} \tag{3}$$

where $\text{diag}(\cdot)$ is a diagonal matrix containing the input elements as diagonal entities. Then, the data term can be rewritten as

$$E_{data}\left(\mathbf{X}; f\right) = \left\|\mathbf{W}\left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right)\right\|_1. \tag{4}$$

Smoothness term: In the smoothness term, local rigidity is assumed: for vertex $\mathbf{v}_i$, the transformations of neighboring vertices $\mathbf{v}_j \in \mathcal{N}_i$ should have very close transformed positions when applied to $\mathbf{v}_i$. Therefore, we define the following smoothness term:

$$E_{smooth}\left(\mathbf{X}\right) = \sum_{\mathbf{v}_i \in \mathcal{V}} \sum_{\mathbf{v}_j \in \mathcal{N}_i} \left\|\mathbf{X}_i\mathbf{v}_i - \mathbf{X}_j\mathbf{v}_i\right\|_1. \tag{5}$$

Define a graph $\mathcal{G} \triangleq (\mathcal{V}, \mathcal{E})$, where the vertices of the graph are the 3D points in $\mathcal{V}$, and the edges of the graph are denoted by $\mathcal{E}$. For a 3D mesh, edges of the graph are simply defined by the edges of the mesh; for 3D point clouds, edges can be defined by connecting each vertex with its K-nearest neighbors (K is typically set to 6). Denote the neighborhood of vertex $\mathbf{v}_i$ by $\mathcal{N}_i$, and an edge e_{ij} is defined between each neighboring vertex $\mathbf{v}_j$ and $\mathbf{v}_i$. So, we have $\mathcal{E} = \{e_{ij} \mid \mathbf{v}_j \in \mathcal{N}_i, \mathbf{v}_i \in \mathcal{V}\}$. Similar to the data term, we define a differential matrix $\mathbf{K} \in \{-1, 1\}^{|\mathcal{E}|\times|\mathcal{V}|}$ on the graph $\mathcal{G}$ for concise presentation. Concretely, each row of $\mathbf{K}$ corresponds to an edge in $\mathcal{E}$ and each column corresponds to a vertex in $\mathcal{V}$. Each row in $\mathbf{K}$ has only two nonzero entries. For example, assuming the r^{th} row in $\mathbf{K}$ is associated with edge e_{ij}, then the entry related to the reference vertex $\mathbf{v}_i$ is set at 1, while the one related to the neighboring vertex v_j is set at -1, i.e. $k_{ri} = 1$ and $k_{rj} = -1$. Let $\mathbf{k}_{i:}$ denote the i^{th} row of $\mathbf{K}$. We introduce a matrix $\mathbf{B} \in R^{|\mathcal{E}|\times 4|\mathcal{V}|}$, where the i^{th} row of $\mathbf{B}$ is defined as $\mathbf{b}_{i:} := \mathbf{k}_{i:} \otimes \mathbf{v}_i^\top$ and $\otimes$ denotes the operator of Kronecker product. Therefore, the cost of transformation smoothness is rewritten as

$$E_{smooth}\left(\mathbf{X}\right) = \left\|\mathbf{B}\mathbf{X}\right\|_1. \tag{6}$$

Orthogonality constraint: Especially for partial meshes with large motions, the problem may be underconstrained leading to large distortions. In this case, the orthogonality constraint as defined below is effective in better preserving local shapes and making the solution more reasonable.

$$\begin{aligned}E_{orth}\left(\mathbf{X}; \mathbf{R}_i\right) &= \sum_{i=1}^{N} \left\|\mathbf{S}\mathbf{X}_i - \mathbf{R}_i\right\|_F^2, \\ \text{s.t.} \quad \mathbf{R}_i^T\mathbf{R}_i &= \mathbf{I},\ det(\mathbf{R}_i) > 0,\end{aligned} \tag{7}$$

where $\mathbf{R}_i$ is a 3×3 rotation matrix, and $\mathbf{S} = \left[\begin{smallmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\end{smallmatrix}\right]$ is a constant 3×4 matrix that extracts the rotation component of $\mathbf{X}_i$. $det(\mathbf{R}_i) > 0$ ensures that $\mathbf{R}_i$ is a rotation matrix, not a reflection matrix.

The final energy function has the following compact form with matrix-vector notations:

$$\begin{aligned}\min_{\mathbf{X},\mathbf{R}_i} \left\|\mathbf{W}\left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right)\right\|_1 + \alpha\left\|\mathbf{B}\mathbf{X}\right\|_1 + \beta\sum_{i=1}^{N}\left\|\mathbf{S}\mathbf{X}_i - \mathbf{R}_i\right\|_F^2, \\ \text{s.t.} \quad \mathbf{R}_i^T\mathbf{R}_i = \mathbf{I},\ det(\mathbf{R}_i) > 0.\end{aligned} \tag{8}$$

Algorithm 1. Algorithm of reweighting non-rigid registration

```
Input: template V, target U.
  While not converged do
    Find correspondence mapping f^(l) : V ↦ U;
    Update W_D^(l) and W_S^(l) acco. to (10) and (11), resp.
    Solve transformations X^(l) via Algorithm (2);
  End while
Output: X
```

Algorithm 2. ADMM algorithm to solve (9)

```
Input: Ũ_{f^(l)} ∈ R^{N×3}, V ∈ R^{N×4N}, B ∈ R^{|E|×4|V|};
  Initialize: X^(l,0) = X^(l−1), Y_1^(0), Y_2^(0) = 0;
              μ1, μ2 > 0, ρ1, ρ2 > 1;
  While not converged do
    Solve C^(l,k+1) by (16);
    Solve A^(l,k+1) by (18);
    Solve R_i^(l,k+1) by (19);
    Solve X^(l,k+1) by (21)∼(22);
    Update μ_1^(k+1), and μ_2^(k+1) according to (15);
    Update Y_1^(k+1), and Y_2^(k+1) according to (15);
  End while
Output: X^(l).
```

Reweighting: In a sparse representation, the gap between the convex ℓ_1-norm and the noncovex ℓ_0-norm in measuring sparseness could be filled by reweighting the ℓ_1-norm [11]. To further promote sparsity, both the data term and the smoothness term are weighted, and the weighting matrices are updated at each iteration of non-rigid registration. The weighted version of the double sparsity model (8) is defined as follows:

$$\begin{aligned}\min_{\mathbf{X},\mathbf{R}_i} \left\|\mathbf{W}_{\text{D}}\left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right)\right\|_1 + \alpha\left\|\mathbf{W}_{\text{S}}\mathbf{B}\mathbf{X}\right\|_1 + \beta\sum_{i=1}^{N}\left\|\mathbf{S}\mathbf{X}_i - \mathbf{R}_i\right\|_F^2, \\ \text{s.t.} \quad \mathbf{R}_i^T\mathbf{R}_i = \mathbf{I},\ det(\mathbf{R}_i) > 0.\end{aligned} \tag{9}$$

where $\mathbf{W}_{\text{D}}$ and $\mathbf{W}_{\text{S}}$ are diagonal weighting matrices for the data term and smoothness term, respectively. The weighting matrices are updated according to the ℓ_1-norm of the corresponding entries. For the data term, the weights are updated as

$$\mathbf{W}_{\text{D}}^{(l)}(i,i) = \begin{cases}\frac{1}{\left\|\mathbf{X}_i^{(l-1)}\mathbf{v}_i - \tilde{\mathbf{u}}_{f(i)}^{(l)}\right\|_1 + \epsilon_{\text{D}}}, & f(i) \neq 0, \\ 0, & f(i) = 0,\end{cases} \tag{10}$$

where l represents the index of iteration, ϵ_{D} is a constant to avoid the division-by-zero issue, and is set as 0.01 in the experiments. Similarly, the weights for the smoothness term are updated as

$$\mathbf{W}_{\text{S}}^{(l)}(i,i) = \frac{1}{\left\|\mathbf{X}_i^{(l-1)}\mathbf{v}_i - \mathbf{X}_j^{(l-1)}\mathbf{v}_i\right\|_1 + \epsilon_{\text{S}}}, \tag{11}$$

where ϵ_{S} is a constant which is set as 0.01 in the experiments, and the r^{th} row of matrix $\mathbf{B}\mathbf{X}$ is associated with edge e_{ij} between $\mathbf{v}_i$ and $\mathbf{v}_j$. The reweighting scheme is incorporated into the iterative registration framework, which only slightly increases the computation to calculate the weights. The reweighted ℓ_1-norm is also related to robust kernels [41] in suppressing the influence of outliers, although we propose a different formulation that works well in our sparse non-rigid registration framework.

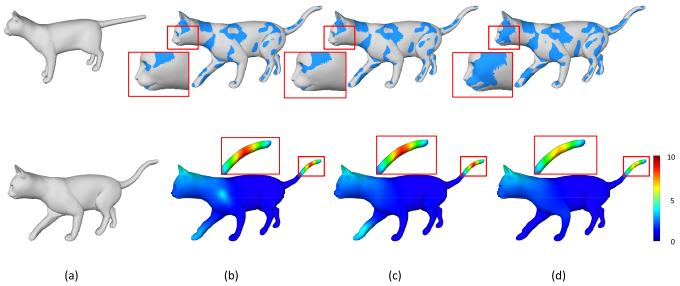


Figure 2. (a) Template (top) and target (bottom) shapes, (b)-(d): Comparison results (top) and registration errors (bottom) of (b) ℓ_2-norm method, (c) SNR method [40] and (d) Our method on *Cat* dataset.

To solve the problem, we first transform the minimization (9) into the following form with auxiliary variables $\mathbf{A}$ and $\mathbf{C}$:

$$\begin{aligned} \min_{\mathbf{X},\mathbf{C},\mathbf{A},\mathbf{R}_i} & \left\|\mathbf{C}\right\|_1 + \alpha \left\|\mathbf{A}\right\|_1 + \beta \sum_{i=1}^{N} \left\|\mathbf{S}\mathbf{X}_i - \mathbf{R}_i\right\|_F^2, \\ \text{s.t.} \quad & \mathbf{C} = \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right), \\ & \mathbf{A} = \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X}, \mathbf{R}_i^T\mathbf{R}_i = \mathbf{I},\ det(\mathbf{R}_i) > 0. \end{aligned} \tag{12}$$

Then, we solve the constrained minimization (12) using the augmented Lagrangian method (ALM) [4]. The ALM method converts the original problem (12) to iterative minimization of its augmented Lagrangian function:

$$\begin{aligned} & L(\mathbf{X}, \mathbf{C}, \mathbf{A}, \{\mathbf{R}_i\}, \mathbf{Y}_1, \mathbf{Y}_2, \mu_1, \mu_2) = \left\|\mathbf{C}\right\|_1 + \alpha \left\|\mathbf{A}\right\|_1 \\ & + \left\langle \mathbf{Y}_1, \mathbf{C} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right) \right\rangle \\ & + \frac{\mu_1}{2} \left\| \mathbf{C} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f\right) \right\|_F^2 \\ & + \left\langle \mathbf{Y}_2, \mathbf{A} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X} \right\rangle + \frac{\mu_2}{2} \left\| \mathbf{A} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X} \right\|_F^2 \\ & + \beta \sum_{i=1}^{N} \left\|\mathbf{S}\mathbf{X}_i - \mathbf{R}_i\right\|_F^2, \\ \text{s.t.} \quad & \mathbf{R}_i^T\mathbf{R}_i = \mathbf{I},\ det(\mathbf{R}_i) > 0, \end{aligned} \tag{13}$$

where $(\mu_1,\ \mu_2)$ are positive constants, $(\mathbf{Y}_1,\mathbf{Y}_2)$ are Lagrangian multipliers, and $\langle \cdot, \cdot \rangle$ denotes the inner product of two matrices considered as long vectors. Under the standard ALM framework, $(\mathbf{Y}_1,\ \mathbf{Y}_2)$ and $(\mu_1,\ \mu_2)$ can be efficiently updated. However, each iteration has to solve $\mathbf{A}$, $\mathbf{C}$, $\{\mathbf{R}_i\}$ and $\mathbf{X}$ simultaneously, which is difficult and computationally demanding. Hence, we resort to the alternate direction method of multipliers (ADM) [9] to optimize $\mathbf{A}$, $\mathbf{C}$, $\{\mathbf{R}_i\}$ and $\mathbf{X}$ separately at each iteration. Detailed derivation of the ADMM algorithm is referred to Appendix A.

The iterative non-rigid registration with reweighting is summarized in Algorithm 1, and the algorithm for minimizing the Eq. (9) is summarized in Algorithm 2 (see the Appendix for the detailed derivation).

5 Experimental Results

In this section, we evaluate the performance of the proposed method on clean datasets (Section 5.1), noisy datasets (Section 5.2), and real scans (Section 5.3). Running times of our method are reported in Section 5.4. All the experiments are performed on a desktop computer with an Intel i5 3.2GHz CPU and 8GB RAM. The numbers of inner and outer iterations of our method are both set as 20.

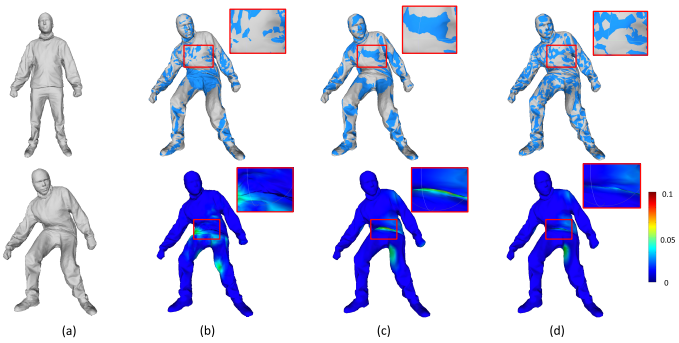


Figure 3. (a) Template (top) and target (bottom) shapes, (b)-(d): Comparison results (top) and registration errors (bottom) of (b) ℓ_2-norm method, (c) SNR method [40] and (d) Our method on *Jumping* dataset.

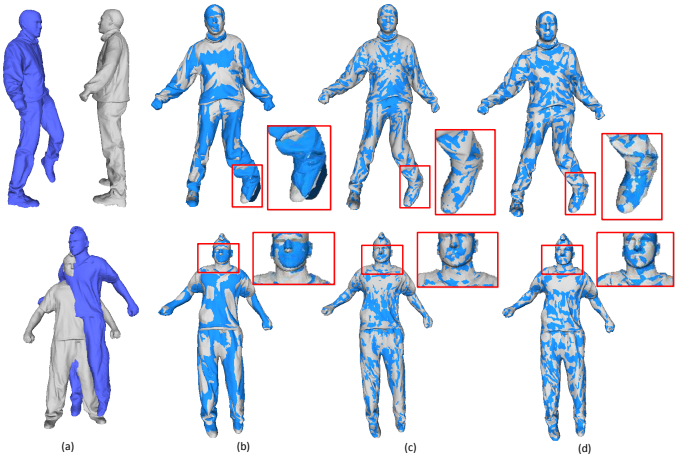


Figure 4. Comparison results on *Bouncing* dataset: (a) Template and target, (b) The method in [21], (c) SNR method [40], and (d) Our method.

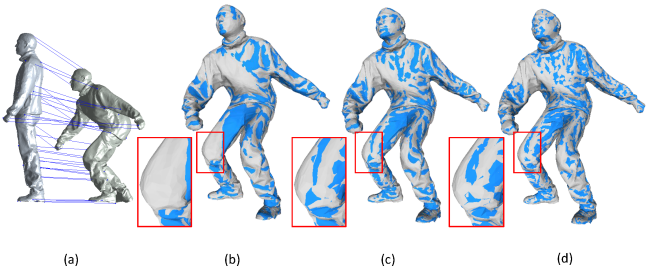


Figure 5. Comparison results on *Jumping* dataset with 35 manually-specified correspondences: (a) Given correspondences, (b) ℓ_2-norm method, (c) SNR method [40], and (d) Our method

5.1 Results on Clean Datasets

We evaluate the proposed method on two datasets: TOSCA high-resolution dataset [10] and a human motion dataset [38]. Fig. 2 and Fig. 3 give the registration results on a particular pair of *cat* and *jumping* datasets, compared with the classic ℓ_2-norm regularized non-rigid ICP method and the SNR method [40]. The results are shown as the overlap of the deformed template shape (blue) and the target shape (gray) and the registration errors are color-coded on the reconstructed mesh for visual inspection. Denote $\mathbf{g}_i$ as the ground-truth correspondence of $\mathbf{v}_i$. For a vertex $\mathbf{v}_i$, the registration error is defined as $\|\mathbf{X}_i\mathbf{v}_i - \mathbf{g}_i\|_2^2$. The compared classic ℓ_2-norm based non-rigid ICP method [7] is formulated as

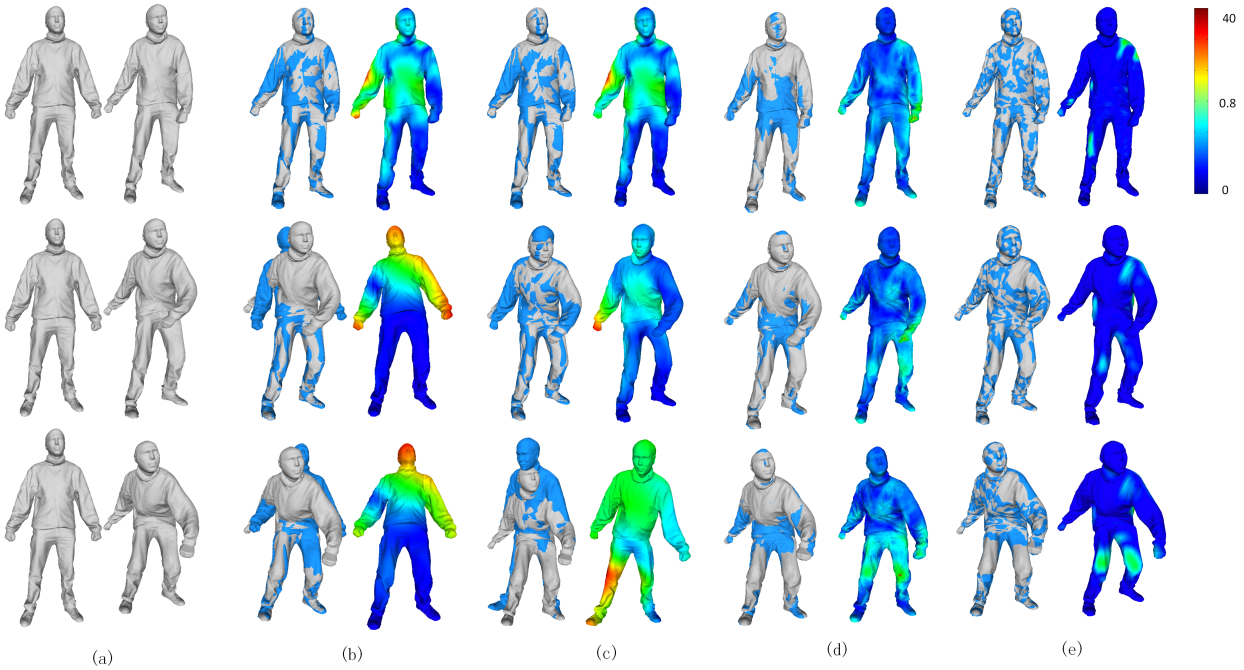


Figure 6. Registration results on *Jumping* dataset with deformation degree increases from top to bottom: (a) Template and target, (b) ICP + ℓ_0-norm method [17], (c) Diffusion pruning (DP) + ℓ_0-norm method [17], (d) The SNR method [40], and (e) Our method.

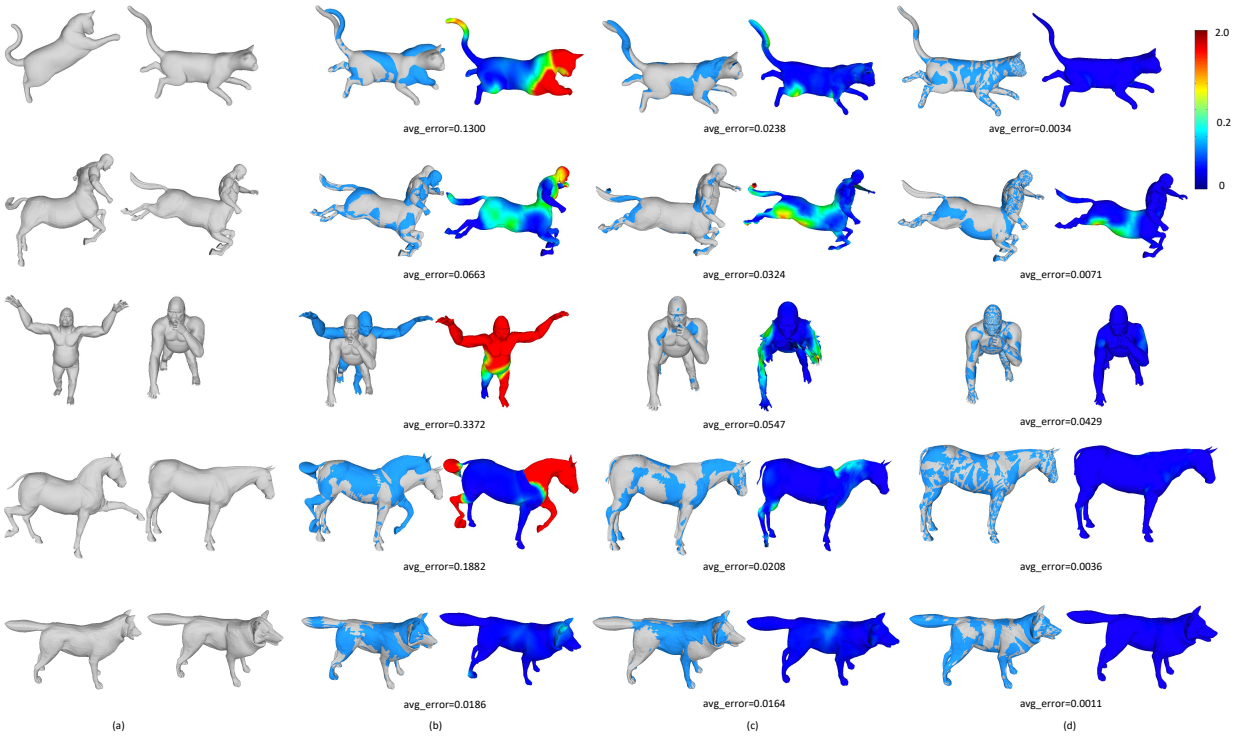


Figure 7. Registration results on TOSCA dataset: (a) Template and target, (b) Diffusion pruning (DP) + ℓ_0-norm method [17], (c) The SNR method [40], and (d) Our method.

optimizing:

$$\min_{\mathbf{X}} \quad \left\|\mathbf{W}(\mathbf{V}\mathbf{X}-\tilde{\mathbf{U}}_f)\right\|_F^2 + \alpha\left\|\mathbf{B}\mathbf{X}\right\|_F^2 . \tag{14}$$

The smoothness constraint of this kind of methods is imposed on the transformation differences. To ensure fair comparison, we adjust the weight α until we get the most accurate registration without loss of smoothness for each method. The result shows that our method achieves the best results with less registration errors in the areas with intensive deformations than the SNR method [40] and the classic ℓ_2-norm regularized non-rigid ICP method, such as the tail of the cat and the wrinkles around the waist of the person highlighted in rectangles.

We compare our method with a state-of-the-art non-rigid registration method [21] in Fig. 4. Obvious registration errors can be seen in the result of the method in [21], especially in the right foot (top) and head (bottom), while the methods with

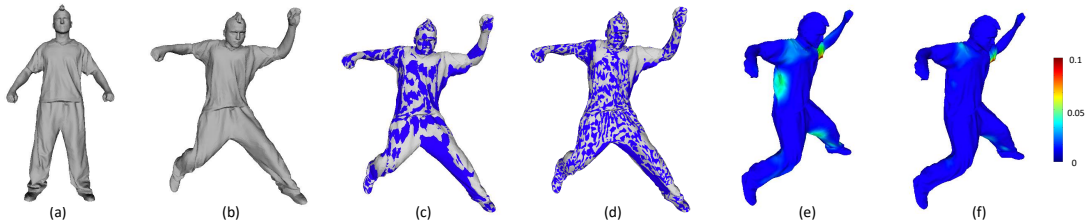


Figure 9. Comparison results with and without reweighting scheme on *Bouncing* dataset: (a) Template, (b) Target, (c) Registration result without reweighting scheme, (d) Registration result with reweighting scheme, (e) Registration errors without reweighting scheme, and (f) Registration errors with reweighting scheme.

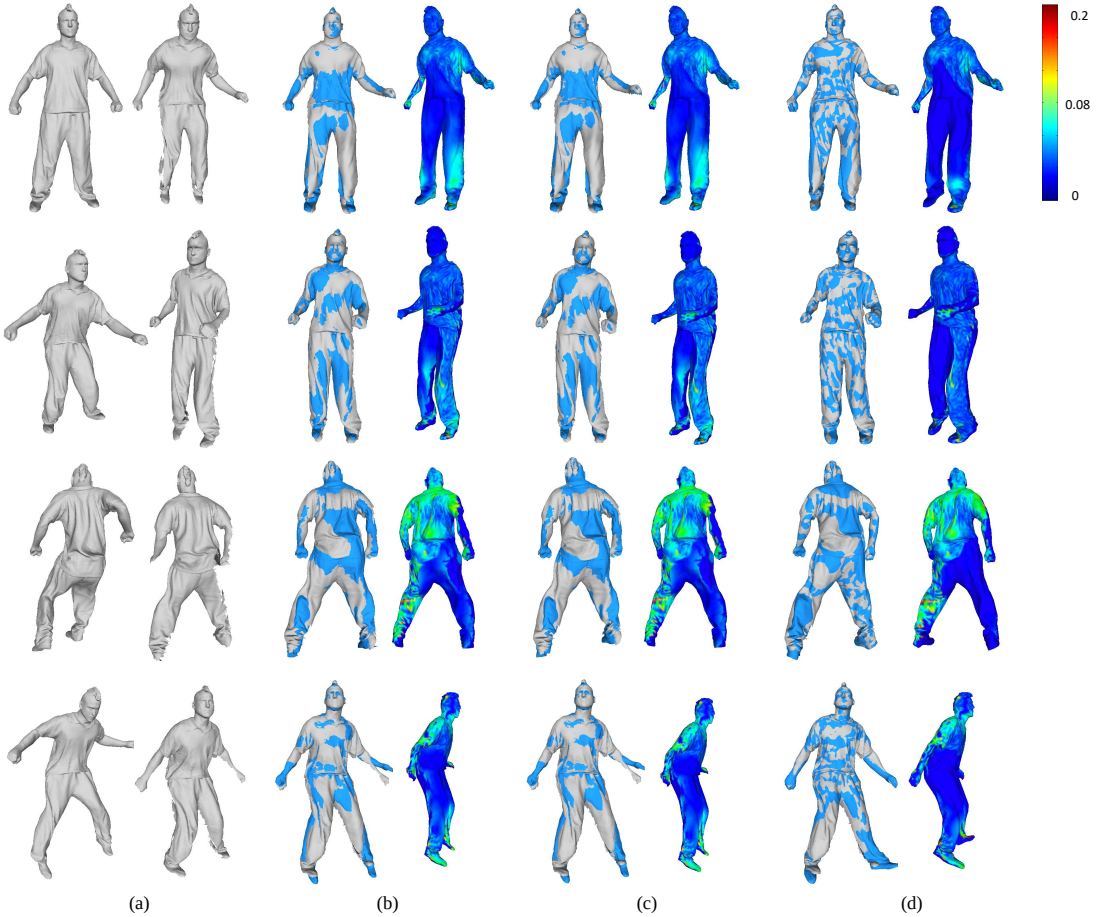


Figure 10. (a) Template (left) and target (right) shapes, (b)-(d): Comparison results (left) and registration errors (right) of (b) ℓ_2-norm method, (c) SNR method [40] and (d) our method on *Bouncing* whole-to-part dataset with frames 1, 21, 51, 84 as templates and frames 87, 107, 137, 171 as targets.

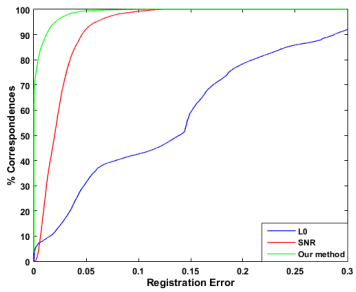


Figure 8. Fraction of correspondences within the error threshold. The graph shows the fraction of correspondences (y-axis) within the error threshold (x-axis).

sparse representation (SNR [40] and our method) achieve better registration results. The method in [21] works effectively when the template and target shapes are close so that good initial correspondences can be obtained, but the pose changes substantially in this example. Moreover, our result is more accurate and better-distributed for the whole body than the SNR method [40], due to the sparse constraint on the position.

To evaluate the robustness of the proposed method, we manually assign 35 correspondences on *Jumping* dataset, and compare the result of our method with the SNR method [40] and the ℓ_2-regularized method. As shown in Fig. 5, our method achieves the best result, especially around the places with substantial deformation, e.g., the right knee.

We also compare our method with a state-of-the-art method [17] that uses ℓ_0 norm in Fig. 6. The code from the authors is used. The original method [17] uses ICP correspondences. When registering scans with large deformations, correspondences

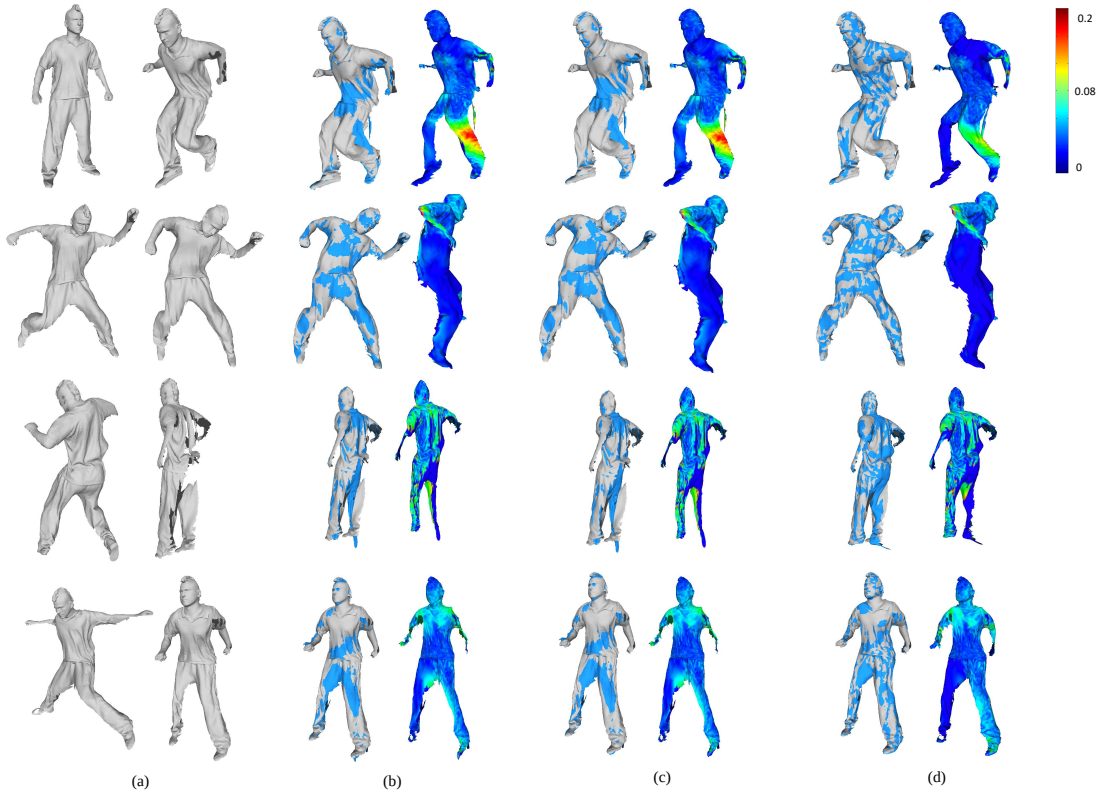


Figure 11. (a) Template (left) and target (right) shapes, (b)-(d): Comparison results (left) and registration errors (right) of (b) ℓ_2-norm method, (c) SNR method [40] and (d) Our method on *Bouncing* part-to-part dataset with frames 1, 21, 51, 84 as templates and frames 87, 107, 137, 171 as targets.

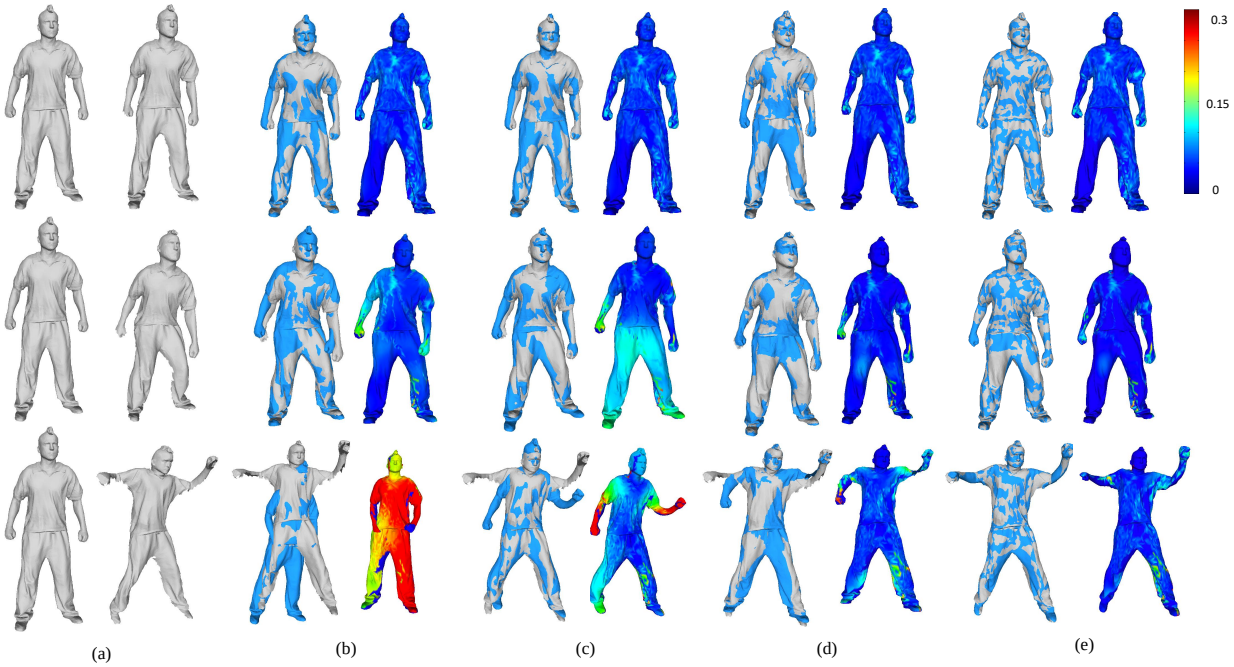


Figure 12. Registration results on *Bouncing* whole-to-part dataset with deformation degree increases from top to bottom: (a) Template and target, (b) ICP + ℓ_0-norm method [17], (c) Diffusion pruning (DP) + ℓ_0-norm method [17], (d) The SNR method [40], and (e) Our method.

derived from intrinsic geometric properties can be more effective. To ensure fair comparison, we compare our method with two versions of the ℓ_0-norm method: ICP + ℓ_0-norm method that uses ICP to compute correspondences, and Diffusion pruning (DP) + ℓ_0-norm method that computes correspondences using [36] as initialization like our method. Fig. 6 gives the registration results for three different degrees of deformation (increasing from top to bottom). It can be seen that with moderate deformation (top row), both variants of [17] work reasonably well. However, when the deformation is large, the method fails to align the two surfaces, resulting in large errors. On the contrary, our method achieves more accurate results than the other methods. The quantitative

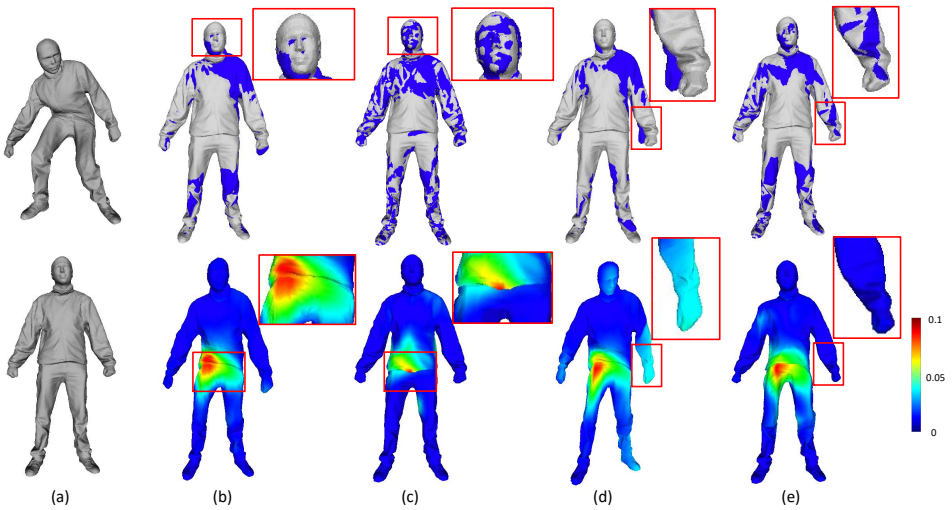


Figure 13. Comparison results on *Jumping* dataset with partially incorrect correspondences: (a) Template and target, (b) SNR method [40] result with one third SHOT correspondences, (c) Our method result with one third SHOT correspondences, (d) SNR method [40] result with all SHOT correspondences, and (e) Our method result with all SHOT correspondences.

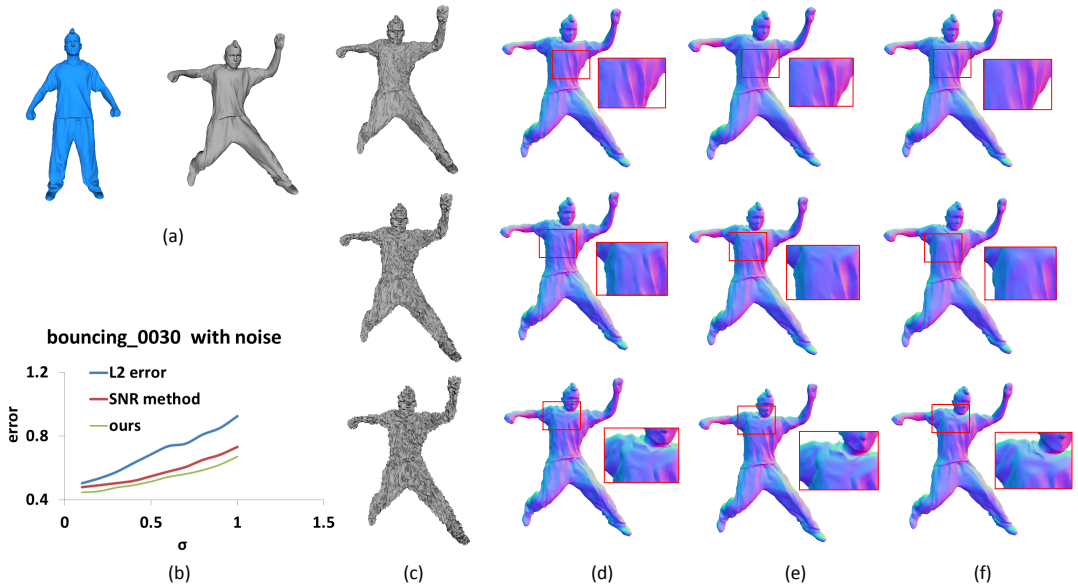


Figure 14. Comparison results on *Bouncing* with noise ($\sigma = 0.3, 0.7, 1$). (a) Template and target, (b) Curves of registration errors vs. normalized noise levels, (c) Target with noise, (d) ℓ_2-norm method, (e) SNR method [40], and (f) Our method.

evaluation (the mean of registration errors over all the vertices) is shown in Table 1. While diffusion pruning helps reduce the errors of [17] in most cases, our method has significantly smaller errors than both variants and the SNR method [40].

Table 1
Quantitative evaluation for Fig. 6.

Deformation	Mean Error			
	ICP + ℓ_0-norm	DP + ℓ_0-norm	SNR	Ours
Small	0.0148	0.0142	0.0121	**0.0003**
Median	0.1707	0.0371	0.0221	**0.0023**
Large	0.2011	0.2029	0.0247	**0.0038**

To more comprehensively evaluate the proposed method, we test our method on five sets of shapes from the TOSCA high-resolution dataset [10], compared with the Diffusion pruning (DP) + ℓ_0-norm method [17] and the SNR method [40]. The initial correspondences are uniformly selected 5% ground-truth correspondences. We test every pair of models in each set (treating one as template and the other as target), and the quantitative evaluation (the mean of registration errors over all the vertices for all the models) is shown in Table 2. Some examples are given in Fig. 7. The average errors are shown in the subfigures. Fig. 8 shows the fraction of correspondences (y-axis) within the error threshold (x-axis) [19]. Our method (green curve) detects nearly 100% correct correspondences for a small threshold of 0.05. The results show that our method achieves the most accurate and robust non-rigid registration.

To evaluate the effectiveness of the proposed reweighting scheme, we compare the registration results with and without reweighting on *Bouncing* dataset in Fig. 9. The parameters ϵ_{D} and ϵ_{S} are set as 0.01. As shown in the figure, the reweighting scheme significantly improves the registration results.

We also evaluate our method on the harder whole-to-part and part-to-part registration problems. Since the 3D models in the

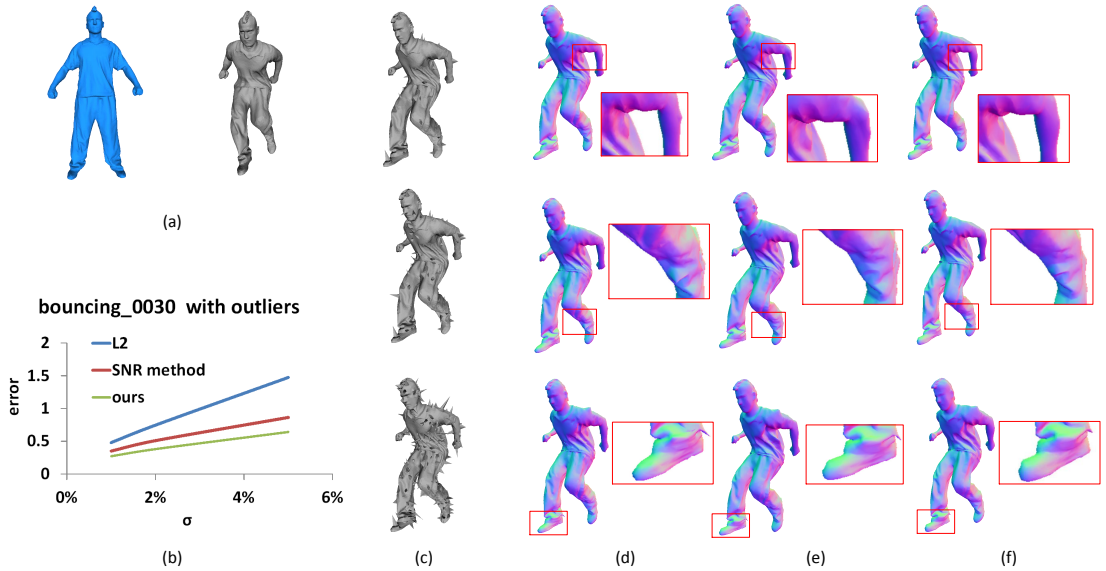


Figure 15. Comparison results on *Bouncing* with 1%, 2%, 5% outliers. (a) Template and target, (b) Curves of registration errors vs. normalized noise levels, (c) Target with noise, (d) ℓ_2-norm method, (e) SNR method [40], and (f) Our method.

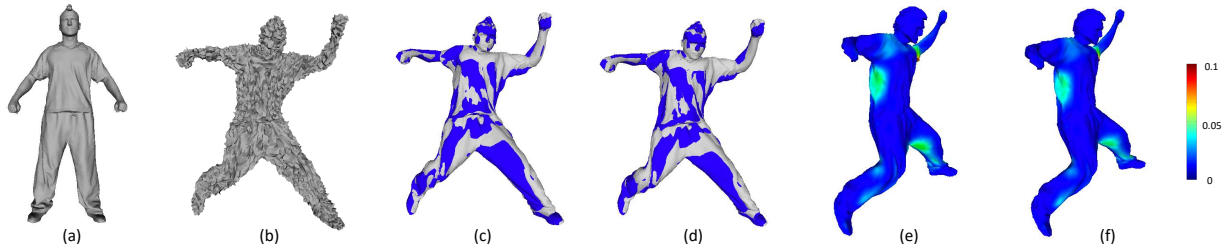


Figure 16. Comparison results with and without reweighting scheme on *Bouncing* dataset with noise ($\sigma = 1$): (a) Template, (b) Target, (c) Registration result without reweighting scheme, (d) Registration result with reweighting scheme, (e) Registration errors without reweighting scheme, and (f) Registration errors with reweighting scheme.

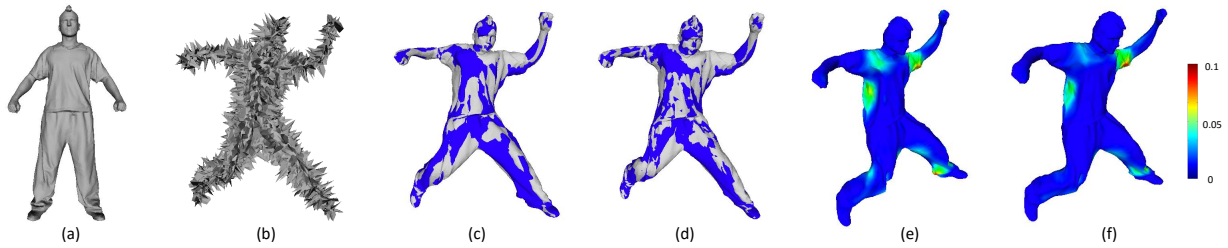


Figure 17. Comparison results with and without reweighting scheme on *Bouncing* dataset with 50% outliers: (a) Template, (b) Target, (c) Registration result without reweighting scheme, (d) Registration result with reweighting scheme, (e) Registration errors without reweighting scheme, and (f) Registration errors with reweighting scheme.

Table 2
Quantitative evaluation for five sets of shapes from the TOSCA dataset.

Method	*Cat*	*Centaur*	*Gorilla*	*Horse*	*Wolf*
DP + ℓ_0-norm	2.6366	0.7311	22.1740	6.6398	0.0545
SNR	0.0902	0.1799	5.364	0.0865	0.0359
Ours	**0.0090**	**0.0297**	**0.5315**	**0.0089**	**0.0012**

public dataset are complete, we obtain partial models by extracting the visible part of each complete model with a virtual depth camera rotating around the model, while keeping the ground truth correspondences. Fig. 10 and Fig. 11 give the whole-to-part and part-to-part registration results on the *Bouncing* dataset which has a total of 172 models. We use a systematic approach that takes the models at frames 1-86 as templates and the models at frames 87-172 as targets such that frame t is registered to frame $t + 86$. We have compared the methods on all the models in this dataset, and Fig. 10 and Fig. 11 show the registration results for 4 frames. The mean of registration errors over all the vertices in the overlapping regions for all the models in the entire dataset of ℓ_2-norm method, SNR method [40] and our method in the whole-to-part case are 0.0323, 0.0324 and 0.0262, respectively. The mean of registration errors over all the vertices in the overlapping regions for all the models in the entire dataset of ℓ_2-norm method, SNR method [40], and our method in the part-to-part case are 0.0380, 0.0381 and 0.0318, respectively. It can be seen that our method achieves the best registration with the smallest errors among these methods. For part-to-part registration, our method reduces errors by more than a half, compared with state-of-the-art sparse non-rigid registration

method [40]. We also compare with the ℓ_0-norm method [17] for the whole-to-part case in Fig. 12. The quantitative evaluation (the mean of registration errors over the whole model) is shown in Table 3. The ℓ_0-norm method [17] has similar limitations to handle large deformations, while our method gives more accurate and robust results.

Table 3
Quantitative evaluation for Fig. 12.

Deformation	Mean Error			
	ICP + ℓ_0-norm	DP + ℓ_0-norm	SNR	Ours
Small	0.0279	0.0275	0.0269	**0.0261**
Median	0.0413	0.0541	0.0249	**0.0195**
Large	0.3390	0.0977	0.0430	**0.0284**

5.2 Results on Noisy Datasets

1) Correspondences with partially incorrect matchings:

It is common to include incorrect correspondences using established methods. We simulate this in two cases. In the first case, we obtain two thirds of correspondences using diffusion pruning [36] and the remaining one third using local geometric feature matching based on SHOT signatures [31]. The majority of correspondences from the former are correct while many correspondences from the latter are incorrect due to the ambiguity of local features. In the second case, we generate all the correspondences using SHOT signatures. Fig. 13 gives the results for the two cases in a difficult situation which involves very complex transformations from template to target. As shown in the figure, our method is significantly more robust than the SNR method [40] with respect to incorrect correspondences. The mean of registration errors over all the vertices for the four cases in Fig. 13 (b-e) are 0.038, 0.012, 0.047, and 0.032, respectively.

2) Target shapes with noise or outliers:

In the first case, 3-D shapes of targets are polluted with dense noise along the normal directions of the associated vertices. All the target vertices are perturbed with Gaussian noise. The standard deviation of the noise σ is normalized by $\bar{l}$, where $\bar{l}$ is the average length of triangle edges on the associated target mesh, and chosen in the range of $[0.1, 1]$. Fig. 14 gives the registration results compared with the SNR method [40] and the ℓ_2-norm regularization method. The results show that our method is more robust to noise, performing significantly better for models with high noise levels.

In the second case, 3-D shapes of targets are polluted with sparse outliers along the normal directions of the associated vertices. Fig. 15 gives the results for the situations when 1%, 2%, 5% of target vertices are perturbed with Gaussian noise. The results show that our method is more robust than the other two methods, particularly for cases with larger proportion of outliers.

To evaluate the effectiveness of the proposed reweighting scheme, we also compare the registration results with and without reweighting for noise and outlier cases on *Bouncing* dataset in Fig. 16 and Fig. 17. The parameters ϵ_{D} and ϵ_{S} are set as 0.01. The standard deviation of the noise σ is set as 1, and the percentage of outliers is set as 50%. It can be seen that the reweighting scheme contributes significantly to improving the registration results for the dataset with noise and outliers.

We compare the registration results with different parameter settings for the reweighting scheme on *Bouncing* dataset with 50% outliers in Fig. 18 to evaluate the influence of the paremeters ϵ_{D} and ϵ_{S}. To make experiments more tractable, we adjust both parameters consistently (i.e. $\epsilon_{\mathrm{D}} = \epsilon_{\mathrm{S}} = \epsilon$). It can be seen that the best setting is 0.006 for this case, which has the smallest registration errors. However, the performance is quite close, and 0.01 is a generally good choice (found in experiments).

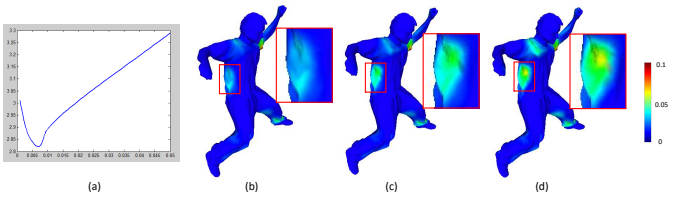


Figure 18. Comparison results with different parameter settings for the reweighting scheme on *Bouncing* dataset with 50% outliers: (a) Curves of registration errors vs. ϵ values, (b) Registration result with $\epsilon = 0.006$, (b) Registration result with $\epsilon = 0.01$, and (d) Registration result with $\epsilon = 0.05$.

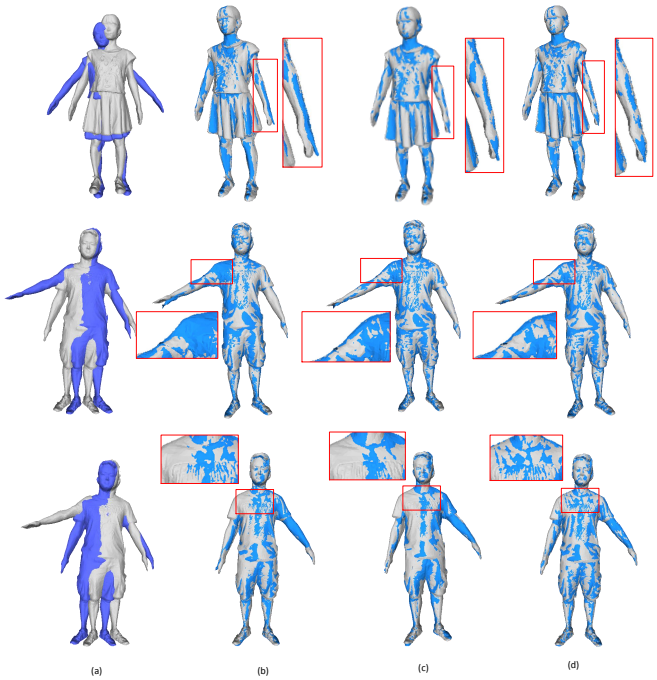


Figure 19. Comparison results on Kinect datasets: (a) Template and target, (b) ℓ_2-norm method, (c) SNR method [40], and (d) Our method.

5.3 Results on Real Scans

Fig. 19 presents the results on real scans generated by Kinect Fusion [23] using Kinect V2.0. The real scans are very challenging, because they have much noise and a large number of outliers. Moreover, each mesh is incomplete and the topology between the template and the target is inconsistent. Hence, it is difficult to obtain sufficient and reliable correspondences. The overlap of the deformed template and the target shows that the ℓ_2-norm regularization method and the SNR method present misalignments around the hands, arms and some other joints which have large deformations, while the result of our method is well-distributed and better registered.

We also compare our method with the ℓ_0-norm method [17] on the *Kongfu* dataset [17] in Fig. 20, where pairs of (non-adjacent)

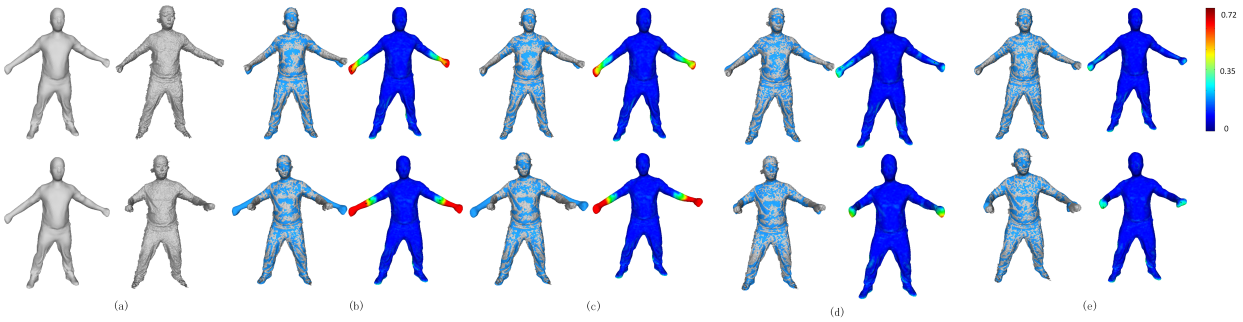


Figure 20. Registration results on *Kongfu* dataset with deformation degree increases from top to bottom: (a) Template and target, (b) ICP + ℓ_0-norm method [17], (c) Diffusion pruning (DP) + ℓ_0-norm method [17], (d) The SNR method [40], and (e) Our method.

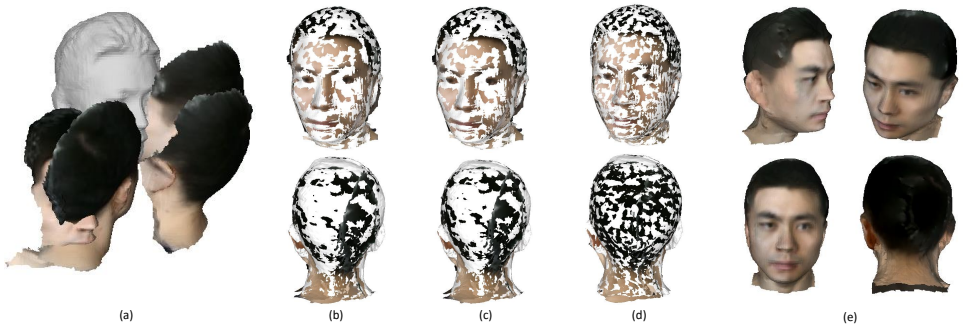


Figure 21. Comparison results on Kinect datasets: (a) Base mesh and four partial color meshes, (b) Registered results of ℓ_2-norm method, (c) Registered results of SNR method [40], (d) Registered results of our method, and (e) Texture fusion results of our method.

frames with increasing degree of deformation are used as input. Our method clearly outperforms both variants of [17], especially for the hands where significant movements exist between scans. The quantitative evaluation (the mean of registration errors over the whole model) is shown in Table 4. Our method has the smallest errors.

Table 4
Quantitative evaluation for Fig. 20.

Deformation	Mean Error			
	ICP + ℓ_0-norm	DP + ℓ_0-norm	SNR	Ours
Small	0.0035	0.0035	0.0031	**0.0033**
Large	0.0039	0.0039	0.0033	**0.0028**

Fig. 21 gives an example of generating a complete color mesh for a human head. A base mesh is scanned by Kinect Fusion using Kinect V2.0, and four partial color meshes are registered to the base mesh using our method. The textures are blended by solving the Poisson equation over the surface of mesh [13]. As shown in the figure, our method correctly registers the input view surfaces with better registration than alternative methods, and successfully generates a watertight color mesh.

5.4 Running times

We compare the running times of the proposed method with the ℓ_2-norm regularized method, SNR method, and group sparsity method on *Crane* dataset. We downsample the meshes into smaller meshes with 1K to 10K vertices. The number of NICP registration iterations for each method is set as 20. The comparison results are shown in Table 5. Our method has similar time complexity as SNR.

Table 5
Comparison on running times

Num. vertices	1000	2000	5000	10000
ℓ_2-norm	1.23s	3.51s	12.88s	29.78s
SNR	8.05s	17.36s	52.48s	119.06s
Group sparsity	7.39s	24.83s	59.96s	126.58s
Ours	7.17s	22.13s	55.68s	122.85s

6 Conclusions

This paper proposes a non-rigid registration method with reweighted sparse position and transformation constraints. We formulate the energy function with position and transformation sparsity on both the data term and the smoothness term, and define the smoothness constraint using local rigidity. The double sparsity based non-rigid registration model is equipped with a reweighting scheme, and solved by the alternating direction method under the augmented Lagrangian multiplier framework which has exact solutions and guaranteed convergence. Experimental results on both public datasets and real scans show that our method provides significantly improved results over alternative methods, especially for more challenging cases, and is more robust to noise and outliers.

Acknowledgements

The authors thank Ke Li for her help with some experiments, and thank Shuai Lin for help with comparative experiments with [21]. The authors are also grateful to the Associate Editor and anonymous reviews for their helps in improving this paper.

This work was supported in part by the National Natural Science Foundation of China (Grant 61571322 and Grant 61771339).

APPENDIX A
DERIVATION OF THE ADMM ALGORITHM

Under the ADMM framework, the augmented Lagrangian function (13) is optimized with respect to the variables alternately, yielding the following subproblems to optimize:

$$\begin{cases} \mathbf{C}^{(k+1)} = & \arg\min_{\mathbf{C}} \ \|\mathbf{C}\|_1 \\ & + \left\langle \mathbf{Y}_1^{(k)}, \mathbf{C} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X}^{(k)} - \tilde{\mathbf{U}}_f \right) \right\rangle \\ & + \frac{\mu_1^{(k)}}{2} \left\| \mathbf{C} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X}^{(k)} - \tilde{\mathbf{U}}_f \right) \right\|_F^2, \\ \mathbf{A}^{(k+1)} = & \arg\min_{\mathbf{A}} \ \alpha\|\mathbf{A}\|_1 + \left\langle \mathbf{Y}_2^{(k)}, \mathbf{A} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X}^k \right\rangle \\ & + \frac{\mu_2^{(k)}}{2} \left\| \mathbf{A} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X}^{(k)} \right\|_F^2, \\ \mathbf{R}_i^{(k+1)} = & \arg\min_{\mathbf{R}_i} \ \beta \sum_{i=1}^N \left\| \mathbf{S}\mathbf{X}_i^{(k)} - \mathbf{R}_i \right\|_F^2 \\ & \text{s.t.} \quad \mathbf{R}_i^T\mathbf{R}_i = \mathbf{I}, \ det(\mathbf{R}_i) > 0 \\ \mathbf{X}^{(k+1)} = & \arg\min_{\mathbf{X}} \ \left\langle \mathbf{Y}_1^{(k)}, \mathbf{C}^{(k+1)} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f \right) \right\rangle \\ & + \frac{\mu_1^{(k)}}{2} \left\| \mathbf{C}^{(k+1)} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X} - \tilde{\mathbf{U}}_f \right) \right\|_F^2 \\ & + \left\langle \mathbf{Y}_2^{(k)}, \mathbf{A}^{(k+1)} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X} \right\rangle \\ & + \frac{\mu_2^{(k)}}{2} \left\| \mathbf{A}^{(k+1)} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X} \right\|_F^2 \\ & + \beta \sum_{i=1}^N \left\| \mathbf{S}\mathbf{X}_i - \mathbf{R}_i^{(k+1)} \right\|_F^2, \\ \mathbf{Y}_1^{(k+1)} = & \mathbf{Y}_1^{(k)} + \mu_1^{(k)} \left(\mathbf{C}^{(k+1)} - \mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X}^{(k+1)} - \tilde{\mathbf{U}}_f \right) \right), \\ \mathbf{Y}_2^{(k+1)} = & \mathbf{Y}_2^{(k)} + \mu_2^{(k)} \left(\mathbf{A}^{(k+1)} - \mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X}^{(k+1)} \right), \\ \mu_1^{(k+1)} = & \rho_1 \mu_1^{(k)}, \ \ \rho_1 > 1, \\ \mu_2^{(k+1)} = & \rho_2 \mu_2^{(k)}, \ \ \rho_2 > 1. \end{cases} \tag{15}$$

The $\mathbf{C}$-subproblem has the following closed solution:

$$\mathbf{C}^{(k+1)} = \text{shrink} \left(\mathbf{W}_{\mathrm{D}} \left(\mathbf{V}\mathbf{X}^{(k)} - \tilde{\mathbf{U}}_f \right) - \frac{1}{\mu_1^{(k)}} \mathbf{Y}_1^{(k)}, \frac{1}{\mu_1^{(k)}} \right), \tag{16}$$

where shrink$(\cdot,\cdot)$ is the shrinkage function applied on the matrix element-wise:

$$\text{shrink}\,(x, \tau) = \text{sign}(x) \max(|x| - \tau, 0). \tag{17}$$

The $\mathbf{A}$-subproblem is solved in a similar way:

$$\mathbf{A}^{(k+1)} = \text{shrink} \left(\mathbf{W}_{\mathrm{S}}\mathbf{B}\mathbf{X}^{(k)} - \frac{1}{\mu_2^{(k)}} \mathbf{Y}_2^{(k)}, \frac{\alpha}{\mu_2^{(k)}} \right). \tag{18}$$

The $\mathbf{R}_i$-subproblem can be explicitly solved using Procrustes projection:

$$\begin{aligned} (\mathbf{U}, \mathbf{D}, \mathbf{V}^\top) &= \text{svd}(\mathbf{S}\mathbf{X}_i^k), \\ \mathbf{R}_i^{k+1} &= \mathbf{U}\mathbf{V}^\top, \end{aligned} \tag{19}$$

where svd$(\cdot)$ is the singular value decomposition. If the obtained matrix has a negative determinant, take $\mathbf{R}_i$ with the opposite sign to turn the matrix into a rotation matrix. This step is similar to [33] for minimizing as-rigid-as-possible energy, although our overall alternating optimization is different and more complicated.

Being quadratic, the $\mathbf{X}$-subproblem can be readily solved by using the first-order optimality condition:

$$\begin{aligned} & \left(\mu_1^{(k)} \mathbf{V}^\top \mathbf{W}_{\mathrm{D}}^\top \mathbf{W}_{\mathrm{D}} \mathbf{V} + \mu_2^{(k)} \mathbf{B}^\top \mathbf{W}_{\mathrm{S}}^\top \mathbf{W}_{\mathrm{S}} \mathbf{B} + \beta \sum_{i=1}^N \mathbf{S}^T\mathbf{S} \right) \mathbf{X} \\ & = \mathbf{B}^\top \mathbf{W}_{\mathrm{S}}^\top \left(\mathbf{Y}_2^{(k)} + \mu_2^{(k)} \mathbf{A}^{(k+1)} \right) \\ & + \mathbf{V}^\top \mathbf{W}_{\mathrm{D}}^\top \left(\mathbf{Y}_1^{(k)} + \mu_1^{(k)} \left(\mathbf{C}^{(k+1)} + \mathbf{W}_{\mathrm{D}} \tilde{\mathbf{U}}_f \right) \right) \\ & + \beta \sum_{i=1}^N \mathbf{S}^T \mathbf{R}_i^{(k+1)}. \end{aligned} \tag{20}$$

However, the straightforward matrix inversion in solving (20) is inefficient or even practically impossible for large-scale problems, e.g., registration of tens of thousands of points. This can be relieved by using the LDL decomposition:

$$\begin{aligned} & (\mathbf{L}, \mathbf{D}) = \\ & \text{ldl} \left(\mu_1^{(k)} \mathbf{V}^\top \mathbf{W}_{\mathrm{D}}^\top \mathbf{W}_{\mathrm{D}} \mathbf{V} + \mu_2^{(k)} \mathbf{B}^\top \mathbf{W}_{\mathrm{S}}^\top \mathbf{W}_{\mathrm{S}} \mathbf{B} + \beta \sum_{i=1}^N \mathbf{S}^T\mathbf{S} \right), \end{aligned} \tag{21}$$

where $\mathbf{L}$ and $\mathbf{D}$ are the lower triangular matrix and the diagonal matrix of the LDL decomposition. Then, the linear equations in (20) is solved by solving the following much easier linear systems:

$$\begin{aligned} \mathbf{L}\mathbf{Q} &= \mathbf{V}^\top \mathbf{W}_{\mathrm{D}}^\top \left(\mathbf{Y}_1^{(k)} + \mu_1^{(k)} \left(\mathbf{C}^{(k+1)} + \mathbf{W}_{\mathrm{D}} \tilde{\mathbf{U}}_f \right) \right) \\ &\quad + \mathbf{B}^\top \mathbf{W}_{\mathrm{S}}^\top \left(\mathbf{Y}^{(k)} + \mu_2^{(k)} \mathbf{A}^{(k+1)} \right) + \beta \sum_{i=1}^N \mathbf{S}^T \mathbf{R}_i^{(k+1)} \\ \mathbf{D}\mathbf{Z} &= \mathbf{Q}, \\ \mathbf{L}^\top \mathbf{X} &= \mathbf{Z}. \end{aligned} \tag{22}$$

REFERENCES

[1] B. Allen, B. Curless, and Z. Popović. Articulated body deformation from range scan data. *ACM Trans. Graph.*, 21(3):612–619, 2002.

[2] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, 2003.

[3] B. Amberg, S. Romdhani, and T. Vetter. Optimal step nonrigid icp algorithms for surface registration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[4] D. P. Bertsekas. Constrained optimization and lagrange multiplier methods. *Computer Science and Applied Mathematics, Boston: Academic Press*, 1, 1982.

[5] P. J. Besl and N. D. McKay. Method for registration of 3-D shapes. In *Robotics-DL tentative*, pages 586–606, 1992.

[6] H. Birkholz. A unifying approach to isotropic and anisotropic total variation denoising models. *JCAM*, 235(8).

[7] S. Bouaziz and M. Pauly. Dynamic 2D/3D registration for the Kinect. In *ACM SIGGRAPH*, page 21, 2013.

[8] S. Bouaziz, A. Tagliasacchi, and M. Pauly. Sparse iterative closest point. *Computer Graphics Forum*, 32(5):113–123, 2013.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[10] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008.

[11] E. J. Candes, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *J. Fourier Analysis and Applications*, 14(5-6):877–905, 2008.

[12] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *IEEE International Conference on Robotics and Automation*, pages 2724–2729. IEEE, 1991.

[13] M. Chuang, L. Luo, B. J. Brown, S. Rusinkiewicz, and M. Kazhdan. Estimating the Laplace-Beltrami operator by restricting 3D functions. *Computer Graphics Forum*, 28(5):1475–1484, 2009.
[14] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114–141, 2003.
[15] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor, P. Kohli, V. Tankovich, and S. Izadi. Fusion4D: Real-time performance capture of challenging scenes. *ACM Trans. Graph.*, 35(4):114:1–114:13, 2016.
[16] S. Esedo$\bar{g}$lu et al. Decomposition of images by the anisotropic Rudin-Osher-Fatemi model. *CPAM*, 57(12).
[17] K. Guo, F. Xu, Y. Wang, Y. Liu, and Q. Dai. Robust non-rigid motion tracking and surface reconstruction using L0 regularization. *IEEE International Conference on Computer Vision (ICCV)*, 2015.
[18] H. Hontani, T. Matsuno, and Y. Sawada. Robust nonrigid icp using outlier-sparsity regularization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 174–181, 2012.
[19] V. G. Kim, Y. Lipman, and T. Funkhouser. Blended intrinsic maps. In *ACM Transactions on Graphics (TOG)*, volume 30, page 79. ACM, 2011.
[20] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph.*, 28(5):175, 2009.
[21] H. Li, R. W. Sumner, and M. Pauly. Global correspondence optimization for non-rigid registration of depth scans. *Computer graphics forum*, 27(5):1421–1430, 2008.
[22] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong. Modeling deformable objects from a single depth camera. In *IEEE International Conference on Computer Vision (ICCV)*, pages 167–174, 2009.
[23] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *10th IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 127–136. IEEE, 2011.
[24] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352. IEEE, 2015.
[25] C. Papazov and D. Burschka. Deformable 3d shape registration based on local similarity transforms. In *Computer Graphics Forum*, volume 30, pages 1493–1502. Wiley Online Library, 2011.
[26] Y. Pekelny and C. Gotsman. Articulated object reconstruction and markerless motion capture from depth video. *Computer Graphics Forum*, 27(2):399–408, 2008.
[27] H. Pottmann, Q.-X. Huang, Y.-L. Yang, and S.-M. Hu. Geometry and convergence analysis of algorithms for registration of 3d shapes. *International Journal of Computer Vision*, 67(3):277–296, 2006.
[28] H. Pottmann, S. Leopoldseder, and M. Hofer. Registration without ICP. *Computer Vision and Image Understanding*, 95(1):54–71, 2004.
[29] M. Rouhani, E. Boyer, and A. D. Sappa. Non-rigid registration meets surface reconstruction. In *Proc. 3D Vision*, 2014.
[30] L. Rudin et al. Nonlinear total variation based noise removal algorithms. *PDNP*, 60(1).
[31] S. Salti, F. Tombari, and L. Di Stefano. SHOT: unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125:251–264, 2014.
[32] K. Sidorov, S. Richmond, D. Marshall, et al. Efficient groupwise non-rigid registration of textured surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2401–2408. IEEE, 2011.
[33] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Proc. Symposium on Geometry Processing*, pages 109–116, 2007.
[34] R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. *ACM Trans. Graph.*, 23(3):399–405, 2004.
[35] J. Süßmuth, M. Winter, and G. Greiner. Reconstructing animated meshes from time-varying point clouds. *Computer Graphics Forum*, 27(5):1469–1476, 2008.
[36] G. K. Tam, R. R. Martin, P. L. Rosin, and Y.-K. Lai. Diffusion pruning for rapidly and robustly selecting global correspondences using local isometry. *ACM Trans. Graph.*, 33(1):4, 2014.
[37] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. Scanning 3d full human bodies using kinects. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):643–50, 2012.
[38] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph.*, 27(3):97, 2008.
[39] M. Wand, B. Adams, M. Ovsjanikov, A. Berner, M. Bokeloh, P. Jenke, L. Guibas, H.-P. Seidel, and A. Schilling. Efficient reconstruction of nonrigid shape and motion from real-time 3D scanner data. *ACM Trans. Graph.*, 28(2):15, 2009.
[40] J. Yang, K. Li, K. Li, and Y.-K. Lai. Sparse non-rigid registration of 3D shapes. In *Computer Graphics Forum*, volume 34, pages 89–99, 2015.
[41] M. Zollhöfer, M. Nießner, S. Izadi, C. Rhemann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Trans. Graph.*, 33(4), 2014.

Kun Li received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2006, and the master and Ph.D. degrees from Tsinghua University, Beijing, in 2011.

She visited École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2012 and from 2014 to 2015. She is currently an Associate Professor with the School of Computer Science and Technology, Tianjin University, Tianjin, China. Her research interests include dynamic scene 3D reconstruction and image/video processing. She was selected into Peiyang Scholar Program of Tianjin University in 2016, and got the Platinum Best Paper award in IEEE ICME 2017.

Jingyu Yang (M10-SM17) received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2003, and Ph.D. (Hons.) degree from Tsinghua University, Beijing, in 2009.

He has been a Faculty Member with Tianjin University, Tianjin, China, since 2009, where he is currently a Research Professor with the School of Electrical and Information Engineering. He was with Microsoft Research Asia (MSRA), Beijing, in 2011, within the MSRA's Young Scholar Supporting Program, and the Signal Processing Laboratory, EPFL, Lausanne, Switzerland, in 2012, and from 2014 to 2015. His research interests include image/video processing, 3D imaging, and computer vision. As a co-author, he got the best 10% paper award in IEEE VCIP 2016 and the Platinum Best Paper award in IEEE ICME 2017. He was selected into the program for New Century Excellent Talents in University (NCET) from the Ministry of Education, China, in 2011, the Peiyang Scholar Program of Tianjin University in 2014, and the Innovation Talent Promotion Program from the Tianjin Municipal Commission of Science and Technology in 2015.

Yu-Kun Lai received his bachelor and Ph.D. degrees in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Senior Lecturer of Visual Computing in the School of Computer Science & Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing and computer vision. He is on the editorial board of The Visual Computer.

Daoliang Guo received the B.E. degree from the School of Electronic Information Engineering, Anhui University, Anhui, China, in 2015. He is currently pursuing the M.E. degree at the School of Electrical and Information Engineering, Tianjin University, Tianjin, China. His research interests are mainly in 3D registration and reconstruction.