Templateless Non-Rigid Reconstruction and Motion Tracking With a Single RGB-D Camera

Kangkan Wang, Guofeng Zhang, *Member, IEEE*, and Shihong Xia, *Member, IEEE*

***Abstract*— We present a novel templateless approach for nonrigid reconstruction and motion tracking using a single RGB-D camera. Without any template prior, our system achieves accurate reconstruction and tracking for considerably deformable objects. To robustly register the input sequence of partial depth scans with dynamic motion, we propose an efficient local-to-global hierarchical optimization framework inspired by the idea of traditional structure-from-motion. Our proposed framework mainly consists of two stages, local nonrigid bundle adjustment and global optimization. To eliminate error accumulation during the nonrigid registration of loop motion sequences, we split the full sequence into several segments and apply local nonrigid bundle adjustment to align each segment locally. Global optimization is then adopted to combine all segments and handle the drift problem through loop-closure constraint. By fitting to the input partial data, a deforming 3D model sequence of dynamic objects is finally generated. Experiments on both synthetic and real test data sets and comparisons with state of the art demonstrate that our approach can handle considerable motions robustly and efficiently, and reconstruct high-quality 3D model sequences without drift.**

***Index Terms*— Templateless, non-rigid reconstruction, motion tracking.**

I. Introduction

NOWADAYS, RGB-D cameras, such as Microsoft Kinect, Intel RealSense, or Asus Xtion Pro, have become an affordable commodity for everyday users. With the availability of these depth sensors, great progress has gained in the field of nonrigid reconstruction and tracking with a single depth camera. To reduce solution space and make this problem tractable, the major route is to take advantage of template priors [1]–[5]. Although tracking accuracy has largely been increased, these approaches require to build template priors beforehand. However, a large number of deformable objects cannot be completely modeled by template priors, e.g., modeling the activity of a person grasping nonrigid deforming clothes using embedded skeleton. On the other hand, some works [6]–[9] tried to track the motion of deforming objects and reconstructed detailed surface geometry only using raw partial data from different views. These approaches are quite appealing and suitable for practical applications because they do not rely on any template priors. However, this problem is much more challenging since only visible parts of the objects can be captured at each time instant. Previous works without templates are limited to handle quasi-rigid objects, small deformation or partial view reconstruction.

In this paper, we propose a templateless nonrigid 3D reconstruction with dynamic motion tracking using a single depth camera. The input of our system is a single-view depth sequence of a deformable object captured from different views at different time instants. Our system tracks the object's motion and reconstructs a 3D object model by fusing visible partial data together. From the reconstructed model and tracked motion, a moving 3D model sequence can be generated that is consistent with the input partial data. Fig. 1 shows a complete example generated by our system. In this example, a person is kicking with large motion. Our system can faithfully recover the time-varying 3D model sequence of the moving person. It should be noted that all our results are obtained by a single depth sensor and no template priors are used. Our proposed method not only can obtain visually comparable results as the state of the art [9], but also is nearly 50 times faster than [9] (about 12 minutes to handle a sequence of a human body with 400 frames using our method). The average processing speed is about $1-2$ seconds per frame, which can provide users immediate feedback of reconstruction results.

Nonrigid objects always deform with time-varying and arbitrary motion, and there is a huge parameter space for nonrigid deformation. So registration error accumulates rather fast and severe drift problem occurs when registering nonrigid loop-closure sequences with noisy and incomplete data from a single depth sensor. Approaches based on structure-from-motion (SfM) [10] have recently obtained good performance for rigid 3D reconstruction and tracking [11], [12] because it poses strong constraints to bundle all frame data

Manuscript received December 15, 2016; revised May 16, 2017, July 13, 2017, and August 9, 2017; accepted August 9, 2017. Date of publication August 16, 2017; date of current version September 21, 2017. This work was supported in part by the Natural Science Foundation of China under Grant 61602444 and Grant 61672457, in part by the Knowledge Innovation Program of the Institute of Computing Technology of the Chinese Academy of Sciences under Grant ICT20166040, and in part by the Science and Technology Service Network Initiative of Chinese Academy of Sciences under Grant KFJ-STS-ZDTP-017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhengguo Li. *(Corresponding author: Kangkan Wang.)*

K. Wang is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China, and also with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: wangkangkan@gmail.com).

G. Zhang is with the State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310058, China (e-mail: zhangguofeng@cad.zju.edu.cn).

S. Xia is with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: xsh@ict.ac.cn).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material contains three videos that show the reconstruction results by our method for various datasets. A few comparisons with other methods are also included. The total size of the videos is 34.2 MB. Contact wangkangkan@gmail.com for further questions about this work.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2740624

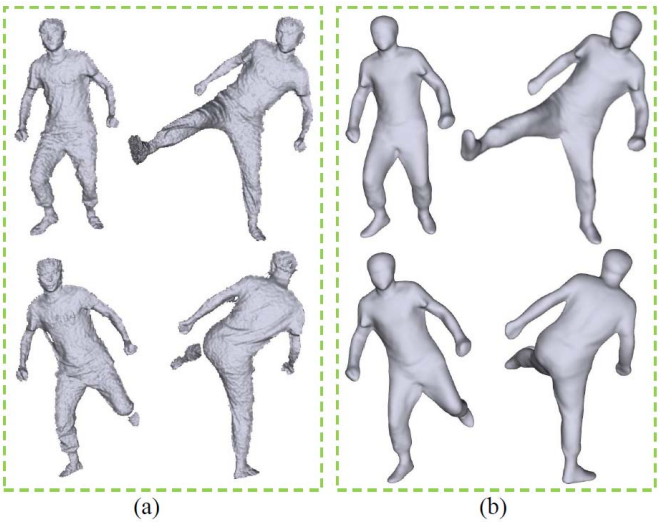


Fig. 1. A person is kicking legs with large deformations in front of a Kinect camera. Without template priors, our system generates the moving 3D model sequence. (a) Input depth scan sequence. (b) Reconstructed 3D model sequence.

together through point correspondences and thus leads to robust tracking results. Motivated by the demonstrated success of traditional SfM [13]–[15], we propose a nonrigid global registration framework for nonrigid depth sequences. Compared with previous templateless methods, our approach not only achieves high accuracy and robustness of nonrigid reconstruction and tracking, but also effectively solves the drift problem under nonrigid registration of loop-closure motion sequences. This is made possible by an efficient local-to-global hierarchical optimization framework. On the first hierarchy level, we register each local segment accurately and mitigate error accumulation using local nonrigid bundle adjustment. On the second hierarchy level, global optimization is applied to refine globally consistent deformations of all segments and merge all depth scans together. During global optimization, we need to compute deformations only for each segment but not for all depth scans which greatly reduces the number of unknowns. Computation and memory cost are tremendously decreased during global optimization for large data. Our method can obtain reconstruction quality that is rather close to the state of the art [9] but with much less computation and memory cost, which achieves a compromise between computational complexity and reconstruction accuracy. The proposed method can handle drift problem more effectively than the state of the art when fully-automatically both reconstructing and tracking for non-rigidly deforming objects from depth data captured by a single RGB-D camera.

In addition, our local-to-global hierarchical optimization framework can easily deal with multiple depth sequences. Our method can quickly and efficiently reconstruct geometry surfaces in different sequences, and then integrate them to obtain the complete model. It allows the user to capture more sequences to cover the whole object without the need of re-computing the data from the beginning, which can significantly improve the user experience since the capturing and processing time is significantly reduced. Due to the low cost and high efficiency of our nonrigid reconstruction system, we believe that our approach can be conveniently employed in many applications about reconstruction and tracking of dynamic objects. In summary, the main contributions of our technique are as follows:

- A novel templateless system for non-rigid reconstruction and motion tracking using a single RGB-D camera. Compared to the state of the art, our method not only can handle considerable deformations and motions robustly, but also can solve drift problem more effectively.
- A local-to-global hierarchical optimization framework which can effectively register non-rigid loop-closure depth sequences with noisy and incomplete data. This framework combines an efficient local nonrigid bundle adjustment and a robust global optimization to eliminate error accumulation, largely improving registration accuracy and computational efficiency for dynamic depth sequences.
- Compared to state-of-the-art methods, our method is much faster and requires less memory cost. Especially, our method can effectively handle multiple depth sequences and provide users immediate feedback of reconstruction results, which significantly improves the user experience.

II. Related Work

The approaches for nonrigid reconstruction and tracking from depth data can be classified into two categories, template-based and templateless methods. Here, we briefly review the recent related works, respectively.

A. Template-Based Methods

Template-based methods need to use template priors such as embedded skeleton, template models, or parametric models, to reconstruct 3D models and motions of dynamic objects with depth cameras. KinectsMocap [1] exploited articulated motion prior of skeleton embedding to capture performance of interacting characters using three handheld Kinects. Zhang et al. [2] proposed to recover dynamic 3D models of human bodies by fitting a pre-trained personalized model to input point clouds of frontal view. Zollhöfer et al. [3] proposed a real-time nonrigid reconstruction system which first created a high-quality base template of the object and then deformed the template to track the object. Guo et al. [4] presented a motion tracking method through L0 regularization to robustly reconstruct non-rigid geometries and motions using a single depth sensor. They did not adopt skeleton information, but needed to obtain 3D mesh templates of the deforming targets. Other methods [5], [16]–[19] need to first learn a parametric model of the target object from a training dataset and use the parametric model to fit input data. However, these methods fail to reconstruct 3D models that cannot be represented by the dataset. For example, Bogo et al. [5] drove a SCAPE-based human body parametric model to a monocular RGB-D sequence containing a moving user. Because the parametric model is learned from a large 3D model dataset of undressed human bodies, their method could not reconstruct 3D models of dressed human bodies or other targets.

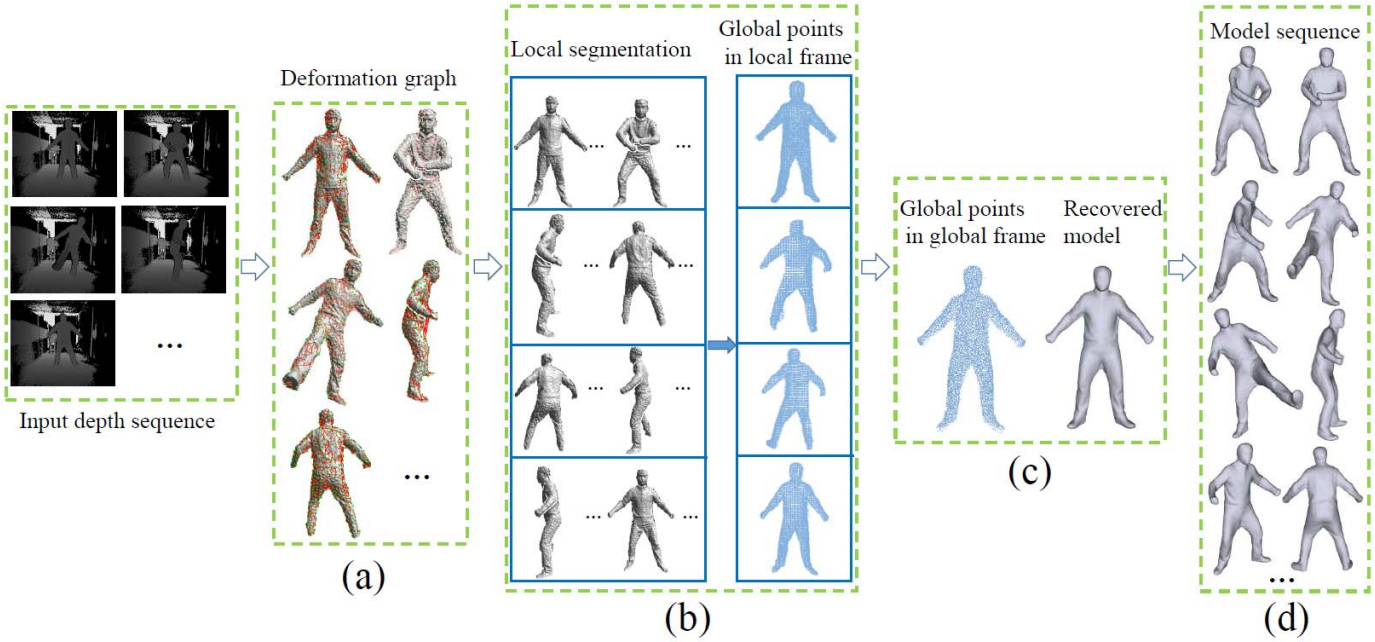


Fig. 2. The pipeline of our system. (a) Data preprocessing. (b) Local nonrigid bundle adjustment. (c) Global optimization. (d) Generating dynamic 3D model sequence. Please refer to Sect. III for a detailed description.

B. Templateless Methods

In recent years, some researchers tried to recover nonrigid surfaces and motions without template priors. The core task of these methods is to accurately register partial surfaces at different time instants together. Liao et al. [6] proposed a linear global warping algorithm which can reconstruct complete 3D deformable models over time using a single depth camera by assuming known point correspondences between partial surfaces. Tong et al. [20] built a system of scanning static full human bodies using three Kinects and a turntable which is complex. Zeng et al. [7] presented a method for quasi-rigid objects modeling from a sequence of depth scans which failed to handle dynamic objects. 3D Self-Portraits [21] allowed ordinary users to capture full 3D models of themselves, but it required users keeping the same pose during scanning. A real-time system, DynamicFusion [8] was proposed for reconstructing and tracking nonrigid scenes. Another real-time system, VolumeDeform [22] preserved fine-scale detail in the reconstruction by using a higher-resolution deformation field than the coarse one in DynamicFusion. Both systems are successful for nonrigid partial view reconstruction, but could not handle drift problem with dynamic motion. When sequentially registering loop-closure motion sequences with noisy and occluded data from a single depth sensor, registration error accumulates rather fast and eventually leads to serious drift problem. However, the implicit loop closure fails to distribute the tremendous accumulated error. Recently, Guo et al. [23] proposed a real-time system for simultaneously recovering object geometry and surface albedo using a single RGB-D camera, however, without considering drift problem. Dou et al. [24] proposed a new system for live multi-view performance capture using 8 depth cameras. Wang et al. [25] presented a system for reconstructing complete watertight and textured models of moving subjects using three or four handheld depth sensors. However, these systems with multiple depth sensors failed to handle drift problem.

Dou et al. [9] dealt with drift problem through error distribution and further used a bundle adjustment algorithm to optimize the recovered shape. Their error distribution detects loop closures through pairwise registration between two scans. However, this loop-closure detection is not robust and accurate because it is hard to recognize overlapping areas among all partial scans especially under occlusion and dynamic motion. So a final nonrigid bundle adjustment was required to refine deformations, surface geometry, and point correspondences simultaneously, which is very expensive in computation and memory. In contrast, our method successfully identifies loop closures and builds accurate correspondences in a robust and effective way, and then optimizes both deformations and surface geometry with fixed correspondences. Through an efficient local-to-global hierarchical optimization framework, we achieve a very high reconstruction accuracy with less computation and memory cost.

III. Overview

We propose an effective approach for templateless nonrigid reconstruction and motion tracking with an RGB-D camera (e.g. Kinect). Fig. 2 gives the pipeline of our system. There are mainly four steps. In the first step, raw depth data is preprocessed to extract deforming targets and generate deformation graphs for extracted depth scans. In the second step, we divide the whole depth sequence into several local segments, and register all depth scans in each segment separately through local nonrigid bundle adjustment. In the third step, global optimization is employed to merge all segments and eliminate the drift problem in a global way.

A watertight or partial 3D model of the object is generated using Poisson surface reconstruction [26] or volumetric fusion [27] from all registered depth scans. Our method registers all scans and tracks the object's motion simultaneously without any template prior. In the fourth step, with the estimated motion and temporal information, the recovered 3D model is deformed back to each depth scan to create a dynamic 3D model sequence which fits to the input depth data well.

IV. Deformation Model

We first introduce a deformation model to describe nonrigid deformation of surface. Due to the success in recent works [4], [7], [9], we employ embedded deformation model [28], [29] to parameterize the non-rigid deformations. Embedded deformation computes a warping field based on a deformation graph. Sparse nodes are uniformly sampled throughout the surface, and neighboring nodes are connected with edges. Each node of the graph induces a deformation within a local influence region. The deformation of the surface is represented by local affine transformations $\mathbf{A}_i, \mathbf{t}_i$ of all nodes $\mathbf{x}_i$. Deformation of each point $\mathbf{v}_j$ is determined by its K nearest nodes using a set of fixed weights $w(\mathbf{v}_j, \mathbf{x}_i) = (1 - d(\mathbf{v}_j, \mathbf{x}_i)/d_{max})^2$, with $d(\mathbf{v}_j, \mathbf{x}_i)$ the geodesic distance between $\mathbf{v}_j$ and $\mathbf{x}_i$, d_{max} the geodesic distance of $\mathbf{v}_j$ to its $K+1$-th nearest node. For point $\mathbf{v}_j$, its new position after the non-rigid deformation is computed as,

$$\mathbf{v}_j^{'} = \sum_{\mathbf{x}_i} \bar{w}(\mathbf{v}_j, \mathbf{x}_i)[\mathbf{A}_i(\mathbf{v}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}_i]. \tag{1}$$

and its new normal is the weighted sum of each normal transformed by the inverse transpose of the node transformations,

$$\mathbf{n}_j^{'} = \sum_{\mathbf{x}_i} \bar{w}(\mathbf{v}_j, \mathbf{x}_i)\mathbf{A}_i^{-1T}\mathbf{n}_j. \tag{2}$$

where $\bar{w}(\mathbf{v}_j, \mathbf{x}_i)$ are the normalized weights of $w(\mathbf{v}_j, \mathbf{x}_i)$. According to [28], the unknown transformations $\mathbf{A}_i, \mathbf{t}_i$ are estimated by minimizing the following energy,

$$E_{tol} = w_{fit}E_{fit} + w_{rigid}E_{rigid} + w_{smooth}E_{smooth}. \tag{3}$$

The fitting term E_{fit} forces $\mathbf{v}_j$ to move to its corresponding point $\mathbf{c}_j$, and is defined as,

$$E_{fit} = \sum_{\mathbf{v}_j^{'} \in C} \alpha_{point} \parallel \mathbf{v}_j^{'} - \mathbf{c}_j \parallel_2^2 + \alpha_{plane} \mid \mathbf{n}_{\mathbf{c}_j}^T(\mathbf{v}_j^{'} - \mathbf{c}_j) \mid^2 . \tag{4}$$

where C contains all points that have correspondences, $\mathbf{n}_{\mathbf{c}_j}$ is the normal of $\mathbf{c}_j$, α_{point} and α_{plane} are weights for point-to-point and point-to-plane metric, respectively. The rigid term E_{rigid} encourages the affine transformation to be as rigid as possible,

$$E_{rigid} = \sum_{\mathbf{x}_i} ((\mathbf{a}_1^T\mathbf{a}_2)^2 + (\mathbf{a}_2^T\mathbf{a}_3)^2 + (\mathbf{a}_3^T\mathbf{a}_1)^2 + (1 - \mathbf{a}_1^T\mathbf{a}_1)^2 + (1 - \mathbf{a}_2^T\mathbf{a}_2)^2 + (1 - \mathbf{a}_3^T\mathbf{a}_3)^2). \tag{5}$$

where $\mathbf{a}_1$,$\mathbf{a}_2$ and $\mathbf{a}_3$ are column vectors of $\mathbf{A}_i$. The smooth term E_{smooth} constrains affine transformations of neighboring nodes to be similar, restricting consistent motion difference on the spatial domain,

$$E_{smooth} = \sum_{\mathbf{x}_j} \sum_{\mathbf{x}_i \in N(\mathbf{x}_j)} w(\mathbf{x}_j, \mathbf{x}_i) \parallel \mathbf{A}_i(\mathbf{x}_j - \mathbf{x}_i) + \mathbf{x}_i + \mathbf{t}_i - (\mathbf{x}_j + \mathbf{t}_j) \parallel_2^2 . \tag{6}$$

Two regularization terms effectively constrain over-flexibility of nonrigid deformation and prevent deforming in unreasonable ways. The correspondences C are built using a method similar to [1], and our method combines point position and normal. Assuming that a source scan is deformed to a target scan, we define a distance measure F between point $\mathbf{p}$ on target scan and point $\mathbf{v}$ on source scan

$$F(\mathbf{p}, \mathbf{v}) = \max(1 - \frac{\parallel x_p - x_v \parallel}{\theta_x}, 0) * \max(n_p * n_v, 0). \tag{7}$$

where x_p, n_p and x_v, n_v denote the position and normal of $\mathbf{p}$ and $\mathbf{v}$, respectively. The maximum distance θ_x is set to $100mm$. For each point $\mathbf{p}$ on target scan, we find the point $\tilde{\mathbf{v}}$ from source scan to maximize F. If the maximum $F(\mathbf{p}, \tilde{\mathbf{v}}) > 0$, the correspondence $(\mathbf{p}, \tilde{\mathbf{v}})$ is correct, otherwise it is pruned. From the definition, a correspondence is valid only if the surface normal difference of the two points is no more than 90° apart and the distance between them is less than $100mm$. The condition of surface normal is used so that front-facing surfaces will not be matched to back-facing surfaces. For each point on the target scan, we searches the correspondence from the source scan. Thus, point $\tilde{\mathbf{v}}$ of source scan may have several corresponding points in the target scan. In general, we use the correspondence with the nearest distance to $\tilde{\mathbf{v}}$ or take the mean value of all corresponding points of $\tilde{\mathbf{v}}$. In our experiments, we find for some large motions the correspondence is prone to be inaccurate and the tracking fails by using the nearest neighbor method or mean value method. Instead, we take the correspondence with the largest distance to $\tilde{\mathbf{v}}$ which can register source scan to target scan more accurately. An example of comparison between two correspondence finding methods is shown in Fig. 3.

A. Deformation Graph Generation for Partial View

Before deformation graph generation, we extract the target objects from each raw depth scan by segmenting background and the ground, and further removing outlier points using Point Cloud Library [30]. Since extracted surfaces only have partial data of deforming objects and the shapes are always varying, it is more difficult to generate deformation graph than for the template model with a fixed shape. We uniformly sample sparse nodes throughout the surface and keep geodesic distance between any two nodes above δ_1. A variant of fast marching method is used to efficiently compute approximate geodesic distances [31]. For each node, we find the nearest r nodes and link it with the neighboring node using an edge if two conditions are satisfied. First, two nodes have a geodesic distance less than δ_2. Second, there are at least p points on the line connecting two nodes to ensure different parts of objects

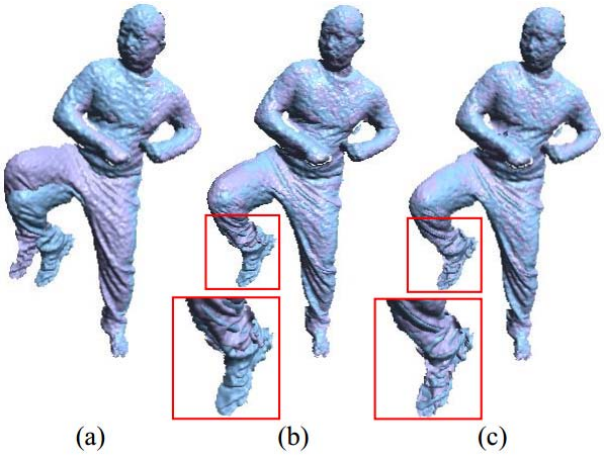


Fig. 3. Comparison of two correspondence finding methods. (a) two depth scans in the "SideKick" data from [4]; (b) registration result using the correspondences with the nearest distance; (c) registration result using the correspondences with the largest distance. The light grey scan is deformed to light blue scan. Registration fails (e.g. left foot) using the correspondences with the nearest distance. In contrast, accurate registration is achieved using the correspondences with the largest distance. Please refer to the last paragraph of Sect. IV for our correspondence finding method.

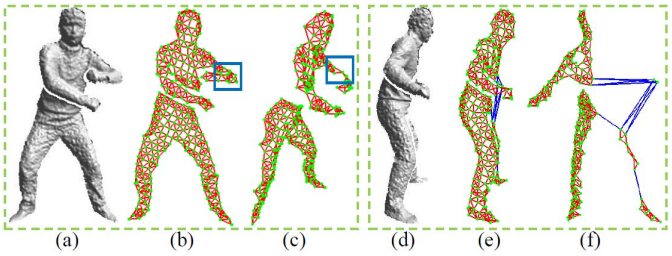


Fig. 4. Two examples of generated deformation graph for partial data. (a, d) extracted depth scan; (b, e) generated deformation graph; (c, f) deformation graph shown in another view. Left: the different parts are not attached even though they are close with each other (surrounded in rectangle). Right: small patches are connected to neighboring patches with relaxed regularization (shown in blue lines).

are not connected. After computing the edges for all nodes, node n_i may have r_l edges less than r. For all remaining $r - r_l$ neighboring nodes, we add an edge if one neighboring node is disconnected with n_i on the deformation graph. In this way, small surface patches caused by occlusion are connected to nearby surface patches so that they can deform together. Without these added node edges, small patches could deform arbitrarily if they do not have correspondences on the target surfaces. To deform small patches to correct positions, these added edges should have relaxed regularization E_{smooth} in Eq. 6 by using a lower smooth weight ($w_{smooth} = w_{smooth} * 0.01$ in our experiments). Two examples of deformation graph generation are shown in Fig. 4.

V. Local Nonrigid Bundle Adjustment

In this section, we register all depth scans in each segment into a local reference frame. We first formulate the nonrigid bundle adjustment, and then introduce a simple yet efficient strategy for local segmentation. Finally, we describe the method for registering local segments.

A. Nonrigid Bundle Adjustment

Inspired by 3D reconstruction for rigid scenes [12], we propose a nonrigid bundle adjustment for nonrigid reconstruction and define the energy as

$$E_{\mathbf{T}^i,\mathbf{g}} = \sum_{i=1}^{n} (w_{fit} E_{data}(\mathbf{T}^i, \mathbf{g}) + w_{rigid} E_{rigid}(\mathbf{T}^i) + w_{smooth} E_{smooth}(\mathbf{T}^i)). \quad (8)$$

where n is the number of depth scans, $\mathbf{T}^i : (\mathbf{A}, \mathbf{t})$ is affine transformation set of all nodes on the ith depth scan D^i, and $\mathbf{g}$ is a global point set in the reference frame. The data term for the ith depth scan is defined as

$$E_{data}(\mathbf{T}^i, \mathbf{g}) = \sum_{(\mathbf{v}_j, \mathbf{g}_j) \in C_i} \alpha_{point} \parallel \mathbf{v}_j^{'} - \mathbf{g}_j \parallel_2^2 + \alpha_{plane} \mid \mathbf{n}_{\mathbf{g}_j}^T (\mathbf{v}_j^{'} - \mathbf{g}_j) \mid^2 . \quad (9)$$

where $\mathbf{v}_j$ is a point on the ith depth scan D^i, $\mathbf{g}_j$ is the corresponding global point of $\mathbf{v}_j$, $\mathbf{n}_{\mathbf{g}_j}$ is the normal of $\mathbf{g}_j$, $\mathbf{v}_j^{'}$ is the deformed position of $\mathbf{v}_j$ through $\mathbf{T}^i$, and C_i denotes correspondence set between D^i and global point set. $E_{rigid}(\mathbf{T}^i)$ and $E_{smooth}(\mathbf{T}^i)$ mean deformation $\mathbf{T}^i$ should satisfy rigid and smooth constraints as defined in Eq. (5) and Eq. (6), respectively. In our experiments, we set $w_{fit} = 1$, $w_{rigid} = 1$, $w_{smooth} = 100$, $\alpha_{point} = 1.0$, and $\alpha_{plane} = 0.1$. We solve this non-linear problem using Levenberg-Marquardt algorithm [14]. The use of a point-to-plane term is a well-known strategy to speed up convergence [32]. We use point-to-plane term in registering local segments, but do not use it in global optimization. That is because in our global optimization, the minimization converges fast with a good initialization. Also, only using point-to-point metric not only can simplify our global optimization, but also can generate accurate results. Our nonrigid bundle adjustment energy aims to refine deformations of all frames and global points simultaneously. In traditional SfM [10], global points are obtained through triangulation on 2D pixels. In our case, since depth information is available, we can initialize global points using raw points of depth scans. There are mainly two advantages for our nonrigid bundle adjustment framework. Firstly, by bundling all frames closely together through correspondences with global points, robust registration is achieved for non-rigid depth sequences. Secondly, compared to partial depth scans especially with serious occlusion, global points in local segments integrate all depth scans and contain more information of modeling targets, which is beneficial for the success of our global optimization.

B. Local Segmentation

Zhang et al. [15] contributed an effective segment-based coarse-to-fine SfM algorithm for robustly handling multiple color sequences. Inspired by their segment-based idea, we segment the entire sequence into several subsequences through local segmentation. We first select some key frames scattered in the sequence with the most number of nodes since they contain the most information of objects. Make sure that there are at least m frames between two key frames.

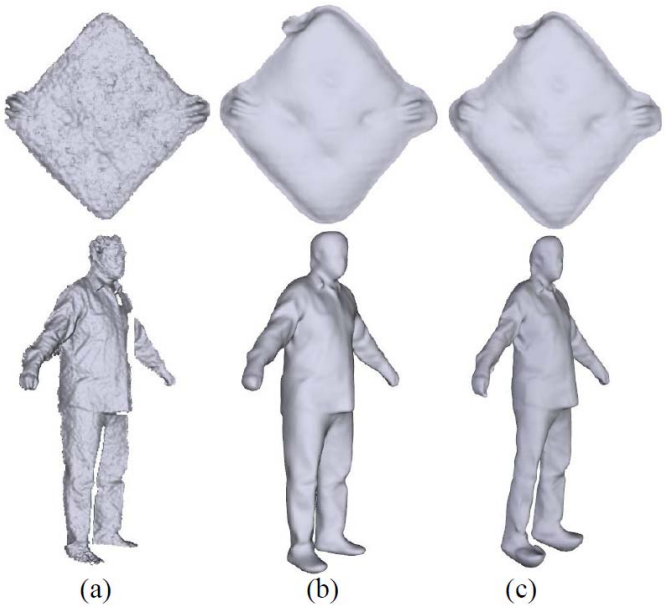


Fig. 5. Comparison of local segmentation with uniform segmentation [9], [12] on "Pillow" data (top row) and a human body data (bottom row). (a) depth scans; (b) the reconstructed 3D models using our local segmentation; (c) the reconstructed 3D models using uniform segmentation.

Initially, all frames between two key frames are classified as a local segment. In each segment, the first key frame is considered as a local reference frame, and local segment is registered sequentially. Large registration error always occurs when registering two partial data with less overlapping region. Thus, during registering each segment, we detect new segments dynamically to eliminate large registration error. Specifically, after registering current depth scan D^i to reference frame in current segment S_l, we compute the overlap ratio $d(\tilde{D}^i, R^l)$ between registered depth scan $\tilde{D}^i$ and the reference scan R^l using the method in section VI. If $d(\tilde{D}^i, R^l) < r$ (we use $r = 40\%$), S_l is then split and a new segment is generated including unprocessed frames in S_l. Registering and detecting local segment continue till the end of the total sequence. An example of detected local reference frames is shown in Fig. 2(b) (the leftmost four depth scans in local segmentation). Under local segmentation, the total sequence is divided into L segments $S_1, S_2, \ldots, S_L$. In each segment, registration is applied separately and all depth scans are aligned.

To testify the advantages of our local segmentation, we compare our method to the segmentation method adopted in [9] and [12] (we call uniform segmentation). The uniform segmentation is to divide the input sequence into many subsegments and each subsegment is with N consecutive frames ($N = 30$ in our test). The test data includes the "Pillow" data and a human body data. The comparison results are shown in Fig. 5. On the "Pillow" data, reconstructed results obtained using uniform segmentation are with similar accuracy as us. However, due to serious occlusion in the human body data, there are clear artifacts on reconstruction results using uniform segmentation. The comparison results demonstrate that our proposed segmentation method is more effective to eliminate accumulated error than the uniform segmentation for nonrigid sequence registration.

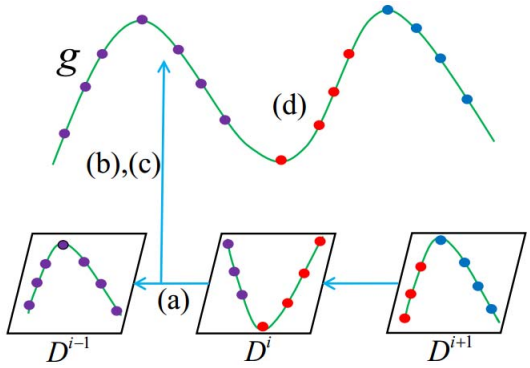


Fig. 6. The procedure of registering local segment. All frames in local segment are registered with global points. The correspondences are denoted using dots of different colors. Please refer to Sect. V-C for a detailed description.

C. Registering Local Segment

The procedure of registering each local segment is illustrated in Fig. 6. The global point set $\mathbf{g}$ is initialized with all points of a reference frame. For the subsequent frames, we repeat four steps to align each depth scan to the reference frame. In the first step (Fig. 6(a)), we register current depth scan D^i to previous one D^{i-1}, and find the point correspondences $(\mathbf{v}_j^i, \mathbf{v}_j^{i-1})$ between them. In the second step (Fig. 6(b)), based on correspondences $(\mathbf{v}_j^{i-1}, \mathbf{g}_j)$ between D^{i-1} and global points $\mathbf{g}$, we can build correspondences $(\mathbf{v}_j^i, \mathbf{g}_j)$ between D^i and $\mathbf{g}$. Using all correspondences $(\mathbf{v}_j^i, \mathbf{g}_j)$, D^i is initially registered to the reference frame. In the third step (Fig. 6(c)), D^i is further registered to the reference frame by newly searched correspondences with global point set $\mathbf{g}$. In the fourth step (Fig. 6(d)), correspondences are established between D^i and $\mathbf{g}$, and global point set is updated by adding unmatched points on D^i. By repeating the four steps, global point set is enlarged as the frame number increases. Both the first and third steps are in an iterative process between correspondence finding and surface deformation. We search correspondences using the method described in section IV. When building correspondences in the third step, we only use global points that overlap with previous several frames to reduce the search range and improve the correspondence accuracy. Based on the established correspondences among depth scans and global points $\mathbf{g}$, we locally refine node transformations of all frames and 3D positions of global points using nonrigid bundle adjustment in Eq. (8) to accurately align all depth scans.

VI. Global Optimization

After finishing local nonrigid bundle adjustment for each segment, we obtain a global point set and deformations for each frame in each segment. In the following, we would like to obtain globally consistent deformations for all frames in the total sequence, and reconstruct a 3D model of the objects based on registered depth scans. One way is to compute a deformation for each frame to align with a global point set, and optimize all deformations of each frame and 3D positions of the global points simultaneously. However, the computation of this method is exhaustive and tremendous memory space is

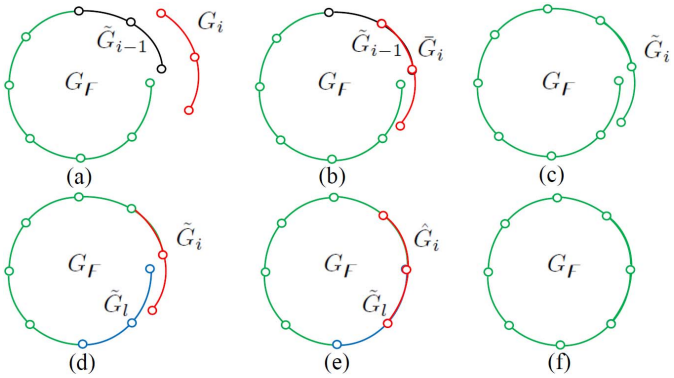


Fig. 7. (a-c) Illustration of registering G_i with G_F. (d-f) Illustration of loop-closure detection between $\tilde{G}_i$ and $\tilde{G}_l$. Please refer to Sect. VI for a detailed description. (a) Registering G_i to G_F through initial registration with $\tilde{G}_{i-1}$. (b) Building correspondences between mapped point set $\bar{G}_i$ and $\tilde{G}_{i-1}$. (c) Updating global point set G_F and denoting the corresponding point set of $\bar{G}_i$ in G_F with $\tilde{G}_i$. (d) Deforming $\tilde{G}_i$ to $\tilde{G}_l$. (e) Loop-closure verification through correspondences between aligned point set $\hat{G}_i$ and $\tilde{G}_l$. (f) Updating global point set G_F and refreshing correspondences between $\{\tilde{G}_k\}_{k=1}^{k=i}$ and G_F.

Notations used in global optimization	
L	the number of local segments
G_l	the global point set of segment l
ED_l	the deformation graph of G_l
F	the reference frame of the total sequence
G_F	the global point set in F
$\bar{G}_i$	the registered point set of G_i into global reference frame F
$\tilde{G}_i$	the corresponding point set of G_i in G_F after registering G_i and adding unmatched points into G_F, which is a subset of G_F
$\hat{G}_i$	the registered point set of $\tilde{G}_i$ to $\tilde{G}_l$

required if a relatively large number of frames are involved in the optimization. In this section, we propose an efficient global optimization to minimize the accumulated error and eliminate the drift.

Through registering local segment, all frames in each segment are non-rigidly aligned with the corresponding global point set in local frame denoted as $G_1, G_2, \ldots, G_L$ for L segments. Since correspondences are established between all frames and the global point set in each segment, we can achieve the goal of registering all frames in the whole sequence by aligning all these global point sets together. In this way, it is only necessary to compute deformations for the global point set of each local segment but not for all frames. Thus, the number of variables is largely reduced, and the computation and memory space is significantly decreased during the optimization. A deformation graph ED_l is computed for each global point set G_l. We employ the same framework in registering local segment to align $G_1, G_2, \ldots, G_L$. The reference frame of the total sequence is set to the first frame denoted as F. We define a global point set G_F in F initialized using G_1. Assuming G_{i-1} has been registered to G_F and unmatched points are added into G_F, we denote the corresponding point set of G_{i-1} in G_F as $\tilde{G}_{i-1}$ which is a subset of G_F. We now describe the process of registering G_i with G_F. An illustration of this process is shown in Fig. 7(a-c). Since there is a common depth scan between neighboring segments, we can register G_i to $\tilde{G}_{i-1}$ using the common scan as initial

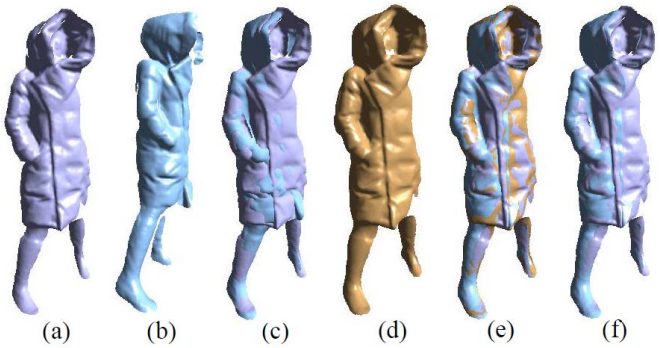


Fig. 8. Loop-closure detection of [9] and our method. (a, b) Two aligned partial scans created as [9]. (c) Due to small overlap, there is high registration error between two scans and loop closure fails to be detected using method in [9]. (d) A global scan formed by global points in reference frame of the total sequence. (e) Each can is registered to the global scan accurately using our method. (f) Registration of two scans by removing global scan from (e). It shows that loop closures between two aligned scans are successfully identified using our method.

correspondences. Then, G_i is further deformed to G_F using newly built correspondences, and obtain a mapped point set $\bar{G}_i$. We match $\bar{G}_i$ with $\tilde{G}_{i-1}$, and G_F is updated by adding unmatched points in $\bar{G}_i$. We use $\tilde{G}_i$ (a subset of G_F) to denote the corresponding points of $\bar{G}_i$ in the global point set G_F. Performing registration on consecutive segments can alleviate but not completely address the drift problem. Hence, we detect loop closures and use global optimization to eliminate the drift.

Loop closures are detected between $\tilde{G}_i$ and each previous aligned point set $\{\tilde{G}_l\}_{l=1}^{i-K}$ (we set $K = 2$ in our experiments to ensure enough movement). Detection between $\tilde{G}_i$ and $\tilde{G}_l$ consists of two steps. In the first step, we search correspondences between $\tilde{G}_i$ and $\tilde{G}_l$ using the method described in section IV. Correspondence is correct if two points have a distance within $\delta = 5cm$ and have a compatible normal angle less than 90°. To measure the overlap of $\tilde{G}_i$ and $\tilde{G}_l$, we define the overlap ratio as

$$d(\tilde{G}_i, \tilde{G}_l) = \frac{C_i}{M_i} \tag{10}$$

where C_i is the found correspondence number and M_i is the point number in $\tilde{G}_i$. In the second step, if $d(\tilde{G}_i, \tilde{G}_l) \geq r_1$ (we use $r_1 = 30\%$), we then perform a direct alignment of $\tilde{G}_i$ towards $\tilde{G}_l$ resulting in aligned point set $\hat{G}_i$. Correspondence is searched within $\delta = 1cm$ and the overlap ratio is recomputed between $\hat{G}_i$ and $\tilde{G}_l$. If $d(\hat{G}_i, \tilde{G}_l) \geq r_2$ (we use $r_2 = 50\%$), we think there is a loop closure between $\tilde{G}_i$ and $\tilde{G}_l$. An illustration of loop-closure detection is given in Fig. 7(d-f). Since the number of segments is small, there are only a few times of loop-closure detection for each segment. Our loop-closure detection is directly applied on the global point set G_F which is more effective to detect all loop closures and solve the drift problem.

Fig. 8 illustrates a comparison between our method and loop-closure detection of [9]. In [9], aligned scans are generated by fusing consecutive 10 frames, and loop closure is identified by registering them. However, because of small overlap between two scans, it is challenging to accurately register them and then detect the loop closure. Thus, it is infeasible to recognize all loop closures through pairwise

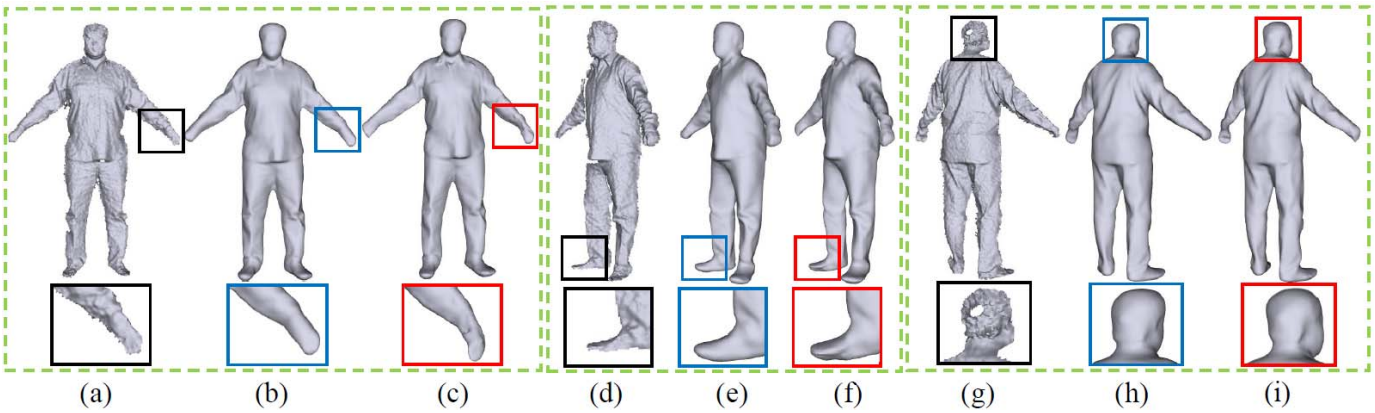


Fig. 9. Comparison with [7]. (a), (d), (g) some depth input; (b), (e), (h) the reconstructed 3D models by our method; (c), (f), (i) the reconstructed 3D models by [7].

registration of scans. In our method, global points G_F fuse more complete data and each local segment is aligned and built point correspondences with G_F. Based on these point correspondences, overlapping regions among all segments are accurately detected.

If a loop closure is detected, the matched correspondences are merged between $\tilde{G}_i$ and $\tilde{G}_l$. G_F is updated, and the correspondences are then refreshed between $\{\tilde{G}_k\}_{k=1}^{k=i}$ and G_F. With the updated global points and correspondences, we globally refine the deformations of $\{ED_k\}_{k=1}^{k=i}$ and 3D points in G_F by employing nonrigid bundle adjustment in Eq. (8). We can uniformly sample G_F to reduce computational complexity using a method similar to [12]. Through iterative loop-closure detection and global optimization, we can eliminate the drift problem. After handling all segments, all depth scans are registered together based on known point correspondences with corresponding global point set in the local frame. Finally, a watertight or partial 3D model is obtained using Poisson surface reconstruction [26] or volumetric fusion [27].

VII. Generating Dynamic 3D Model Sequence

In this section, we generate a dynamic 3D model sequence that fits to the input depth scans by deforming the recovered model back to each frame. After global optimization in previous section, all depth scans are aligned with the reconstructed model and the correspondences can be established between each mapped depth scan and the reference model. Note that the correspondences have been built during registering scans. The reconstructed model is then deformed to each raw depth scan using the correspondences. In order to solve the jittering problem of the recovered model sequence, we introduce a temporal constraint to smooth the recovered motion and predict reasonable positions for occluded parts. We compute the model deformation by minimizing the following energy

$$\begin{aligned} E_{\mathbf{T}^i} = \sum_{i=1}^{N} (w_{fit} \sum_{(\mathbf{v}_j, \mathbf{p}_j^i) \in C_i} \| \mathbf{v}_j^i - \mathbf{p}_j^i \|_2^2 \\ + w_{rigid} E_{rigid}(\mathbf{T}^i) + w_{smooth} E_{smooth}(\mathbf{T}^i)) \\ + w_{temp} \sum_{j=1}^{V_n} \sum_{i=2}^{N-1} \| 2\mathbf{v}_j^i - \mathbf{v}_j^{i-1} - \mathbf{v}_j^{i+1} \|_2^2 . \end{aligned} \quad (11)$$

where $\mathbf{v}_j$ is a vertex on the recovered model, $\mathbf{p}_j^i$ is the correspondence point for $\mathbf{v}_j$ on ith depth scan, and $\mathbf{v}_j^i$ is the transformed position of $\mathbf{v}_j$ to ith depth scan. V_n is the vertex number of the model, w_{temp} is the weight of the temporal constraint, and $\mathbf{T}^i$ is the deformation of the model to ith depth scan. $E_{rigid}(\mathbf{T}^i)$ and $E_{smooth}(\mathbf{T}^i)$ impose that model deformation to ith depth scan should satisfy rigid and smooth constraints, respectively. In each minimization, N consecutive frames are refined together. Then, for each depth scan, there are N estimated model deformations. The final model is computed by averaging all deformed models. Some reconstruction results are shown in Fig. 16.

VIII. Experiments

In this section, we evaluate our approach on both real and synthetic data, and compare our method to state-of-the-art techniques.

A. Comparison With Implicit Loop Closure [7]

Implicit loop closure is adopted in both [7] and DynamicFusion [8] to distribute accumulation error. The two systems have a similar framework for nonrigid depth sequence registration. Zeng et al. [7] proposed a reconstruction method for quasi-rigid objects, while DynamicFusion [8] was proposed to reconstruct partial surfaces but not watertight models for dynamic objects. Here, we compare our method to [7]. We capture a test sequence of a full human body with a single Kinect. During scanning, the subject keeps his pose while turning around in the front of the depth camera. Fig. 9 shows the comparison of our method with [7] in reconstruction quality. There are clear artifacts in the reconstructed result using [7] because implicit loop closure fails to build correct loop-closure constraint if the accumulation error is too large. In contrast, our method faithfully reconstructs the high-quality 3D model.

B. Comparison With Error Distribution of [9]

Dou et al. [9] proposed a reconstruction system for continuously deforming objects. In their method, error distribution is used to detect loop closures, and a final nonrigid bundle adjustment is applied to refine results of error distribution.

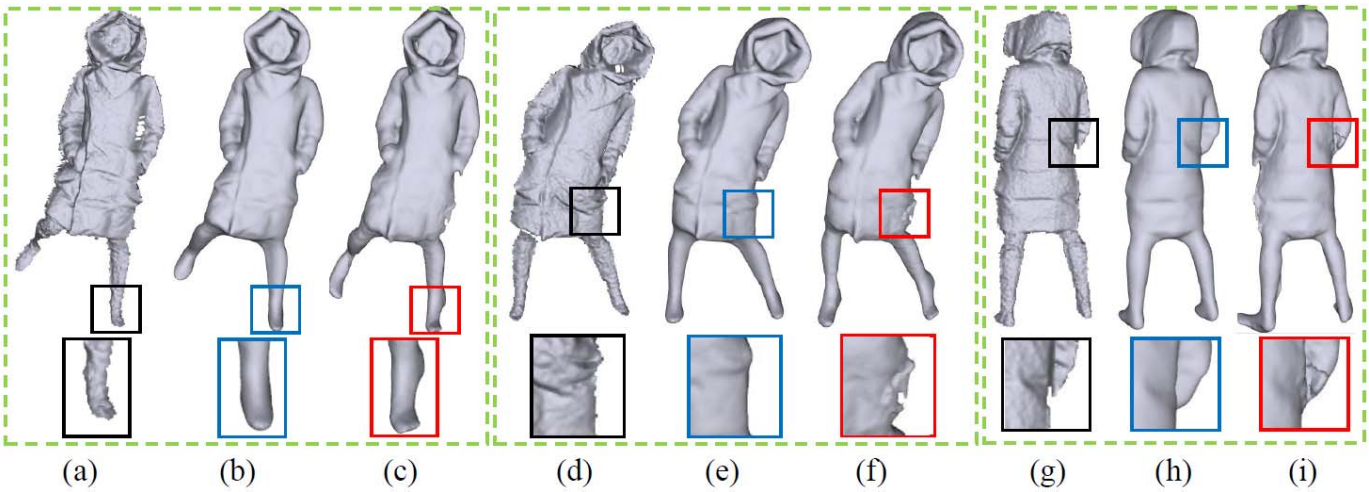


Fig. 10. Comparison with Error Distribution of [9]. (a), (d), (g) some depth input; (b), (e), (h) the reconstructed 3D models by our method; (c), (f), (i) the reconstructed 3D models by Error Distribution of [9].

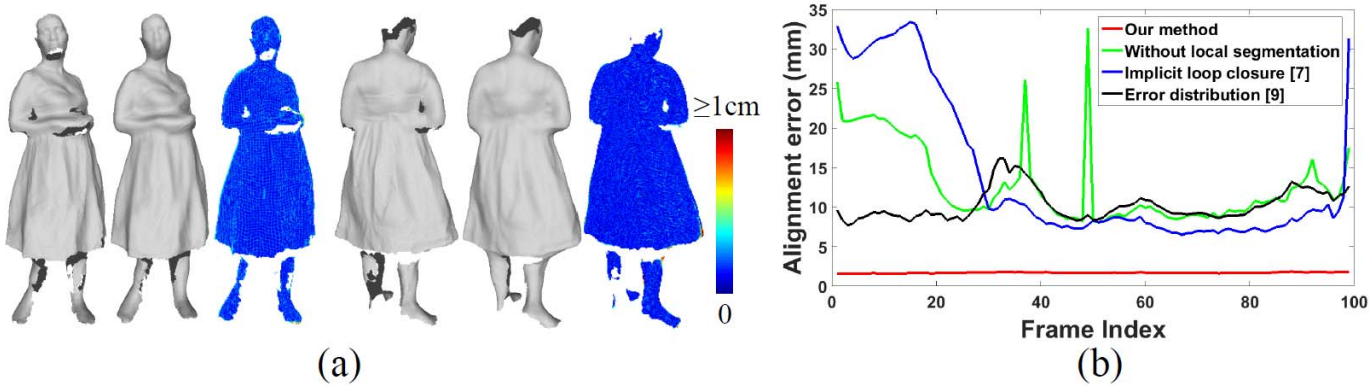


Fig. 11. Comparison results on Saskia dataset. (a) Alignment error map of our method. The first model in each example is the ground truth, the second one is our reconstructed model, and the third image shows the alignment error. (b) The alignment error of each frame using different methods.

In this experiment, we compare our method with loop-closure detection of [9], error distribution (ED), to demonstrate our method still can achieve a good reconstruction based on a robust and accurate loop-closure detection without their final computation exhaustive step. The test data used in this experiment is a dynamic human body sequence. Comparison results in reconstruction quality are shown in Fig. 10. The results demonstrate better robustness and accuracy of our method over error distribution of [9]. In [9], loop closures are detected by registering two scans. The method inevitably misses some true loop closures and detects false ones even with careful detection. In comparison, our method adopts the framework of traditional SfM in which global points fuse more complete depth information and closely bundle all frames together. In addition, by employing the local-to-global hierarchical optimization framework, drift problem is eliminated successfully.

C. Synthetic Data

Saskia dataset [33] is a public 3D model sequence which contains dramatic deformations. We synthesize a depth sequence from each 3D model rotating around the subject as [9]. We test our method with and without local segmentation on the synthesized depth data, and compare to error redistribution of both [7] and [9]. We further compare our method to final result reported in [9]. We deform the reconstructed models to each frame data using the estimated motions of different methods. The alignment error is then

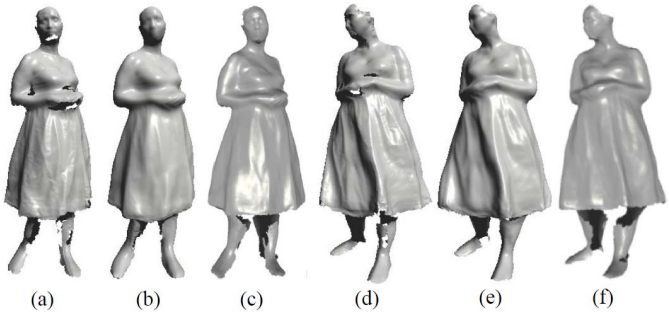


Fig. 12. Comparison with [9] on Saskia dataset. (a) and (d) the ground truth; (b) and (e) the reconstructed 3D models by our method; (c) and (f) the reconstructed 3D models from [9]. Our method can obtain visually comparable results with [9].

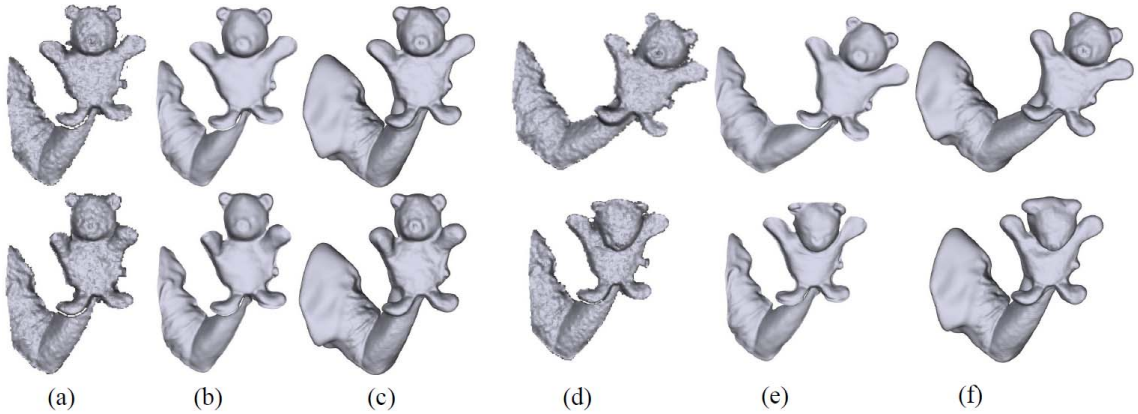


Fig. 13. Comparison with template-based method [4] on "Puppet" data. (a) and (d) two depth scans; (b) and (e) the reconstructed 3D models by our method; (c) and (f) the reconstructed 3D models by [4]. In [4], a pre-scanned template model is deformed to track the object motion. Our method can reconstruct the 3D model and track the motion simultaneously without a template model.

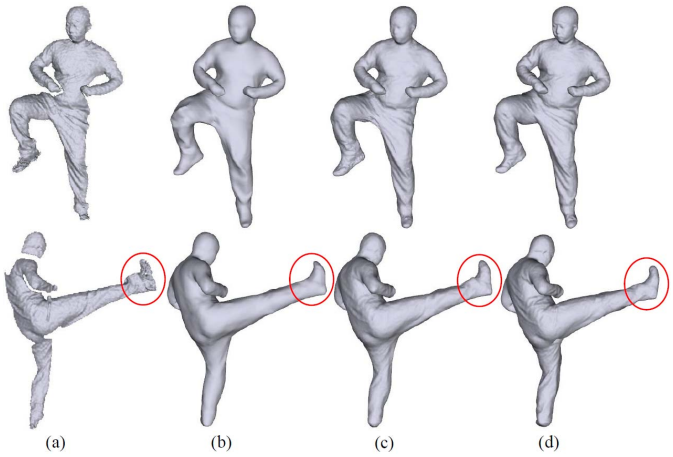


Fig. 14. Comparison with template-based method [4] on "SideKick" data. (a) two depth scans; (b) the reconstructed 3D models by our method; (c) the reconstructed 3D models by our method with surface refinement; (d) the reconstructed 3D models by [4]. Without a template model, our method can track the motion with similar or even better accuracy than [4] (e.g. foot shown in the marked area).

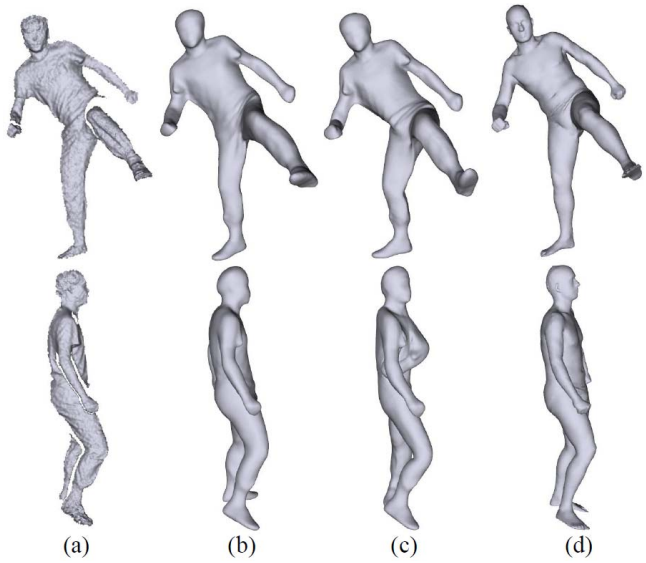


Fig. 15. Comparison with skeleton-driven deformation [1] and data-driven method [34]. (a) some input depth scans; (b) the reconstructed 3D models by our method; (c) the reconstructed 3D models using skeleton-driven deformation [1]; (d) the reconstructed models using data-driven method [34]. Compared with the above two template-based methods, our results are more consistent with input depth scans in both reconstruction accuracy and personalized details. Note that our method does not need to create a template model with embedded skeleton or build a model database of the modeling targets.

measured between deformed models and synthesized depth data as [9]. Fig. 11(a) shows two examples of the error maps between our results and the ground truth. Please refer to our supplementary video to watch more results. We also draw alignment error of each frame using different methods in Fig. 11(b). The alignment error of [9] reported in Fig. 11(b) is after the loop-closure detection step (error distribution) in [9]. We compare our method with error distribution in [9] to demonstrate that our method can detect loop closures with much higher accuracy. Alignment error of our method is much lower than other methods which demonstrates that our method achieves nonrigid error redistribution with higher accuracy than state of the art. Merit of local segmentation in our approach is testified from the comparison results.

A final bundle adjustment (BA) is employed in [9] to simultaneously optimize deformations of all scans, recovered shape, and point correspondences after error distribution (ED), while our proposed method optimizes deformations and surface geometry with fixed correspondences. Our approach is a compromise between computational effort and geometry quality compared to [9]. Without optimizing correspondences, our method can obtain an average error (about $1.6mm$) that is rather close to the final error reported in [9] (about $1mm$). The visual results are comparable with [9] as shown in Fig. 12 (results of [9] are from their paper). Although the accuracy of [9] is a bit higher than our method, our method is much faster and requires less memory. Since the data of [9] is not released publicly, we capture a sequence of a human body with the same number of frames and similar motions as [9] to compare computational efficiency. For a sequence with 400 frames, our method takes about 12 minutes to register all

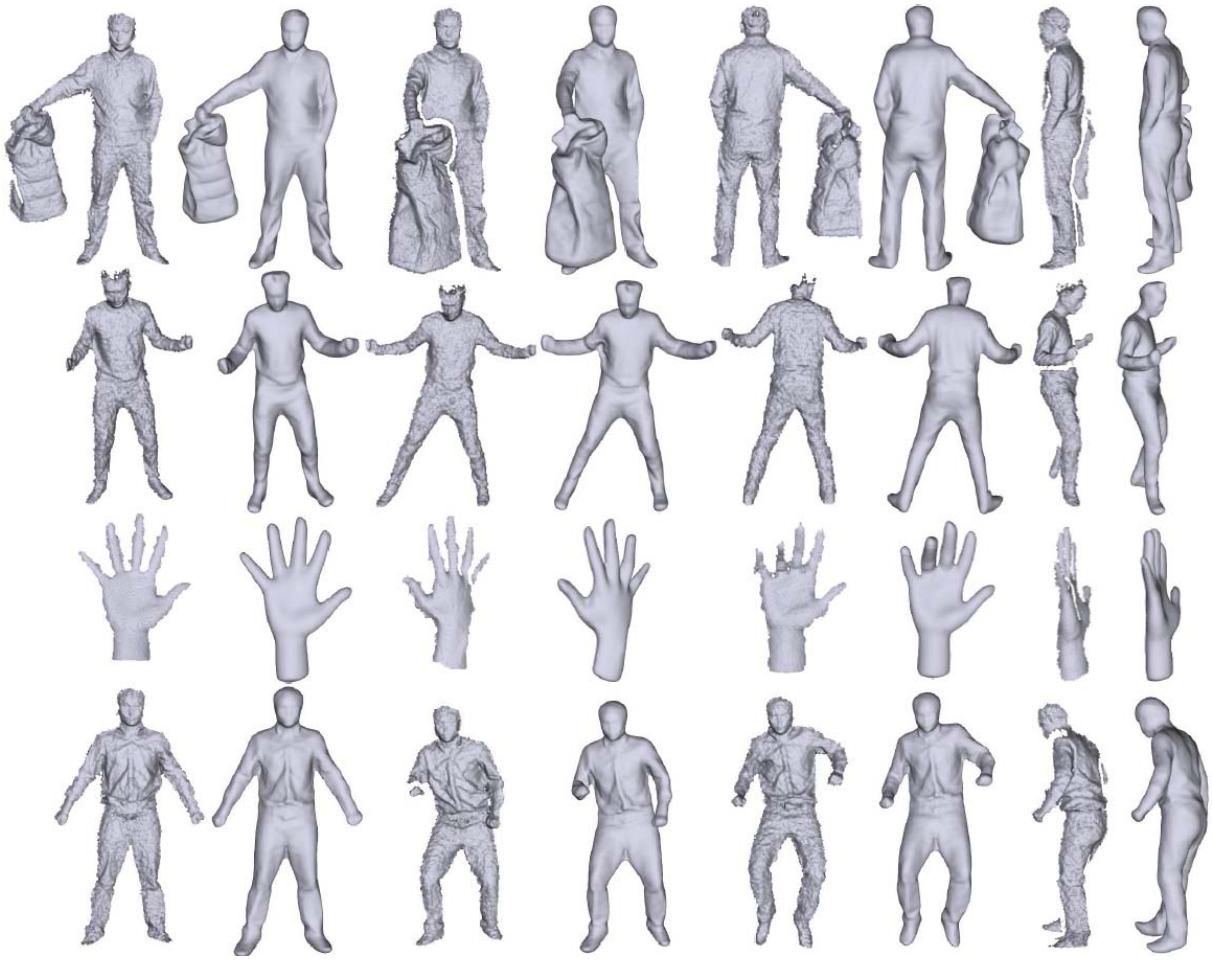


Fig. 16. Results of our method. For each result, we show the input depth scan and the reconstructed model. From top to bottom: "Grasping" data, "Dancing" data, "Hand" data, and "Kungfu" data.

depth scans, while in [9] partial scan preprocessing stage takes around 30 seconds per frame, ED takes about 1 hour and BA up to 5 hours (nearly 10 hours in total). Also, BA of [9] needs a machine with $64G$ memory, but $20G$ is completely enough for our method. The comparison with [7] demonstrates that our method significantly outperforms [7] in both accuracy and robustness. In addition, both [7] and [9] focus on recovering a reference model not the whole model sequence.

D. Comparison With Template-Based Approaches

We first compare against the template-based method [4]. We use "Puppet" data from [4] as test data. In [4], a template model of the dynamic target is scanned beforehand and then used to track motion through L0 regularization. The comparison results on "Puppet" data are shown in Fig. 13. Without template models, our method still can reconstruct similar quality as [4]. In addition, we test our method on a more challenging "SideKick" data from [4] that contain strong non-rigid deformations and fast motions. The comparison results on "SideKick" data are shown in Fig. 14. The reconstruction results of [4] have more details because surface details are transferred from raw depth scans to the tracked template models directly using a final surface refinement. Surface refinement is not the focus of our method, but we also can refine the details using a postprocessing step as [4] (shown in Fig. 14(c)). Without a template model, our method can obtain similar or even better tracking results (e.g. foot) than [4].

We further compare our method to skeleton-driven deformation [1] and data-driven method [34] on the "Kicking" data. In the BlendSCAPE model proposed in [34], rotation of each template triangle is a linear blend of the rotations of body parts, which introduces less artifacts near joints compared with SCAPE model [16]. We adopt the BlendSCAPE model and define the linear blend weights manually as [34]. In [34], a BlendSCAPE model is trained from multiple people in multiple poses using a coregistration method. Instead, we train the BlendSCAPE model on the dataset of SCAPE using the training method of [16]. The comparison with data-driven method [34] aims to demonstrate the advantage of our method in recovering personalized details even without a 3D model dataset. The comparison results are shown in Fig. 15. Please refer to the supplementary video for the complete frames. Skeleton-driven deformation [1] requires the embedding of a predefined skeleton into a template model and can track articulated motions. But, the tracking is easy to fail under occlusion with a single depth camera. There are also some distortions in the reconstruction results because deformations of some parts

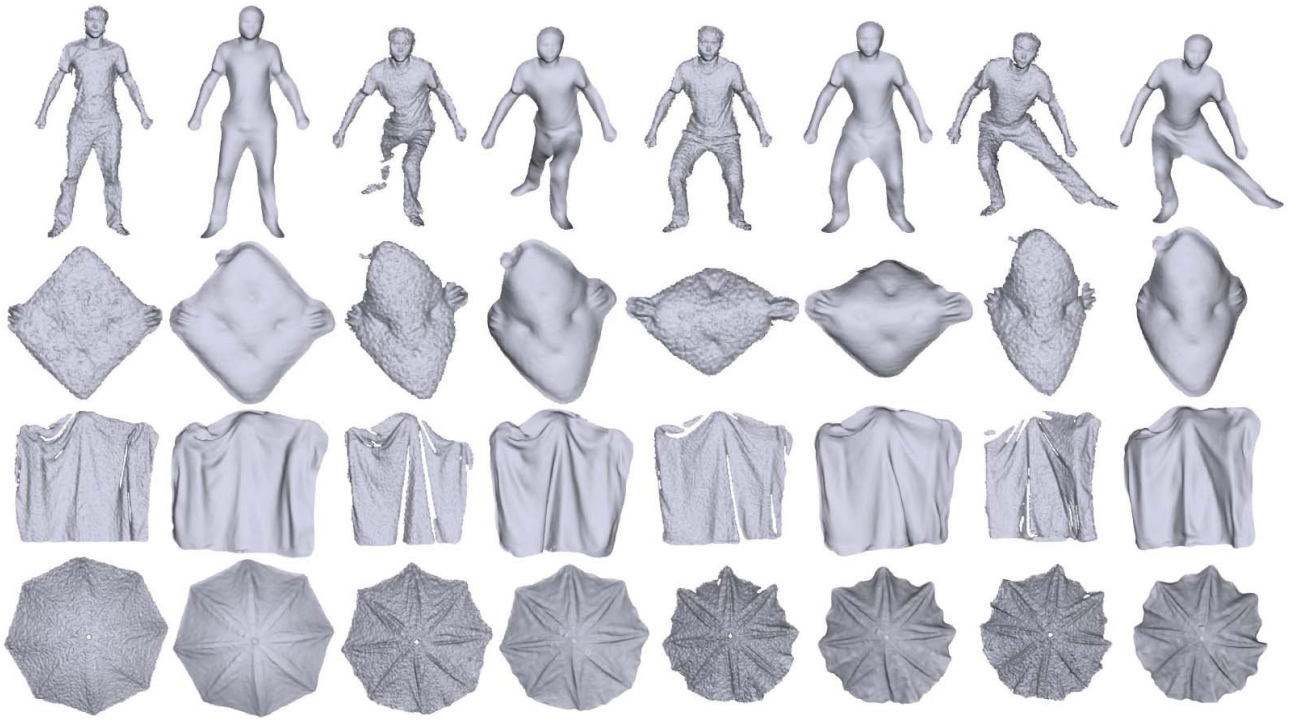


Fig. 17. Reconstruction results of our method on general motion and deformation. For each result, we show the input depth scan and the reconstructed model. From top to bottom: "Crouching" data, "Pillow" data, "Cloth" data, and "Umbrella" data.

(e.g. waist) cannot be accurately modeled even with a carefully embedded skeleton. In data-driven method [34], a parametric model is learned from a 3D model dataset of undressed human bodies, so the method cannot recover personalized details of modeling targets. In contrast, our method can robustly and accurately reconstruct a deforming 3D model sequence of the dynamic objects.

E. Real Data

We test our system using various real data. The boy "Dancing" data is from [4], the "Hand" data is scanned using a Artec Scanner, and other data are captured by a Kinect. Our reconstruction results are shown in Fig. 16. In "Grasping" data, it is very inconvenient to build template priors for both the human body and the clothes. So using template-based methods is unpractical. Without templates, our method still can recover expressive results. More results on general non-rigid examples shown in Fig. 17 support that our method can handle a variety of motion and deformation. Please refer to the supplementary video for the complete results. The reference models are generated by merging all depth data. We run our experiments on a desktop PC with 8-core 4.0 GHz Intel i7 CPU and 20GB RAM, and a NVIDIA GeForce GTX Titan X display card. The running time relates to the size of modeling targets. For human body data, data preprocessing stage takes around $0.1s$ per frame, local nonrigid bundle adjustment takes about $1.2s$ per frame, global optimization takes about $3m$ in total, and generation of model sequence takes about $0.3s$ per frame. Multi-thread acceleration is applied on data preprocessing stage and model sequence generation, and GPU acceleration is used in deformation graph generation and correspondence finding. Local nonrigid bundle adjustment are performed in parallel for all segments. Note that these results are obtained with our unoptimized code and there is much room for acceleration. We can speed up optimization process of our energy function through GPU [35]. Since some steps such as local nonrigid bundle adjustment can be easily parallelized, we believe the computation time can be significantly reduced with GPU acceleration.

F. Application: Reconstruction From Multiple Sequences

Our method can reconstruct a non-rigid object in a fast manner from a single monocular depth camera and is suitable for handling multiple depth sequences. In practice, multiple sequences of a target may be captured when scanning process interrupts or complete data is collected from various sides of the object. An intermediate reconstruction of partial data is beneficial to continue the scanning. In this experiment, we test our method on three depth sequences of a human body. The human rotates by himself in front of a Kinect. During the process, the scanning is broken and one sequence with 101 frames is captured. Our method can immediately recover a surface from this sequence in about 2 minutes. Under the guidance of generated surface, we start another data scanning on the same human body. In total, three sequences are captured to cover the whole human body. The second sequence has 85 frames, and the third sequence 53 frames. Fig. 18 shows reconstructed results on three sequences and each sequence only captures a part of the human body. By combining global points of three sequences, we reconstruct a complete 3D model

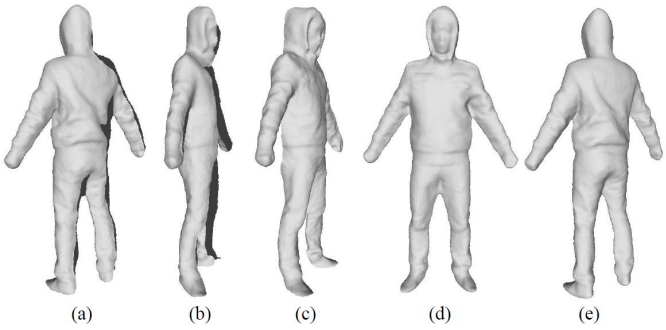


Fig. 18. Reconstruction from multiple depth sequences using our method. (a) reconstruction result of the first sequence; (b) reconstruction result of the second sequence; (c) reconstruction result of the third sequence; (d) and (e) the reconstructed complete model shown from two different views by integrating three sequences. Our method is convenient to handle multiple sequences in a fast manner.

Fig. 19. Reconstruction from multiple depth sequences using our method on "Bag" data (top row) and "Bear" data (bottom row). (a) reconstruction result of the first sequence; (b) reconstruction result of the second sequence; (c) reconstruction result of the third sequence; (d) reconstruction result of the fourth sequence; (e) and (f) the reconstructed complete model shown from two different views by integrating four sequences.

of the human body (totally about 6 minutes). In comparison, it is unpractical to merge multiple sequences using method [9] by waiting several hours to obtain an intermediate reconstruction. We test our method on another two examples (i.e. "Bag" data and "Bear" data) of multiple sequences. In these examples, we need capture multiple depth sequences to recover the full models of the targets. The reconstruction results are shown in Fig. 19. Please refer to the accompanying video for the complete frames.

IX. Conclusion

In this paper, we present a novel templateless nonrigid reconstruction and tracking system for deforming objects with an RGB-D camera. We propose an effective local-to-global hierarchical optimization framework motivated by traditional structure-from-motion. Our framework combines an efficient local nonrigid bundle adjustment and a robust global optimization to register all partial depth scans together and eliminate drift successfully under dynamic motion. Compared to the state of the art, our templateless method not only can effectively handle considerable motions in a robust way, but also can recover a 3D object model and track the object's motion simultaneously with much less computational cost and memory space.

Our approach is ineffective to handle complex and occluded motions which is still an open problem. These motions may pose challenge for nonrigid registration due to large discrepancy of shapes and loss of data. Multiple depth cameras can effectively mitigate the occlusion challenge. There is also a lack of details on the recovered models. This is because details differ dramatically on different depth scans under large deformations (e.g. crouching), and merging them smooths out high-frequency geometry details. We can refine geometry details from raw depth scans. Our method cannot handle strong topological changes because currently used deformation model does not allow for topological variance on the same modeling target. There are also some motions that cannot be recovered accurately by the deformation model such as twist deformation (e.g. the hand rotating a bag in "Bag" data). For very fast motions, it is challenging to build correspondences between consecutive frames. The most recent works [24], [36] present correspondence finding methods for very fast motions based on machine learning or deep learning. We think these methods are promising and can provide us accurate correspondences even under large frame-to-frame motions. In the near future, we will extend our method to deal with very fast motions and topological changes. Also, we will accelerate our system through GPU to achieve near real-time performance. We believe our templateless method will pave the way for many new and interesting applications in augmented and virtual reality, where interaction with non-rigidly deforming objects is of fundamental importance.

Acknowledgment

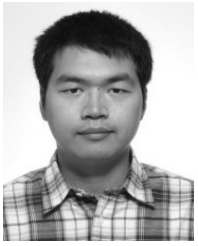
The authors would like to thank all the reviewers for their constructive comments to improve this paper. The authors are grateful for hardware donations from NVIDIA Corporation. They also thank the authors of [4] for providing their results for comparison.

References

[1] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt, "Performance capture of interacting characters with handheld kinects," in *Proc. ECCV*, Oct. 2012, pp. 828–841.
[2] Q. Zhang, B. Fu, M. Ye, and R. Yang, "Quality dynamic human body modeling using a single low-cost depth camera," in *Proc. CVPR*, 2014, pp. 676–683.
[3] M. Zollhöfer *et al.*, "Real-time non-rigid reconstruction using an RGB-D camera," *ACM Trans. Graph.*, vol. 33, no. 4, p. 156, 2014.
[4] K. Guo, F. Xu, Y. Wang, Y. Liu, and Q. Dai, "Robust non-rigid motion tracking and surface reconstruction using L0 regularization," in *Proc. ICCV*, 2015, pp. 3083–3091.
[5] F. Bogo, M. J. Black, M. Loper, and J. Romero, "Detailed full-body reconstructions of moving people from monocular RGB-D sequences," in *Proc. ICCV*, 2015, pp. 2300–2308.
[6] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong, "Modeling deformable objects from a single depth camera," in *Proc. ICCV*, Sep. 2009, pp. 167–174.
[7] M. Zeng, J. Zheng, X. Cheng, and X. Liu, "Templateless quasi-rigid shape modeling with implicit loop-closure," in *Proc. CVPR*, Jun. 2013, pp. 145–152.
[8] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. CVPR*, 2015, pp. 343–352.
[9] M. Dou, J. Taylor, H. Fuchs, A. Fitzgibbon, and S. Izadi, "3D scanning deformable objects with a single RGBD sensor," in *Proc. CVPR*, 2015, pp. 493–501.

[10] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2004.
[11] Q.-Y. Zhou and V. Koltun, "Dense scene reconstruction with points of interest," *ACM Trans. Graph.*, vol. 32, no. 4, p. 112, 2013.
[12] K. Wang, G. Zhang, and H. Bao, "Robust 3D reconstruction with an RGB-D camera," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4893–4906, Nov. 2014.
[13] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, "Multi-view stereo for community photo collections," in *Proc. ICCV*, 2007, pp. 1–8.
[14] M. I. A. Lourakis and A. A. Argyros, "SBA: A software package for generic sparse bundle adjustment," *ACM Trans. Math. Softw.*, vol. 36, no. 1, p. 2, Mar. 2009.
[15] G. Zhang, H. Liu, Z. Dong, J. Jia, T.-T. Wong, and H. Bao, "Efficient non-consecutive feature tracking for robust structure-from-motion," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5957–5970, Dec. 2016.
[16] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, and J. Rodgers, "SCAPE: Shape completion and animation of people," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 408–416, Jul. 2005.
[17] Y. Chen, Z. Liu, and Z. Zhang, "Tensor-based human body modeling," in *Proc. CVPR*, 2013, pp. 105–112.
[18] B. Allen, B. Curless, Z. Popović, and A. Hertzmann, "Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis," in *Proc. ACM SIGGRAPH*, 2006, pp. 147–156.
[19] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel, "A statistical model of human pose and body shape," *Comput. Graph. Forum*, vol. 28, no. 2, pp. 337–346, Mar. 2009.
[20] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, "Scanning 3D full human bodies using Kinects," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 4, pp. 643–650, Apr. 2012.
[21] H. Li, E. Vouga, A. Gudymv, L. Luo, J. T. Barron, and G. Gusev, "3D self-portraits," *ACM Trans. Graph.*, vol. 32, p. 187, Nov. 2013.
[22] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time volumetric non-rigid reconstruction," in *Proc. ECCV*, 2016, pp. 362–379.
[23] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo, and motion reconstruction using a single RGB-D camera," *ACM Trans. Graph.*, vol. 36, no. 3, p. 32, 2017.
[24] M. Dou *et al.*, "Fusion4D: Real-time performance capture of challenging scenes," in *ACM SIGGRAPH*, 2016, pp. 1–13.
[25] R. Wang *et al.*, "Capturing dynamic textured surfaces of moving targets," in *Proc. ECCV*, 2016, pp. 271–288.
[26] M. Kazhdan, M. Bolitho, and H. H. Hoppe, "Poisson surface reconstruction," in *Proc. Eurograph. Symp. Geometry Process.*, 2006, pp. 61–70.
[27] M. Dou, J.-M. Frahm, and H. H. Fuchs, "Scanning and tracking dynamic objects with commodity depth cameras," in *Proc. Int. Symp. Mixed Augmented Reality*, 2013, pp. 99–106.
[28] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Trans. Graph.*, vol. 26, no. 3, p. 80, 2007.
[29] H. Li, B. Adams, L. J. Guibas, and M. Pauly, "Robust single-view geometry and motion reconstruction," *ACM Trans. Graph.*, vol. 28, p. 175, Dec. 2009.
[30] *Point Cloud Library*. Accessed on Aug. 10, 2014. [Online]. Available: http://www.pointclouds.org/
[31] R. Kimmel and J. A. Sethian, "Computing geodesic paths on manifolds," *Proc. Nat. Acad. Sci. USA*, vol. 95, no. 15, pp. 8431–8435, 1998.
[32] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *J. Image Vis. Comput.*, vol. 10, no. 3, pp. 144–155, 1992.
[33] D. Vlasic *et al.*, "Dynamic shape capture using multi-view photometric stereo," *ACM Trans. Graph.*, vol. 28, no. 5, p. 174, 2009.
[34] D. A. Hirshberg, M. Loper, E. Rachlin, and M. J. Black, "Coregistration: Simultaneous alignment and modeling of articulated 3D shape," in *Proc. ECCV*, 2012, pp. 242–255.
[35] D. Zachary *et al.* (Apr. 2016). "Opt: A domain specific language for non-linear least squares optimization in graphics and imaging." [Online]. Available: https://arxiv.org/abs/1604.06525
[36] L. Wei, Q. Huang, D. Ceylan, E. Vouga, and H. Li, "Dense human body correspondences using convolutional networks," in *Proc. CVPR*, 2016, pp. 1544–1553.

Kangkan Wang received the B.S. degree in computer science from Northwestern Polytechnical University in 2009, and the Ph.D. degree from the State Key Lab of CAD&CG, Zhejiang University, in 2015. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. Prior to that, he was an Assistant Professor with the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include face modeling, human body reconstruction, 3D reconstruction, nonrigid reconstruction, and motion tracking.



Guofeng Zhang (M'07) received the B.S. and Ph.D. degrees in computer science from Zhejiang University in 2003 and 2009, respectively. He is currently an Associate Professor with the State Key Laboratory of CAD&CG, Zhejiang University. His research interests include structure-from-motion, SLAM, 3D reconstruction, augmented reality, and video segmentation and editing. He received the National Excellent Doctoral Dissertation Award and the Excellent Doctoral Dissertation Award of China Computer Federation.

Shihong Xia received the B.S. degree in mathematics from Sichuan Normal University in 1996, and the M.S. degree in applied mathematics and the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, in 1999 and 2002, respectively. He is currently a Professor with the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include computer graphics, virtual reality and artificial intelligence, and focus on modeling and understanding human motion.