UnstructuredFusion: Realtime 4D Geometry and Texture Reconstruction Using Commercial RGBD Cameras

Lan Xu, Zhuo Su, Lei Han, Tao Yu, Yebin Liu, and Lu Fang, *Senior Member, IEEE*

Abstract—A high-quality 4D geometry and texture reconstruction for human activities usually requires multiview perceptions via highly structured multi-camera setup, where both the specifically designed cameras and the tedious pre-calibration restrict the popularity of professional multi-camera systems for daily applications. In this paper, we propose *UnstructuredFusion*, a practicable realtime markerless human performance capture method using unstructured commercial RGBD cameras. Along with the flexible hardware setup using simply three unstructured RGBD cameras without any careful pre-calibration, the challenge 4D reconstruction through multiple asynchronous videos is solved by proposing three novel technique contributions, i.e., online multi-camera calibration, skeleton warping based non-rigid tracking, and temporal blending based atlas texturing. The overall insights behind lie in the solid global constraints of human body and human motion which are modeled by the skeleton and the skeleton warping, respectively. Extensive experiments such as allocating three cameras flexibly in a handheld way demonstrate that the proposed UnstructuredFusion achieves high-quality 4D geometry and texture reconstruction without tiresome pre-calibration, liberating the cumbersome hardware and software restrictions in conventional structured multi-camera system, while eliminating the inherent occlusion issues of the single camera setup.

Index Terms—4D reconstruction, performance capture, multi-camera, atlas texturing, skeleton warping, online calibration

1 Introduction

Over the last decade, Virtual Reality (VR) and Augmented Reality (AR) technologies have provided innovative solutions to present information in a way that was unthinkable just few years ago, extending its applications from entertainment to commerce, from gaming to education, and from military to art. In particular, the live 4D (3D spatial plus 1D time) content generation or reconstruction evolves as a cutting-edge yet bottleneck technique in VR/AR applications, restricted by the imperfect 3D sensing using existing RGBD sensors as well as the imperfect reconstruction especially when handling challenging dynamic scenes such as non-rigid human motions. How to reconstruct the 4D models of human activities for better VR/AR experience has recently attracted substantive attention of both the computer vision and computer graphics communities.

- *L. Xu and L. Han are with Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Beijing 100091, China, and also with the Department of ECE, Hong Kong University of Science and Technology, Hong Kong. E-mail: lxuan@connect.ust.hk, lhanaf@connect.ust.hka.*
- *Z. Su and L. Fang are with Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Beijing 100091, China. E-mail: su-z18@mails.tsinghua.edu.cn, fanglu@sz.tsinghua.edu.cn.*
- *Y. Liu is with Department of Automation, Tsinghua University, Beijing 100091, China. E-mail: liuyebin@mail.tsinghua.edu.cn.*
- *T. Yu is with Beihang University, Beijing 100091, China. E-mail: ytrock@buaa.edu.cn.*

Manuscript received 12 Oct. 2018; revised 4 Feb. 2019; accepted 21 Apr. 2019. Date of publication 7 May 2019; date of current version 2 Sept. 2020. (Corresponding author: Lu Fang.) Recommended for acceptance by M. Bennamoun and Y. Guo. Digital Object Identifier no. 10.1109/TPAMI.2019.2915229

Within the category of using RGBD sensors for 4D reconstruction, recent technological advances have led to a profound progress in terms of both effectiveness and efficiency, by leveraging the high-end GPUs. A number of reconstruction techniques using the most handy single depth camera setup [1], [2], [3], [4], [5] usually adopt a temporal fusion pipeline to solve the incomplete observation challenges, yet the reconstruction still suffers from the inherent self-occlusion issue due to lack of camera view resource. Although the state-of-the-art DoubleFusion [6] method takes advantage of human shape prior and achieves robust dynamic reconstruction on casual human body motions, it merely provides geometry reconstruction results, and is still incapable of generating compelling texture due to the limited input. To reconstruct high quality 4D geometry and texture simultaneously, one solution is to rely on collaborative multiple cameras systems like Fusion4D [7] and Motion2Fusion. [8] However, such systems are expensive and difficult to be deployed due to the requirement of non-commercial depth cameras, which are developed using tens of RGB/infra-red cameras integrated with structured lights. More importantly, all of the cameras and lightings are required to be synchronized and pre-calibrated in advance, leading to the high restriction of its wide applications for daily usage.

In this paper, we propose *UnstructuredFusion*, which allows realtime, high-quality, complete reconstruction of 4D textured models of human performance via only three commercial RGBD cameras. The three depth cameras cover the

overall human body in a relatively compensated yet flexible way, i.e., they are allocated in an unstructured manner without any pre-calibration or synchronization in advance. Compared with, [7], [8] our deployment is much more easy to setup with a low budget, yet such convenience presents the following challenges on the algorithm side for high quality 4D geometry and texture reconstruction: 1) unsynchronized data capture exists not only among RGBD cameras, but also inside each camera (internal asynchronous RGB video and depth video); 2) Since the camera array is sparse and unsynchronized, traditional pre-calibration will be very tedious and difficult. To refrain from it, the multi-camera system requires an on-line registration of the unstructured depths and videos based on the human performance.

As analyzed above, from the algorithm perspective, the given three RGB video streams and the three depth streams are fully unstructured, i.e., all the six streams are temporally and spatially misaligned. To solve this challenge, our key idea is to find a proper anchor for aligning the depth and color streams. For depth alignment, we leverage the skeleton and surface information of the human character, and propose a coarse-to-fine alignment scheme by first utilization of skeleton information, followed with a non-rigid optimization for warping the multiview depths. For the texture reconstruction, we propose to fuse a canonical texture atlas as an anchor for guiding the blending and updating of the temporal dynamic texture. The technique contributions of our UnstructuredFusion system are summarized as follows.

- We propose an unstructured multiple RGBD camera system using only three commercial RGBD cameras for realtime human performance capture.
- We propose a skeleton warping based non-rigid tracking scheme for unstructured multiview depth alignment. This scheme can be used in both the online calibration step and the tracking step.
- We propose a dynamic atlas texturing scheme for warping and updating dynamic texture on the fused geometry, leading to a high-quality appearance reconstruction in realtime.

Given the aforementioned distinctiveness, UnstructuredFusion serves as a good compromising settlement between over-demanding hardware setup and high-quality reconstruction, promoting potential applications of 4D reconstruction in immersive telepresence and supporting better immersive/interactive experiences.

2 Related Work

This section presents an overview of research works related with the proposed UnstructuredFusion system. We first provide an overview from the system setting aspect of human performance capture technologies, followed by an overview of representative dynamic reconstruction algorithms, and a brief summary of recent progress on reconstructing textured dynamic targets. Note that here we constrain the scope to full human body motion and geometry capture.

Performance Capture System For decades, marker based performance capture [9] has been a mature technique successfully used in many fields such as the movie industry, sport science and virtual reality. However, marker based performance capture suffers from the requirements of a controlled capture environment setup and marker suits with sensors like optical markers, [10] inertial devices, [11] pressure sensors, [12] or mounted cameras, [13] making them unable to capture motions of people wearing everyday apparel. To mitigate the above intrusive characteristics, markerless performance capture technologies have been recently investigated. For markerless system, earlier setups required multi-view video camera systems with controlled chromakey backgrounds [14], [15] to reconstruct the temporal varying geometry of the human body with a embedded skeletal model. Recent developments using hundreds of cameras and a controlled high-quality imaging environment [16], [17], [18] have even been able to produce extremely high quality skeletal motion, surface motion and even appearance reconstruction. Some recent systems [7], [8], [19] only utilize 8 static depth cameras to capture challenging motions in real-time with the sacrifice of temporal coherent results. However, most of these multi-view systems require considerable setup time for camera calibration, image segmentation or a pre-scanned 3D model of the actor with a manually embedding skeleton. Some recent works only rely on a light-weight single-view setup, [6], [20], [21], [22], [23], [24], [25] which enrich more practical application of performance capture. However, these single-stream systems are fragile to self-occlusion due to the lack of capture view resources.

Besides the above systems using static cameras, the research on hand-held camera based performance capture attacks the limitation of fixed capture volume. Hasler et al. [26] introduced an approach for motion capture outdoor from multiple handheld RGB cameras. Ye et al. [27] presented the approach using multiple handheld depth cameras to capture interacting motions, while Wu et al. [28] proposed using binocular cameras to capture human motion and at the same time derive the surface geometry detail of the actor. Xu et al. [29] proposed to use multiple drones to capture the motion in a wide space. All these methods fit a 3D scan of the actor to silhouettes or depths estimated in each of the moving cameras, which relies on a pre-scanned model or a pre-embeded skeleton and thus they have to be performed in an off-line manner. Recently, Wang et al. [30] presented how to reconstruct the 3D models of moving subjects using a new pairwise registration algorithm to register partial scans with little overlap. However, they still needed 15% to 20% overlap of different views, which can be difficult in the unstructured setting.

Comparably, the proposed UnstructuredFusion is the first to perform real-time performance capture in the unstructured and sparse multi-view setting, without the limitation of severe self-occlusion and any pre-scanning or pre-calibration efforts.

Dynamic Reconstruction Algorithm From the algorithm aspect, markerless motion reconstruction can be mainly classified into two categories: discriminative approaches [31], [32] and generative approaches. [33], [34] The former takes advantage of data driven machine learning strategies to convert the motion capture problem into a regression or pose classification problem. In contrast, generative approaches often rely on temporal information and solve a tracking problem. Many of these approaches parameterize the high dimensional human body by a low-dimensional skeleton embedded in the body model template. The motion reconstruction process is

then formulated as a frame-by-frame optimization to deform the skeletal pose [15] or the surface geometry [14], [35] or both of these together, [26], [27], [36] to be consistent with the observed input. The generative strategy is the preferred choice when accurate results are desired. However, they share limitations such as the requirement of a pre-scanned model template and a skeletal embedded and aligned initial pose, and they struggle to recover from tracking errors. Some research have tried to solve the above limitations. Non-rigid surface registration methods [37], [38], [39] deform the model vertices instead of the skeletal structure, which provides an appealing solution for dynamic scene modeling since it does not require the processing of skeleton embedding and surface skinning. Guo et al. [35] proposed a L0 based motion regularizer to regularize the large parameter space of non-rigid deformation, improving the performance capture robustness.

Only recently, free-form dynamic reconstruction methods with real-time performance have been proposed by combining the volumetric fusion [40], [41] and the nonrigid tracking techniques. DynamicFusion [1] utilized an approximate direct GPU solver to fuse the geometry information of a non-rigid scene in real-time without the need for any preprocessing. The following work [2] added the SIFT features to improve the accuracy of motion reconstruction. Guo et al. [3] proposed a realtime pipeline that utilized shading information to improve non-rigid registration, meanwhile the accurate temporal correspondences are used to estimate surface appearance. Slavcheva et al. [5], [42] proposed more constraints on the motion field to support topology changes, while some research [4], [43] proposed to combine the skeleton motion or IMU sensors for more robust reconstruction. Yu et al. [6] utilized the human shape prior and proposed a double-layer node-graph to reconstruct human motion efficiently. Fusion4D [7] and Motion2Fusion [8] extended the non-rigid tracking pipeline to a static rig with 8 depth cameras to capture dynamic scenes with challenging motions in realtime. However, neither of these methods was suitable for the unstructured multi-view setting, in which the misalignment between views will cause accumulated tracking error leading to uncanny reconstruction results.

Our reconstruction pipeline is the first to extend the real-time non-rigid fusion pipeline to the unstructured and sparse multi-view setting. We further demonstrate that, through our proposed optimization method to modeling the unstructured influence and the motion field together, the RGBD streams can be accurately aligned in the global view, enabling robust full body surface geometry and motion reconstruction.

Textured Model Reconstruction A high quality texturing scheme plays a critical role in realistic human performance capture. Many existing works utilized or optimized per-vertex or per-voxel color information for more realistic reconstruction of both static [44] and dynamic scenes. [3], [7] However, tying the color sampling to the geometry resolution gives rise to blur color mesh output. It became clear that using an atlas map would avoid this trade-off. The literature on atlas mapping is vast. Generally, a surface has to be first either cut through a seam (e.g, [45]) or segmented into charts (e.g, [46]). While atlas mapping has been widely utilized to static scene reconstruction, [47], [48], [49], [50] only limited work [8], [51] supports atlas texturing for dynamic scene reconstruction. What's worse, their atlas mapping schemes work in a frame-by-frame manner. Thus in their methods, for each independent frame, the camera views have to be sufficient enough to assure the color information covering any vertex, which is impractical in the sparse multi-view or even single-view setting.

Fig. 1. Illustration of the system setup and the real-time reconstruction results of our UnstructuredFusion. The red circles indicate the unstructured sparse multi-view RGBD cameras.

In our UnstrutureFusion system, we propose the first dynamic atlas mapping scheme for the sparse multi-view and even single view setting. We further demonstrate that through our atlas optimization scheme, a more realistic textured model with sharp and complete atlas can be obtained.

3 Overview of UnstructuredFusion

The proposed UnstructuredFusion attempts to bring aspects inherent in realtime human performance capture system to unstructured and sparse multi-view setting as illustrated in Fig. 1. In so doing we need to design a new pipeline which is not only robust to the unstructured misalignment among different views, but also makes full use of the multiview depth and color information for realistic reconstruction, whilst still maintaining real-time rates.

Fig. 2 illustrates the high-level components of our system pipeline, which achieves considerably more vivid results than previous real-time performance capture systems under the sparse multiview setting. Our system takes RGBD images as input from sparse and unstructured multiple views, and generates textured meshes as output. Specifically, three Kinect v2 sensors are utilized to generate three uncalibrated and unsynchronized RGBD streams with 512×424 resolution in 30 fps. A temporal fusion strategy to accumulate the 3D reconstruction is adopted and Truncated Signed Distance Function (TSDF) [40] volume is utilized as the underlying data structure. Similar to, [6], [43] both the embedded deformation (ED) model [38] and the linear human body model (SMPL) [52] are combined for non-rigid motion representation (see Section 4.1). Furthermore, a brief introduction of each main component of our pipeline is provided as follows.

Initialization. For the initialization in the first frame, to align the uncalibrated RGBD streams and automatically embed the SMPL human prior model simultaneously, we propose a novel online multiview calibration scheme, which jointly optimize the camera poses, initial pose and shape parameters of SMPL model (see Section 4.2). Our online calibration scheme is robust to align the multiple RGBD streams, even considering the lack of sufficient overlap regions between nearby views in the sparse multiview setting. It is worth noting that during the online calibration, none of the external

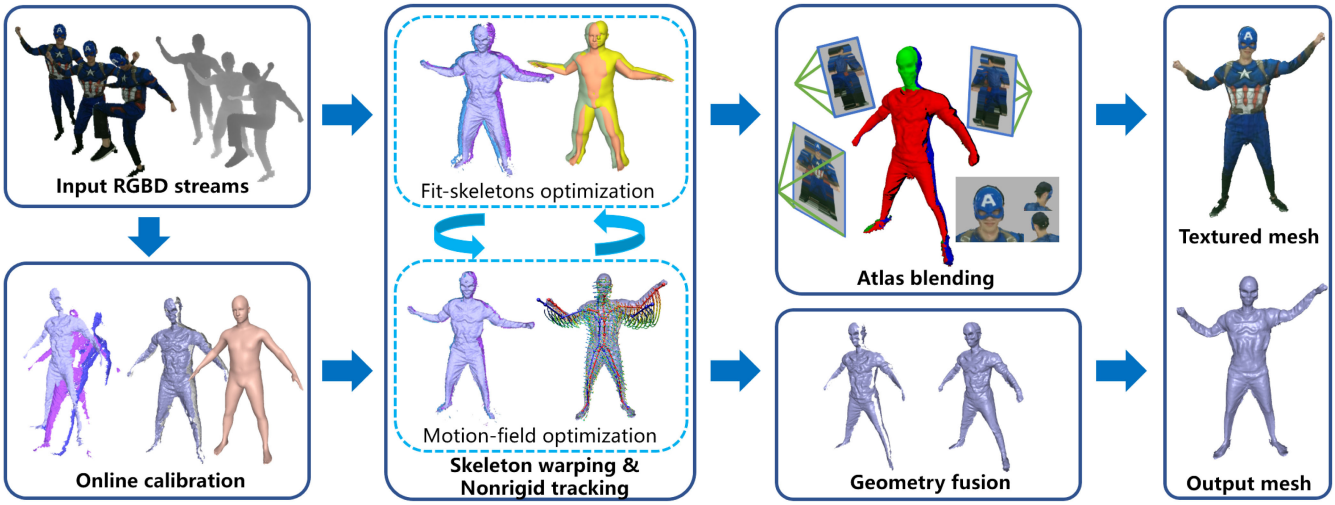

Fig. 2. The system pipeline of UnstructuredFusion. We first initialize our system at the first frame by performing the online multi-camera calibration (Section 4.2). Then for each frame, we sequentially perform the next 3 steps: skeleton warping based non-rigid tracking (Section 4.3), geometric fusion and atlas texturing based on temporal blending (Section 4.4). Finally, live textured meshes with geometry details are obtained.

devices like checkerboard, IMU, etc., or process like pre-scanning and manually rigging is needed. The performer only need to start with a rough A-pose as the initialization.

Non-Rigid Tracking. The core of our pipeline is to solve the non-rigid alignment parameters from the canonical frame to the camera views of current RGBD inputs. We propose a novel skeleton warping based non-rigid tracking scheme, which utilizes the unique and solid human shape prior to separate the non-rigid tracking problem into two sub-problems, and employs an iterative flip-flop strategy to optimize the fit-skeletons and the hybrid motion-fields (See Section 4.3). Our skeleton warping scheme is efficient and robust for modeling the non-rigid motion and attacking the misalignment problem in the unstructured sparse multiview setting.

Geometry Fusion. After estimating the non-rigid motions and aligning the unstructured RGBD input streams, we fuse the depth observations into a global canonical TSDF volume, which is maintained to provide temporal coherent reconstruction results. When updating the canonical volume, similar to previous works, [7], [8] we discard the data of the voxels which are warped into invalid area in current inputs, and also explicitly detect collided voxels to avoid erroneously fused geometry. The body shape and pose are also optimized in the fused signed distance field using the efficient volumetric shape-pose optimization in previous work [6] to obtain better canonical body fitting and skeleton embedding. Finally, marching cubes is used to extract a triangle mesh.

Atlas Texturing. To provide more vivid performance capture results, we propose a novel atlas texturing scheme in the sparse multiview setting. To stay within realtime computational budget, we construct a high-efficient projective atlas map with virtual camera views bound to the canonical volume. The projective atlas is utilized to texture the reconstructed mesh extracted from the canonical volume. A novel grid warping based dynamic atlas blending scheme is provided to blend a complete and sharp atlas map (See Section 4.4). Our method provides the first realtime atlas texturing solution for dynamic reconstruction in sparse multiview setting, which works for single-view input as well.

4 Technique Details of UnstructuredFusion

On the contrary to the conventional structured multi-camera system which requires fixed allocation and careful calibration of cameras offline, our UnstructuredFusion allows cameras to be allocated in an unstructured and even hand-held manner. In other words, we propose a novel online calibration scheme to liberate the tedious overhead of camera array system, followed by the skeleton warping based non-rigid tracking algorithm. Meanwhile, aiming for vivid human performance capture, a temporal blending based online atlas texturing scheme is proposed to generate a high-quality appearance. In the following subsections, we first introduce the motion representation in our method, followed by elaborating the online calibration, non-rigid tracking, and online atlas texturing, respectively.

4.1 Human Motion Representation

Since our method focuses on human performance capture, we adopt the efficient and robust double-layer surface representation for motion representation, [6] which combines the embedded deformation model and the linear human body model SMPL. In this subsection, we overview these two motion parameterization schemes briefly and define the mathematical notation in our method.

Embedded Deformation Model. The ED model in our method is represented by a non-rigid motion field $G = \{\mathbf{dq}_j, \mathbf{x}_j\}$, consisting of the dual quaternions $\{\mathbf{dq}_j\}$ and the corresponding sparse ED nodes $\{\mathbf{x}_j\}$. Let $SE3(\mathbf{dq}_j)$ denote the rigid transformation associated with the jth dual quaternion. Neighboring ED nodes are connected together to form an ED graph and then for any 3D vertex $\mathbf{v}_c$ in the canonical volume, the ED warping operation is formulated as follows:

$$\tilde{\mathbf{v}}_c = ED(\mathbf{v}_c; G) = SE3\left(\sum_{i \in \mathcal{N}(v_c)} w(\mathbf{x}_i, \mathbf{v}_c)\mathbf{dq}_i\right)\mathbf{v}_c, \tag{1}$$

where $\mathcal{N}(v_c)$ represents a set of node neighbors of $\mathbf{v}_c$, and $w(\mathbf{x}_i, \mathbf{v}_c) = \exp(-\|\mathbf{v}_c - \mathbf{x}_i\|_2^2/(2r_k^2))$ is the influence weight of the ith node $\mathbf{x}_i$ to $\mathbf{v}_c$. The influence radius r_k is set as 0.075m

for all the ED nodes. Similarly, $\tilde{\mathbf{n}}_{v_c} = ED(\mathbf{n}_{v_c}; G)$ denotes the warped normal of $\mathbf{v}_c$ using the ED motion field G.

SMPL Inner Body Model. SMPL model [52] associates with $N = 6890$ vertices and a skeleton with $K = 24$ joints. Before posing, the body model $\bar{\mathbf{T}}$ deforms into the morphed model $T(\boldsymbol{\beta}, \boldsymbol{\theta})$ with the shape parameters $\boldsymbol{\beta}$ and pose parameters $\boldsymbol{\theta}$ to accommodate for different identities and pose-dependent deformations. Mathematically, the body shape $T(\boldsymbol{\beta}, \boldsymbol{\theta})$ is morphed according to:

$$T(\boldsymbol{\beta}, \boldsymbol{\theta}) = \bar{\mathbf{T}} + B_s(\boldsymbol{\beta}) + B_p(\boldsymbol{\theta}), \tag{2}$$

where $B_s(\boldsymbol{\beta})$ and $B_p(\boldsymbol{\theta})$ represent the shape blendshapes and pose blendshapes respectively. Let $T(\bar{\mathbf{v}}; \boldsymbol{\beta}, \boldsymbol{\theta})$ denotes the morphed 3D position for any vertex $\bar{\mathbf{v}} \in \bar{\mathbf{T}}$. The posed function of SMPL model is further formulated as $W(T(\boldsymbol{\beta}, \boldsymbol{\theta}), J(\boldsymbol{\beta}), \boldsymbol{\theta}, \mathcal{W})$, which is a general blend skinning function in terms of the morphed body $T(\boldsymbol{\beta}, \boldsymbol{\theta})$, pose parameters $\boldsymbol{\theta}$, joint locations $J(\boldsymbol{\beta})$ and the skinning weights $\mathcal{W}$. Then for any 3D vertex $\mathbf{v}_c$, the Linear Blend Skinning (LBS) operation with the SMPL skeleton motions is formulated as follows:

$$\begin{aligned} \hat{\mathbf{v}}_c =& \mathbf{G}(\mathbf{v}_c, \boldsymbol{\theta})\mathbf{v}_c, \mathbf{G}(\mathbf{v}_c, \boldsymbol{\theta}) = \sum_{i \in \mathcal{B}} w_{i,v_c} \mathbf{G}_i, \\ \mathbf{G}_i =& \prod_{k \in \mathcal{K}_i} \exp(\theta_k \hat{\xi}_k), \end{aligned} \tag{3}$$

where $\mathbf{G}(\mathbf{v}_c, \boldsymbol{\theta})$ is the posed rigid transformation of $\mathbf{v}_c$, $\mathcal{B}$ is index set of bones, $\mathbf{G}_i$ is the cascaded rigid transformation of ith bone, $\mathcal{K}_i$ are the parent indices of ith bone in the backward kinematic chain, $\exp(\theta_k \hat{\xi}_k)$ is the exponential map of the twist associated with kth bone, and w_{i,v_c} is the skinning weight associated with ith bone and point $\mathbf{v}_c$. For w_{i,v_c} setting, if $\mathbf{v}_c$ is on SMPL model, w_{i,v_c} is pre-defined in $\mathcal{W}$. If $\mathbf{v}_c$ is on the fused surface, w_{i,v_c} is given by the weighted average of the skinning weights of its knn-nodes. If $\mathbf{v}_c$ is on the depth input, we find its knn-nodes by warping the ED-nodes into the camera view first.

4.2 Online Multi-Camera Calibration

Recall that under the unstructured multi-camera setting of our system, all the RGBD cameras are uncalibrated and unsynchronized. Our online calibration scheme is illustrated in Fig. 3. For the first frame, the performer needs to start with a rough A-pose as the initialization. Then we jointly optimize the initial camera poses $\boldsymbol{T} = \{\boldsymbol{T}_i\}, i = 1, 2, 3$, initial skeleton pose $\boldsymbol{\theta}_0$ and the shape parameters $\boldsymbol{\beta}_0$ of SMPL model as follows:

$$\begin{aligned} E_{\text{init}}(\boldsymbol{T}, \boldsymbol{\beta}_0, \boldsymbol{\theta}_0) = & \lambda_{\text{vdata}} E_{\text{vdata}} + \lambda_{\text{sdata}} E_{\text{sdata}} \\ & + \lambda_{\text{pdata}} E_{\text{pdata}} + \lambda_{\text{prior}} E_{\text{prior}}. \end{aligned} \tag{4}$$

Here the volumetric data term E_{vdata} measures misalignment error between the SMPL model and the reconstructed mesh in the reference volume [6]:

$$E_{\text{vdata}}(\boldsymbol{\beta}_0, \boldsymbol{\theta}_0) = \sum_{\bar{\mathbf{v}} \in \bar{\mathbf{T}}} \psi(\mathbf{D}(W(T(\bar{\mathbf{v}}; \boldsymbol{\beta}_0, \boldsymbol{\theta}_0); J(\boldsymbol{\beta}_0), \boldsymbol{\theta}_0)), \tag{5}$$

where $\mathbf{D}(\cdot)$ takes a point in the canonical volume and returns the bilinear interpolated TSDF, and $\psi(\cdot)$ is the robust Geman-McClure penalty function.

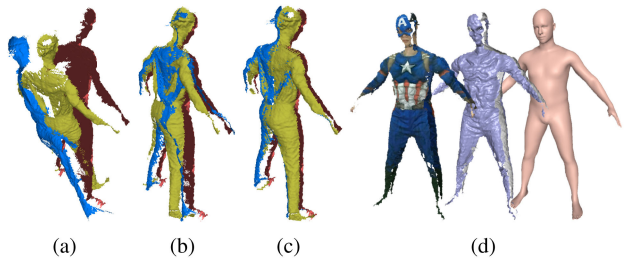

Fig. 3. Illustration of the online calibration. (a,b,c) are the depth inputs with the initial camera poses, the camera poses after solving Eqn. (7), and the one after solving Eqn. (4). (d) The final output of online calibration. From left to right: the aligned color frames, the aligned depth frames and the embedded SMPL model.

The dense projective data term E_{pdata} forces the warped vertices on the SMPL model to move to the corresponding depth point of the input depth data based on a point-to-plane distance metric, which is formulated as:

$$E_{\text{pdata}}(\boldsymbol{T}) = \sum_{i=1}^{3} \sum_{(\bar{\mathbf{v}}, \mathbf{u}_i) \in \mathcal{C}_i} \psi(\mathbf{n}_{\bar{\mathbf{v}}}^T (\boldsymbol{T}_i W(T(\bar{\mathbf{v}}; \boldsymbol{\beta}_0, \boldsymbol{\theta}_0)) - \mathbf{u}_i)), \tag{6}$$

where $(\bar{\mathbf{v}}, \mathbf{u}_i)$ is a correspondence pair found via a projective look-up method in the ith camera view; $\mathbf{u}_i$ is a sampled point on the depth map while $\bar{\mathbf{v}}$ is a vertex on the SMPL model. Besides the dense alignment, we detect the global human skeleton using the Kinect SDK. Let $\mathbf{J}_{p,i}$ denotes the p-th 3D joint position of detected skeleton in the ith camera view which indicates additional global constraints for fitting the SMPL model to current depth maps, formulated as the following sparse feature term:

$$E_{\text{sdata}}(\boldsymbol{T}) = \sum_{1 \le i < j \le 3} \sum_{p=1}^{N_p} \tau(p; i, j) \| \boldsymbol{T}_i^{-1} \mathbf{J}_{p,i} - \boldsymbol{T}_j^{-1} \mathbf{J}_{p,j} \|_2^2, \tag{7}$$

where N_p is the amount of 3D skeleton joints and $\tau(p; i, j)$ is the indicator function which equals to 1 only if the pth joint is observable in both the ith and jth camera views.

Similar to, [6], [53] we utilize a pose prior to penalize the unnatural poses, which is defined as

$$E_{\text{prior}}(\boldsymbol{\theta}_0) = -log\left(\sum_j w_j N(\boldsymbol{\theta}_0; \mu_j, \delta_j) \right). \tag{8}$$

This term is formulated as a Gaussian Mixture Model (GMM), where w_j, μ_j and δ_j are the mixture weight, the mean and the variance of jth Gaussian model, respectively.

The resulting energy in Eqn. (4) is solved in an Iterative Closest Point (ICP) framework with a custom designed highly efficient Preconditioned Conjugate Gradient (PCG) solver on GPU. [3], [7] Similar to, [6] we ignore the pose blend shape in E_{vdata} (Eqn. (5)) to make the convergence faster. To further pick a good initial value during the first ICP iteration, we solve Eqn. (5) to find the initial guess of both the skeleton pose $\boldsymbol{\theta}_0$ and shape parameters $\boldsymbol{\beta}_0$ following. [6] The initial guess of camera poses $\boldsymbol{T}$ is obtained by solving Eqn. (7). At last, after the optimization we embed the body shape and pose into the canonical frame, and initialize the motion field and skeleton motions.

4.3 Skeleton Warping Based Non-Rigid Tracking

It is worth noting that in our commodity unstructured setup, due to the asynchronous nature of the consumer-oriented hardware, the RGBD input streams suffer from error due to lack of synchronization, let alone the extrinsic camera localization error and the distortion of raw depth maps. These issues cause misalignment of raw RGBD inputs, which further increases the difficulty of solving non-rigid tracking, leading to uncanny reconstruction results. To attack these misalignment problems when designing non-rigid tracking strategy, we propose to align all the asynchronous raw RGBD streams to a global 'reference' through the skeleton warping, based on the insight that the unique prior of human shape can serve as a quite reliable 'reference'. In this way, both skeletal motions and non-rigid ED deformations are jointly investigated via jointly optimizing current SMPL skeleton pose $\boldsymbol{\theta}$ and the ED motion field G in a frame-by-frame manner, given that the ED node-graph is bound tightly to the SMPL model.

Mathematically, we introduce the "*fit-skeleton*" $\hat{\boldsymbol{\theta}}_i$, denoting the SMPL skeleton pose fit to current live RGBD image input $\mathcal{D}_i$ in the ith camera view, which corresponds to sub-frame level asynchronous capture time under the unstructured setting. Let $\boldsymbol{\theta}$ denotes the optimized skeleton for current frame without the influence of unstructure error. Then for each pixel $\mathbf{u}_i \in \mathcal{D}_i$, we can warp it from the fit-skeleton pose $\hat{\boldsymbol{\theta}}_i$ to the global pose $\boldsymbol{\theta}$, using the "skeleton warping" operation formulated as:

$$\mathbf{u}'_i = \mathbf{G}(\mathbf{u}_i, \boldsymbol{\theta})\mathbf{G}(\mathbf{u}_i, \hat{\boldsymbol{\theta}}_i)^{-1}\mathbf{u}_i, \tag{9}$$

where $\mathbf{G}(\cdot)$ is the Linear Blend Skinning rigid transformation of the SMPL skeleton. Note that the skinning weight of $\mathbf{u}_i$ is given by the weighted average of the skinning weights of its knn-nodes. To align all the raw RGBD streams using skeleton warping, we combine a dense data term and the pose prior term [53] to optimize the fit-skeleton $\hat{\boldsymbol{\theta}}_i$, formulated as:

$$E_{\text{skwarp}}(\hat{\boldsymbol{\theta}}_i) = \lambda_{\text{fit}} E_{\text{fit}}(\hat{\boldsymbol{\theta}}_i) + \lambda_{\text{prior}} E_{\text{prior}}(\hat{\boldsymbol{\theta}}_i). \tag{10}$$

Here the definition of pose prior term E_{prior} resembles Eqn. (8) in terms of $\hat{\boldsymbol{\theta}}_i$ instead of $\boldsymbol{\theta}_0$. The dense data term measures the skeleton fitting between the reconstructed double layer surface and the depth map:

$$\begin{aligned} E_{\text{fit}}(\hat{\boldsymbol{\theta}}_i) = \sum_{(v_c, u_i)\in\mathcal{P}_i} & \tau_1(\mathbf{v}_c)\psi(\hat{\mathbf{n}}_{v_c}^{\mathbf{T}}(\mathbf{G}(\mathbf{v}_c, \hat{\boldsymbol{\theta}}_i)\mathbf{v}_c - \mathbf{u}_i)) \\ & + \tau_2(\mathbf{v}_c)\psi(\tilde{\mathbf{n}}_{v_c}^{\mathbf{T}}(\mathbf{G}(\mathbf{v}_c, \hat{\boldsymbol{\theta}}_i)\mathbf{G}(\mathbf{v}_c, \boldsymbol{\theta})^{-1}\tilde{\mathbf{v}}_c - \mathbf{u}_i)), \end{aligned} \tag{11}$$

where $\mathcal{P}_i$ is the correspondence set of the ith camera view; u_i is a sampled point on the depth map and its closest point v_c can be on either the body shape or the fused surface. $\tau_1(\cdot)$ and $\tau_2(\cdot)$ are correspondence indicator functions: $\tau_1(\mathbf{v}_c)$ equals to 1 only if $\mathbf{v}_c$ is on the body shape; $\tau_2(\mathbf{v}_c)$ equals to 1 only if $\mathbf{v}_c$ is on the fused surface. We follow the same correspondences searching scheme on the double layer surface as. [6] Please refer to [6] for more detail.

After solving all the fit-skeletons $\hat{\boldsymbol{\theta}}_i$ of current frame, we further jointly optimize the global skeleton pose $\boldsymbol{\theta}$ and current ED non-rigid motion field G as follows:

$$\begin{aligned} E_{\text{mot}}(G, \boldsymbol{\theta}) = & \lambda_{\text{data}} E_{\text{data}} + \lambda_{\text{bind}} E_{\text{bind}} + \lambda_{\text{reg}} E_{\text{reg}} \\ & + \lambda_{\text{prior}} E_{\text{prior}} + \lambda_{\text{skele}} E_{\text{skele}}. \end{aligned} \tag{12}$$

Again, the pose prior term E_{prior} resembles Eqn. (8). Following, [6] the binding term E_{bind} constrains both motions to be consistent while the geometry regularity term E_{reg} produces locally as-rigid-as-possible (ARAP) motions to prevent overfitting to depth inputs. These two terms are detailed in. [3], [6]

The dense projective data term E_{data} is formulated as the sum of point-to-plane distances in our multi-view setting:

$$E_{\text{data}}(G, \boldsymbol{\theta}) = \sum_{i=1}^{3} \sum_{(v_c, u_i)\in\mathcal{P}_i} (\tilde{\mathbf{n}}_{v_c}^{\mathbf{T}}(\tilde{\mathbf{v}}_c - \mathbf{u}'_i))^2, \tag{13}$$

where $\mathbf{u}_i$ is a sampled point in the depth map, and $\mathbf{v}_c$ denotes its closest point on the fused surface. $\mathcal{P}_i$ is the set of correspondences found via a projective local search [1], [29] in the ith camera view. $\mathbf{u}'_i$ denotes the aligned depth pixel after skeleton warping using Eqn. (9). To further bridge the fit-skeletons $\hat{\boldsymbol{\theta}}_i$ and current global skeleton $\boldsymbol{\theta}$, we introduce the following skeleton term:

$$E_{\text{skele}}(\boldsymbol{\theta}) = \sum_{i=1}^{3} \sum_{u_i\in\mathcal{P}_i} \|W_{\mathbf{u}_i}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_i)\|_2^2, \tag{14}$$

where $W_{\mathbf{u}_i}$ is the LBS skinning weight vector of the depth point $\mathbf{u}_i$. Note that we first warp the ED-nodes into the ith camera view to find the knn-nodes of $\mathbf{u}_i$; then $W_{\mathbf{u}_i}$ is given by the weighted average of the skinning weights of its knn-nodes.

We solve the optimization problem in Eqn. (12) under the ICP framework. The non-linear least squares problem is solved using Levenberg-Marquardt (LM) method. During each iteration, twist representation is utilized for both the bone and node transformations and the transformations are approximated using one-order Taylor expansion around the latest values. The resulting linear system is solved using the same custom designed highly efficient Preconditioned Conjugate Gradient solver on GPU. [3]

4.4 Temporal Blending Based Atlas Texturing

A high quality texturing plays a critical role in reconstructing the vivid appearance for human motion capture. Prior work [3], [7] adopt per-vertex colors in the final output, but simply associating the color sampling with the geometry resolution requires unfavorable trade-offs that produce blurred results. Recent work [8], [17], [51] propose atlas texturing to liberate the constraint of the geometry resolution, leading to sharper results. However, these methods conduct atlas texturing frame by frame independently, implying that the texture atlas changes for each live frame. Thus the camera views have to be sufficient enough to assure the color information covering any vertex for any possible motions of model in the live frame (i.e., 8 and 106 camera views for previous work [8], [17] respectively). In view of this, we propose a novel temporal blending based atlas texturing scheme, allowing for generating a sharp and realistic textured model in the commodity sparse multi-view set-up and even for single-view input. More specifically, to stay within real-time computational budget, we choose to construct a high-efficient projective atlas

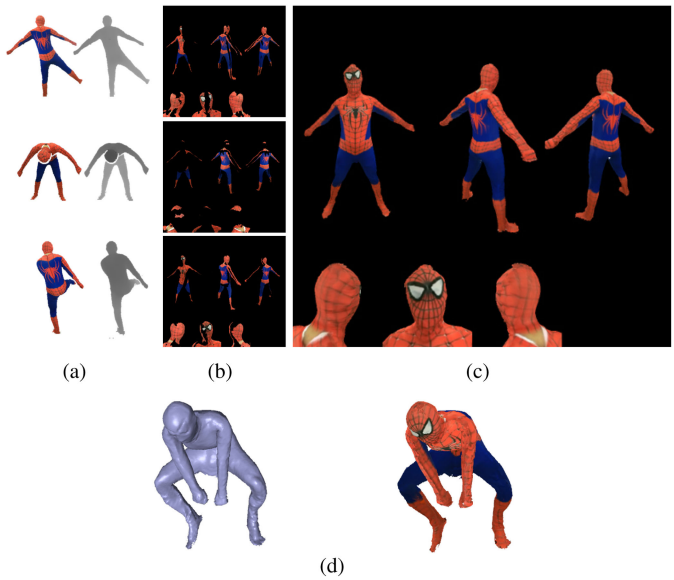

Fig. 4. Illustration of our projective atlas scheme. (a) The input RGBD image examples. (b) The corresponding partial atlas $\{\boldsymbol{a}_i\}$. (c) The global blended atlas $\{\boldsymbol{A}_i\}$. (d) The textured mesh output in the live frame.

map. To utilize the intra-frame information and blend a global atlas, all the projective poses are bound to the global canonical model. In particular, to capture more details in the head region, we build two kinds of virtual projective poses in the first frame: $\boldsymbol{T}_{H,i}$ and $\boldsymbol{T}_{B,i}$ for texturing the head and body regions respectively, where $i \in 1, 2, 3$ denotes the index of virtual projective views since three views can cover the static canonical model in the A-pose setting overall.

As shown in Fig. 4, for every input color frame, we first project all the visible canonical vertices with $\boldsymbol{T}_{H,i}$ and $\boldsymbol{T}_{B,i}$, followed by writing the color at the projected position into the corresponding texture coordinates to build the partial projective textures $\boldsymbol{a}_i$ of the ith virtual view for head and body region respectively. The generation of such partial texture can be easily achieved using the OpenGL rasterization pipeline. We then temporally blend all the partial texture images $\{\boldsymbol{a}_i\}$ into a complete and global atlas $\{\boldsymbol{A}_i\}$ in a frame-by-frame manner, as illustrated in Fig. 4b and 4c respectively. For texturing the un-covered and occluded areas of canonical model, we also fuse all the color frames into a global color volume $\mathbf{C}$, similar as the TSDF volume. Such hybrid texturing scheme can easily be achieved in OpenGL pipeline. The faces in the canonical model with valid projective UV coordinates are textured using $\{\boldsymbol{A}_i\}$, while those without valid UV coordinates extract per-vertex color values directly from the color volume $\mathbf{C}$.

Dynamic Projected Texture Blending: In order to obtain a sharp and complete atlas from the partial projective atlas, we solve the texture blending in a frame-by-frame and per-texel manner. Similar to TSDF fusion in the volume space, we perform temporal atlas blending as follows. For each texel p, $\boldsymbol{a}_i(p)$ and $\boldsymbol{A}_i(p)$ denote the corresponding color values from the partial and blended atlas in the ith virtual camera view respectively; $\mathbf{W}_i(p)$ denotes its accumulated blending weight; $w_i(p) = cos(\theta)$ is the view-dependent weight of current frame, where θ is the angle between the projected normal into the camera view, and the view direction of the camera. Finally, the projected atlas is dynamically blended with the weight truncation as follows:

$$\begin{aligned} \boldsymbol{A}_i(p) &\leftarrow \frac{\boldsymbol{A}_i(p)\mathbf{W}_i(p) + \boldsymbol{a}_i(p)w_i(p)}{\mathbf{W}_i(p) + w_i(p)}, \\ \mathbf{W}_i(p) &\leftarrow min(\mathbf{W}_i(p) + w_i(p), w_{max}). \end{aligned} \tag{15}$$

The above maximum blending weight w_{max} enables the moving average texture blending scheme to support dynamic texturing. It is set to be 4 for head, and 8 for body region to obtain more dynamic texture detail in the face region.

Examining the blurry effect in atlas texturing, the motion blur caused by fast motion can be simply eliminated by discarding bad color frames through selecting views using the blurriness measure by Crete et al. [54] For the atlas blur caused by the non-rigid misalignment between the live mesh and the color images, a 2D as-similar-as-possible (ASAP) grid-based warping scheme, denoted as grid-based warping, is adopted between the partial atlas $\boldsymbol{a}_i(p)$ and the temporally blended atlas $\boldsymbol{A}_i(p)$ during the atlas blending using Eqn. (15). Let $\{p_A, p_a\}$ denote all the feature pairs between $\boldsymbol{A}_i$ and $\boldsymbol{a}_i$, based on *ORB* descriptors and *GMS* matching method. [55] Regular grid cells are then sampled in $\boldsymbol{A}_i$ and each cell is split into two triangles. Similar as, [56], [57], [58] the 2D warping from $\boldsymbol{A}_i$ to $\boldsymbol{a}_i$ is modeled as the positions of the deformed grid vertices, denoted as $\hat{V}$. Mathematically, we optimize the 2D deformation using the following energy function:

$$E(\hat{V}) = E_{\mathrm{d}}(\hat{V}) + \alpha E_{\mathrm{s}}(\hat{V}). \tag{16}$$

The data term $E_{\mathrm{d}}(\hat{V})$ that sums the distances of all the feature pairs in atlas domain after warping $\hat{V}$ is formulated as:

$$E_{\mathrm{d}}(\hat{V}) = \sum_{p_A} \|w_{p_A}\hat{V}_{p_A} - p_a\|_2^2, \tag{17}$$

where $\hat{V}_{p_A}$ are the warped grid vertices enclosed p_A, and w_{p_A} is the corresponding weight of bilinear interpolation.

The ASAP regular term $E_{\mathrm{s}}(\hat{V})$ is formulated as:

$$E_{\mathrm{s}}(\hat{V}) = \sum_{\hat{v}} \tau(\hat{v})\|\hat{v} - \hat{v}_1 - sR_{90}(\hat{v}_0 - \hat{v}_1)\|_2^2, R_{90} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{18}$$

where v, v_0, v_1 are the neighboring triangle vertices clockwise, and $s = \|v - v_1\|/\|v_0 - v_1\|$ is a known scalar of initial grid. $\tau(\hat{v})$ equals to 1 only if $\hat{v}$ is on the valid area of the blended atlas $\boldsymbol{A}_i$. For more details of ASAP warping, please refer to. [56], [57], [58] Before blending $\boldsymbol{A}_i$ and $\boldsymbol{a}_i$ using Eqn. (15), we warp $\boldsymbol{A}_i$ using bilinear interpolation with the grid deformation $\hat{V}$, which can be accomplished in the OpenGL rasterization pipeline efficiently. The proposed texturing scheme is evaluated in Fig. 5, which can produce sharper and more realistic textured results, compared to the per-vertex scheme.

5 Experimental Results

In this section, we first report the implementation details of our UnstructuredFusion, followed by the evaluation of our main technical contributions as well as the comparison with previous state-of-the-art methods, both qualitatively and quantitatively. The limitation and discussion regarding our UnstructuredFusion are provided in the last subsection.

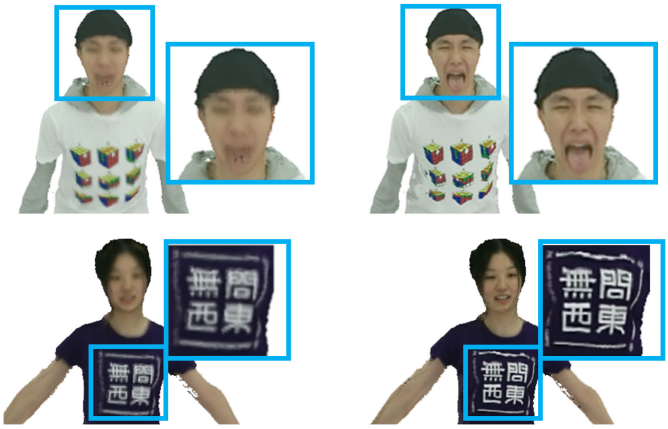

Fig. 5. Our atlas texturing scheme (right) compared to per-vertex color scheme (left). Per-vertex colors suffer from block artifacts while our method provides sharper textures with more dynamic facial expression.

Implementation Details UnstructuredFusion is implemented on a single NVIDIA GeForce GTX TITAN X GPU and a 3.2 GHz 4-core Xeon E3-1230 CPU with 16 GB of memory. The input live RGBD streams are captured from three Kinect v2 sensors in 30 fps with 512×424 resolution. The entire pipeline runs at 33 ms per frame, where skeleton warping based motion tracking takes around 16 ms with 5 ICP iterations, the TSDF fusion takes around 6 ms, the atlas texturing takes around 8 ms, and 3 ms for all the remaining computations. For online calibration, the parameters λ_{vdata}, λ_{sdata}, λ_{pdata} and λ_{prior} are set as 1.0, 2.0, 1.0 and 0.01, respectively. For motion tracking, we choose $\lambda_{\text{fit}} = 1.0$, $\lambda_{\text{data}} = 1.0$, $\lambda_{\text{bind}} = 1.0$, $\lambda_{\text{reg}} = 5.0$ and $\lambda_{\text{skele}} = 10.0$, respectively. Note that these parameters are set empirically for the balance of the cost of each term. For ED model, we use the 4 nearest node neighbors for ED warping, and the 8 nearest node neighbors to construct the ED graph following previous work [3], [7]. The TSDF voxel size is set as 4 mm in each dimension to preserve sufficient geometry detail of the target.

5.1 Evaluation

Several representative sequences reconstructed by UnstructuredFusion are illustrated in Fig. 6, where both the challenging motions and high-quality textures are reconstructed. In particular, we further evaluate our technique contributions, i.e., the skeleton warping based non-rigid tracking, the temporal blending based atlas texturing, as well as the online multi-camera calibration in the following contents respectively.

Skeleton Warping Based Non-rigid Tracking. In Fig. 7, we take three sequences as examples to present the effectiveness of the proposed skeleton warping scheme during the non-rigid tracking process qualitatively. As we expected, without skeleton warping, the fused model suffers from severe accumulated errors especially for the contents highlighted with the red circles, due to the misalignment between the unstructured RGBD streams from different camera views. On the contrary, with the proposed skeleton warping, our approach succeeds to align the unstructured sequences, leading to visually pleasant 4D geometry and texture reconstruction.

Camera Movement. We further evaluate the effectiveness of our method using three sequences captured by hand-held moving cameras as illustrated in Fig. 8. For these three sequence, during capturing the cameras maintain the target performer in the capture views by slightly moving in roughly fixed positions, moving forward in a roughly straight line and circling around the target, respectively. Note that in such hand-held setting, the movement of different cameras aggravates the difficulty of the consistent registration of unstructured cameras. Thus the reconstruction without skeleton warping fails quickly with uncanny geometry due to accumulated misalignment error as highlighted in the regions with the red circles in Fig. 8. In contrast, our method with skeleton warping can handle the misalignment caused by highly unstructured inputs of moving cameras with different kinds of motions, obtaining high quality reconstruction mesh.

Temporal Blending Based Atlas Texturing. Recall that in Fig. 5, the representative sequences qualitatively illustrate that the proposed atlas texturing method outperforms traditional per-vertex scheme in terms of producing sharper and more realistic textured results. To further evaluate the effectiveness of the particular grid-based warping procedure in our atlas texturing method, we take one representative sequence as an example to demonstrate the results of our atlas texturing scheme w/o the proposed grid-based warping in Fig. 9b and 9c, respectively. The result produced by per-vertex texturing scheme [3] is shown in Fig. 9d as well. The blue map visualizes the color-coded residual generated by comparing the textured result with the input color image. As we expected, the proposed atlas texturing scheme outperforms the per-vertex scheme, inducing much less residue and obtaining sharper textured results, while the particular grid-based warping procedure further improves the sharpness of the final textured result notably.

Furthermore, the corresponding quantitative error curves of our atlas texturing scheme and the per-vertex scheme are depicted in Fig. 9e. Note that the residuals are calculated as the per-pixel euclidean distances of the RGB values between the textured results and the color image inputs, where each color channel is normalized to [0,1]. It can be shown that our proposed atlas texturing scheme achieves around 0.33 average normalized error compared with 0.42 of the one without grid-based warping and 0.54 of per-vertex scheme, which illustrates the effectiveness of both our atlas blending method and the grid-based warping optimization. Note that the per-vertex scheme is even the best at the beginning of the sequence, because the blended atlas is still incomplete for the proposed atlas texturing scheme. Moreover, to further illustrate the effectiveness of the proposed atlas texturing scheme, we artificially demonstrate the result of atlas texturing using a sequence from DoubleFusion [6] with only single-view RGBD input, in Fig. 10, showing that our atlas texturing method allows for generating sharp and realistic textured results for both single-view input and commodity sparse multi-view input.

Online Multi-Camera Calibration. To evaluate the proposed online multi-camera calibration scheme, the state-of-the-art global registration methods 4PCS [59] and Go-ICP [60] are adopted for comparison. Go-ICP [60] combines a local ICP method with a branch-and-bound search to find the global minimum and 4PCS [59] performs global registration by constructing a congruent set of 4 points between range images. Fig. 11a presents the original depth inputs from the unstructured three camera views, while Fig. 11b, 11c, 11d are the

Fig. 6. Several examples that demonstrate the quality and fidelity of the reconstructed 4D geometry and texture results of the proposed UnstructuredFusion system.

registration results of 4PCS, [59] Go-ICP [60] and our online multi-camera calibration scheme in different render views for qualitative visualization, respectively. As highlighted by the circles and boxes, 4PCS and Go-ICP fail to align the three unstructured views and cause weird interlacement of the partial meshes from different camera views, especially for the head and limbs regions due to the small overlap between different camera views. In contrast our proposed method

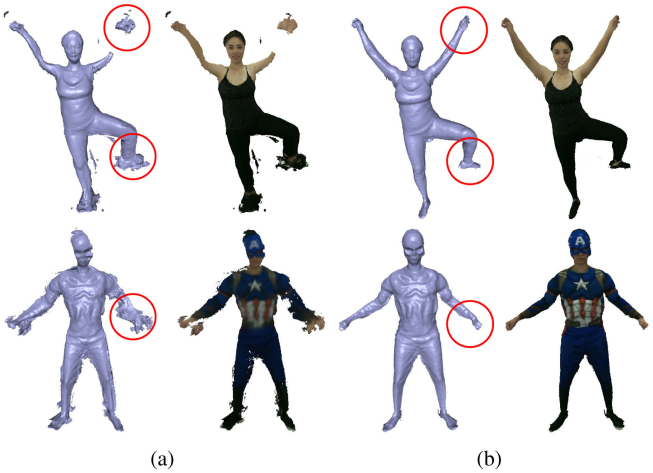

Fig. 7. Evaluation of skeleton warping. (a) Geometric and textured results without skeleton warping. (b) The corresponding results with skeleton warping.

obtains considerably better registration results in Fig. 11d by utilizing the solid human shape prior.

Camera Number. Recall that comparing to existing high-end multi-view capture systems [8], [17] our system utilizes only 3 commercial RGBD cameras for overall coverage of the target. Note that the reconstruction algorithm in our system can be extended to more cameras, but current light-weight sparse multi-view setup is more convenient for daily usage. In Fig. 12, we further evaluate the influence of camera numbers in our system. Our method achieve good fidelity reconstructions even using less cameras. However, in the cases with less cameras, our method fails to track the challenge motions in

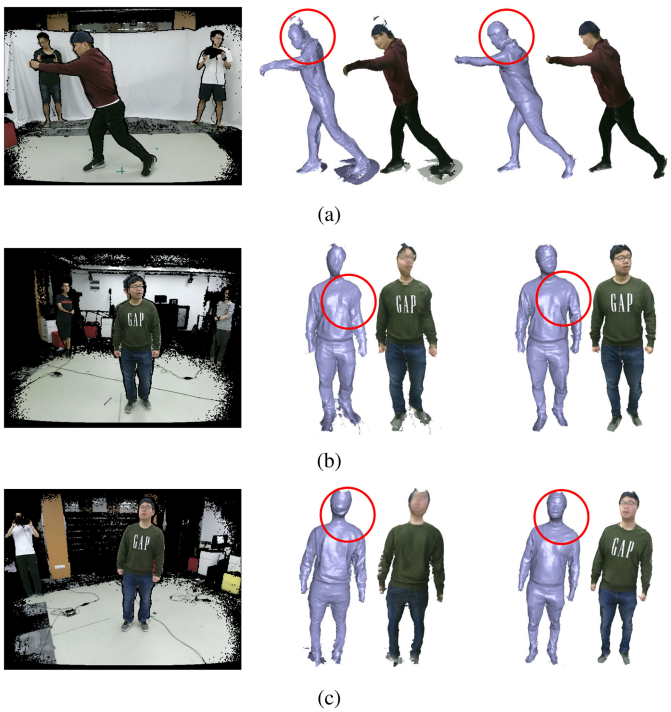

Fig. 8. Evaluation of our method using moving cameras. (a, b, c) The three example sequences in which the three cameras move in roughly fixed positions, move forward in a roughly straight line and circle around the captured target, respectively. From left to right: the captured scene; the reconstruction results without skeleton-warping and our results.

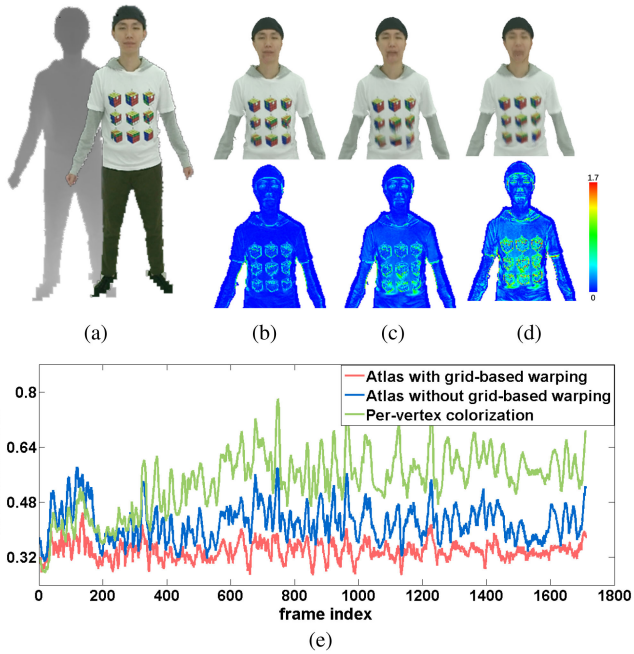

Fig. 9. Evaluation of the atlas blending. (a) Input depth and color image. (b) The reconstructed result of our atlas texturing with grid-based warping, where the blue map indicates the color-coded residual compared with the input color image. (c) The reconstructed result of our atlas texturing without grid-based warping. (d) The reconstructed result using per-vertex scheme. (e) The corresponding quantitative error curves.

the regions like elbows and knees due to the lack of enough valid constraints provided by the input depth streams. With more camera resources, more accurate target motions can be tracked, suffering from less accumulated misalignment errors.

5.2 Comparison

Apart from the evaluation of each technique contribution separately in the previous subsection, we demonstrate the overall performance of the proposed UnstructuredFusion by comparing it against other state-of-the-art methods both qualitatively and quantitatively in this subsection.

While the latest DoubleFusion [6] also utilizes SMPL model to regularize the embedded deformation, it is a single-view

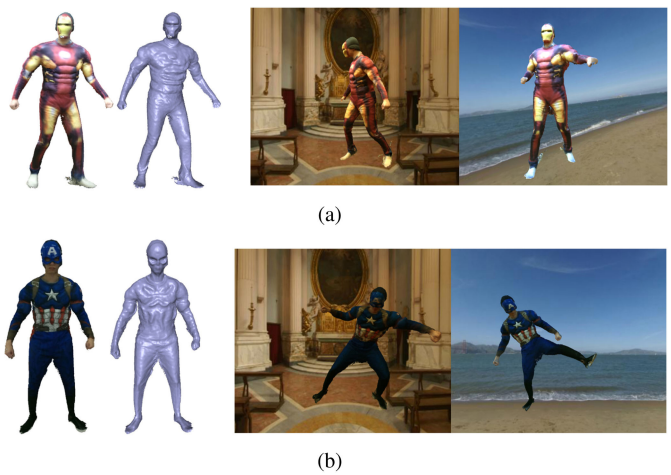

Fig. 10. Evaluation of atlas blending in terms of the number of input views. (a) The textured and relighting results with atlas blending using the single-view sequence from DoubleFusion.[6] (b) The results using the three-views sequence captured by UnstructuredFusion.

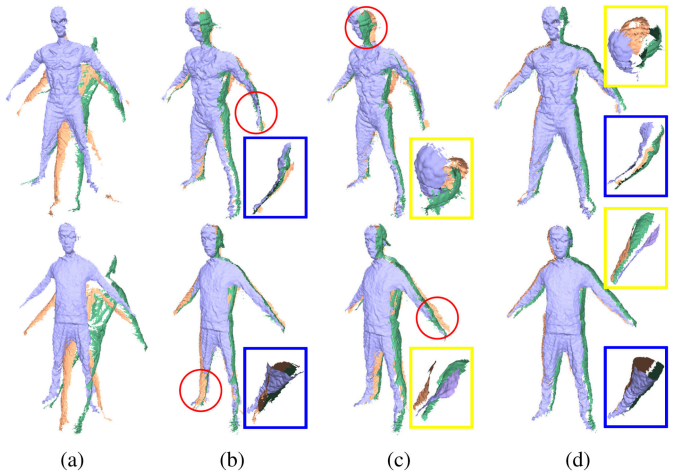

Fig. 11. Evaluation of online calibration. (a) The original depth inputs. (b, c,d) The registration results of 4PCS,[59] Go-ICP[60] and our online multi-camera calibration scheme, respectively. The red circles highlight the misaligned regions, while the corresponding boxes further visualize the misalignment in different render views. Note that different colors of the boxes encode different body regions.

method. For fair comparison of real-time dynamic reconstruction under the unstructured and sparse multi-view setting, we extend DoubleFusion [6] to the sparse multi-view setting, by directly formulating the data term in DoubleFusion [6] into multi-view depth inputs, denoted as Multi-DoubleFusion. In this basic extension, we adopt the same online calibration which is used in our method to obtain the initial camera poses and the embedded SMPL model. Note that similar as the other state-of-the-art multi-view methods, [7], [8] in Multi-DoubleFusion, we estimate the global rigid motion first using the rigid-ICP algorithm for each frame, and then the non-rigid parameters are estimated with fixed global rigid motion parameters.

Fig. 13 shows the qualitative comparison of our UnstructuredFusion against the other methods under consideration. The geometry results of DoubleFusion [6] suffer from fast self-occluded motions and challenging loop closure due to the limited capture view resource and incomplete geometry. While Multi-DoubleFusion tends to be more robust to the occlusion, it still suffers from accumulated misalignment errors between different views, leading to unnatural reconstruction results especially in the head and limb regions as

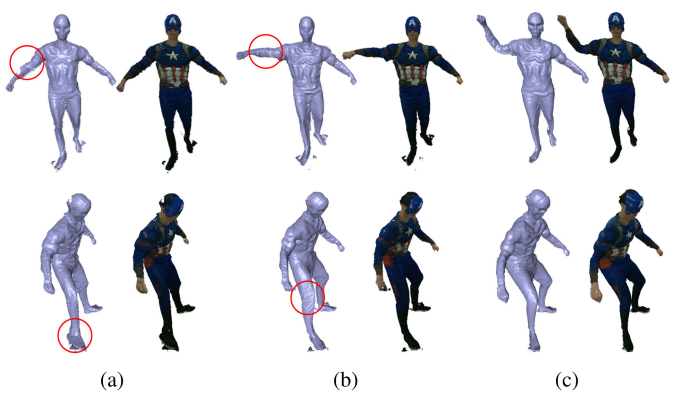

Fig. 12. Evaluation of the camera numbers. (a)-(c) are the reconstructed geometry and texture results using a single camera, 2 and 3 cameras, respectively.

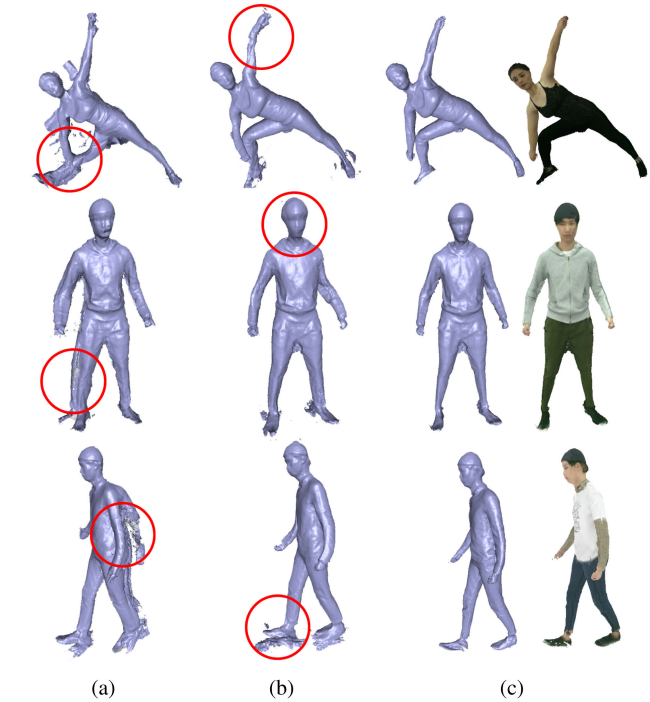

Fig. 13. Qualitative comparison. (a)-(c) are the reconstructed geometry/texture results of DoubleFusion,[6] Multi-DoubleFusion[6] and our UnstructuredFusion, respectively.

highlighted in the red circle regions. In contrast, our method is capable to handle the unstructured inputs and provide loop-closed results with fine geometric details. Besides, our method also performs dynamic and complete atlas texturing, ensuring more realistic reconstruction.

For quantitative comparison, we render the reconstructed geometry result into a 2D depth map in the camera view, and compute its MAE (Mean Absolute Error) by taking the depth input as the reference only in the visible surface regions. Note that even without ground truth reconstruction, this MAE metric encodes the reconstruction error for both the rigid ICP

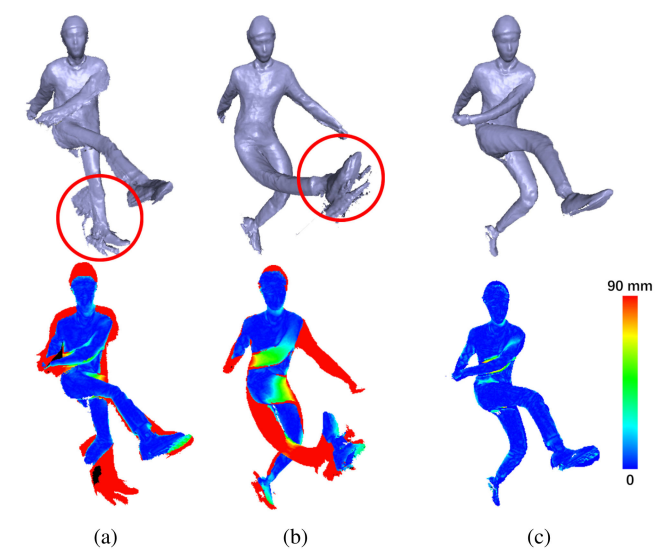

Fig. 14. Quantitative comparison. (a)-(c) are the reconstructed geometry/texture results of DoubleFusion,[6] Multi-DoubleFusion[6] and our UnstructuredFusion, respectively. The color-coded maps in bottom row indicate the projective error maps.

TABLE 1
Average Projective Numerical Errors of All the Captured Sequences for the Concerned Methods: DoubleFusion, [6] Multi-DoubleFusion [6] and Our UnstructuredFusion

	DoubleFusion	Multi-DoubleFusion	UnstructuredFusion
Sports	51.01 mm	45.15 mm	21.13 mm
Yoga	54.68 mm	48.07 mm	27.71 mm
Captain America	39.22 mm	30.47 mm	25.49 mm
Dancing	43.18 mm	40.54 mm	16.38 mm
Kicking	37.52 mm	36.03 mm	17.47 mm
Walking	34.13 mm	37.10 mm	20.96 mm
Spiderman	50.36 mm	43.27 mm	22.07 mm
Waving	45.83 mm	41.23 mm	23.74 mm
Crossing	44.38 mm	47.52 mm	26.09 mm

and nonrigid ICP processes of each method, providing a reliable quantitative comparison. As shown in Fig. 14, our method achieves high quality reconstruction results with less accumulated artifacts. The MAE for the entire sequence of our method is around 17.47 mm, compared with 37.52 mm of DoubleFusion [6] and 36.03 mm of Multi-DoubleFusion, respectively. Moreover, the MAE of all the captured sequences in our experiments are listed in Table 1, where the errors are computed in the visible surface regions only. It can be shown that our method leads to considerably less error, i.e., 22.34 mm average MAE, compared with 44.48 mm of DoubleFusion [6] and 39.04 mm of Multi-DoubleFusion. These quantitative comparisons reveal the effectiveness of our method for better non-rigid tracking during dynamic reconstruction in the unstructured and sparse multi-view setting.

To further evaluate our method quantitatively, we particularly compare our results with the marker based motion capture results using the OptiTrack system. Note that the capture sequences from the OptiTrack system and our system are synchronized by flashing the infrared LED. Similar to, [6], [43] the two system are calibrated using manually pre-selected corresponding pairs. After calibration, the detected marker positions from the OptiTrack coordinates system are transformed into the camera coordinates system of the first frame. Then for each concurrent frame, we further track the motions of these markers using the reconstructed motion field and compare the per-frame positions with the OptiTrack-detected ground-truth. Fig. 15 presents the numerical curves of per-frame maximum error of DoubleFusion, [6] Multi-DoubleFusion and

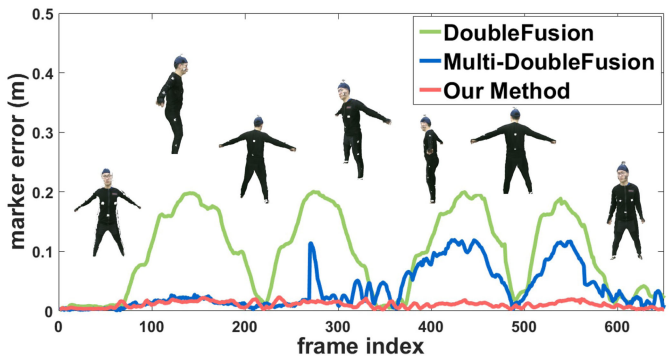

Fig. 15. Numerical error curves of our method, compared against DoubleFusion[6] and Multi-DoubleFusion. Note that the groud truth is obtained via the Vicon system.

TABLE 2
Average and Maximal Numerical Errors on the Entire Sequence Compared to the Ground Truth Observation from the OptiTrack System, for These Three Methods: DoubleFusion, [6] Multi-DoubleFusion and Our Method, Respectively

	DoubleFusion	Multi-DoubleFusion	Our Method
max error	0.2001 m	0.1195 m	0.0231 m
average error	0.0976 m	0.0368 m	0.0107 m

our method on one sequence, in which the performer is circling periodically for challenge loop closure motions. Note that the numerical error of DoubleFusion [6] gets worse periodically especially in the side-view of the performer, due to the limitation of DoubleFusion for tracking the non-rigid motions of the self-occluded regions in the side-view. Multi-DoubleFusion is more robust to the self-occlusion problem and achieves smaller error than DoubleFusion, [6] and even produces comparable results to our method until around frame 250, since for these frames the target motion is till slow and the accumulated error is still acceptable for reconstruction. However, Multi-DoubleFusion still suffers from the accumulated error caused by the misalignment of the unstructured inputs. In contrast, our method can handle the misalignment of unstructured inputs without the self-occlusion problem, and further achieves the smallest numerical error against the ground truth provided by the OptiTrack system. Besides, both the maximal and average errors of DoubleFusion, [6] Multi-DoubleFusion and our method for the sequences captured via the OptiTrack system are listed in Table 2 for better comparison. These numerical results illustrate that our method achieve higher tracking accuracy in the challenging unstructured and sparse multi-view setting. For more sequential results, we highly recommend viewing our accompanying video for more comprehensive evaluation of our approach.

5.3 Limitation and Discussion

As the first trial to explore the problem of real-time dynamic reconstruction for both geometry and texture in the unstructured sparse multi-view setting, the proposed UnstructuredFusion still owns limitations as follows.

From the reconstructed geometry aspect, due to the limited resolution of the depth input, our method cannot reconstruct the extremely fine details of the target especially in the face region. Data-driven techniques can be adopted to further generate synthetic geometry details in those model-specific regions. Besides, we cannot handle surface splitting topology changes, which we plan to address in future work by incorporating the key-volume update technique. [7] Our method is also restricted to human reconstruction, without modelling human-object interactions, which is critical for many practical applications. We are going to combine static object reconstruction method [61], [62] into our current framework. In addition, the reconstructed meshs suffer from the jittery effect in the regions without valid depth input like feets and fits. A further post-processing strategies in the 4D meshes like temporal filtering would alleviate such jittery effect. For the atlas texturing, our scheme relies on the fused color volume for those occluded regions when projecting into the camera views, which causes discrete colors near the boundary of the occluded regions. We plan to combine the atlas processing

method [63] in gradient domain as a post-processing step for seamless blending. It would also be promising to utilize generative model and data-driven methods to fill a more complete and sharp atlas. In addition, our texturing scheme is based on projective atlas, which is efficient but not compact enough. A texture atlas compression technique is needed to stream all the textured meshes to enable more efficient applications. For the overall system setup, our system cannot work outside due to the limitation of the available commercial RGBD sensors. We plan to combine the binocular solution with the available learning technique [64] to enhance the quality of the captured raw data. Another issue of our system is that it relies on a rough A-pose initialization of the performer. We are going to utilize the data driven technique to detect the human shape and pose during initialization. Such human shape detector can further be applied to our non-rigid tracking pipeline to prevent accumulated tracking error.

6 Conclusion

Motivated by alleviating the strict requirements (such as highly structured multi-camera setup, tedious pre-calibration and synchronization procedure) when generating a high-quality 4D geometry and texture for human activities, we proposed to use unstructured commercial RGBD cameras to realize a practicable realtime markerless human performance capture system, denoted as UnstructuredFusion. Under the flexible hardware setup using simply three unstructured RGBD cameras, we mainly solved the challenge online multi-camera calibration, non-rigid tracking, as well as atlas texturing problems based on multiple asynchronous videos. The proposed solution stands on the solid global constraints of human body and human motion modeled by the skeleton and the skeleton warping, respectively. We have conducted extensive experiments to evaluate the effectiveness of UnstructuredFusion in high-quality geometry and texture 4D reconstruction without tiresome pre-calibration, even allocating three cameras flexibly in a handheld way. Our UnstructuredFusion succeeds to liberate the cumbersome hardware and software restrictions in conventional structured multi-camera system, while eliminating the inherent occlusion issues under the single camera setup.

Acknowledgments

This work is supported in part by Natural Science Foundation of China (NSFC) under contract No. 61722209, 6181001011 and 61827805.

References

[1] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 343–352.

[2] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time volumetric non-rigid reconstruction," *Comput. Vis. – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Springer International Publishing, pp. 362–379, 2016.

[3] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo and motion reconstruction using a single rgbd camera," *ACM Trans. Graph.*, vol. 36, 2017, Art. no. 44a.

[4] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu, "Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 910–919.

[5] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic, "KillingFusion: Non-rigid 3D reconstruction without correspondences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5474–5483.

[6] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu, "Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018.

[7] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu, "DoubleFusion: Real-time capture of human performances with inner body shapes from a single depth sensor," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun., 2018, pp. 7287–7296, doi: 10.1109/CVPR.2018.00761.

[8] M. Dou, P. Davidson, S. R. Fanello, S. Khamis, A. Kowdle, C. Rhemann, V. Tankovich, and S. Izadi, "Motion2fusion: Real-time volumetric performance capture," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 246:1–246:16, Nov. 2017.

[9] H. Woltring, "New possibilities for human motion studies by real-time light spot position measurement," *Biotelemetry*, vol. 1, no. 3, 1974, Art. no. 132.

[10] Vicon, "Vicon systems," 2016. [Online]. Available: http://www.vicon.com

[11] L. A. Schwarz, D. Mateus, and N. Navab, "Multiple-activity human body tracking in unconstrained environments," in *Articulated Motion and Deformable Objects*. New York, NY, USA: Springer, 2010, pp. 192–202.

[12] P. Zhang, K. Siu, J. Zhang, C. K. Liu, and J. Chai, "Leveraging depth cameras and wearable pressure sensors for full-body kinematics and dynamics capture," *ACM Trans. Graph.*, vol. 33, no. 6, 2014, Art. no. 221.

[13] T. Shiratori, H. S. Park, L. Sigal, Y. Sheikh, and J. K. Hodgins, "Motion capture from body-mounted cameras," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 30, no. 4, 2011, Art. no. 31.

[14] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, "Performance capture from sparse multi-view video," *ACM Trans. Graph.*, vol. 27, no. 3, 2008, Art. no. 98.

[15] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt, "Fast articulated motion tracking using a sums of gaussians body model," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 951–958.

[16] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3334–3342.

[17] 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 Trans. Graph.*, vol. 34, no. 4, 2015, Art. no. 69.

[18] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8320–8329.

[19] S. Orts-Escolano, C. Rhemann, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, S. Khamis, M. Dou, V. Tankovich, C. Loop, Q. Cai, P. A. Chou, S. Mennicken, J. Valentin, V. Pradeep, S. Wang, S. B. Kang, P. Kohli, Y. Lutchyn, C. Keskin, and S. Izadi, "Holoportation: Virtual 3d teleportation in real-time," in *Proc. 29th Annu. Symp. User Interface Softw. Technol.*, 2016, pp. 741–754. [Online]. Available: http://doi.acm.org/10.1145/2984511.2984517

[20] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, P. Fua, H. Seidel, and C. Theobalt, "Mo2cap2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera," *CoRR*, vol. abs/1803.05959, 2018. [Online]. Available: http://arxiv.org/abs/1803.05959

[21] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, D. Mehta, H.-P. Seidel, and C. Theobalt, "Monoperfcap: Human performance capture from monocular video," *ACM Trans. Graph.*, vol. 37, no. 2, pp. 27:1–27:15, May 2018.

[22] M. Habermann, W. Xu, M. Zollhöfer, G. Pons-Moll, and C. Theobalt, "Livecap: Real-time human performance capture from monocular video," *ACM Trans. Graph.*, vol. 38, no. 2, pp. 14:1–14:17, Mar. 2019. [Online]. Available: http://doi.acm.org/10.1145/3311970

[23] Z. Zheng, T. Yu, Y. Wei, Q. Dai, and Y. Liu, "Deephuman: 3D human reconstruction from a single image," *arXiv preprint arXiv:1903.06473*, 2019.

[24] D. Xiang, H. Joo, and Y. Sheikh, "Monocular total capture: Posing face, body, and hands in the wild," *arXiv preprint arXiv:1812.01598*, 2018.

[25] T. Yu, Z. Zheng, Y. Zhong, J. Zhao, Q. Dai, G. Pons-Moll, and Y. Liu, "Simulcap : Single-view human performance capture with cloth simulation," *arXiv preprint arXiv:1903.06323*, 2019.

[26] N. Hasler, B. Rosenhahn, T. Thormahlen, M. Wand, J. Gall, and H.-P. Seidel, "Markerless motion capture with unsynchronized moving cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 224–231.

[27] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt, "Performance capture of interacting characters with handheld kinects," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 828–841.

[28] C. Wu, C. Stoll, L. Valgaerts, and C. Theobalt, "On-set performance capture of multiple actors with a stereo camera," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 32, no. 6, 2013, Art. no. 161.

[29] L. Xu, Y. Liu, W. Cheng, K. Guo, G. Zhou, Q. Dai, and L. Fang, "Flycap: Markerless motion capture using multiple autonomous flying cameras," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 8, pp. 2284–2297, Aug. 2018.

[30] R. Wang, L. Wei, E. Vouga, Q. Huang, D. Ceylan, G. Medioni, and H. Li, "Capturing dynamic textured surfaces of moving targets," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 271–288.

[31] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time Human Pose Recognition in Parts from Single Depth Images," in *Proc. 24th IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1297–1304.

[32] V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun, "Real time motion capture using a single time-of-flight camera," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 755–7692.

[33] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics," *Int. J. Comput. Vis.*, vol. 56, no. 3, pp. 179–194, 2004.

[34] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H. Seidel, "Motion capture using joint skeleton tracking and surface estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1746–1753.

[35] K. Guo, F. Xu, Y. Wang, Y. Liu, and Q. Dai, "Robust non-rigid motion tracking and surface reconstruction using L0 regularization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3083–3091.

[36] Y. Liu, J. Gall, C. Stoll, Q. Dai, H.-P. Seidel, and C. Theobalt, "Markerless motion capture of multiple characters using multiview image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2720–2735, Nov. 2013.

[37] H. Li, B. Adams, L. J. Guibas, and M. Pauly, "Robust single-view geometry and motion reconstruction," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 28, no. 5, 2009, Art. no. 175.

[38] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Trans. Graph.*, vol. 26, no. 3, 2007, Art. no. 80.

[39] 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 Trans. Graph.*, vol. 33, no. 4, 2014, Art. no. 156.

[40] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interactive Tech.*, 1996, pp. 303–312. [Online]. Available: http://doi.acm.org/10.1145/237170.237269

[41] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-Time Dense Surface Mapping and Tracking," in *Proc. 10th IEEE Int. Symp. Mixed Augmented Reality*, 2011, pp. 127–136.

[42] M. Slavcheva, M. Baust, and S. Ilic, "SobolevFusion: 3D reconstruction of scenes undergoing free non-rigid motion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2646–2655.

[43] Z. Zheng, T. Yu, H. Li, K. Guo, Q. Dai, L. Fang, and Y. Liu, "HybridFusion: Real-time performance capture using a single depth sensor and sparse IMUs," *Comput. Vis. – ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., pp. 389–406, 2018.

[44] Q.-Y. Zhou and V. Koltun, "Color map optimization for 3d reconstruction with consumer depth cameras," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 155:1–155:10, Jul. 2014.

[45] M. Soucy, G. Godin, and M. Rioux, "A texture-mapping approach for the compression of colored 3d triangulations," *Visual Comput.*, vol. 12, no. 10, pp. 503–514, Dec. 1996.

[46] K. Zhou, J. Synder, B. Guo, and H.-Y. Shum, "Iso-charts: Stretch-driven mesh parameterization using spectral analysis," in *Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Process.*, 2004, pp. 45–54. [Online]. Available: http://doi.acm.org/10.1145/1057432.1057439

[47] M. Waechter, N. Moehrle, and M. Goesele, "Let there be color! large-scale texturing of 3d reconstructions," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 836–850.

[48] Y. Fu, Q. Yan, L. Yang, J. Liao, and C. Xiao, "Texture mapping for 3d reconstruction with rgb-d sensor," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4645–4653.

[49] J. Huang, A. Dai, L. Guibas, and M. Niessner, "3dlite: Towards commodity 3d scanning for content creation," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 203:1–203:14, Nov. 2017.

[50] S. Bi, N. K. Kalantari, and R. Ramamoorthi, "Patch-based optimization for image-based texture mapping," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 106:1–106:11, Jul. 2017.

[51] R. Du, M. Chuang, W. Chang, H. Hoppe, and A. Varshney, "Montage4d: Interactive seamless fusion of multiview video textures," in *Proc. ACM SIGGRAPH Symp. Interactive 3D Graph. Games*, 2018, pp. 5:1–5:11.

[52] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 248:1–248:16, Oct. 2015.

[53] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 561–578.

[54] F. Crete, T. Dolmiere, P. Ladret, and M. Nicolas, "The blur effect: perception and estimation with a new no-reference perceptual blur metric," in *Human Vis. Electron. Imaging XII*, vol. 6492, Feb. 2007, Art. no. 64920I.

[55] J. Bian, W.-Y. Lin, Y. Matsushita, S.-K. Yeung, T. D. Nguyen, and M.-M. Cheng, "Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2828–2837.

[56] T. Igarashi, T. Moscovich, and J. F. Hughes, "As-rigid-as-possible shape manipulation," in *Proc. ACM SIGGRAPH Papers*, 2005, pp. 1134–1141.

[57] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, "Content-preserving warps for 3d video stabilization," in *Proc. ACM SIGGRAPH Papers*, 2009, pp. 44:1–44:9.

[58] S. Liu, L. Yuan, P. Tan, and J. Sun, "Bundled camera paths for video stabilization," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 78:1–78:10, Jul. 2013.

[59] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust surface registration," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 27, no. 3, 2008, Art. no. 85.

[60] J. Yang, H. Li, D. Campbell, and Y. Jia, "Go-icp: A globally optimal solution to 3d icp point-set registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2241–2254, Nov. 2016.

[61] L. Han and L. Fang, "Flashfusion: Real-time globally consistent dense 3d reconstruction using cpu computing," *Proc. Robot.: Sci. Syst.*, Jun. 2018, doi: 10.15607/RSS.2018.XIV.006.

[62] L. Han, L. Xu, D. Bobkov, E. Steinbach, and L. Fang, "Real-time global registration for globally consistent rgb-d slam," *IEEE Trans. Robot.*, vol. 35, no. 2, pp. 498–508, Apr. 2019.

[63] F. Prada, M. Kazhdan, M. Chuang, and H. Hoppe, "Gradient-domain processing within a texture atlas," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 154:1–154:14, Jul. 2018. [Online]. Available: http://doi.acm.org/10.1145/3197517.3201317

[64] S. Yan, C. Wu, L. Wang, F. Xu, L. An, K. Guo, and Y. Liu, "Ddrnet: Depth map denoising and refinement for consumer depth cameras using cascaded cnns," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 155–171.

Lan Xu received the BS degree from the Department of Information and Communication, College of Information Science & Electronic Engineering, Zhejiang University, Hangzhou, China, in 2015. He is currently working toward the PhD degree in the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong SAR. His current research interests include computer vision and computer graphics.

Zhuo Su received the BS degree from the Department of Automation, College of Information Science and Engineering, Northeastern University, China, in 2018. He is currently working toward the MS degree in the Department of Automation, Tsinghua University, China. His current research interests include computer vison and graphics.

Lei Han received the degree in electrical engineering from the Hong Kong University of Science and Technology, China and Tsinghua University, China, and the BS degree in July 2013 and joined the Department of Electrical Computing Engineering, Hong Kong University of Science and Technology in September 2016, where he is working toward the PhD degree. His current research focuses on multiview geometry and 3D computer vision.

Tao Yu received the BS degree in measurement and control from Hefei University of Technology, China, in 2012. He is currently working toward PhD degree in instrumental science at Beihang University, China. His current research interests include computer vision and computer graphics.

Yebin Liu received the BE degree from the Beijing University of Posts and Telecommunications, China, in 2002, and the PhD degree from the Automation Department, Tsinghua University, Beijing, China, in 2009. He is currently an associate professor with Tsinghua University. His research areas include computer vision, computer graphics and computational photography. He was a research fellow with the Computer Graphics Group, Max Planck Institute for Informatik, Germany, in 2010.

Lu Fang received the BE degree from University of Science and Technology of China, in 2007 and the PhD degree from Hong Kong University of Science and Technology, in 2011. She is currently an associate professor with Tsinghua University. Her research interests include computational photography and 3D vision. She used to received Best Student Paper Award in ICME 2017, Finalist of Best Paper Award in ICME 2017 and ICME 2011 etc. She is a senior member of the IEEE.