Function4D: Real-time Human Volumetric Capture from Very Sparse Consumer RGBD Sensors

Tao Yu[1], Zerong Zheng[1], Kaiwen Guo[2], Pengpeng Liu[3], Qionghai Dai[1], Yebin Liu[1]
[1]Department of Automation, Tsinghua University, Beijing, China   [2]Google, Switzerland
[3]Institute of Automation, Chinese Academy of Sciences, Beijing, China

Figure 1: Volumetric capture results of various challenging scenarios using Function4D.

Abstract

Human volumetric capture is a long-standing topic in computer vision and computer graphics. Although high-quality results can be achieved using sophisticated off-line systems, real-time human volumetric capture of complex scenarios, especially using light-weight setups, remains challenging. In this paper, we propose a human volumetric capture method that combines temporal volumetric fusion and deep implicit functions. To achieve high-quality and temporal-continuous reconstruction, we propose dynamic sliding fusion to fuse neighboring depth observations together with topology consistency. Moreover, for detailed and complete surface generation, we propose detail-preserving deep implicit functions for RGBD input which can not only preserve the geometric details on the depth inputs but also generate more plausible texturing results. Results and experiments show that our method outperforms existing methods in terms of view sparsity, generalization capacity, reconstruction quality, and run-time efficiency.

1. Introduction

Real-time volumetric capture of human-centric scenarios is the key to a large number of applications ranging from telecommunications, education, entertainment, and so on. And the underlying technique, volumetric capture, is a challenging and long-standing problem in both computer vision and computer graphics due to the complex shapes,

fast motions, and changing topologies (e.g., human-object manipulations and multi-person interactions) that need to be faithfully reconstructed. Although high-end volumetric capture systems [4, 13, 25, 5, 27, 36] based on dense camera rigs (up to 100 cameras [8]) and custom-designed lighting conditions [47, 14] can achieve high-quality reconstruction, they all suffer from complicated system setups and are limited to professional studio usage.

In contrast, light-weight volumetric/performance capture systems are more practical and attractive. Given a pre-scanned template, [22, 59, 15] track dense surface deformations from single-view RGB input [17, 18]. However, the prerequisite of a fixed-topology template restricts their applications for general volumetric capture. In 2015, DynamicFusion [33] proposed the first template-free and single-view dynamic 3D reconstruction system. The following works [53, 54, 51, 42] further improve the reconstruction quality for human performance capture by incorporating semantic body priors. However, it remains challenging for them to handle large topological changes like dressing or taking-off clothes. Recently, a line of research [32, 37, 38, 23] leverages deep implicit functions for textured 3D human reconstruction only from a single RGB image. However, they still suffer from off-line reconstruction performance [38, 37] or over-smoothed, temporally discontinuous results [23]. State-of-the-art real-time volumetric capture systems are volumetric fusion methods like Fusion4D [10] and Motion2Fusion [9]. But both of them rely on custom-designed high-quality depth sensors (up to

120 fps and 1k resolution) and multiple (up to 9) high-end GPUs, which is infeasible for consumer usage.

In this paper, we propose Function4D, a volumetric capture system using very sparse (as sparse as 3) consumer RGBD sensors. Compared with existing systems, our system is able to handle various challenging scenarios, including human-object manipulations, dressing or taking off clothes, fast motions and even multi-person interactions, as shown in Fig. 1.

Our key observations are: To generate complete and temporal consistent results, current volumetric fusion methods have to fuse as much temporal depth observations as possible. This results in heavy dependency on accurate and long-term non-rigid tracking, which is especially challenging under severe topology changes and large occlusions. On the contrary, deep implicit functions are good at completing surfaces, but they cannot recover detailed and temporal continuous results due to the insufficient usage of depth information and severe noise from consumer RGBD sensors.

To overcome all the limitations above, we propose a novel volumetric capture framework that organically combines volumetric fusion with deep implicit functions. By introducing dynamic sliding fusion, we re-design the volumetric fusion pipeline to restrict tracking and fusion in a sliding window and finally got noise-eliminated, topology-consistent, and temporally-continuous volumetric fusion results. Based on the sliding fusion results, we propose detail-preserving deep implicit functions for final surface reconstruction to eliminate the heavy dependency on long-term tracking. Moreover, by encoding truncated projective SDF (PSDF) values explicitly and incorporating attention mechanism into the multi-view feature aggregation stage, our networks not only achieve detailed reconstruction results but also orders of magnitude faster than existing methods.

Our contributions can be summarized as:

- The first real-time volumetric capture system which combines volumetric fusion with deep implicit functions using very sparse consumer RGBD sensors.

- Dynamic Sliding Fusion for generating noise-eliminated and topology consistent volumetric fusion results.

- Detail-preserving Implicit Functions specifically designed for sufficient utilization of RGBD information to generate detailed reconstruction results.

- The training and evaluation dataset, which contains 500 high-resolution scans of various poses and clothes, will be publicly available to stimulate future research.

2. Related Work

In the following, we focus on 3D human volumetric/performance capture and classify existing methods into 4 categories according to their underlying techniques.

Volumetric capture from multi-view stereo. Multi-view volumetric capture is an active research area in the computer vision and graphics community. Previous works use multi-view images for human model reconstruction [20, 41, 25]. Shape cues like silhouette, stereo, shading, and cloth priors have been integrated to improve reconstruction/rendering performance [41, 25, 49, 48, 47, 4, 31, 36, 50]. State-of-the-art methods build extremely sophisticated systems where up to 100 cameras [8] and even custom-designed gradient lighting [14] for high-quality volumetric capture. In particular, the methods by [8] and [14] first perform multi-view stereo for the point cloud generation, followed by mesh construction, simplification, tracking, and post-processing steps such as UV mapping. Although the results are compelling, the reliance on well-controlled multi-camera studios and a huge amount of computational resources prohibit them from being used in living spaces.

Template-based Performance Capture. For performance capture, some of the previous works leverage pre-scanned templates and exploit multi-view geometry to track the motion of the templates. For instance, the methods in [46, 13, 5] adopted a template with an embedded skeleton driven by multi-view silhouettes and temporal feature constraints. Their methods are then extended to handle multiple interacting characters in [27, 26]. Besides templates with an embedded skeleton, some works adopted non-rigid deformation for template motion tracking. Li *et,al* [22] utilized embedded deformation graph in [43] to parameterize the non-rigid deformations of the pre-scanned template. Guo *et,al.* [15] adopted an ℓ_0 norm constraint to generate articulated motions for bodies and faces without explicitly constructing a skeleton. Zollhöfer *et, al.* [59] took advantage of GPU to parallelize the non-rigid registration algorithm and achieved real-time performance of general non-rigid tracking. Recently, capturing 3D dense human body deformation with coarse-to-fine registration from a single RGB camera has been enabled [52] and improved for real-time performance [17]. DeepCap [18] introduced a deep learning method that jointly infers the articulated and non-rigid 3D deformation parameters in a single feed-forward pass. Although template-based approaches require less input than multi-view stereo methods, they are incapable of handling topological changes due to the prerequisite of a template with fixed topology.

Volumetric Fusion for Dynamic 3D Reconstruction. To get rid of template priors and realize convenient deployment, researchers turn to use one or sparse depth sensors for 3D reconstruction. In 2015, DynamicFusion [33] proposed the first template-free, single-view, real-time dynamic 3D reconstruction system, which integrates multiple frames into a canonical model to reconstruct a complete surface model. However, it only handles controlled and relatively slow motions due to the challenges of real-time non-rigid tracking. In order to improve the robustness of Dynam-
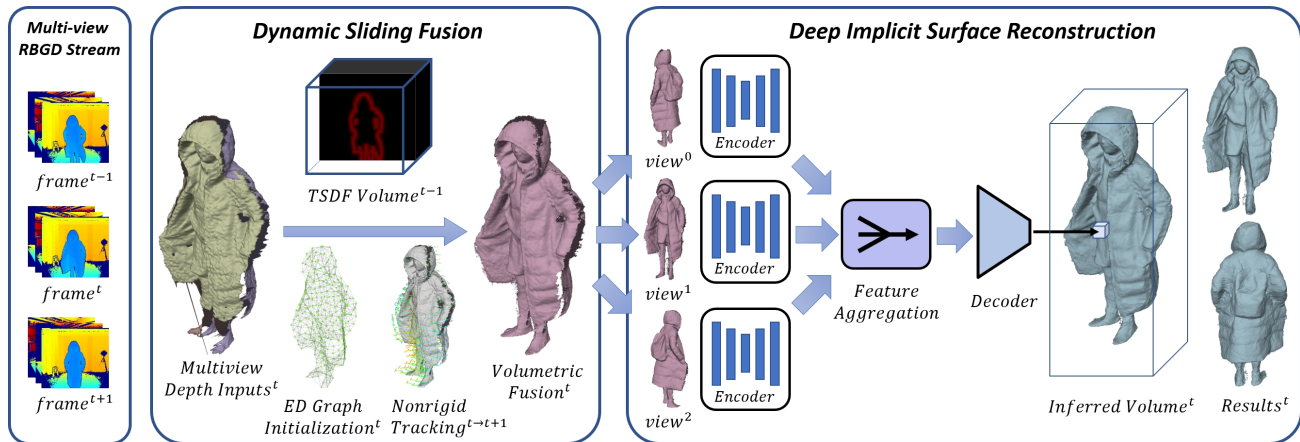
Figure 2: Overview. We focusing on geometry reconstruction in this figure. For color inference please refer to Fig.5.

icFusion, the following works incorporated killing/sobolev constraints [39, 40], articulated priors [21], skeleton tracking [53], parametric body models [54], sparse inertial measurements [55], data-driven prior [42] or learned correspondences [3] into the non-rigid fusion pipeline. However, all of these methods are prone to tracking failure in invisible areas, which is an inherent drawback of single-view systems. To overcome this limitation, Fusion4D [10] and Motion2fusion [9] focused on real-time multi-view setups using high-end custom-design sensors, with the notion of key volume updating and learning-based surface matching. Even though the pipelines were carefully designed in [10] and [9], they still suffer from incomplete and noisy reconstructions when severe topological changes occur especially under very sparse system setups.

Learning-based 3D Human Reconstruction. Fueled by the rapid developments in neural 3D representations(e.g., [34, 35, 6, 29] etc.), a lot of data-driven methods for 3D human reconstruction have been proposed in recent years. Methods in [58, 1] proposed to deform a parametric body model to fit the image observations including keypoints, silhouettes, and shading. DeepHuman [57] combined the parametric body model with a coarse-scale volumetric reconstruction network to reconstruct 3D human models from a single RGB image. Some methods infer human shapes on 2D image domains using multi-view silhouettes [32] or front-back depth pairs [12]. PIFu [37] proposed to regress an implicit function using pixel-aligned image features. Unlike voxel-based methods, PIFu is able to reconstruct high-resolution results thanks to the compactness of the implicit representations. PIFuHD [38] extended PIFu to capture more local details. However, both PIFu [37] and PIFuHD [38] fail to reconstruct plausible models in cases of challenging poses and self-occlusions. PaMIR [56] resolved this challenge by using the SMPL model as a prior but suffers from run-time inefficiency since it requires

a post-processing optimization step. IFNet [7] and IP-Net [2] can recover impressive 3D humans from partial point clouds, but the dependency on multi-scale 3D convolutions and parametric body models block the realization of real-time reconstruction performance.

3. Overview

As shown in Fig. 2, the proposed volumetric capture pipeline mainly contains 2 steps: Dynamic Sliding Fusion and Deep Implicit Surface Reconstruction. Given a group of synchronized multi-view RGBD inputs, we first perform dynamic sliding fusion by fusing its neighboring frames to generate noise-eliminated and temporal-continuous fusion results. After that, we re-render multi-view RGBD images using the sliding fusion results in the original viewpoints. Finally, in the deep implicit surface reconstruction step, we propose detail-preserving implicit functions (which consists of multi-view image encoders, a feature aggregation module, and an SDF/RGB decoder) for generating detailed and complete reconstruction results.

4. Dynamic Sliding Fusion (DSF)

Different from previous volumetric fusion methods, the proposed DSF method aims at augmenting current observations but not completing surfaces. So we re-design the fusion pipeline to make sure we can get topologically consistent and noise-eliminated results for current observations.

The proposed DSF mainly contains 3 steps: topology-aware node graph initialization, non-rigid surface tracking, and observation-consistent truncated SDF (TSDF) fusion. To eliminate the heavy dependency on long-term tracking, instead of fusing all frames as in previous methods, we only perform fusion in a sliding window in DSF. Specifically, we allow a 1-frame delay for the reconstruction pipeline and fuse the current frame (indexed by t) only with its preced-
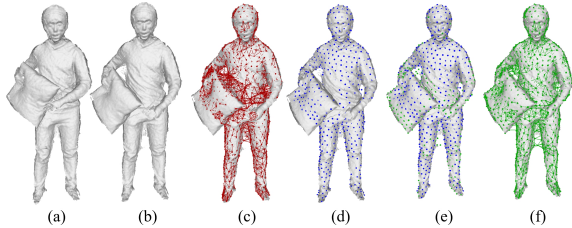
Figure 3: Topology-aware node graph initialization. (a) and (b) are the preceding and the current frame, respectively. (c) is the overlay of the current frame and the warped node graph of the preceding frame; erroneous nodes can be found around the pillow and arms where large topological changes occur. (d), (e) and (f) are the cleared node graph, the additional nodes (green) for newly observed surfaces, and the final node graph for the current frame, respectively.
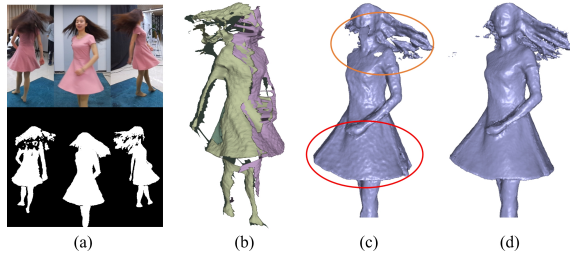

Figure 4: Evaluation of dynamic sliding fusion. From (a) to (d) are multi-view RGB references and depth masks, multi-view depth input rendered in a side viewpoint, results without and with dynamic sliding fusion, respectively.

ing frame (t-1) as well as its succeeding frame (t+1) to minimize the topological changes and tracking error accumulation. Note that we only need to perform non-rigid surface tracking for the succeeding frame since the deformation between the current frame and the preceding frame has already been tracked. Regarding the TSDF fusion stage, we propose observation-consistent TSDF fusion to fuse multi-view observations of frame t+1 into frame t.

4.1. Topology-aware Node Graph Initialization

Previous fusion methods initialize the embedded deformation graph (ED graph) [43] in the canonical frame [33]. However, such an ED graph cannot well describe the topological changes in live frames. Different from previous methods, we have to initialize the ED graph exactly for the current frame to guarantee that the topology of the node graph is consistent with the current observations. However, it is inefficient to initialize the node graphs from scratch for every frame because of the complexity of node selection, graph connection, and volume KNN-field calculation. To overcome this limitation, we propose topology-aware node graph initialization, which can not only leverage the node graph of the previous frame for fast initialization but also has the ability to generate a topologically-consistent node graph for the current observations.

As shown in Fig. 3, we first initialize the current node graph using the live node graph from the preceding frame. This is achieved by warping the node graph of the preceding frame to the current frame directly. Due to tracking errors and topological changes, the live node graph may not be well aligned with current observations (Fig. 3(c)). So we clear those nodes that are located far from current observations by constraining their TSDF values in the current TSDF volume (Fig. 3(d)). Specifically, if the magnitude of the normalized TSDF value corresponding to a node is greater than δ_t, we suppose that it is relatively far from the observations

and we delete this node to maintain the current mesh topology. Finally, considering that there may still exist newly observed surfaces that are not covered by the cleared node graph, we further refine it based on the current observations by sampling additional nodes (Fig. 3(e)) as in DynamicFusion [33] to make sure that all of the surfaces can be covered by the final node graph (Fig. 3(f)).

4.2. Non-rigid Surface Tracking

For non-rigid surface registration, we follow previous methods to search projective point-to-plane correspondences between the surface at frame t and the multi-view depth observations at frame t+1. The definition of the non-rigid tracking energy is:

$$E_{\text{tracking}} = \lambda_{\text{data}} E_{\text{data}} + \lambda_{\text{reg}} E_{\text{reg}}, \qquad (1)$$

where E_{data} and E_{reg} are the energies of data term and regularization term respectively. The data term measures the fitting error between the deformed surface and the depth observations, while the regularization term enforces local as-rigid-as-possible surface deformations. A Gauss-Newton solver with Preconditioned Conjugate Gradient algorithm (PCG) is used to solve this non-linear optimization problem efficiently on GPU. Please refer to [33] for more details.

4.3. Observation-consistent TSDF Fusion

After non-rigid surface tracking, we warp the volume of frame t to the depth observations at frame t+1 for TSDF fusion. Since we are focusing on augmenting the current observations but not completing surfaces, we propose an aggressive TSDF fusion strategy to eliminate the impact of tracking errors and severe topological changes. Specifically, in our fusion method, a voxel in the volume of frame t will update its TSDF value if and only if: i) the tracking error corresponding to this voxel is lower than a threshold δ_e, and ii) there exist valid voxels (voxels that in the truncated band of current observations) located around this voxel (the searching radius is set to 3 in our implementation).

Calculating the tracking error for a specific voxel is not straightforward since non-rigid tracking is established between the reference surface and the depth maps [9]. So we first calculate the tracking error for each node on the ED graph and then calculate the tracking error for each voxel by interpolation. Suppose $\mathcal{C} = (v_i, p_i)$ is the correspondence set after the last iteration of non-rigid tracking, where v_i is the ith vertex on the reference surface and p_i is the corresponding point of v_i on live depth inputs. The tracking error corresponding to the jth node can be calculated as:

$$\mathbf{e}(n_j) = \frac{\sum_{v_i \in \mathcal{C}_j} \mathbf{r}(v_i', p_i)}{\sum_{v_i \in \mathcal{C}_j} \mathbf{w}(v_i, n_j) + \epsilon}, \qquad (2)$$

where $\mathbf{r}(v_i', p_i) = \|pn_i^T \cdot (v_i' - p_i)\|_2^2$ is the residue of v_i after non-rigid tracking, v_i' the warped position of v_i, pn_i the surface normal of p_i, $\mathcal{C}_j$ a subset of $\mathcal{C}$ which includes all the reference vertices controlled by node j, $\mathbf{w}(v_i, n_j) = exp(-\|v_i - n_j\|_2^2/(2d^2))$ the blending weight of node j on v_i, in which d is the influence radius of n_j and $\epsilon = 1e-6$ is used to avoid zero division.

For a voxel x_k, its tracking error is then interpolated using its K-Nearest-Neighbors on the node graph $\mathcal{N}(x_k)$ (where $K = 4$) as:

$$\mathbf{e}(x_k) = \sum_{j \in \mathcal{N}(x_k)} \mathbf{w}(x_k, n_j) \cdot \mathbf{e}(n_j). \qquad (3)$$

5. Deep Implicit Surface Reconstruction

After dynamic sliding fusion, we can get noise-eliminated surfaces. However, the surfaces are by no means complete due to the very sparse inputs and occlusions. The goal of the deep implicit surface reconstruction step is to generate complete and detailed surface reconstruction results using deep implicit functions. Since we have already fused a 3D TSDF volume in dynamic sliding fusion, a straightforward idea is to use a 3D convolution-based encoder-decoder network to "inpaint" the volume. And the methods in [7, 2] have achieved complete 3D surface reconstruction results by proposing multi-scale 3D convolution networks. However, the dependency on inefficient 3D convolution limits their applications in real-time systems, and the huge memory consumption restricts them from generating high-resolution results. In contrast, real-time implicit surface reconstruction can be achieved using 2D pixel-aligned local features combined with positional encoding as shown in [23]. However, it was designed for using only RGB images and can only generate over-smoothed results. Finally, regarding the RGBD-based implicit functions proposed in [24], we can see that simply adding depth input as an additional input channel still cannot preserve the geometric details on the depth inputs.

To resolve the limitations above, we propose a new deep implicit surface reconstruction method that is specifically
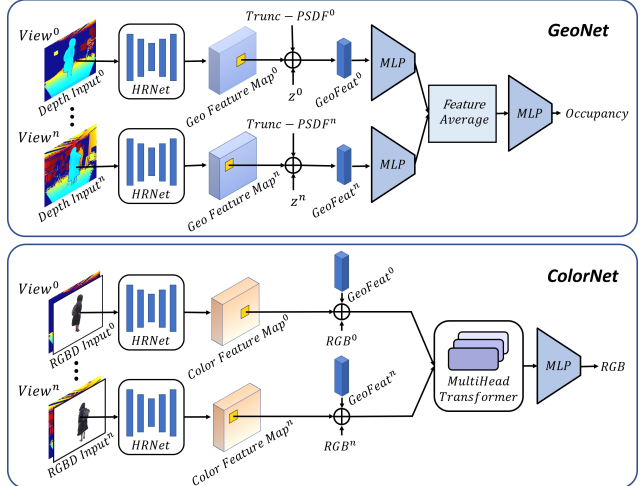

Figure 5: Network structures of GeoNet and ColorNet.

designed for RGBD input. The implicit surface reconstruction contains two steps: First, we re-render the multi-view RGBD images from the fused surface after dynamic sliding fusion. And then, given multi-view RGBD inputs, we propose detail-preserving implicit functions to reconstruct a complete surface with texture for the current frame.

5.1. Multi-view RGBD Re-rendering

In this step, we re-render multi-view RGBD images from the fused surfaces using input camera viewpoints. The re-rendered RGBD images contain much less noise than the original inputs thanks to the dynamic sliding fusion step. Note that another benefit of multi-view RGBD re-rendering is that we can manually fix the perspective projection parameters for all the rendered RGBD images to make sure they are consistent with the projection parameters that were used for rendering the training dataset.

5.2. Detail-preserving Implicit Functions

We propose two networks, *GeoNet* and *ColorNet*, for inferring detailed and complete geometry with color from multi-view RGBD images. As shown in Fig. 5, GeoNet and ColorNet share similar network architectures. Different from [37, 23], we explicitly calculate the truncated PSDF feature in *GeoNet* for preserving geometric details on depth maps. Moreover, we use a multi-head transformer network for multi-view feature aggregation in *ColorNet* to generate more plausible color inference results. Empirically, we found that using only depth images is enough for training the *GeoNet*, so we eliminate the RGB information for geometry reconstruction for efficient reconstruction.
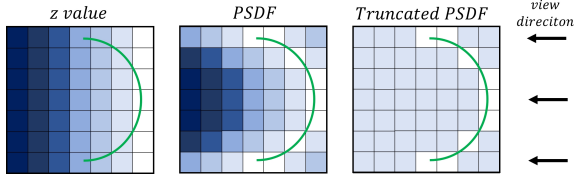
Figure 6: Illustration of the truncated PSDF feature. The green line represents the depth input. Note that we visualize the absolute of PSDF values here for simplicity, and the darker the grid, the larger the absolute PSDF values.
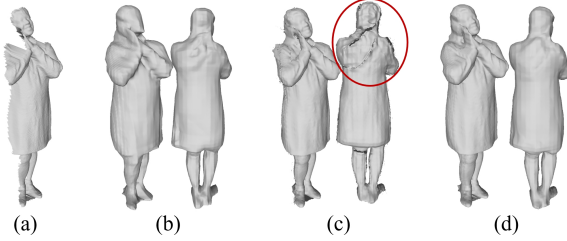

Figure 7: Evaluation of the truncated PSDF. (a) is single-view depth input, (b,c,d) are results generated without PSDF, with PSDF and with truncated PSDF, respectively.

5.2.1 Truncated PSDF Feature

The feature used for decoding occupancy values in pixel-aligned implicit functions can be decomposed into a 2D image feature and a positional encoding (z value in [37] or the one-hot mapping in [23]). The previous method [24] augments the 2D image feature by enhancing the 2D image feature using RGBD images as input. Although this can successfully guide the network to resolve the z-ambiguity of using only RGB images, it does not preserves the geometric details on the depth inputs. This is due to the fact that the variation of the geometric details on depth inputs is too subtle (when compared with the global range of depth inputs) for the networks to "sense" by 2D convolutions. To fully utilize the depth information, we propose to use truncated PSDF values as an additional feature dimension. The truncated PSDF value is calculated by:

$$f(q) = \mathbf{T}(q_{.z} - \mathbf{D}(\mathbf{\Pi}(q))), \qquad (4)$$

where q is the coordinate of the query point, $\mathbf{\Pi}(\cdot)$ is the perspective projection function, $\mathbf{D}(\cdot)$ is a bi-linear sampling function used for fetching depth values on the depth image, and $T(\cdot)$ is used to truncate the PSDF values in $[-\delta_p, \delta_p]$.

As shown in Fig. 6, the truncated PSDF value is a strong signal corresponding to the observed depth inputs. Moreover, it also eliminates the ambiguities of using global depth values. Fig. 7(b) demonstrates that without using the PSDF values, we can only get over-smoothed results even for the visible regions with detailed observations. Moreover, without truncation, the depth variations of the visible regions

(the arms on top of the body) will be misleadingly transferred to the invisible regions (Fig. 6) and finally leads to ghost artifacts (the ghost arm in the red circle of Fig. 7(c)).

5.2.2 Multi-view Feature Aggregation

Although PIFu [37] has demonstrated multi-view reconstruction results by averaging the intermediate feature of the SDF/RGB decoders, we argue that the average pooling operation has limited capacity and cannot capture the difference in inference confidence between different viewpoints. For color inference, the network should have the ability to "sense" the geometric structure and also the visibility in different viewpoints for a query point.

To fulfill this goal, we propose to leverage the attention mechanism in [45] for multi-view feature aggregation in *ColorNet*. Compared with direct averaging, the attention mechanism has the advantage of incorporating the inter-feature correlations between different viewpoints, which is necessary for multi-view feature aggregation. Intuitively, for a query point that is visible in $view^0$ but fully occluded in other views, the feature from the $view^0$ should play a leading role in the final decoding stage. As shown in Fig. 10, direct averaging of the multi-view features may lead to erroneous texturing results. On the contrary, using attention-based feature aggregation can enable more effective feature merging and thus generate more plausible color inference results. In practice, we follow [45] to use multi-head self-attention with 8 heads and 2 layers without positional encoding. The input is the concatenation of multi-view geometry features, color features, and RGB values as in Fig. 5. And we fuse the output multi-head features through a two layers FC and weighted summation. Moreover, we found that the attention mechanism has limited improvement for geometry reconstruction since the visibility has been encoded by the truncated PSDF feature in *GeoNet*.

6. Results

The results of our system are shown in Fig. 1 and Fig. 8. Note that the temporal-continuous results are reconstructed by our system under various challenging scenarios, including severe topological changes, human-object manipulations, and multi-person interactions.

6.1. Real-time Implementation

In order to achieve real-time performance, we implement our run-time pipeline fully on GPU. Specifically, for deep implicit surface reconstruction, we use TensorRT with mixed precision for fast inference. After that, the major efficiency bottleneck of geometry inference lies in the evaluation of an excessive number of voxels when evaluating every voxel in the volume. Since we already have multi-view depth maps as input, we can leverage the depth infor-

Figure 8: Temporal reconstruction results of a fast dancing girl. Our system can generate temporally-continuous and high-quality reconstruction results under challenging deformations (of the skirt) and severe topological changes (of the hair).

mation directly for acceleration without using the surface localization method in [23]. Specifically, we first use the depth images to filter out empty voxels. Then we follow the octree-based reconstruction algorithm [30] to perform inference for the rest voxels in a coarse-to-fine manner, which starts from a resolution of 64^3 to the final resolution of 256^3. To further improve the run-time efficiency, we simplify the network architectures as follows. For the image encoders, we follow [23] and use HRNetV2-W18-Small-v2 [44] as the backbone, setting its output resolution to 64×64 and channel dimension to 32. For the SDF/color decoders, we use MLPs with skip connections and the hidden neurons as $(128, 128, 128, 128, 128)$. For dynamic sliding fusion, we set $\delta_t = 0.5$, $\delta_e = 0.1$ and $\delta_p = 0.01m$ for all the cases. We refer readers to [19, 16] for real-time implementation details. For multi-view RGBD re-rendering, we render multi-view RGBD images in a single render pass with original color images as textures to improve efficiency. Finally, our system achieves reconstruction at $25fps$ with $21ms$ for dynamic sliding fusion, $17ms$ for deep implicit surface reconstruction (using 3 viewpoints) and $2ms$ for surface extraction using Marching-Cubes Algorithm [28].

6.2. Network Training Details

We use 500 high-quality scans for training *GeoNet* and *ColorNet*, which contains various poses, clothes and human-object interactions. We rotate each scan around the yaw axis, apply random shifts and render 60 views of the scan with image resolution of 512×512. For color image rendering, we use the PRT-based rendering as in [37]. For depth rendering, we first render ground truth depth maps and then synthesis the sensor noises of TOF depth sensors on top of the depth maps according to [11]. Note that we render all the RGBD images using perspective projection to keep consistent with real world sensors. During network training, gradient-based adaptive sampling (in which we use discrete gaussian curvature and rgb gradient as reference for query point sampling in *GeoNet* and *ColorNet* respectively) is used for more effective sampling around detailed regions. We randomly select 3 views from the rendered 60 views of a subject for multi-view training.
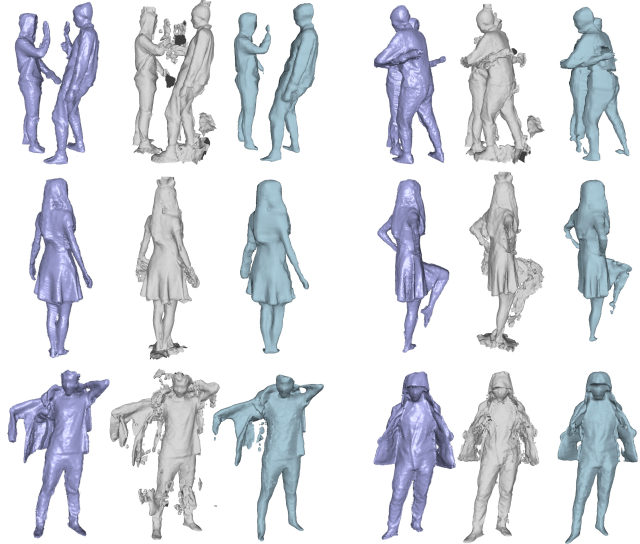

Figure 9: Qualitative comparison. For each subject, from left to right are results of our method, Motion2Fusion [9] and Multi-view PIFu [37], respectively.

Method	P2S$\times10^{-3}\downarrow$	Chamfer$\times10^{-3}\downarrow$	Normal-Consis$\uparrow$
Multi-view PIFU [37]	4.594	4.657	0.862
IPNet [2]	3.935	3.858	0.902
GeoNet	**1.678**	**1.719**	**0.941**

Table 1: Quantitative comparison of geometry reconstruction with multi-view PIFU and IPNet.

6.3. Comparisons

Qualitative Comparison The qualitative comparison with Motion2Fusion [9] and Multi-view PIFu [37] is shown in Fig. 9. Given very sparse and low frame rate depth inputs from consumer RGBD sensors, Motion2Fusion generates noisy results under severe topological changing regions and fast motions due to the deteriorated non-rigid tracking performance. Moreover, the lack of depth information in Multi-view PIFu leads to over-smoothed results.

Quantitative Comparison We compare with 2 state-of-the-art deep implicit surface reconstruction methods, Multi-view PIFu [37](with RGB images as input) and IP-

Method	P2S$\times 10^{-3}\downarrow$	Chamfer$\times 10^{-3}\downarrow$	Normal-Consis$\uparrow$
w/o PSDF w.RGBD	2.36	2.458	0.916
w/o PSDF w.depth-only	2.264	2.359	0.918
w. PSDF w.depth-only	**1.678**	**1.719**	**0.941**

Table 2: Ablation study on the truncated PSDF feature.

Net [2](with voxelized point clouds as input). We retrain their networks using our training dataset and perform the evaluation on a testing dataset that contains 116 high-quality scans with various poses, clothes, and human-object interactions. Tab. 1 shows the quantitative comparison results. We can see that the lack of depth information deteriorates the reconstruction accuracy of Multi-view PIFu. Moreover, even with multi-view depth images as input, the heavy dependency on SMPL initialization (which is difficult to get with large poses and human-object interactions) restricts the IPNet from generating highly accurate results. Finally, by explicitly encoding depth observations using the truncated PSDF values, the proposed *GeoNet* can not only achieves accurate reconstruction results but also orders of magnitude faster than IPNet (approximately 80 seconds for reconstruction). For a detailed description of the comparison, please refer to the supplementary material.

6.4. Ablation Studies

Dynamic Sliding Fusion As shown in Fig. 4(a) and (b), the depth inputs in different views are not consistent with each other due to the challenging hair motion. This results in incomplete results (the orange circle). More importantly, without using dynamic sliding fusion, the result is much noisy (the red circle). By using dynamic sliding fusion, we can get more complete and noise-eliminated reconstruction results as shown in Fig. 4(d). Please refer to the supplementary video for more clear evaluations.

Truncated PSDF Feature The qualitative evaluation of the truncated PSDF feature is shown in Fig.7. Tab. 2 also provides quantitative evaluation results for the networks with and without using truncated PSDF values. We conduct two experiments with RGBD images and depth-mask images as input, respectively. We can see that without using the truncated PSDF feature, the depth-only model and RGBD model produces similar results. Benefiting from the truncated PSDF feature, our GeoNet achieves much accurate results, which demonstrates the effectiveness of our method.

Attention-based Feature Aggregation In Fig. 10, we compare the models without using a multi-view self attention mechanism for color inference qualitatively. Benefiting from the multi-view self-attention mechanism, the color inference results becomes much sharper and plausible especially around observation boundaries. This is because the self-attention enables dynamic feature aggregation rather than the simple average-based feature aggregation, which enforces the MLP-based-decoder to learn how multi-view features (including geometric features and texture features)
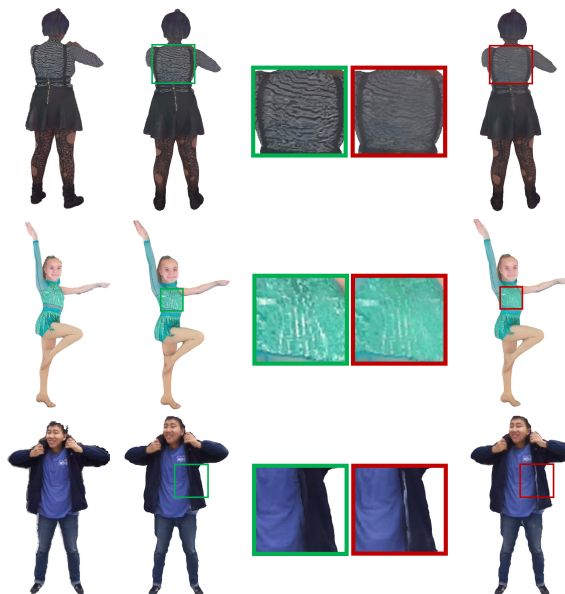

Figure 10: Evaluation of the attention mechanism in *Color-Net*. From left to right are: input RGB images, texture results with (green) and without (red) attention, respectively. are correlated with each other in the 3D space.

7. Conclusion

In this paper, we propose Function4D, a real-time volumetric capture system using very sparse consumer RGBD sensors. By proposing dynamic sliding fusion for topology consistent volumetric fusion and detail-preserving deep implicit functions for high-quality surface reconstruction, our system achieves detailed and temporally-continuous volumetric capture even under various extremely challenging scenarios. We believe that such a light-weight, high-fidelity, and real-time volumetric capture system will enable many applications, especially consumer-level holographic communications, on-line education, and gaming, etc.

Limitations and Future Work Although we can preserve the geometric details in the visible regions, generating accurate and detailed surfaces&textures for the fully occluded regions remains challenging. This is because current deep implicit functions are mainly focus on per-frame independent reconstruction. Expanding deep implicit functions for using temporal observations may resolve this problem in the future. Moreover, specific materials like black hair may cause the lack of observations of depth sensors and therefore severely deteriorate current system, incorporating RGB information for geometry reconstruction may resolve this limitation and we leave this as a future work.

References

[1] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2293–2303, 2019. 3

[2] Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Combining implicit function learning and parametric models for 3d human reconstruction. In *European Conference on Computer Vision (ECCV)*. Springer, August 2020. 3, 5, 7, 8

[3] Aljaž Božič, Michael Zollhöfer, Christian Theobalt, and Matthias Nießner. Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2020. 3

[4] Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. Markerless garment capture. In *ACM SIGGRAPH 2008 papers*, pages 1–9. 2008. 1, 2

[5] Thomas Brox, Bodo Rosenhahn, Juergen Gall, and Daniel Cremers. Combined region and motion-based 3d tracking of rigid and articulated objects. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):402–415, 2009. 1, 2

[6] Rohan Chabra, Jan E. Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 608–625, Cham, 2020. Springer International Publishing. 3

[7] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020. 3, 5

[8] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Trans. Graph*, 34(4):69, 2015. 1, 2

[9] Mingsong Dou, Philip L. Davidson, Sean Ryan Fanello, Sameh Khamis, Adarsh Kowdle, Christoph Rhemann, Vladimir Tankovich, and Shahram Izadi. Motion2fusion: real-time volumetric performance capture. *ACM Trans. Graph.*, 36(6):246:1–246:16, 2017. 1, 3, 5, 7

[10] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (TOG)*, 35(4):1–13, 2016. 1, 3

[11] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, and R. Siegwart. Kinect v2 for mobile robot navigation: Evaluation and modeling. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 388–394, 2015. 7

[12] Valentin Gabeur, Jean-Sébastien Franco, Xavier Martin, Cordelia Schmid, and Gregory Rogez. Moulding humans: Non-parametric 3d human shape estimation from single images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2232–2241, 2019. 3

[13] Juergen Gall, Carsten Stoll, Edilson De Aguiar, Christian Theobalt, Bodo Rosenhahn, and Hans-Peter Seidel. Motion capture using joint skeleton tracking and surface estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1753. IEEE, 2009. 1, 2

[14] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, Danhang Tang, Anastasia Tkach, Adarsh Kowdle, Emily Cooper, Mingsong Dou, Sean Fanello, Graham Fyffe, Christoph Rhemann, Jonathan Taylor, Paul Debevec, and Shahram Izadi. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Trans. Graph.*, 38(6), Nov. 2019. 1, 2

[15] Kaiwen Guo, Feng Xu, Yangang Wang, Yebin Liu, and Qionghai Dai. Robust non-rigid motion tracking and surface reconstruction using l0 regularization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3083–3091, 2015. 1, 2

[16] Kaiwen Guo, Feng Xu, Tao Yu, Xiaoyang Liu, Qionghai Dai, and Yebin Liu. Real-time geometry, albedo and motion reconstruction using a single rgbd camera. *ACM Transactions on Graphics*, 36(3):32:1–32:13, 2017. 7

[17] Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics (TOG)*, 38(2):1–17, 2019. 1, 2

[18] Marc Habermann, Weipeng Xu, Michael Zollhofer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5052–5063, 2020. 1, 2

[19] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568, 2011. 7

[20] Takeo Kanade, Peter Rander, and P. J. Narayanan. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE MultiMedia*, 4(1):34–47, 1997. 2

[21] Chao Li, Zheng Zhang, and Xiaohu Guo. Articulatedfusion: Real-time reconstruction of motion, geometry and segmentation using a single depth camera. In *European Conference on Computer Vision (ECCV)*, pages 324–40, Munich, 2018. SPRINGER. 3

[22] Hao Li, Bart Adams, Leonidas J Guibas, and Mark Pauly. Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics*, 28(5):1–10, 2009. 1, 2

[23] Ruilong Li, Yuliang Xiu, Shunsuke Saito, Zeng Huang, Kyle Olszewski, and Hao Li. Monocular real-time volumetric performance capture. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 5, 6, 7

[24] Zhe Li, Tao Yu, Chuanyu Pan, Zerong Zheng, and Yebin Liu. Robust 3d self-portraits in seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1344–1353, 2020. 5, 6

[25] Yebin Liu, Qionghai Dai, and Wenli Xu. A point-cloud-based multiview stereo algorithm for free-viewpoint video. *IEEE transactions on visualization and computer graphics*, 16(3):407–418, 2009. 1, 2

[26] Yebin Liu, Juergen Gall, Carsten Stoll, Qionghai Dai, Hans-Peter Seidel, and Christian Theobalt. Markerless motion capture of multiple characters using multiview image segmentation. *IEEE T-PAMI*, 35(11):2720–2735, 2013. 2

[27] Yebin Liu, Carsten Stoll, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. Markerless motion capture of interacting characters using multi-view image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1249–1256. IEEE, 2011. 1, 2

[28] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 7

[29] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[30] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 7

[31] A. Mustafa, H. Kim, J-Y. Guillemaut, and A. Hilton. Temporally coherent 4d reconstruction of complex dynamic scenes. In *CVPR*, 2016. 2

[32] Ryota Natsume, Shunsuke Saito, Zeng Huang, Weikai Chen, Chongyang Ma, Hao Li, and Shigeo Morishima. Siclope: Silhouette-based clothed people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4480–4490, 2019. 1, 3

[33] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352, 2015. 1, 2, 4

[34] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[35] Songyou Peng and Michael Niemeyer *et,al*. Convolutional occupancy networks. In *ECCV*, 2020. 3

[36] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. Clothcap: Seamless 4d clothing capture and retargeting. *ACM Transactions on Graphics (TOG)*, 36(4):1–15, 2017. 1, 2

[37] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitiza-

tion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2304–2314, 2019. 1, 3, 5, 6, 7

[38] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2020. 1, 3

[39] Miroslava Slavcheva, Maximilian Baust, Daniel Cremers, and Slobodan Ilic. Killingfusion: Non-rigid 3d reconstruction without correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1395, 2017. 3

[40] Miroslava Slavcheva, Maximilian Baust, and Slobodan Ilic. Sobolevfusion: 3d reconstruction of scenes undergoing free non-rigid motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2646–2655, Salt Lake City, June 2018. IEEE. 3

[41] Jonathan Starck and Adrian Hilton. Surface capture for performance-based animation. *IEEE Computer Graphics and Applications*, 27(3):21–31, 2007. 2

[42] Zhuo Su, Lan Xu, Zerong Zheng, Tao Yu, Yebin Liu, and Lu Fang. Robustfusion: human volumetric capture with data-driven visual cues using a rgbd camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 3

[43] Robert W. Sumner, Johannes Schmid, and Mark Pauly. Embedded deformation for shape manipulation. *ACM Transactions on Graphics*, 26(3), July 2007. 2, 4

[44] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 7

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc., 2017. 6

[46] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popovic. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph.*, 27(3):97:1–97:9, 2008. 2

[47] Daniel Vlasic, Pieter Peers, Ilya Baran, Paul E. Debevec, Jovan Popovic, Szymon Rusinkiewicz, and Wojciech Matusik. Dynamic shape capture using multi-view photometric stereo. *ACM Trans. Graph.*, 28(5):174:1–174:11, 2009. 1, 2

[48] Michael Waschbüsch, Stephan Würmlin, Daniel Cotting, Filip Sadlo, and Markus H. Gross. Scalable 3d video of dynamic scenes. *The Visual Computer*, 21(8-10):629–638, 2005. 2

[49] Chenglei Wu, Kiran Varanasi, Yebin Liu, Hans-Peter Seidel, and Christian Theobalt. Shading-based dynamic shape refinement from multi-view video under general illumination. In *IEEE ICCV*, pages 1108–1115, 2011. 2

[50] Minye Wu, Yuehao Wang, Qiang Hu, and Jingyi Yu. Multi-view neural human rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[51] Lan Xu, Zhuo Su, Lei Han, Tao Yu, Yebin Liu, and Lu Fang. Unstructuredfusion: Realtime 4d geometry and texture reconstruction using commercial RGBD cameras. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2508–2522, 2020. 1

[52] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018. 2

[53] Tao Yu, Kaiwen Guo, Feng Xu, Yuan Dong, Zhaoqi Su, Jianhui Zhao, Jianguo Li, Qionghai Dai, and Yebin Liu. Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera. In *IEEE International Conference on Computer Vision (ICCV)*, pages 910–919, Venice, 2017. IEEE. 1, 3

[54] Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 7287–7296, Salt Lake City, June 2018. IEEE. 1, 3

[55] Zerong Zheng, Tao Yu, Hao Li, Kaiwen Guo, Qionghai Dai, Lu Fang, and Yebin Liu. Hybridfusion: Real-time performance capture using a single depth sensor and sparse imus. In *European Conference on Computer Vision (ECCV)*, pages 389–406, Munich, Sept 2018. SPRINGER. 3

[56] Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *arXiv*, 2020. 3

[57] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7739–7749, 2019. 3

[58] Hao Zhu, Xinxin Zuo, Sen Wang, Xun Cao, and Ruigang Yang. Detailed human shape estimation from a single image by hierarchical mesh deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4491–4500, 2019. 3

[59] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (ToG)*, 33(4):1–12, 2014. 1, 2