Robust 3D Human Motion Reconstruction Via Dynamic Template Construction

Zhong Li[1,2] Yu Ji[2] Wei Yang[1,2] Jinwei Ye[3] Jingyi Yu[1,2,4]

[1]University of Delaware,Newark,USA
lizhong@udel.edu

[2]Plex-VR
{yu.ji,wei.yang}@plex-vr.com

[3]Louisiana State University,Baton Rouge,USA
jinweiye@lsu.edu

[4]ShanghaiTech University,Shanghai,China
yujy1g@shanghaitech.edu

Abstract

In multi-view human body capture systems, the recovered 3D geometry or even the acquired imagery data can be heavily corrupted due to occlusions, noise, limited field-of-view, etc. Direct estimation of 3D pose, body shape or motion on these low-quality data has been traditionally challenging.In this paper, we present a graph-based non-rigid shape registration framework that can simultaneously recover 3D human body geometry and estimate pose/motion at high fidelity.Our approach first generates a global full-body template by registering all poses in the acquired motion sequence.We then construct a deformable graph by utilizing the rigid components in the global template.We directly warp the global template graph back to each motion frame in order to fill in missing geometry. Specifically,we combine local rigidity and temporal coherence constraints to maintain geometry and motion consistencies.Comprehensive experiments on various scenes show that our method is accurate and robust even in the presence of drastic motions.

1. Introduction

Despite tremendous efforts and advances in 3D shape and motion reconstruction [8, 38, 43, 39, 32, 2, 49, 46], reliable estimation of 3D pose, body geometry and motion trajectory remains challenging. 3D reconstruction produced by traditional photogrammetry or multi-view geometry can be heavily corrupted due to occlusions, noise, limited field-of-view, etc. It is possible to add additional cameras to improve the reconstruction but would lead to higher computational and equipment cost. One possible solution is to complete the missing data via geometric operators such as filtering and hole filling (*e.g.*, Poisson surface completion) [20, 19, 26]. By far these methods can only handle small holes. It is also possible to adopt a template-based approach [22, 50, 15] by using pre-reconstructed full body 3D geometry. In reality, generating the template requires special acquisition system that is inaccessible to commodity users. Further, such techniques cannot handle strong deformations caused by drastic motion.

In this paper, we present a graph-based non-rigid shape registration framework that can simultaneously recover 3D human body geometry and estimate motion at high fidelity. Our approach first generates a global full-body template by registering all poses in the acquired motion sequence. We observe that missing body geometry in one frame may appear in other frames in the motion sequence. This implies that we can generate the complete body template by aligning each individual partial reconstruction. To do so, we conduct multi-frame correspondence matching by imposing a temporal coherence constraint. We consider both forward and backward motions to formulate the temporal regularization. We then construct a deformable graph by utilizing the rigid components in the global template. Although the human body is non-rigidity, it can be effectively decomposed into piece-wise rigid components. We hence segment the global template into connected rigid patches and build a deformable graph with centroid of rigid patches as nodes. Finally, we develop a patch surface expansion approach for fitting the global template in terms of each node's motion estimation. We also impose temporal consistency to maintain local rigidity and motion smoothness. The reconstruction pipeline of our algorithm is shown in Fig. 1. Comprehensive experiments on a multi-view system show that our method is accurate and robust even in the presence of drastic motions.

2. Related Work

There is an emerging trend on using multi-view acquisition techniques for reconstructing 3D human body geometry and motion. Notable examples include techniques using a multi-view camera system [32, 1, 29, 28],shape-from-silhouette [14, 5], multiple-view stereo matching [37], and photometric stereo [43, 8]. The focus has been on conduct-

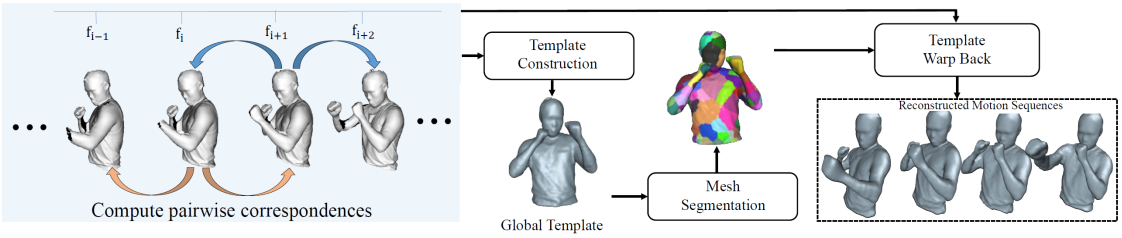

Figure 1. A diagram showing the pipeline of our reconstruction algorithm.

ing non-rigid registration [18, 23] for mesh sequence tracking and 3D reconstruction from the captured data. Most previous work falls into two categories, *i.e*. template-free shape alignment approach and template-based deformation approach.

Template-free shape alignment. This class of methods performs per-frame reconstruction without using a global full body template. Süssmuth *et al*. [41] map all input scans onto a 4D space-time volume and conduct high-dimensional shape reconstruction. Mitra *et al*. [27] also use a 4D space-time representation to compute the motion of the scanned object. They recover the volume by estimating globally consistent motion instead of pairwise alignment. Wand *et al*. [45] applied a statistic framework to conduct pairwise shape alignment if the topology remains consistent. They further improves the template-free shape alignment by using volumetric deformation model [44]. However, the computational complexity is very high and the performance is limited by the running time. The algorithm is also sensitive to corrupted input data such as large shape deformation and/or truncated geometry. To summarize, above template-free approaches can only handle small motion due to the accumulation of tracking errors.

Template-based approaches. This class of methods attempts to utilize geometric template as a shape prior for mesh sequences tracking. Some focus on tracking and reconstructing the model to accommodate general scenarios. Offline approaches such as [22] acquire a coarse low-resolution template via static acquisition and then track the input sequence using embedded deformation [40]. Dou *et al*. [10] use an eight-depth camera system to reconstruct the full body geometry and track the motion by deforming a pre-captured human body template. Zollhöfer *et al*. [50] perform online template acquisition for mesh tracking and use GPU acceleration to achieve real-time performance. However, acquiring the online template requires the motion to be rigid and is prone to errors in case of drastic motions. Newcombe *et al*. [30] extended the Kinect fusion algorithm [31] to perform template-based reconstruction. Their approach is able to capture the non-rigid partial views of a moving person. However, their system can only handle relatively slow motion. Guo *et al*. [15] use L0 based regularizer to achieve more accurate and robust result. More recent approaches [7, 33, 17, 2] adopt a keyframe-based mesh tracking and similarity tree scheme and are able to handle topology changes and significantly reduce the tracking failure rate.

Other non-rigid tracking approaches tackle elastic objects and are suitable for emulating Cartoon style avatars. Vlasic *et al*. [42] apply shape-from-silhouette and deformed a statically acquisition template via linearly blended skinning [21]. Huang *et al*. [16] use a skeleton-based hybrid deformation approach. Rhodin *et al*. [35] and Robertini *et al*. [36] present pleasant results on outdoor motion capture, however, their methods are based on articulated skeleton thus can't applied to general shape. Cargniart *et al*. [3] propose a patch-based approach. [47] and [1] explore fitting 3D body model database onto the acquired data. Similar methods have been also applied to face and hand tracking [25, 4, 34]. Another seminal work of Holoportation by Dou *et al*. [11] achieves real-time performance capture,however, their results are sensitive to background segmentation errors.

Our approach falls into the category of template-based approach. However, different from [50, 22], we do not require a separate process for building the template. Instead, we construct our global template by accumulating individual frames during the capture process. Our system uses a multi-view stereo capture system for data acquisition. However, our input data is corrupted due to viewing frustum truncation and drastic motion. Direct reconstruction from multi-view stereo approach exhibits large holes and even truncations. In our approach, we propose to exploit the temporal redundancies to solve this problem.

3. 3D Human Shape/Motion Reconstruction

Our algorithm consists of four major steps. We first build a global human body template from a motion sequence with incomplete body geometry. In order to achieve this, we establish pairwise correspondences between adjacent motion frames by imposing a temporal regularization term. By minimizing our global deformation energy function, we align the incomplete poses from all frame to a global template. Next, we use the graph-cut algorithm to segment the global template into multiple connected rigid patches and use the segmentation results to determine the global nodes.

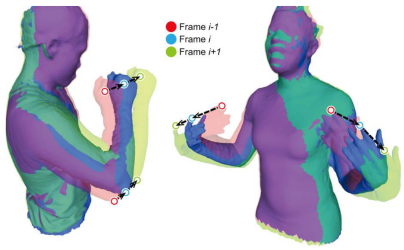

Figure 2. Human motions exhibit temporal smoothness between adjacent frames.

Finally, we estimate the rotation parameters to warp the piece-wise rigid global template back to each input frame in order to recover the full body geometry for the entire motion sequence.

3.1. Pairwise Surface Matching

To build the global template, we first need to register the surfaces from adjacent motion frames. We use a deformation graph technique similar to [40]. Given a sequence of captured N motion frames $\{\boldsymbol{P}^n|_{n=1,...,N}\}$, where a frame P^n has κ vertices $\{v_i|_{i=1,...,\kappa}\}$, where $v_i \in \mathbb{R}^3$, we first uniformly sample a set of graph nodes $\boldsymbol{G} = \{g_1, g_2, ..., g_m\}$ (where $m << \kappa$) on the surface P^n. Once we have graph nodes, we use the deformation of graph nodes to represent the movement of vertex. Specifically, we use affine transformation $\boldsymbol{A} = \{\boldsymbol{A}_t\}_{t=1}^m$ and $\boldsymbol{b} = \{\boldsymbol{b}_t\}_{t=1}^m$ to parameterize the deformable movement of a graph node. After deformation, the new position of a vertex v can be written as:

$$v' = f(v, \boldsymbol{A}, \boldsymbol{b}) = \sum_{t=1}^{m} w_t(v)[\boldsymbol{A}_t(v - g_t) + g_t + \boldsymbol{b}_t] \quad (1)$$

where $w_t(v)$ is the weighing factor of a graph node g_t on the vertex v. In particular, $w_t(v) = max(0, (1 - d^2(v, g_t)/r^2)^3)$, where $d(v, g_t)$ is geodesic distance between v and g_t and r is the distance between v and its K-nearest nodes in the geodesic distance domain (we use K=4 in our experiments).

Once we have constructed the deformation graph, we align the surface onto other frames by finding the optimal affine transformation of its graph nodes. Recall that our input is a sequence of deformable surfaces. To align all the surfaces, a brute-force approach is to use non-rigid registration [6]. A major drawback of using this approach is the lack of stability: deformation errors would accumulate over the frames and can result in failure of the algorithm.

An alternative solution is to perform pairwise correspondence matching [22]. This approach attempts to find correspondences between adjacent meshes. Compared with their inputs, our surface sequences are more challenging due to incomplete geometry and drastic motions. As shown in Fig. 2, although largely overlapped, adjacent surface meshes exhibit temporal smoothness between adjacent motions. Furthermore, the non-rigidity of human body geometry can cause large errors even in the presence of small motions since affine transform is no longer sufficient to characterize the motion. We propose to solve these challenges by exploiting the temporal coherence. As shown Fig. 1, adjacent motion frames are highly consistent due to the motion smoothness. We therefore add a temporal smoothness term to the pairwise correspondence energy function in order to enforce the motion continuity. In particular, we register three consecutive frames (*i.e.* we consider both forward and backward motion) at the same time. As shown in Fig. 1, we warp a frame i onto its previous $(i-1)$ and successive $(i+1)$ frames. Therefore, our pairwise correspondence matching energy function is defined as:

$$\boldsymbol{E}_{total} = \lambda_1 \boldsymbol{E}^{\pm}_{rigid} + \lambda_2 \boldsymbol{E}^{\pm}_{smooth} + \lambda_3 \boldsymbol{E}^{\pm}_{fit} + \lambda_4 \boldsymbol{E}_{tempo} \quad (2)$$

In this equation, we omit the frame stamp n in superscript and use "+" for forward motion "$n \rightarrow n+1$" and "$-$" for backward motion "$n \rightarrow n-1$". $\lambda_1 \sim \lambda_4$ are weighing factors for balancing the regularization terms. In our experiments, we use $\lambda_1 = 100$, $\lambda_2 = 20$, $\lambda_3 = 1$ and $\lambda_4 = 5$. Next, we explain each energy term in Eqn. 2 in details.

The first term $\boldsymbol{E}_{rigid}$ constraints the rigidity enforced by the affine transformation,and thus is defined as:

$$\begin{aligned}\boldsymbol{E}_{rigid} = \sum_{G} \big(&(\boldsymbol{a}_1^T \boldsymbol{a}_2)^2 + (\boldsymbol{a}_2^T \boldsymbol{a}_3)^2 + (\boldsymbol{a}_1^T \boldsymbol{a}_3)^2 \\ &+ (1 - \boldsymbol{a}_1^T \boldsymbol{a}_1)^2 + (1 - \boldsymbol{a}_2^T \boldsymbol{a}_2)^2 + (1 - \boldsymbol{a}_3^T \boldsymbol{a}_3)^2 \big) \end{aligned} \quad (3)$$

where $\boldsymbol{a}_1$, $\boldsymbol{a}_2$ and $\boldsymbol{a}_3$ are the three column vectors that form the 3×3 matrix $\boldsymbol{A}_t$.

The second term $\boldsymbol{E}_{smooth}$ enforces the spatial smoothness of the geometric deformation in one frame and it is written as:

$$\begin{aligned}\boldsymbol{E}_{smooth} = \sum_{t=1}^{m} \sum_{k \in \nu(t)} \hat{w}_{(t,k)} || \boldsymbol{A}_t(g_k - g_t) + g_t + \\ \boldsymbol{b}_t - (g_k + \boldsymbol{b}_k) ||^2 \end{aligned} \quad (4)$$

where $\nu(t)$ is node g_t's neighbor that shares the same edge in the sub-sample graph.

We adopt a data fitting term E_{fit} similar to Iterated Closest Point (ICP) to measure vertex displacements between the reference frame and the target frame. The fitting cost consists of two components: one for minimizing the point-to-point distances and the other for minimizing the point-to-plane distances. Further, instead of using the closest points as correspondences, we trace an undirected ray n_i along the normal direction of the source vertex v_i and choose the vertex that is the closest to the ray-target surface intersection

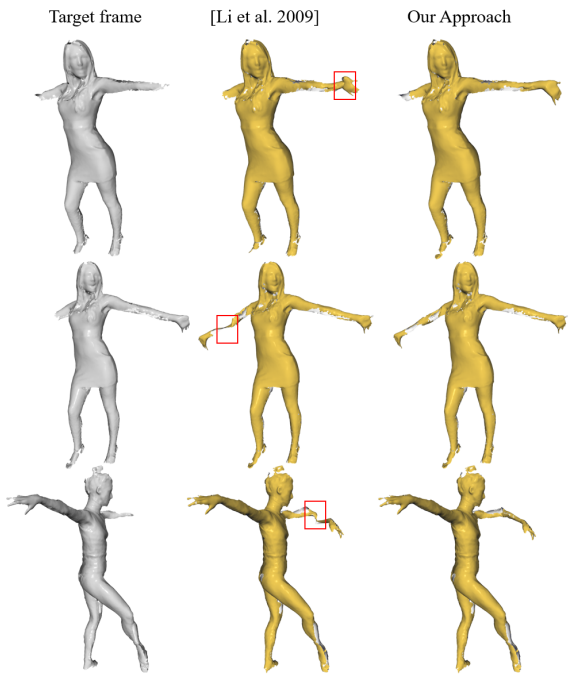

Figure 3. Pairwise correspondence matching results. We show the target frames in the first column. We compare our registration results (third column) with results of [22] (second column).

as the temporary correspondence c_i:

$$\boldsymbol{E}_{fit} = \sum_{i \in P} \lambda_{point} ||v_i - c_i||^2 + \lambda_{plane} |n_i^T (v_i - c_i)|^2 \quad (5)$$

In our experiments, we use $\lambda_{point} = 0.1$ and $\lambda_{plane} = 1$.

Finally, we propose a temporal regularization term $\boldsymbol{E}_{tempo}$ to preserve the motion continuity among three consecutive frames, *i.e.* from frame n to frame $n-1$ as well as frame $n+1$. More specifically, we constrain the current-to-next motion $\{A_t^+, \boldsymbol{b}_t^+\}_{t=1}^m$ by current-to-previous motion $\{A_t^-, \boldsymbol{b}_t^-\}_{t=1}^m$. Since motions between adjacent frames are similar, we formulate a new energy term $\boldsymbol{E}_{tempo}$ to force $A_t^- A_t^+$ close to an identity matrix, and minimize $A_t^- \boldsymbol{b}_t^+ + \boldsymbol{b}_t^-$:

$$\boldsymbol{E}_{tempo} = \sum_{t=1}^{m} ||\mathbb{I} - A_t^+ A_t^-||_F^2 + ||A_t^- b_t^+ + b_t^-||_2^2 \quad (6)$$

where $\mathbb{I}$ is an identity matrix.

In our implementation, we solve Equation. 2 in an iterative manner by using the Gauss-Newton method.

To illustrate the effectiveness of pairwise correspondence optimization algorithm, we show our frame alignment results in Fig. 3 and compare with [22]. Notice that the input frames exhibit severe occlusions and/or geometric truncations. Our algorithm still generates accurate alignment results with fewer artifacts due to the consideration of temporal smoothness term.

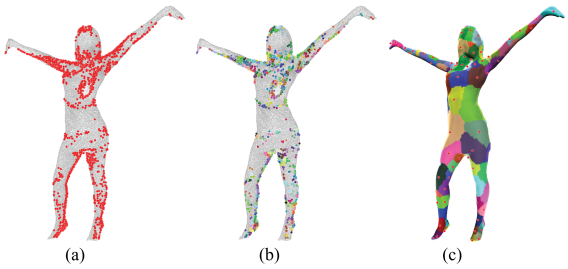

Figure 4. Patch segmentation. (a) All aligned graph nodes in the global template; (b) Grouped nodes (color-coded) after K-means clustering; (c) Final segmentation result after graph-cut.

3.2. Global Template Construction

Recall that our input frames are incomplete and exhibit many missing parts due to occlusions/truncations. We observe that the occluded geometry may appear in later frames as the pose changes. So we set out to align all input frames $\{\boldsymbol{P}^n|_{n=1,\dots,N}\}$ into an optimized pose $\boldsymbol{P}^0$ where nearly all occluded regions are filled. Notice that we have already obtained dense correspondences using the optimization algorithm described in Section 3.1, similar to [24, 12],we further define an energy function E_{global} as follow to construct a global template:

$$\boldsymbol{E}_{global} = \sum_{n=1}^{N} (\lambda_r \boldsymbol{E}_{rigid}^{n \to 0} + \lambda_s \boldsymbol{E}_{smooth}^{n \to 0}) + \lambda_c \boldsymbol{E}_{corr} \quad (7)$$

where $\boldsymbol{E}_{rigid}$ and $\boldsymbol{E}_{smooth}$ are the same as in Eqn. 2. $\boldsymbol{E}_{corr}$ is a data term to impose the distant consistency between corresponding vertices in adjacent frames. $\boldsymbol{E}_{corr}$ is defined as:

$$\begin{aligned} \boldsymbol{E}_{corr} = \sum_{n=1}^{N-1} ||f(\boldsymbol{P}^n, \boldsymbol{A}^{n \to 0}, \boldsymbol{b}^{n \to 0}) \\ - f(f(\boldsymbol{P}^n, \boldsymbol{A}^+, \boldsymbol{b}^+), \boldsymbol{A}^{n+1 \to 0}, \boldsymbol{b}^{n+1 \to 0})||^2 \end{aligned} \quad (8)$$

where $f(\cdot)$ is the deformed position from Eq.1.

In our experiment, we use $\lambda_r = 150$, $\lambda_s = 5$ and $\lambda_c = 1$. We iteratively solve the equation via Gauss-Newton optimization to sequentially align consecutive frames to obtain a global optimal alignment.

Once we align all input frames, we then "stitch" them together to form the final global template. Notice that directly fusing the point clouds can lead to large errors such as discontinuity. We instead fuse their gradients and then reintegrate the surface. The process is analogous to image completion in the gradient domain and in our solution we apply poisson surface reconstruction [19] to obtain the reconstructed template mesh.

3.3. Patch Segmentation

Once we have the global template, we map all input frames $\{\boldsymbol{P}^n|_{n=1,\dots,N}\}$ onto the global template mesh $\boldsymbol{P}^0$

through a common deform graph $\boldsymbol{G}^0$. We assume that the topology (e.g., the number of nodes and edge connectivity) remain consistent across frames. Specifically, we segment the global template mesh $\boldsymbol{P}^0$ into patches and treat the geometry of each patch relatively rigid. We then use the centroid of each patch as the node in the global deform graph $\boldsymbol{G}^0$. In contrary to [3] in which the patch segmentation is performed based on geodesic distance, we also consider the motion similarity among vertices.

In particular, we set out to partition the vertices $\widetilde{v}$ in $\boldsymbol{P}^0$ into relatively rigid subsets. For an input frame $\boldsymbol{P}^n$, we use v' and g' to represent the vertex and graph node respectively after the global registration. We then perform K-means clustering for all aligned graph nodes according to their Euclidean distances. We set the pre-defined number of clusters $\boldsymbol{K}$ as the maximum number of deform graph nodes in all N frames.

For each vertex $\widetilde{v_i}$ in global template mesh $\boldsymbol{P}^0$, we first find its K-nearest neighbors $\Omega(\widetilde{v_i})$ in the aligned vertices v' of all frames $\{\boldsymbol{P}^n|_{n=1,..,N}\}$. We then calculate the weight between $\widetilde{v_i}$ and cluster c_j using the mean value of all weights between vertices v' in $\Omega(\widetilde{v_i})$ and graph nodes g' in c_j:

$$w(\widetilde{v_i}, c_j) = \sum_{v' \in \Omega(\widetilde{v_i})} \sum_{g' \in c_j} w(v', g')/S \tag{9}$$

where S is the total number of valid $w(v', g')$.

Since $w(v', g')$ corresponds to the weighing factor of a graph node g' on vertex v', we can also use $w(\widetilde{v_i}, c_j)$ to determine how significance of cluster c_j with respect to $\widetilde{v_i}$. The set of vertices affected most by the same cluster should have a relative similar rigid motion. Therefore, we can simply treat weight $w(\widetilde{v_i}, c_j)$ as the data cost for assigning $\widetilde{v_i}$ to cluster c_j. We further use the pots form smoothness cost: $p(\widetilde{v_i}, \widetilde{v_k}) = 0$ when $\widetilde{v_i}, \widetilde{v_k}$ have the same label and belong to the same triangle in $\boldsymbol{P}^0$ and 1 otherwise. Finally, we formulate the energy function as:

$$E = -\sum_{\widetilde{v_i}} w(\widetilde{v_i}, c_j) + \lambda \sum_{\{\widetilde{v_i}, \widetilde{v_k}\} \in \mathcal{N}} p(\widetilde{v_i}, \widetilde{v_k}) \tag{10}$$

where λ is a weighting factor. To find an optimal solution, we apply the graph-cut algorithm [9] and we group vertices with the same label into a patch. An example of segmentation result is shown in Fig. 4.

3.4. Surface Expansion and Patch Warping

Once we partition the global template $\boldsymbol{P}^0$ into $\boldsymbol{K}$ patches, We treat each patch $patch_\tau$'s centroid g_t as the graph node.

To warp the global template $\boldsymbol{P}^0$ back to each frame $\boldsymbol{P}^n$, we adopt a two-step approach to first approximate and then optimize the graph motion parameter $\{\{A_t^{0,n}, b_t^{0,n}\}_{t=1}^{\boldsymbol{K}}\}_n^N$ respectively. We first conduct the closest point approximation, second step is to further optimize them which is constrain by adjacent temporal information.

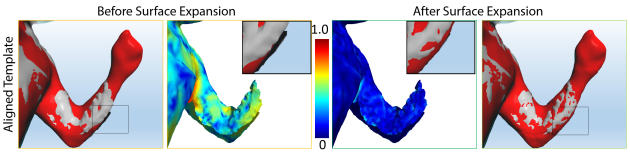

Figure 5. Surface expansion result. Our patch expansion algorithm can effectively reduce the misalignment between deformed frames and the global template.

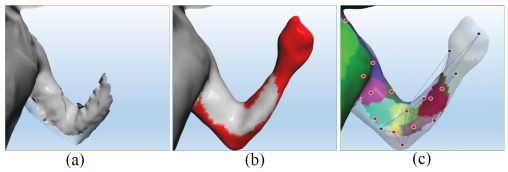

Figure 6. Weighted node estimation using geodesic distance. (a) original input mesh which is truncated; (b) the global template after surface expansion, Where corrupted regions are marked in red; (c) our weighted node estimation, where each black node is weighed by its distances between three closet neighbors (red nodes).

Recall that we have already aligned each input frame $\{\boldsymbol{P}^n\}_{n=1}^N$ to an optimal position $\{\boldsymbol{P}^{n\to0}\}_{n=1}^N$ when building the global template. We can thus directly convert $\boldsymbol{P}^{n\to0}$'s graph node's motion $\{\boldsymbol{A}_t, \boldsymbol{b}_t\}_{t=1}^m$ to each vertex v_i's rigid rotation $\boldsymbol{R}_i$ and translation $\boldsymbol{T}_i$ by further decompose Eq. 1.

Every vertex v_i^n in $\boldsymbol{P}^n$ can be viewed to go through a rigid motion $\boldsymbol{R}_i^{n\to0}, \boldsymbol{T}_i^{n\to0}$ to an optimized target $v_i^{n\to0}$ after deformation and we can then warp $v_i^{n\to0}$ back through $v_i^n = \boldsymbol{R}_i^{0\to n} v_i^{n\to0} + \boldsymbol{T}_i^{0\to n}$, where $\boldsymbol{R}_i^{0\to n} = inv(\boldsymbol{R}_i^{n\to0})$ and $\boldsymbol{T}_i^{0\to n} = -inv(\boldsymbol{R}_i^{n\to0})\boldsymbol{T}_i^{n\to0}$. To approximate the each g_t's motion parameter when warping it to $\boldsymbol{P}^n$, we locate the graph node g_t's closest point $v_{i'}^{n\to0}$ in $\boldsymbol{P}^{n\to0}$ and use $v_{i'}^{n\to0}$'s $\{\boldsymbol{R}_i^{0\to n}, \boldsymbol{T}_i^{0\to n}\}$ as the motion parameter. However, from Fig. 5, we observe that the deformed frame and the reconstructed global template can still exhibit relatively large misalignments. To better approximate the motion parameters, we present a patch based surface expansion approach based on [48] to better fit global template onto the deformed frame $\boldsymbol{P}^{n\to0}$:

$$\begin{aligned} \boldsymbol{E}_{expan} &= \sum_{\gamma \in T_k} ||v_\gamma^0 + d_\gamma n_\gamma^0 - c_\gamma||^2 \\ &+\lambda_{patch} \sum_{patch_\tau}^{\boldsymbol{K}} \sum_{\nu \in patch_\tau} \sum_{k \in \eta(\nu_\gamma)} |d_i - d_k|^2 \end{aligned} \tag{11}$$

Specifically, we trace a ray from each vertex v_γ^0 on the global template mesh along its normal direction n_γ^0 to the target deformed mesh $\boldsymbol{P}^{n\to0}$. Since $\boldsymbol{P}^{n\to0}$ may be trun-

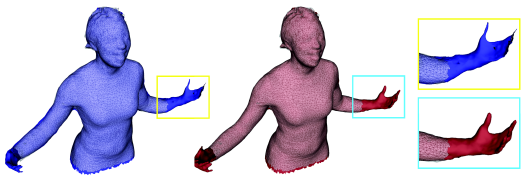

Figure 7. Our warping result (right) in comparison with direct non-rigid registration result (left).

cated due to occlusion, not all v_i^0 will be able to find intersections with the deformed mesh $\boldsymbol{P}^{n\to 0}$. We denote the ones we manage to find the corresponding points as $\{v_\gamma^0, c_\gamma, n_\gamma^0\}_{\gamma\in T_k}$. The first term of the $\boldsymbol{E}_{expan}$ aims to minimize the distance d_γ between vertex v_γ^0 and its intersection point c_γ. The second part regularization term ensures smoothness. We enforce it by setting d_i close to its K-nearest neighbors d_k in its patch $patch_\tau$. It will also propagate d to non-correspondence vertex. Fig. 5 shows the results before and after optimizing d_i.

A further comparison between the expanded global template $\boldsymbol{P}_{expan}^n$ with each deformed input frame $\boldsymbol{P}^n$ as shown in Fig. 5 illustrates that the expanded global template recovers the occluded parts. We again trace a ray from the global node g_t along its normal direction to determine whether g_t intersects with the target deformed frame $\boldsymbol{P}^{n\to 0}$. If yes, we adopt $\{\boldsymbol{R}_t^{0\to n}, \boldsymbol{T}_t^{0\to n}\}$ and further convert it to $\{\boldsymbol{A}_t^{0\to n}, \boldsymbol{b}_t^{0\to n}\}$ from its closest point in $\boldsymbol{P}^{n\to 0}$ as nodes motion parameter. If not, we approximate its motion parameter as weighted average of its K-nearest (K=3 in our experiment) known motion parameters where the weights correspond to geodesic distance, as shown in Fig. 6.

After we warp the expanded global template $\boldsymbol{P}_{expan}^n$ back to each input frame $\boldsymbol{P}^n$, we obtain the initial warped position of the global template $\{\boldsymbol{P}^{0,n}\}_{n=1}^N$. To ensure the temporal coherency between each frame, we further adjust the motion parameter $\{\{\boldsymbol{A}_t^{0,n}, \boldsymbol{b}_t^{0,n}\}_{t=1}^K\}_n^N$ globally from each expanded global template $\boldsymbol{P}_{expan}^n$ to each input frame $\boldsymbol{P}^n$ by introducing a temporal term $\tilde{\boldsymbol{E}}_{tempo}$ and a data term $\tilde{\boldsymbol{E}}_{data}$ to improve smoothness:

$$\begin{aligned}\tilde{\boldsymbol{E}}_{tempo} = \sum_{n=2}^{N-1} ||f(\boldsymbol{P}_{expan}^{n+1}, \boldsymbol{A}^{0,n+1}, \boldsymbol{b}^{0\to n+1}) \\ +f(\boldsymbol{P}_{expan}^{n-1}, \boldsymbol{A}^{0\to n-1}, \boldsymbol{b}^{0\to n-1}) \\ -2f(\boldsymbol{P}_{expan}^{n}, \boldsymbol{A}^{0\to n}, \boldsymbol{b}^{0\to n})||^2\end{aligned} \tag{12}$$

$$\tilde{\boldsymbol{E}}_{data} = \sum_{n=2}^{N-1} ||f(\boldsymbol{P}_{expan}^{n}, \boldsymbol{A}^{0\to n}, \boldsymbol{b}^{0\to n}) - \boldsymbol{P}_{expan}^{n}||^2 \tag{13}$$

where $\tilde{\boldsymbol{E}}_{data}$ forces adjust vertices to be close to the initial approximation.

Table 1. Data Information

Data	Avg vertices	Frames	Avg nodes
Dance	32K	160	353
Yoga	31K	361	340
Ballet	29K	200	350
Ballet2	27K	230	332
Boxing	23K	20	323
Singing	19K	23	342
Guitar	29K	21	356

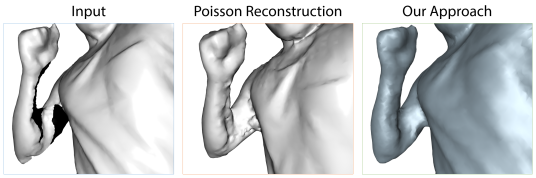

Figure 8. Our reconstruction in comparison with the poisson reconstruction.

Finally, we construct the energy function as $\tilde{\boldsymbol{E}}_{ajust} = \alpha_r\tilde{\boldsymbol{E}}_{rigid} + \alpha_s\tilde{\boldsymbol{E}}_{smooth} + \alpha_t\tilde{\boldsymbol{E}}_{tempo} + \alpha_d\tilde{\boldsymbol{E}}_{data}$. In our experiment, $\alpha_r = 100$, $\alpha_s = 30$, $\alpha_t = 1$ and $\alpha_d = 5$ and solve for the optimal results via Gauss-Newton technique.

Fig. 7 shows the warping back result use our weighted node approximation. And we also use non-rigid registration as the comparison which directly finds the warp back motion parameters from the global template to each frame. The non-rigid registration works well in overlapped regions. However, it causes severe critical bending effect in non-overlapped area. Our weighted node approximation can estimate the warping parameters accurately by combining the remaining nodes not in missing part weighted by their geodesic distance and we further use the temporal coherence constraint for further ensure the motion smoothness between adjacent frames.

4. Experiment

We perform experiments on captured real-life human motion sequences to validate the effectiveness of our algorithm.

To capture high fidelity motion, we build a multi-camera system for data acquisition. Our system equipped with 20 Point Grey cameras. Each camera has resolution 1280×720. We have captured seven motion sequences to test our algorithm. Five sequences (Yoga, Dance, Ballet, Ballet2 and Guitar) contain full human body, eg. and the other two sequences (Boxing and Singing) only contain the upper body. Detailed information of our test data is shown in Table. 1. All input sequences are suffered from heavy occlusion and truncation.

In pre-processing steps, we first recover a sparse point cloud using the Patch-Based Multi-View Stereo (PMVS) [13]. We then use Poisson surface reconstruc-

tion [19] with the surface trimmer to generate an initial surface mesh. Due to the limited camera field-of-view and/or occlusions, the initial surface mesh might be truncated or have large holes. By taking these incomplete initial surface meshes as input, our algorithm restores the complete surface shape for every motion frame and hence recover the motion sequence with high-fidelity. We performed reconstruction using the four-step algorithm described in section 3. All computations are performed off-line on a PC with CPU Intel Core i7-5820K and 32 GB memory. In average, the running time (per frame) of our algorithm is as follow: pairwise surface matching takes around 20 seconds, global template alignment and patch segmentation costs 65 and 30 seconds respectively(both only perform once for the entire sequence), and template warping takes 10 seconds.

We also compare our algorithm with the Poisson surface reconstruction for hole completion. The results are shown in Fig. 8. Due to large chunk of missing data, the poisson reconstruction cannot complete the hole (*e.g.*, the arm regions) correctly. By utilizing a global template that contains the full body geometry, our algorithm generates accurate and smooth reconstruction.

4.1. Global Template Reconstruction Results

In the first step, we register incomplete surfaces from the entire input sequence to generate a complete full body global template.To generate the global template, we first initiate a deformable graph for the body surface mesh of every input motion frame. We then find pairwise correspondences by imposing our temporal coherence constraint. Finally, we compute the affine transformations for every input surface mesh to align different poses and generate the global template. Fig.9 illustrates the global template generated by our algorithm for three different input sequences (*i.e.*, Boxing, Dance, and Ballet). The results demonstrate that our global templates are smooth and preserve some fine details at the same time. In the second column of Fig. 9, we show the composition of our global template by color-coding each frame. It shows that the poses in a motion sequences are complementary in geometry and by combining them, we are able to obtain the complete shape geometry.When sequence is too long, Eq.7 may hard to converge.So in our experiments, we set maximum frame number under 370.

4.2. Motion Reconstruction Results

Next, we segment the global template into connected rigid patches and build a deformable graph by taking the centroid of rigid patches as graph nodes. Finally, we warp the global template back to every input motion pose to restore the complete body surface meshes. Our reconstruction results are shown in Fig. 11, Fig. 12 and Fig. 14. Fig. 14 shows that our approach is capable of reconstruction full or partial body motion from heavily corrupted input data

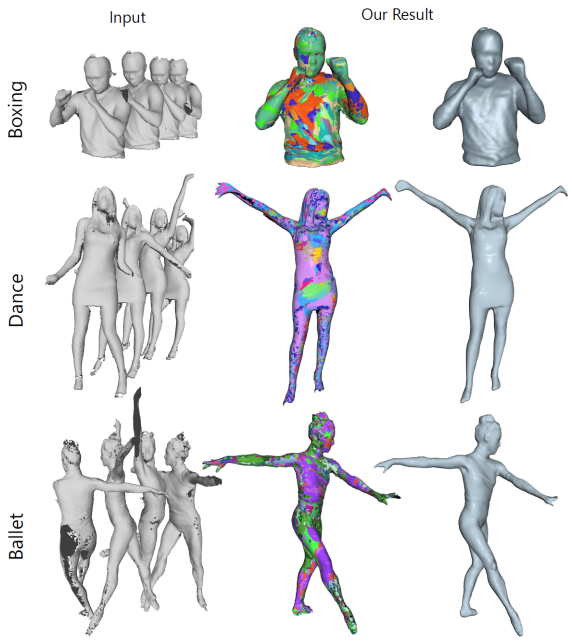

Figure 9. Global template reconstruction. The first column shows corrupted input meshes from several frames. The second column shows the alignment result of the entire sequence (every frame is coded by a different color). The third column shows the global template generate by our algorithm.

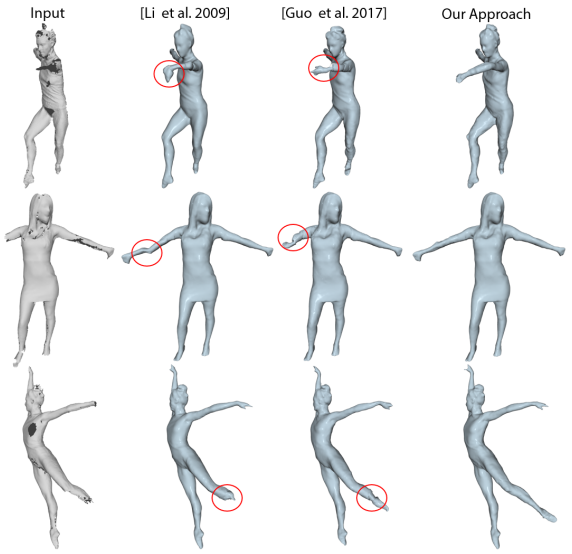

Figure 10. Compare our reconstruction result with [22] in various of scene.

in various scenarios. Fig. 11 demonstrate that our algorithm could also handle fast, drastic and rotating motion. We can see that our approach can successfully restore truncations and fill in large holes (*e.g.*, face in the Yoga scene, arms in the Dance scene, belly in the Ballet2 and legs in the Ballet scene). Further, because we warp the global tem-

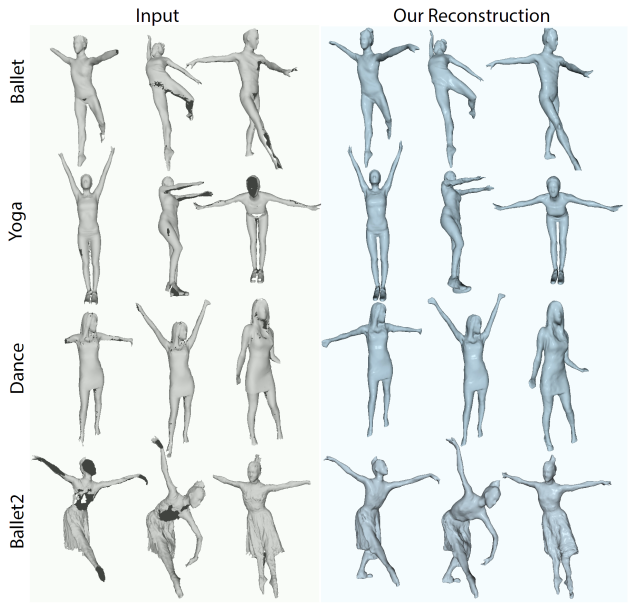

Figure 11. Our motion reconstruction results. Our approach is able to restore truncated areas and fill in large holes. Please refer to the supplemental material for more results.

Figure 12. The recovered geometry is consistent throughout the entire sequences. Corresponding vertices are color-coded.

plate through entire sequences, our reconstruction results are consistent in geometry throughout the entire sequences as shown in Fig. 12. Such consistency implies that our reconstruction could be beneficial for future applications such as consistent texture generation and data compression.

We perform experiments to compare our algorithm with the state-of-the-art method [22] and [15]. Fig. 10 shows the reconstruction comparison result. We can see that our algorithm provide more accurate reconstruction in presence of large holes/truncations caused by fast motions and occlusion. This is mainly because we consider the temporal coherence in surface alignment. We also perform quantitative evaluation to illustrate performance. In particular, we compute the mean Hausdorff distance between each pair of visible input and reconstructed surface and use it as the metric for quantitative evaluation. We compare the distance plot of our algorithm with [22], [15], and our method without patch segmentation on two input sequences(*i.e.* Dance and Ballet). As shown in Fig. 13, our reconstructions have lower error for most of the frames.

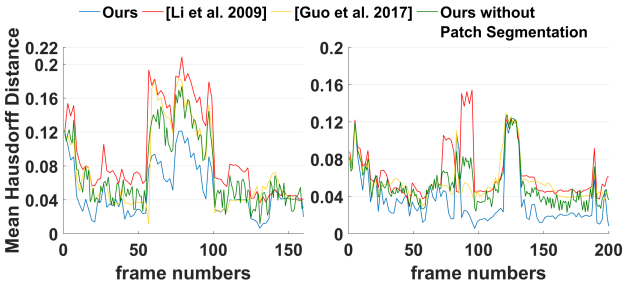

Figure 13. Quantitative evaluation in comparison with [22] on Ballet (a) and Dance(b).

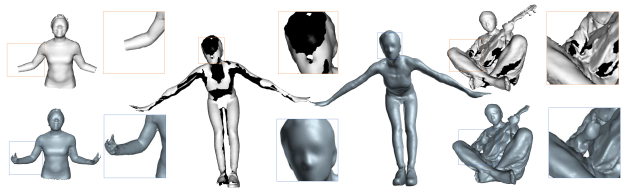

Figure 14. Warping back results of singing, yoga and guitar with close-up view (from left to right).

5. Conclusions and Discussions

We have presented a graph-based non-rigid shape registration framework that can simultaneously recover 3D human body geometry and estimate motion at high-fidelity.Our approach is especially effective in presence of large holes and truncated areas.We propose a temporal regularization term to get more accurate pairwise correspondence than the state-of-the-art method to generate a global body template by registering all poses in the acquired motion sequence. We also developed a new segmentation algorithm to divide the global template into locally rigid patches and built a deformable graph using the rigid patches.

Our approach has several limitations. First, our proposed global template generation algorithm cannot handle topology change such as cross arms and hands. One possible solution could be first automatically detecting topology change and then splitting the sequence into segments with the same topology and constructing separate global template. Second, subtle details(*e.g.*, fingers) are lost in our reconstruction since they are not effectively represented in our deformable graph. To achieve even higher-fidelity, we can recover the subtle motions in a separate pass and then add them back to our reconstruction.

Acknowledgement

This research is supported by National Science Foundation Grant IIS-1218156 and Army Research Office under the grant W911NF14-1-0338.

References

[1] F. Bogo, M. J. Black, M. Loper, and J. Romero. Detailed full-body reconstructions of moving people from monocular rgb-d sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2300–2308, 2015.

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

[3] C. Cagniart, E. Boyer, and S. Ilic. Free-form mesh tracking: a patch-based approach. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1339–1346. IEEE, 2010.

[4] C. Cao, Y. Weng, S. Lin, and K. Zhou. 3d shape regression for real-time facial animation. *ACM Transactions on Graphics (TOG)*, 32(4):41, 2013.

[5] K. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, volume 1, pages I–77. IEEE, 2003.

[6] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114–141, 2003.

[7] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (TOG)*, 34(4):69, 2015.

[8] P. Debevec. The light stages and their applications to photoreal digital actors. *SIGGRAPH Asia Technical Briefs*, 2, 2012.

[9] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *International journal of computer vision*, 96(1):1–27, 2012.

[10] M. Dou, H. Fuchs, and J.-M. Frahm. Scanning and tracking dynamic objects with commodity depth cameras. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pages 99–106. IEEE, 2013.

[11] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (TOG)*, 35(4):114, 2016.

[12] M. Dou, J. Taylor, H. Fuchs, A. Fitzgibbon, and S. Izadi. 3d scanning deformable objects with a single rgbd sensor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 493–501, 2015.

[13] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2010.

[14] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis, and I. Pitas. The i3dpost multi-view and 3d human action/interaction database. In *Visual Media Production, 2009. CVMP'09. Conference for*, pages 159–168. IEEE, 2009.

[15] K. Guo, F. Xu, Y. Wang, Y. Liu, and Q. Dai. Robust non-rigid motion tracking and surface reconstruction using l 0 regularization. *IEEE transactions on visualization and computer graphics*, 2017.

[16] C.-H. Huang, E. Boyer, and S. Ilic. Robust human body shape and pose tracking. In *3DV 2013*, pages 287–294. IEEE, 2013.

[17] P. Huang, M. Tejera, J. Collomosse, and A. Hilton. Hybrid skeletal-surface motion graphs for character animation from 4d performance capture. *ACM Transactions on Graphics (TOG)*, 34(2):17, 2015.

[18] Q.-X. Huang, B. Adams, M. Wicke, and L. J. Guibas. Non-rigid registration under isometric deformations. In *Computer Graphics Forum*, volume 27, pages 1449–1457. Wiley Online Library, 2008.

[19] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.

[20] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(3):29, 2013.

[21] J. P. Lewis, M. Cordner, and N. Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 165–172. ACM Press/Addison-Wesley Publishing Co., 2000.

[22] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. In *ACM Transactions on Graphics (TOG)*, volume 28, page 175. ACM, 2009.

[23] H. Li, R. W. Sumner, and M. Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer graphics forum*, volume 27, pages 1421–1430. Wiley Online Library, 2008.

[24] H. Li, E. Vouga, A. Gudym, L. Luo, J. T. Barron, and G. Gusev. 3d self-portraits. *ACM Transactions on Graphics (TOG)*, 32(6):187, 2013.

[25] H. Li, J. Yu, Y. Ye, and C. Bregler. Realtime facial animation with on-the-fly correctives. *ACM Transactions on Graphics (TOG)*, 32(4):42–1, 2013.

[26] P. Liepa. Filling holes in meshes. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 200–205. Eurographics Association, 2003.

[27] N. J. Mitra, S. Flöry, M. Ovsjanikov, N. Gelfand, L. J. Guibas, and H. Pottmann. Dynamic geometry registration. In *Symposium on geometry processing*, pages 173–182, 2007.

[28] A. Mustafa, H. Kim, J.-Y. Guillemaut, and A. Hilton. General dynamic scene reconstruction from multiple view video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 900–908, 2015.

[29] A. Mustafa, H. Kim, J.-Y. Guillemaut, and A. Hilton. Temporally coherent 4d reconstruction of complex dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2016.

[30] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 343–352, 2015.

[31] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011.

[32] S. Orts-Escolano, C. Rhemann, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, S. Khamis, M. Dou, et al. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 741–754. ACM, 2016.

[33] F. Prada, M. Kazhdan, M. Chuang, A. Collet, and H. Hoppe. Motion graphs for unstructured textured meshes. *ACM Transactions on Graphics (TOG)*, 35(4):108, 2016.

[34] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1106–1113, 2014.

[35] H. Rhodin, N. Robertini, D. Casas, C. Richardt, H.-P. Seidel, and C. Theobalt. General automatic human shape and motion capture using volumetric contour cues. In *European Conference on Computer Vision*, pages 509–526. Springer, 2016.

[36] N. Robertini, D. Casas, H. Rhodin, H.-P. Seidel, and C. Theobalt. Model-based outdoor performance capture. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 166–175. IEEE, 2016.

[37] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016.

[38] J. Starck and A. Hilton. Model-based multiple view reconstruction of people. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 915–922. IEEE, 2003.

[39] J. Starck and A. Hilton. Surface capture for performance-based animation. *IEEE Computer Graphics and Applications*, 27(3), 2007.

[40] R. W. Sumner, J. Schmid, and M. Pauly. Embedded deformation for shape manipulation. In *ACM Transactions on Graphics (TOG)*, volume 26, page 80. ACM, 2007.

[41] J. Süßmuth, M. Winter, and G. Greiner. Reconstructing animated meshes from time-varying point clouds. In *Computer Graphics Forum*, volume 27, pages 1469–1476. Wiley Online Library, 2008.

[42] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. In *ACM Transactions on Graphics (TOG)*, volume 27, page 97. ACM, 2008.

[43] D. Vlasic, P. Peers, I. Baran, P. Debevec, J. Popović, S. Rusinkiewicz, and W. Matusik. Dynamic shape capture using multi-view photometric stereo. *ACM Transactions on Graphics (TOG)*, 28(5):174, 2009.

[44] M. Wand, B. Adams, M. Ovsjanikov, A. Berner, M. Bokeloh, P. Jenke, L. Guibas, H.-P. Seidel, and A. Schilling. Efficient reconstruction of nonrigid shape and motion from real-time 3d scanner data. *ACM Transactions on Graphics (TOG)*, 28(2):15, 2009.

[45] M. Wand, P. Jenke, Q. Huang, M. Bokeloh, L. Guibas, and A. Schilling. Reconstruction of deforming geometry from time-varying point clouds. In *Symposium on Geometry processing*, pages 49–58, 2007.

[46] R. Wang, L. Wei, E. Vouga, Q. Huang, D. Ceylan, G. Medioni, and H. Li. Capturing dynamic textured surfaces of moving targets. In *European Conference on Computer Vision*, pages 271–288. Springer, 2016.

[47] J. Yang, A. Shehu, F. Hétroy-Wheeler, J.-S. Franco, and S. Wuhrer. Computing temporal alignments of human motion sequences in wide clothing using geodesic patches. In *3DV 2016*, 2016.

[48] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: High-resolution capture for modeling and animation. In *Data-Driven 3D Facial Animation*, pages 248–276. Springer, 2008.

[49] Q. Zhang, B. Fu, M. Ye, and R. Yang. Quality dynamic human body modeling using a single low-cost depth camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 676–683, 2014.

[50] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (TOG)*, 33(4):156, 2014.