Robust Non-rigid Motion Tracking and Surface Reconstruction Using L_0 Regularization

Kaiwen Guo[1,2], Feng Xu[1,3], Yangang Wang[4], Yebin Liu[1,2*], Qionghai Dai[1,2*]
[1]Tsinghua National Laboratory for Information Science and Technology, Beijing, China
[2]Department of Automation, Tsinghua University, Beijing, China
[3]School of Software, Tsinghua University, Beijing, China
[4]Microsoft Research

Abstract

We present a new motion tracking method to robustly reconstruct non-rigid geometries and motions from single view depth inputs captured by a consumer depth sensor. The idea comes from the observation of the existence of intrinsic articulated subspace in most of non-rigid motions. To take advantage of this characteristic, we propose a novel L_0 based motion regularizer with an iterative optimization solver that can implicitly constrain local deformation only on joints with articulated motions, leading to reduced solution space and physical plausible deformations. The L_0 strategy is integrated into the available non-rigid motion tracking pipeline, forming the proposed L_0-L_2 non-rigid motion tracking method that can adaptively stop the tracking error propagation. Extensive experiments over complex human body motions with occlusions, face and hand motions demonstrate that our approach substantially improves tracking robustness and surface reconstruction accuracy.

1. Introduction

Acquiring 3D models of deforming objects in real-life is attractive but remains challenging in computer vision and graphics. One kind of approach is to explore the inner structure of deforming objects and use skeleton-based strategy to perform the tracking and reconstruction, *e.g.* human body tracking [10, 14], hand motion capture [17, 8]. However, there are large number of deforming objects which cannot be completely modeled by skeletons, *e.g.* the activity of people grasping a non-rigid deforming pillow (Fig. 6). Besides, the tracking performance is sensitive to the skeleton embedding and the surface skinning [2] strategies, which usually require manual operations to achieve high quality motion tracking [10, 11].

Non-rigid deformation [23, 22, 19] provides an appealing solution for dynamic objects modeling since it does not require the build-in skeletons. The basic idea of non-rigid deformation for objects motion reconstruction is to deform the model vertices to fit the observations at each time step. However, since the parameter space of non-rigid deformation is much larger than that of the skeleton space, and non-rigid deformation usually employs a local optimization, available non-rigid motion tracking methods are easy to fall into local minimums. Furthermore, it suffers from error accumulation, and would usually fail when tracking long motion sequences from noisy and incomplete data obtained by a single depth sensor [26]. Robustly tracking of complex human body and hand motions using non-rigid motion tracking techniques (without embedded skeleton) is still an open problem.

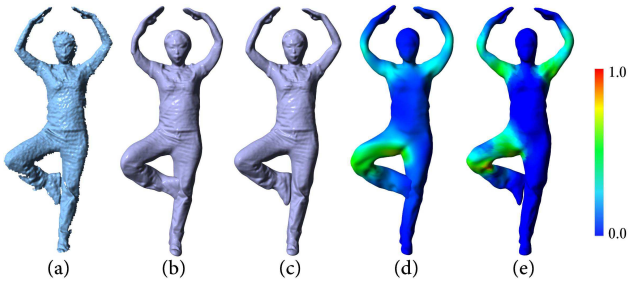


Figure 1. Reconstruction result compared to [12]. (a) input depth; (b) reconstruction result of [12]; (c) reconstruction result of our approach; (d) and (e) color coded normalized motion energy of result (b) and (c), respectively.

In this paper, we observe that most of the non-rigid motions implicitly contain articulated motions, which have strong deformation changes around the joint regions while remain unchanged in other regions. This phenomena indicates that different regions should introduce different degree of smooth deformation priors. When calculating spatial deformation gradient on the object surface, only some joints regions have non-zero gradient values while other surface regions keep zero. Fig.1(d) and (e) show the magnitude of the gradient on two reconstruction results.

Based on this key observation, we contribute a novel

* Corresponding authors: {liuyebin, qhdai}@tsinghua.edu.cn

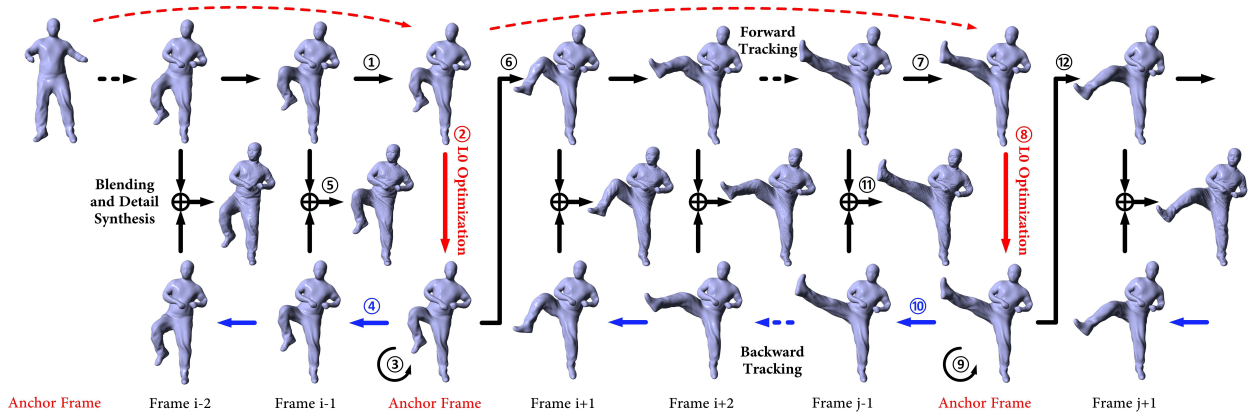


Figure 2. The pipeline of the proposed method. Basically, it is a forward-backward tracking scheme using combined L_2 and L_0 regularization for non-rigid deformation. The first row shows the forward tracking while the last row shows the backward tracking. The middle row is the blending between the two results for temporal smoothness. Please refer to Sect.3 for detailed description.

sparse non-rigid deformation framework to deform a template model to reconstruct non-rigid geometries and motions from a single view depth input via L_0-based motion constraint. In contrast to the widely used L_2 regularizer which sets a smooth constraint for the motion differences between neighboring vertices, the L_0 regularizer constrains local non-rigid deformation only on several significant deformation parts, *i.e.* joints of articulated motion, which greatly reduces the solution space and yields a more physically plausible and therefore a more robust and high quality deformation.

For temporal successive frames, however, the articulated motion deformation is so small that the proposed L_0 regularizer is incapable to distinguish it from non-rigid human surface motion. To this end, we accumulate the motion for temporal frames until the motion change is large enough for articulated motion detection and propose a combined L_0-L_2 tracking strategy that bears L_0 optimization on a small number of anchor frames while keeping other frames being optimized by L_2. To guarantee temporal consistency, we finally refine the non-rigid tracking and reconstruction results between anchor frames in a bidirectional way.

We demonstrate that with monocular depth input captured by a consumer depth sensor, the proposed approach achieves accurate and robust reconstruction of complex non-rigid motions such as human body motions, face expressions, hand motions and the body motion interacting with objects (One example is shown in Fig.1.). Our approach shows more robustness on tracking long sequences (up to 800 frames) with complex motion and significant occlusions, compared with the state-of-the-art non-rigid deformation methods. Furthermore, the technique does not rely on skeleton embedding and skinning weight calculation, thus dramatically reducing the workload of motion reconstruction. The data and source code of our work are made public on the project website.

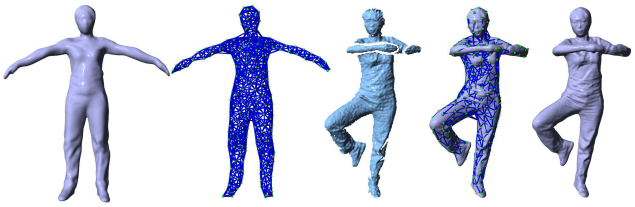
2. Related work

Techniques of non-rigid motion reconstruction have been widely used in recent years. For example, in movie and game industry, motion marker systems (*e.g.*, Vicon[1]) are successfully applied to capture non-rigid motions of human bodies and faces. Nevertheless, these systems are quite expensive and require actors/actresses to stick a large set of optical beacons on bodies or faces. To overcome this drawback, marker-less solutions with video input are extensively investigated in academia in recent decades. Early works on this topic are well surveyed in [16] and [15].

For multi-view video input, the shape of moving objects can be directly reconstructed by shape-from-silhouette [25] or stereo matching [20] methods for each frame. After that, techniques like [4] are able to calculate the correspondences among all frames by a non-sequential registration scheme. Besides, a predefined template model can also be used to reconstruct the motion of an object by deforming it to fit the multi-view video input [3, 7, 5, 6]. Beyond that, a skeleton can be further embedded into the template to better capture kinematic motions of moving objects [24, 10, 14, 21]. Besides color cameras, systems with multiple depth cameras are also proposed in recent years [29, 9]. With the help of the additional depth information, complex motions are expected to be better reconstructed. Although the above solutions reconstruct articulated and/or non-rigid motions without motion markers, the sophisticated multi-view systems are still not easy to build and cannot be applied to general environment, which strictly limit their applications.

Monocular color or depth camera is the most facilitative device for capturing moving objects. For kinematic body motions, Zhu *et al.* [33] reconstructed 3D body skeletons by modeling human actions as a union of subspace. Baak *et al.* [1] and Ye *et al.* [30] identified a similar pose

[1] http://www.vicon.com/



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

Figure 3. Non-rigid registration. (a,b) initial model with nodes and their connectivity; (c) input depth; (d) result of the non-rigid registration; (e) surface refinement.

in a prerecorded database to reconstruct the human pose for a video frame. Wei *et al*. [26] formulated the pose estimation problem as a *Maximum A Posteriori* (MAP) framework to achieve more robust skeleton estimation. However, these techniques only estimate kinematic motions of moving objects, the full surface non-rigid deformations are not reconstructed.

More recently, Wu *et al*. [27] reconstructed the non-rigid body motion with stereo input by exploring BRDF information and scene illumination. Ye and Yang [31] proposed an exponential-maps-based parametrization to estimate 3D poses and shapes. However, these techniques utilize a skeleton to constrain the kinematic motion space, which requires skeleton embedding and skinning weight calculation. These two steps are crucial to the quality of the final result and are difficult to be precisely achieved by automatic methods. Furthermore, the skeleton restricts the techniques to be applied only to articulated objects rather than general objects.

On the other hand, pure non-rigid registration technique is an alternative solution to avoid using skeleton. Liao *et al*. [13] achieved this by stitching partial surfaces at different time instances, thus limited to continuous and predictable motions. Popa *et al*. [18] achieved space-time reconstruction with a gradual change prior, which caused it difficult to handle fast motions and long sequences. Li *et al*. [12] and Zollhöfer *et al*. [34] reconstructed complex motions using template tracking based on ICP-defined correspondence, which achieved the state-of-the-art reconstruction. However, as only smooth motion prior is involved in their deformation models, strong articulated motions and large occlusions are difficult to be handled especially for noisy depth input captured by a consumer Kinect camera. In this paper, we propose a method that combines the benefits of the skeleton based and non-rigid registration based methods and demonstrate robust and accurate surface motion reconstruction from a single-view depth input.

3. Overview

The goal of this work is to reconstruct the non-rigid motion of deforming objects from a single-view depth sequence. Different from existing solutions for reconstructing articulated motions [24, 10], our method does not require the embedding of a predefined skeleton, while has the ability to robustly output 3D deforming mesh sequences of the dynamic objects. In addition to the input depth sequence, the 3D mesh templates of the deforming targets (Fig.3(a)) are needed and can be obtained by depth fusion using a single depth sensor [32]. In this way, the whole pipeline only relies on one off-the-shelf depth camera. After data recording, a rigid and rough alignment of the template to the initial frame of the captured sequence is automatically performed using sample-based global optimization method [10].

The motion tracking and surface reconstruction pipeline is then processed fully automatically as illustrated in Fig.2. Overall, it uses a forward-backward tracking strategy. The traditional L_2 based nonrigid deformation is first performed frame by frame sequentially (step 1 in Fig.2). The reconstructed motion is accumulated until prominent articulated motion is detected at one frame, called *anchor frame*. Then the L_0 based motion regularization is triggered to regularize and refresh the motion in this anchor frame using the reference from the previous anchor frame (step 2 in Fig.2). Such a refreshment effectively stops the cumulative non-rigid deformation error while implicitly refines the underlying articulated motion. An L_2 based nonrigid deformation is further introduced to refine the non-rigid shape on this anchor frame to approximate the input depth (step 3 in Fig.2). To propagate the refreshment to the previous frames, the non-rigid deformation is performed backwards (step 4 in Fig.2) from the latest anchor frame to the previous anchor frame. The final result of one frame in-between is a weighted blending of the reconstruction results of the forward and backward tracking (step 5 in Fig.2), followed by a surface detail refinement (see Fig.3(e)). This strategy goes on from one anchor frame to the next detected anchor frame till the end of the sequence (step 6 to 11 in Fig. 2).

4. Combined L_0-L_2 Tracking

Given the captured depth sequence $\{D^1, D^2, ..., D^n\}$, the proposed L_0-L_2 tracking strategy selects between L_2 based regularizer and L_0 based regularizer for each frame D^t. In the following, we will first overview the available L_2 based non-rigid registration and then introduce our proposed L_0 based motion regularization, followed by our scheme to select between these two regularizers and the overall tracking strategy. The reason why L_0 regularizer can not be applied on all the frames is explained in Sect.4.2 and Sect 4.3.

4.1. L_2 based non-rigid registration

Given a depth frame D^t $(t = 1, ..., n)$, as a temporal tracking strategy, we have a mesh M^{t-1} which is roughly aligned with the current depth D^t. The L_2 based non-rigid registration then takes M^{t-1} as an initialization to further fit it to D^t through non-rigid deformation. For conciseness,

we ignore the time stamp t in the following derivations. Following the state-of-the-art method [12], the deformation of a mesh M is represented by affine transformations $\{\mathbf{A}_i, \mathbf{t}_i\}$ of some sparse nodes $\{\mathbf{x}_i\}$ on the mesh (Fig.3(b)). For a particular mesh vertex $\mathbf{v}_j$, its new position after the non-rigid deformation is formulated as:

$$\mathbf{v}'_j = \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{v}_j)} w(\mathbf{v}_j, \mathbf{x}_i)[\mathbf{A}_i(\mathbf{v}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}_i], \quad (1)$$

where $w(\mathbf{v}_j, \mathbf{x}_i)$ measures the influence of the node $\mathbf{x}_i$ to the vertex $\mathbf{v}_j$. Please refer to [12] for details about extracting $\mathbf{x}_i$ from the mesh and calculating w for all mesh vertices. Given the deformation model, the estimation of $\{\mathbf{A}_i, \mathbf{t}_i\}$ is achieved by minimizing the following energy:

$$E_{\text{tol}} = E_{\text{fit}} + \alpha_{\text{rigid}} E_{\text{rigid}} + \alpha_{\text{smo}} E_{\text{smo}}, \quad (2)$$

where

$$E_{\text{fit}} = \sum_{\mathbf{v}'_j \in \mathcal{C}} \alpha_{\text{point}} \|\mathbf{v}'_j - \mathbf{c}_j\|_2^2 + \alpha_{\text{plane}} |\mathbf{n}_{\mathbf{c}_j}^T (\mathbf{v}'_j - \mathbf{c}_j)|^2. \quad (3)$$

which forces vertex $\mathbf{v}_j$ to move to its corresponding depth point $\mathbf{c}_j$ especially along the normal direction of $\mathbf{c}_j$. $\mathcal{C}$ includes all vertices that have correspondences in the depth D. E_{rigid} restricts the affine transformation to be as rigid as possible, which is formulated as:

$$\begin{aligned} E_{\text{rigid}} = & R(\mathbf{A}_i) = \sum_i \Big((\mathbf{a}_{i1}^T \mathbf{a}_{i2})^2 + (\mathbf{a}_{i2}^T \mathbf{a}_{i3})^2 + (\mathbf{a}_{i3}^T \mathbf{a}_{i1})^2 + \\ & (1 - \mathbf{a}_{i1}^T \mathbf{a}_{i1})^2 + (1 - \mathbf{a}_{i2}^T \mathbf{a}_{i2})^2 + (1 - \mathbf{a}_{i3}^T \mathbf{a}_{i3})^2 \Big), \end{aligned} \quad (4)$$

where $\mathbf{a}_{i1}$, $\mathbf{a}_{i2}$ and $\mathbf{a}_{i3}$ are column vectors of $\mathbf{A}_i$. E_{smo} defines the L_2 regularizer which constrains the consistent motion difference on the spatial domain, namely, the affine transformation of a node should be as similar as possible to its neighboring nodes:

$$E_{\text{smo}} = \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} w(\mathbf{x}_j, \mathbf{x}_i) \|\mathbf{A}_i(\mathbf{x}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}_i - (\mathbf{x}_j + \mathbf{t}_j)\|_2^2. \quad (5)$$

The neighborhood of the nodes is shown as edges in Fig.3(b) and is defined by the method in [12]. The minimization of E_{tol} is performed in an Iterative Closest Point (ICP) framework, where $\mathcal{C}$ is updated by closest point searching and parameters are also updated during the iterations. We exactly follow [12] to set parameters in our implementation. Please refer to their paper for details.

4.2. L_0 based motion regularization

As illustrated in Sect.1, from single-view low quality depth input captured by a consumer depth sensor, pure non-rigid deformation can not robustly and accurately reconstruct objects like human body or human hand, whose motions may have strong occlusions which lead to inaccurate point-to-depth correspondences. But on the other hand, this kind of objects majorly performs articulated motion besides non-rigid motion. To pursue good tracking results, previous works adopt skeleton embedding to explicitly exploit the articulated motion prior, which strictly restrict that possible motion changes only happen on pre-defined skeleton joints and prevent motion changes on other regions. This skeleton embedding is similar to constrain the L_0 norm of spatial motion variation with a pre-defined distribution on the object. Based on this observation, we propose an L_0 based motion regularizer over existing non-rigid surface deformation framework to implicitly utilize the articulated motion prior without the requirement of skeleton embedding.

Attention should be paid here that, the proposed L_0 regularizer can not be applied on every input frame. Intuitively, although the deformation change between two temporal successive frames contains both articulated motion and non-rigid motion, the magnitude of the articulated motion is too small and ambiguous to be distinguished from the non-rigid motion. If L_0 regularizer is applied on these tiny motions, the articulated motions will also be pruned with the non-rigid motions by the L_0 regularizer which will lead to tracking failure. As such, we only apply L_0 regularizer on some anchor frames, and track the kinematic motion and shape of an anchor frame using the previous anchor frame as a reference.

Specifically, given the initial vertex positions $\{\mathbf{v}'_j\}$ of the new anchor frame obtained by the L_2 non-rigid tracking in Sect.4.1, we estimate the refined implicit articulated transformation $\{\mathbf{A}'_i, \mathbf{t}'_i\}$ by minimizing the following energy function:

$$E'_{\text{tol}} = E'_{\text{data}} + \alpha'_{\text{rigid}} E'_{\text{rigid}} + \alpha'_{\text{reg}} E'_{\text{reg}}. \quad (6)$$

Here, E'_{data} constrains that the refined transformation should deform the target object to a similar pose by the L_2 optimization, thus the result still fits the input depth:

$$E'_{\text{data}} = \sum_j \|\mathbf{v}''_j - \mathbf{v}'_j\|_2^2, \quad (7)$$

where $\mathbf{v}''_j$ is the vertex position defined by the refined transformation:

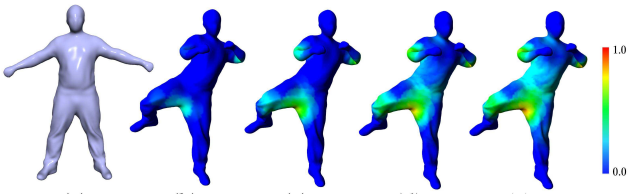
$$\mathbf{v}''_j = \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{v}_j)} w(\mathbf{v}_j, \mathbf{x}_i)[\mathbf{A}'_i(\mathbf{v}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}'_i]. \quad (8)$$

E'_{rigid} has the same formulation as shown in Eqn.4:

$$E'_{\text{rigid}} = R(\mathbf{A}'_i). \quad (9)$$

E'_{reg} brings the articulated motion prior into the optimization. It constrains that motions defined on the nodes do not change smoothly over the object but only change between sparse pairs of neighboring nodes. This is a plausible assumption because of the fact that the nodes on the same body part mostly share the same motion transform. We therefore formulate this term as an L_0 regularizer as:

$$\begin{aligned} E'_{\text{reg}} = & \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \|\|\mathbf{D}x_{ij}\|_2\|_0, \\ \mathbf{D}x_{ij} = & \mathbf{A}'_i(\mathbf{x}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}'_i - (\mathbf{x}_j + \mathbf{t}'_j). \end{aligned} \quad (10)$$



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

Figure 4. Color coded normalized magnitude of $\mathbf{k}_{ij}$ on the vertices during iterations in solving L_0 minimization. Blue color stands for lowest (0.0) magnitude, green for higher and red for the highest (1.0) magnitude. (a) the previous L_0 anchor frame; (b-e) some of the intermediate iteration steps.

Here $\|\mathbf{D}x_{ij}\|_2$ represents the magnitude of the motion difference, and E'_{reg} measures the L_0 norm of the motion difference between all pairs of neighboring nodes. In our implementation, α'_{rigid} is set to 1000, and α'_{reg} is set to 1.

Eqn.6 is difficult to be optimized as the E'_{reg} term brings a discrete counting metric. Inspired by the solver described in [28], we split the optimization into two subproblems by introducing auxiliary variables into the energy function. Notice that the original L_0 optimization is computational intractable, and our solution is only an approximation. However, the proposed method is effective to get a good enough solution.

We introduce auxiliary variables $\mathbf{k}_{ij}$ and reformulate the optimization problem as:

$$\min_{\mathbf{A}'_i,\mathbf{t}'_i,\mathbf{k}_{ij}} E'_{\text{data}} + \alpha'_{\text{rigid}} E'_{\text{rigid}} + \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \lambda \|\|\mathbf{k}_{ij}\|_2\|_0 + \beta \|\mathbf{D}x_{ij} - \mathbf{k}_{ij}\|_2^2. \tag{11}$$

Here $\mathbf{k}_{ij}$ is an approximation to $\mathbf{D}x_{ij}$. To solve this problem, we alternatively fix $\{\mathbf{A}'_i, \mathbf{t}'_i\}$ to solve $\{\mathbf{k}_{ij}\}$ and fix $\{\mathbf{k}_{ij}\}$ to solve $\{\mathbf{A}'_i, \mathbf{t}'_i\}$. If $\{\mathbf{A}'_i, \mathbf{t}'_i\}$ are fixed, the minimization is formulated as:

$$\min_{\mathbf{k}_{ij}} \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \lambda \|\|\mathbf{k}_{ij}\|_2\|_0 + \beta \|\mathbf{D}x_{ij} - \mathbf{k}_{ij}\|_2^2. \tag{12}$$

As $\mathbf{D}x_{ij}$ is pre-fixed, Eqn.12 has a close form solution:

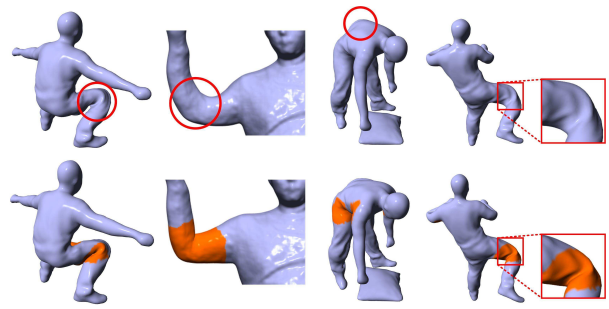
$$\mathbf{k}_{ij} = \begin{cases} 0 & \text{if } \|\mathbf{D}x_{ij}\|_2^2 < \lambda/\beta \\ \mathbf{D}x_{ij} & \text{if } \|\mathbf{D}x_{ij}\|_2^2 \geq \lambda/\beta \end{cases} \tag{13}$$

If $\mathbf{k}_{ij}$ are fixed, Eqn.11 has the following formulation:

$$\min_{\mathbf{A}'_i,\mathbf{t}'_i} E'_{\text{data}} + \alpha'_{\text{rigid}} E'_{\text{rigid}} + \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \beta \|\mathbf{D}x_{ij} - \mathbf{k}_{ij}\|_2^2. \tag{14}$$

Eqn.14 formulates a pure L_2 based optimization problem. We solve it by the Gauss-Newton method.

In solving Eqn.11 with this iterative method, the parameter λ and β needs to be changed in the iterations. In all our experiments, we fix λ to be 0.02 and set β to be 1 in the first iteration and multiplied by 2 after each iteration until β exceeds 10^6. Fig.4 illustrates the vertex motion magnitudes during the L_0 iteration updates. Comparing with the pose at previous anchor frame, we see that the crotch between two legs has noticeable motion. Correspondingly, this region is successfully detected by the algorithm as an articulated region at the beginning of the iterations. With iterations going on, more articulated regions are implicitly detected, as shown in Fig.4(b-e).



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

Figure 5. Comparison of L_0 and L_2 based motion regularization on some anchor frames. The first row shows the tracking results of using L_2, while the second row shows the results of using L_0. The vertices with non-zero motion difference ($\mathbf{k}_{ij} \neq \mathbf{0}$) in the first L_0 iteration are marked orange.

It is also important to note, after the L_0 minimization, the articulated motions are well reconstructed while other non-rigid motions are removed. To reconstruct those non-rigid motions, we run the L_2 based non-rigid registration again on the anchor frame using the refined result from L_0 as an initial with the depth input as target. Notice that, the new initial refined result has got rid of the accumulated error of the non-rigid tracking and thereby achieves better results.

Some results on the effectiveness of our proposed L_0 regularization are illustrated in Fig.5. Compared with the traditional non-rigid registration (the top row) which smoothly blends the relative deformation across the human body joints, our L_0 based regularizer (the second row) effectively concentrates these motions to the right joints, thereby substantially removes the deformation artifacts on both the joint regions (Fig.5(a,b,d)) and on the rigid body parts (Fig.5(c)).

4.3. Anchor frame detection

As stated in Sect.4.2, since the articulated motion between two neighbor frames are usually small, the pruning based L_0 regularization may wrongly prune the articulated motion, causing the ineffectiveness of the L_0 optimization. Our key idea to overcome this problem is to accumulate motions of every frame since the previous anchor frame:

$$\widetilde{\mathbf{A}}_i^t = \mathbf{A}'_i * \widetilde{\mathbf{A}}_i^{t-1}, \quad \widetilde{\mathbf{t}}_i^t = \mathbf{t}'_i + \widetilde{\mathbf{t}}_i^{t-1}. \tag{15}$$

where $\{\mathbf{A}'_i, \mathbf{t}'_i\}$ and $\{\widetilde{\mathbf{A}}_i^t, \widetilde{\mathbf{t}}_i^t\}$ denote the current and accumulated motion of node i at time t, respectively. With the accumulation, if the object is performing some articulated motion, the spatial motion variation around the joint of the

articulated motion will become larger and larger while the spatial motion variation caused by other non-rigid deformation stays at the same level. By analyzing the distribution of the spatial motion variation, we detect an anchor frame that has large enough articulated motion. The L_0 regularization is then triggered and the pruning algorithm in Sect.4.2 is operated on the detected anchor frame by referring to the previous anchor frame.

In practice, we calculate the variance for all $\|\mathbf{D}x_{ij}\|_2$, where $\mathbf{D}x_{ij}$ is calculated by the accumulated motion $\{\widetilde{\mathbf{A}}_i, \widetilde{\mathbf{t}}_i\}$. If the variance is larger than θ at a particular frame, we set this frame as an anchor frame where the L_0 based motion regularization will be performed. The value of θ in $[0.01, 0.03]$ usually gives reasonable results, while smaller or larger values may bring artifacts. In all our experiments, we set θ to be 0.02. Our supplementary material shows all the detected anchor frames in several motion sequences.

4.4. Bidirectional tracking and surface refinement

After refining the newly detected anchor frame, we need to update the frames between the previous anchor frame and the current anchor frame. We perform a backward tracking from the current anchor frame using the L_2 based non-rigid deformation method. For those frames which are close to the current anchor frame, the backward tracking results should be more accurate, while for those frames which are close to the former anchor frames, the original forward tracking results should be more accurate. As a consequence, we use a position dependent linear weight to blend the two results of each frame (see step 4 and 5 in Fig.2). Notice that directly blending vertex positions may cause artifacts when the bidirectional results are with large shape differences. In our implementation, we blend the affine transformation at each node and then apply the motion transform to get the final results.

After intermediate frames have been blended, we further reconstruct surface details of the captured objects. To achieve this, we first subdivide the current mesh model and then utilize the method in [12] to synthesize surface details from the captured depth. After that, we take the result of current anchor frame as an initialization to perform L_2 non-rigid tracking for the following frames and detect the next anchor frame. Such tracking cycle goes on until the end of the sequence.

5. Experiments

We recorded 10 test sequences consisting of over 6000 frames using a single Kinect camera or an Intel IVCam camera. The Kinect camera is used for capturing full human body motions while the IVCam camera is for capturing hand motions and facial expressions. During data capture, the camera remains fixed. Table 1 shows the details of our captured data. The experiment sequences include fast human motions, e.g. "Sliding" and "SideKick", multiple kinds of objects, e.g. "Puppet" "Pillow$_1$" "Pillow$_2$" "Face" and "Hand", and motions with heavy occlusions, e.g. "Pillow$_2$" and "Hand". Besides, we also use synthesized data with and without noise for quantitative evaluation.

After data capture, our motion reconstruction method is performed offline. The template modeling step reconstructs a mesh model with about 9000 vertices. After roughly aligning the template with the first depth frame, the tracking system runs with about 3 frames per-minute. For each frame, about 18s is taken by the bidirectional non-rigid registration. The L_0 based refinement requires 60s for one frame, which does not contribute too much to the total time as it is only performed on a small amount of anchor frames. Notice that we implemented our method by C++ on a PC with an 3.20 GHz four core CPU and 16GB memory.

	No. Frames	No. Anchors	No. Vertices	No. Nodes	Source
Dance	800	16	9427	260	Kinect
Kongfu	752	26	8734	249	Kinect
Pillow$_1$	623	9	10446	249	Kinect
Pillow$_2$	419	5	9848	281	Kinect
Puppet	800	2	9995	206	Kinect
Sliding	800	35	8734	249	Kinect
Girl	800	14	9501	270	Kinect
SideKick	400	17	8378	239	Kinect
Face	400	2	9850	299	IVCam
Hand	300	6	8923	260	IVCam

Table 1. Statistics of the captured dataset in the experiments.

5.1. Reconstruction results

Our technique is capable to reconstruct various motions of different objects, including human body motion, hand motion and their interaction with objects. Some of the results are demonstrated in Fig.6, where the first column shows the results of pure body motion in the "Sliding" and "Dance" sequence, which indicates that our technique is capable for reconstructing fast motions and handling self-occlusion caused by articulated motions. The second column shows the results of the "Pillow$_1$" and "Pillow$_2$" sequences with human-object interactions, where the actor is manipulating a non-rigid pillow. The third column demonstrates human motion with loose cloth and motion of an interactive toy. Together with the successful tracking of the human face and the hand motion in Fig.8, it demonstrates the fact that our method supports various object types with different shapes and topologies, regardless of the existence of articulated structure or not. Our method is also well compatible with surface detail reconstruction method, see the sophisticated geometry obtained on the "Girl" models. For more sequential reconstruction showing our temporal coherency, please refer to our supplementary video.

Figure 6. Results of our technique. For each result, we show a color image, the input depth and the reconstruction result. Notice that the color image is only for viewing the captured motion. It is not used by our system.

5.2. Evaluation and comparison

We quantitatively evaluate our method with Vicon motion capture system, and also compare our results with [12] and [34]. First, we synchronize Vicon and Kinect using infrared flash, and register markers of Vicon system with landmarkers on the template. Then for each frame, we calculate the average L_2 norm error between the markers and the corresponding vertices. Numerical error curves for all the three methods are shown in Fig.7. Average numerical

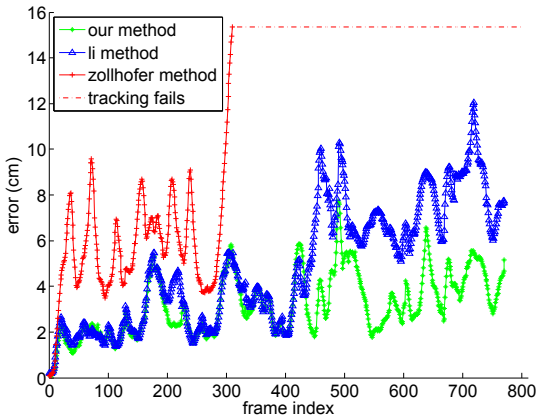


Figure 7. Average numerical errors w.r.t. Vicon system for our method, [12] and [34].

error of our method for the whole sequence is 3.19cm, compared with 4.88cm of [12] and 12.79cm of [34]. For longer time range (after frame 400), the average errors of the three methods are 3.93cm, 7.37cm and 17.24cm respectively.

We evaluate our method using intial object templates of different qualities. We downsample the original model to 75% and 50% and reconstruct coarsened templates. We test our method using these templates on "Sliding" sequence. One frame of reconstructed results is shown in Fig.9. For the 75% and 50% reconstructed templates, only synthesized details appear to be a little different. In practice, our method can tolerate a large range of smoothness. Therefore, our proposed method does not require high quality templates, which makes it more useful in real cases.

In Fig.10, we compare our method with [12] and [34] on real captured data. From the comparison, we see that our method outperforms [12] on the left foot, while [34] fails to track this pose caused by fast motion in the sequence. In Fig.8, we compare our method with [12] on face, body and hand sequences. Since there is no strong articulated motion in the face sequence, our method is similar to [12]. However, on articulated sequences of body and hand, our method prevents tracking failure and local misalignment which appear in the result of [12]. More comparisons on motion sequences are shown in the supplementary video.

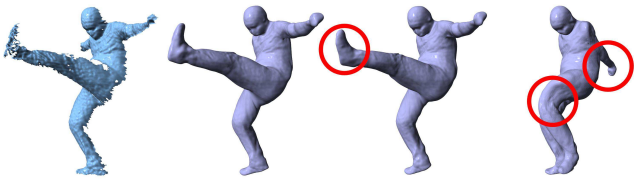
We compare L_1 sparsity constraint with the proposed L_0 method. Similar to formula 6, the new regularizer is $E'_{\text{reg}} = \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \|\mathbf{D}x_{ij}\|_1$. We solve it using primal-dual internal point method. Comparison results are shown in Fig.11. Our L_0 solver reconstructs motion and joints more accurately and avoid artifacts.

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

Figure 8. Comparison with [12] on Kinect and IVCam inputs. (a, d, g) depth input; (b, e, h) reconstruction results of our method; (c, f, i) reconstruction results of [12].

(a) (b) (c)

Figure 9. Our results using initial templates of different qualities. (a)-(c) show original template and 75% and 50% reconstructed ones respectively. The first row shows templates of different qualities, and second row shows one frame of results.



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

Figure 10. Comparison with [12] and [34] on Kinect input. (a) input depth; (b) results of our method; (c) results of [12]; (d) results of [34].

(a) (b)

Figure 11. Comparison with L_1 constraint. Left images in (a) and (b) are our L_0 results and right ones are approximation of L_1.

5.3. Other types of depth input

In addition to data captured by a single consumer depth sensor, our technique is also applicable for other depth acquisition techniques such as structure light [12] and binocular cameras [27]. This provides the extensive practicalities and enables more appealing applications. Results are shown in the supplementary video.

5.4. Limitations

The proposed L_0-L_2 non-rigid tracking approach is still limited in tracking extremely fast motions. For instance, the supplementary video shows a failure case that the tracking cannot catch up the up-moving leg of a character. This is mainly because of the frangibility of the vertex-to-point matching in dealing with fast motions. Our method is also incapable of motions with serious or long term occlusions. However, it naturally supports multiple view depth inputs, which will effectively mitigate the occlusion challenge.

6. Discussion

We have presented a novel non-rigid motion tracking method using only a single consumer depth camera. Our method outperforms the state-of-the-art methods in terms of robustness and accuracy. The key contribution of our technique is the combined L_0-L_2 tracking strategy which takes advantage of the intrinsic properties of articulated motion to constrain the solution space. According to experiment results, our method outperforms two previous state-of-the-arts of non-rigid tracking algorithms and can robustly capture full body human motions using a single depth sensor without embedding skeleton manually.

Our L_0 regularization is performed on the result of non-rigid registration but not related to algorithms for getting the results, which means it can be flexibly applied to other non-rigid registration techniques for better reconstructions.

Acknowledgements

This work was supported by the National key foundation for exploring scientific instrument No. 2013YQ140517, the 863 Program (No. 2013AA01A604), the open funding project of state key laboratory of virtual reality technology and systems of Beihang University (Grant No. BUAA-VR-14KF-08), and NSFC (No. 61120106003, No. 61272235, No. 91315302, and No. 61522111).

References

[1] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *Consumer Depth Cameras for Computer Vision*. 2013.

[2] I. Baran and J. Popovic. Automatic rigging and animation of 3d characters. *ACM Trans. Graph*, 26(3):72, 2007.

[3] D. Bradley, T. Popa, A. Sheffer, W. Heidrich, and T. Boubekeur. Markerless garment capture. *ACM Trans. Graph*, 27(3):99, 2008.

[4] C. Budd, P. Huang, M. Klaudiny, and A. Hilton. Global non-rigid alignment of surface sequences. *IJCV*, 102(1-3):256–270, Mar. 2013.

[5] C. Cagniart, E. Boyer, and S. Ilic. Free-form mesh tracking: a patch-based approach. In *CVPR*, 2010.

[6] C. Cagniart, E. Boyer, and S. Ilic. Probabilistic deformable surface tracking from multiple videos. In *Computer Vision–ECCV 2010*, pages 326–339. Springer, 2010.

[7] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun. Performance capture from sparse multi-view video. *ACM Trans. Graph*, 27(3):98, 2008.

[8] M. de La Gorce, D. J. Fleet, and N. Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(9):1793–1805, 2011.

[9] M. Dou, H. Fuchs, and J.-M. Frahm. Scanning and tracking dynamic objects with commodity depth cameras. In *Mixed and Augmented Reality (ISMAR)*, 2013.

[10] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *CVPR*, 2009.

[11] A. Jacobson, I. Baran, J. Popovi, and O. Sorkine. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph*, pages 1–8, 2011.

[12] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph*, 28(5):175, 2009.

[13] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong. Modeling deformable objects from a single depth camera. In *ICCV*, 2009.

[14] Y. Liu, C. Stoll, J. Gall, H.-P. Seidel, and C. Theobalt. Markerless motion capture of interacting characters using multi-view image segmentation. In *CVPR*, 2011.

[15] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer vision and image understanding*, 81(3):231–268, 2001.

[16] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126, 2006.

[17] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV*, 2011.

[18] T. Popa, I. South-Dickinson, D. Bradley, A. Sheffer, and W. Heidrich. Globally consistent space-time reconstruction. *Computer Graphics Forum*, 2010.

[19] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, 2007.

[20] J. Starck and A. Hilton. Surface capture for performance-based animation. *Computer Graphics and Applications*, 27(3):21–31, 2007.

[21] J. Starck and A. Hilton. Model-based human shape reconstruction from multiple views. *Computer Vision and Image Understanding*, 111(2):179–194, 2008.

[22] R. W. Sumner, J. Schmid, and M. Pauly. Embedded deformation for shape manipulation. *ACM Trans. Graph*, 26(3):80, 2007.

[23] R. Szeliski and S. Lavallée. Matching 3-d anatomical surfaces with non-rigid deformations using octree-splines. *IJCV*, 18(2):171–186, 1996.

[24] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph*, 27(3):97, 2008.

[25] M. Waschbüsch, S. Würmlin, D. Cotting, F. Sadlo, and M. Gross. Scalable 3d video of dynamic scenes. *The Visual Computer*, 21(8-10):629–638, 2005.

[26] X. Wei, P. Zhang, and J. Chai. Accurate realtime full-body motion capture using a single depth camera. *ACM Trans. Graph*, 31(6):188, 2012.

[27] C. Wu, C. Stoll, L. Valgaerts, and C. Theobalt. On-set performance capture of multiple actors with a stereo camera. *ACM Trans. Graph*, 32(6):161, 2013.

[28] L. Xu, C. Lu, Y. Xu, and J. Jia. Image smoothing via l0 gradient minimization. *ACM Trans. Graph*, 30(6):174, 2011.

[29] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt. Performance capture of interacting characters with handheld kinects. In *ECCV*. 2012.

[30] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys. Accurate 3d pose estimation from a single depth image. In *ICCV*, 2011.

[31] M. Ye and R. Yang. Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera. In *CVPR*, 2014.

[32] Q. Zhou, S. Miller, and V. Koltun. Elastic fragments for dense scene reconstruction. In *ICCV*, 2013.

[33] Y. Zhu, D. Huang, F. De La Torre, and S. Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. June 2014.

[34] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, and C. Theobalt. Real-time Non-rigid Reconstruction using an RGB-D Camera. *ACM Trans. Graph*, 33(4):156, 2014.