

High-Resolution Volumetric Reconstruction for Clothed Humans

SICONG TANG*, Simon Fraser University, Canada

GUANGYUAN WANG*, Alibaba Group, China

QING RAN, Alibaba Group, China

LINGZHI LI, Alibaba Group, China

LI SHEN, Alibaba Group, China

PING TAN, Simon Fraser University, Canada

We present a novel method for reconstructing clothed humans from a sparse set of, e.g., 1–6 RGB images. Despite impressive results from recent works employing deep implicit representation, we revisit the volumetric approach and demonstrate that better performance can be achieved with proper system design. The volumetric representation offers significant advantages in leveraging 3D spatial context through 3D convolutions, and the notorious quantization error is largely negligible with a reasonably large yet affordable volume resolution, e.g., 512. To handle memory and computation costs, we propose a sophisticated coarse-to-fine strategy with voxel culling and subspace sparse convolution. Our method starts with a discretized visual hull to compute a coarse shape and then focuses on a narrow band nearby the coarse shape for refinement. Once the shape is reconstructed, we adopt an image-based rendering approach, which computes the colors of surface points by blending input images with learned weights. Extensive experimental results show that our method significantly reduces the mean point-to-surface (P2S) precision of state-of-the-art methods by more than 50% to achieve approximately 2mm accuracy with a 512 volume resolution. Additionally, images rendered from our textured model achieve a higher peak signal-to-noise ratio (PSNR) compared to state-of-the-art methods.

CCS Concepts: • **Computing methodologies** → **Mesh models; Volumetric models; Reconstruction.**

Additional Key Words and Phrases: Clothed Human, 3D Reconstruction, Holoportation

ACM Reference Format:

Sicong Tang, Guangyuan Wang, Qing Ran, Lingzhi Li, Li Shen, and Ping Tan. 2018. High-Resolution Volumetric Reconstruction for Clothed Humans. *ACM Trans. Graph.* 37, 4, Article 111 (August 2018), 15 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 INTRODUCTION

Automatic 3D reconstruction of clothed humans using image inputs has gained increasing significance due to its potential applications in

*Both authors contributed equally to this research.

Authors' addresses: Sicong Tang, sta105@sfu.ca, Simon Fraser University, Vancouver, Canada; Guangyuan Wang, yixuan.wgy@alibaba-inc.com, Alibaba Group, Hang Zhou, China; Qing Ran, ranqing.rq@alibaba-inc.com, Alibaba Group, Hang Zhou, China; Lingzhi Li, llz273714@alibaba-inc.com, Alibaba Group, Hang Zhou, China; Li Shen, lshen.lsh@gmail.com, Alibaba Group, Hang Zhou, China; Ping Tan, Simon Fraser University, Vancouver, Canada, pingtan@sfu.ca.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Association for Computing Machinery.

0730-0301/2018/8-ART111 \$15.00

<https://doi.org/XXXXXXX.XXXXXXX>

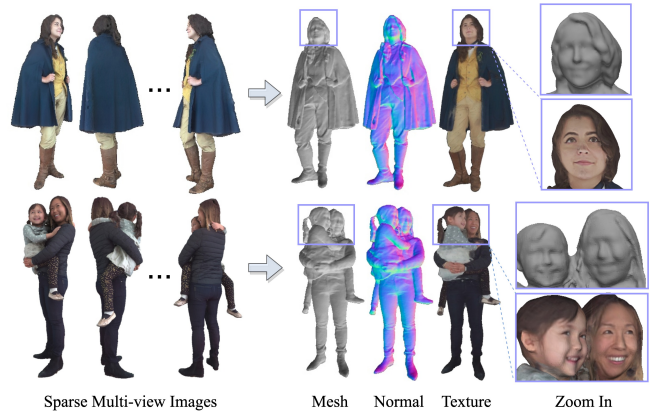


Fig. 1. Our method reconstructs a textured 3D model of clothed humans from sparse multi-view images. It recovers detailed geometry with vivid texture, despite complexities caused by garments, poses, and occlusions.

a wide array of AR/VR scenarios. High-fidelity reconstructions typically depend on sophisticated capture systems, which are developed with dense camera arrays [Collet et al. 2015; Joo et al. 2019, 2018], programmable light-stages [Guo et al. 2019; Vlastic et al. 2009], and depth sensors [Dou et al. 2016; Newcombe et al. 2015, 2011; Yu et al. 2017, 2020]. However, stringent capture environments equipped with complex hardware pose significant challenges for consumer-level applications.

In this context, considerable research effort has been dedicated to developing methods that allow for more flexible capture configurations, such as utilizing a few RGB inputs. Among these works, learning implicit functions [Hong et al. 2021; Saito et al. 2019, 2020] has proven effective in achieving highly detailed reconstructions by integrating the advancements of deep neural networks. These methods employ large multi-layer perceptrons (MLPs) to predict the occupancy probability or truncated signed distance function (TSDF) value of every queried 3D point based on its associated local feature, which is extracted from images. They can recover a continuous surface at arbitrary resolutions without topology restrictions.

However, in typical MLP-based implicit networks, the occupancy or TSDF value at each location is solved independently with planar image features, rendering them less capable of addressing challenging cases such as occlusions. Consequently, these methods suffer from generalization and robustness issues, particularly when tackling strong occlusions caused by large motion or multiple interacting humans. Some follow-up studies [Huang et al. 2020; Zheng et al. 2021, 2022] utilize an extra geometric model, SMPL [Loper et al. 2015], to improve robustness by introducing strong shape priors.

Their success typically relies on the assumption of geometrical similarity [Huang et al. 2020] between the shape prior and target reconstruction, making them intractable for handling complex cases with loose clothes and sensitive to errors in SMPL model fitting.

We instead revisit the 3D volumetric representation and resort to 3D convolutional neural networks (CNNs) for feature learning, due to their impressive performance in feature learning and the ability to incorporate spatial context. However, volumetric methods and 3D convolution involve discretization, which might raise concerns regarding whether a discretized volume can preserve subtle geometric details as continuous representations learned in implicit functions. We investigate the relationship between volume resolution and quantization error on synthetic data by converting target mesh objects to TSDF volumes, as shown in Figure 3. We observe that the quantization errors are significantly reduced by increasing volume resolution and become nearly negligible when reaching a relatively high resolution (e.g., 512 or higher). In other words, achieving fine-detailed reconstruction is not supposed to be restricted by the use of volume representations as long as a proper volume resolution is utilized. Therefore, we present a method with high-resolution feature volumes, e.g., 256 and 512, while traditional volumetric methods [Gilbert et al. 2018; Varol et al. 2018] are often limited to much lower resolutions, such as 32 or 128.

On the other hand, an increase in volume resolution may lead to a cubic growth of memory overhead [Ge et al. 2017]. Reducing memory costs while guaranteeing the granularity of volumetric representations is necessary for pursuing high-quality reconstruction. Thus, we adopt a coarse-to-fine approach and cull away irrelevant voxels to build a sparse high-resolution feature volume. At the coarse level, the network computes an initial TSDF by applying a U-Net with sparse 3D CNN [Graham et al. 2018] on the sparse feature volume, which is carved by a visual hull. Through our experiments, it turns out that more than 95% of the volume grids are discarded by the visual hull culling, making the sparse 3D CNN efficient. At the fine level, the network focuses on a narrow band near the zero-level set of the initial TSDF and discretizes the narrow band with smaller voxels. By employing this narrow-band culling, we further shrink the sampling space, resulting in a relatively small range of grid numbers (usually 300K–500K in our experiments) even with a high volume resolution of 512. The remaining voxels in the narrow band are associated with features that fuse high-frequency information from the computed normal maps upon the low-frequency shape from the coarse level to compute the TSDF at high resolution. The final mesh is then extracted from the TSDF using the Marching-Cube algorithm [Lorensen and Cline 1987].

In addition to geometry, high-quality mesh texture is also a crucial factor contributing to visual appearance. Directly computing a color field in 3D space, as in [Saito et al. 2019], struggles to capture high-frequency texture details, while the neural radiance field (NeRF) [Yu et al. 2021a] or the DoubleField [Shao et al. 2022a] require expensive per-instance optimization and are often unstable for sparse input images. In contrast, we adopt an image-based rendering approach to compute a texture atlas map, which is efficient and widely supported in existing computer graphics tools. Specifically, we compute a blending weight at each 3D point on the mesh surface to determine its color as a weighted average of the colors

at its image projections. The blending weights can be computed at a relatively coarse resolution, e.g., 512 volume resolution in our case, and leave texture details to the high-resolution images, such as 1K or 2K. Unlike previous methods that generate blurry texturing results under sparse input, our method generalizes well on both synthetic and real data with just a few input views. Figure 1 shows two examples reconstructed by our method. Despite the challenging garment, pose, and occlusion, our method recovers faithful shape, normal, and texture on the right.

In summary, the main contributions of this paper are as follows:

- We revisit the 3D volumetric representation and demonstrate that it can support clothed human reconstruction with equal or even better performance compared to implicit representation.
- We develop a memory and computation-efficient method for high-resolution volumetric reconstruction using sophisticated sparse 3D CNN, coarse-to-fine estimation, and voxel culling by visual hull and narrow bands.
- We introduce a novel method to compute a texture atlas map, which captures rich appearance details from high-resolution input images.
- We achieve impressive results on standard benchmark datasets Twindom and MultiHuman, significantly reducing the point-to-surface (P2S) precision to approximately 0.2cm from just six input views, with more than 50% error reduction compared to the state-of-the-art methods, including DoubleField [Shao et al. 2022a] and PIFuHD [Saito et al. 2020].

2 RELATED WORK

Parametric Model Based Methods. Parametric human models such as SCAPE [Anguelov et al. 2005], SMPL [Loper et al. 2015], and SMPL-X [Pavlakos et al. 2019] have been widely adopted to recover human pose and shapes. SMPLify [Bogo et al. 2016] estimates the SMPL model from a single image using 2D keypoint detection. Recently, deep neural networks [Kanazawa et al. 2018; Omran et al. 2018; Pavlakos et al. 2018; Xu et al. 2019] have been trained to directly regress SMPL model parameters from a single image. The accuracy is further improved by combining bottom-up optimization [Güler and Kokkinos 2019; Kolotouros et al. 2019] and using temporal generation networks [Kocabas et al. 2020]. However, the SMPL estimation from a single image suffers from shape ambiguity. Huang et al. [2017] and Liang et al. [2019] generalize SMPL model fitting to multiple input images. To capture cloth shape details, methods like [Alldieck et al. 2019; Bhatnagar et al. 2019] use SMPL+D representation to explain high-frequency details. However, this representation struggles to handle loose clothes and long hair. In contrast to these methods, we aim to reconstruct a clothed human without relying on any parametric model to achieve better generalization to different poses and garments.

Implicit Function Based Methods. Implicit functions [Chen and Zhang 2019; Mescheder et al. 2019; Park et al. 2019] provide a powerful shape representation for 3D reconstruction, enabling surface reconstruction at arbitrary resolutions and topologies. Many recent works utilize implicit functions to reconstruct clothed humans. Huang et al. [2018] and PIFu [Saito et al. 2019] introduce implicit functions for clothed human reconstruction, where a 3D

point is projected onto the input images to gather features for occupancy regression. PIFuHD [Saito et al. 2020] improves PIFu by extracting high-resolution features from estimated normal maps. PHORHUM [Alldieck et al. 2022] also predicts shading parameters to generate more realistic rendering results. MonoPort [Li et al. 2020] accelerates the occupancy evaluation in a coarse-to-fine manner using Octree structures.

However, the aforementioned methods compute the occupancy or TSDF of a 3D point using point-wise inference and planar image features, which are limited in exploring 3D context information, especially for self-occlusions and challenging poses. As a result, Arch [Huang et al. 2020], Arch++ [He et al. 2021], PaMIR [Zheng et al. 2022], and ICON [Xiu et al. 2022] employ the SMPL model as a shape prior to improve robustness to different body poses. DeepMultiCap [Zheng et al. 2021] further employs a spatial attention network and a temporal fusion method for multi-view input videos. However, the SMPL model estimation itself is fragile, StereoPIFu [Hong et al. 2021] takes stereo images as input to exploit the geometric constraints of stereo vision. The recent work, DoubleField [Shao et al. 2022a], combines the neural radiance field with an implicit surface field to generate high-quality results. DiffuStereo [Shao et al. 2022b] further introduces a diffusion-based stereo algorithm to enhance shape accuracy by enforcing multi-view correspondences.

As observed in [Chibane et al. 2020; Peng et al. 2020], implicit function based 3D reconstruction can benefit from a 3D convolutional feature encoder. While these two methods are designed for 3D reconstruction from point clouds or sparse voxel inputs, we extend a similar idea for image-based reconstruction of clothed humans. 3D convolutions can easily encode geometric contexts and compute the TSDF values at nearby points jointly. However, it requires a high-resolution feature volume to capture shape details.

Volumetric Methods. There are relatively fewer works adopting volumetric representation for clothed human reconstruction, as it is known to have expensive memory and running time costs. To reduce memory and computation costs, earlier works [Jackson et al. 2018; Varol et al. 2018] employ 2D convolutions to regress the occupancy volume from a single RGB image, but only recover limited shape details and suffer from challenging poses. DeepHuman [Zheng et al. 2019] employs 3D convolution and uses SMPL models as shape priors to guide the volume regression. However, like parametric model-based methods, its SMPL estimation tends to fail at challenging poses and loose garments. Gilbert et al. [2018] use multi-view images to recover a visual hull, and then apply 3D CNN to compute the occupancy values at the discretized visual hull. However, they do not involve any image features in the 3D CNN, which is crucial to recover shape details, and only generate over-smoothed results. Similar 3D convolution is also applied to the discretized SMPL model in [Zheng et al. 2021, 2022] to facilitate learning implicit functions, but image features are not involved in the 3D convolution. Furthermore, most previous methods [Jackson et al. 2018; Varol et al. 2018; Zheng et al. 2021, 2022, 2019] use low volume resolution of 128, except the method in [Gilbert et al. 2018] uses 256 volume resolution without including image features.

We find it is important to use high-resolution volumes, e.g. 512 or higher, for accurate 3D reconstruction of clothed humans. Furthermore, it is important to include image features in the 3D convolution

to reconstruct shape details. To make these ideas feasible, we design a sophisticated volumetric method by combining efficient sparse 3D convolution, voxel culling, and coarse-to-fine computation.

Surface Color Prediction. Traditional methods like [Waechter et al. 2014] color surface points according to images with front parallel viewing directions. For human modeling tasks, PIFu [Saito et al. 2019] and its follow-up works [Yu et al. 2021b; Zheng et al. 2021] often use an additional implicit function to compute a continuous color field, where each 3D point is associated with a color. Implicit functions can hallucinate colors in unobserved regions. However, it is also difficult to capture appearance details like high-frequency textures due to the compact representation. Recently, neural radiance field (NeRF) [Mildenhall et al. 2020] has shown its great potential of generating high quality view synthesis. NeuralBody [Peng et al. 2021] further introduces a SMPL model to aggregate color constraints in the canonical frame, extending the NeRF-based human reconstruction to sparse multi-view inputs. To capture 3D shape details while generating realistic rendering, Doublefield [Shao et al. 2022a] combines the advantages of implicit surface field [Saito et al. 2019] and neural radiance field [Mildenhall et al. 2020], which further speedup the convergence of NeRF models. While these methods generate high-quality results, NeRF-based methods still require expensive per-instance optimization, which is undesirable in many real applications. To address these problems, we follow the spirit of traditional methods to compute a texture map on the mesh surface. Instead of computing the texture color directly, we design a network to estimate a blending weight to evaluate the color according to the input images. In this way, our texture map can easily inherit high-frequency details from high-resolution input images.

3 MOTIVATION

Given sparse view RGB images $\{I_i\}$ capturing a clothed human and their calibrations, our goal is to estimate the truncated signed distance function (TSDF) \mathbb{D} which describes the clothed human shape. This TSDF \mathbb{D} might be directly discretized as a 3D volume, where each voxel grid stores the TSDF value. In contrast, recent learning based reconstruction methods [Chen and Zhang 2019; Mescheder et al. 2019; Park et al. 2019] employ a multi-layer perceptron (MLP) as an implicit representation of the TSDF or occupancy function, which is continuous and free from resolution and topology limitations. Following this idea, PIFu [Saito et al. 2019] computes the occupancy function with pixel-aligned features as,

$$\mathbb{D}(\mathbf{X}) = f(\mathbf{X}, \mathbf{F}^{2D}(\Pi(\mathbf{X}))) = s, \quad s \in [-1, 1], \quad (1)$$

where f is an MLP, \mathbf{X} is a 3D point, and $\Pi(\cdot)$ projects 3D points to the input image. The feature map \mathbf{F}^{2D} is computed from the input image with 2D convolutions.

On the other hand, the simple fully-connected network architecture of MLPs is inefficient in integrating context information as studied in [Chibane et al. 2020; Peng et al. 2020]. These studies suggest combining a 3D convolutional encoder with an MLP decoder for 3D reconstruction from point clouds. In the same spirit, we might solve the TSDF \mathbb{D} as

$$\mathbb{D}(\mathbf{X}) = f(\mathbf{X}, \mathbf{F}^{3D}(\mathbf{X})) = s, \quad s \in [-1, 1], \quad (2)$$

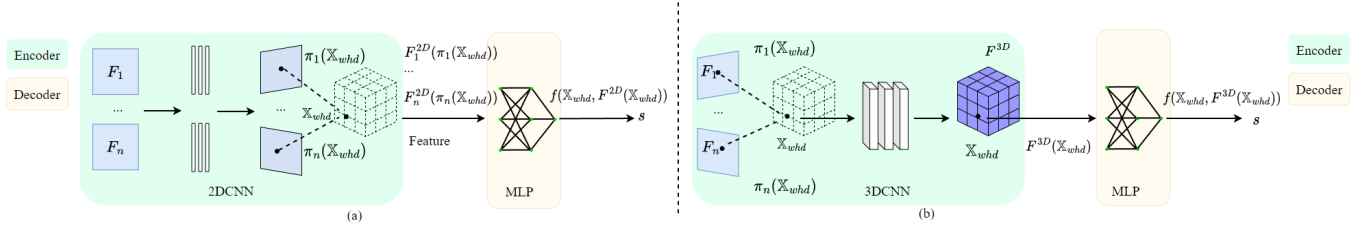


Fig. 2. The network architecture of the two toy networks in Section 3.1. (a) features are encoded in the 2D image plane, similar to the multi-plane encoder proposed in [Peng et al. 2020]. (b) features are encoded in the 3D volume.

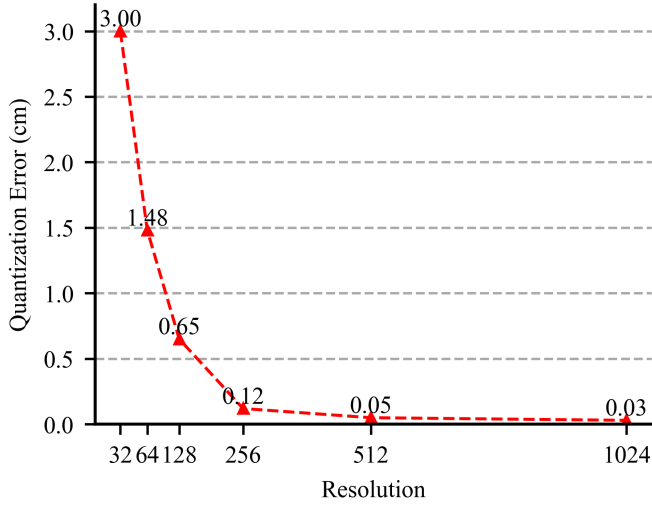


Fig. 3. Quantization error of TSDF volume at different resolutions. The quantization error becomes negligible for volume resolutions beyond 512.

	2D Features	3D Features
Chamfer/P2S	0.601/0.549	0.404/0.358

Table 1. Chamfer and P2S precision errors of the two toy networks with 2D or 3D features tested on the Twindom[Twindom [n. d.]] dataset.

where the feature volume F^{3D} is computed in 3D by a 3D convolutional encoder. As reported in [Peng et al. 2020], learning this 3D feature volume is superior to its counterpart 2D feature maps (i.e. the single-plane or multi-plane feature encoder), which are commonly employed in earlier clothed human reconstruction methods including PIFu[Saito et al. 2019], PaMIR[Zheng et al. 2022], and DeepMultiCap[Zheng et al. 2021].

3.1 3D Feature Volume

To demonstrate the strength of 3D convolution in Equation 2, we experiment with a toy network by directly appending a 3D convolutional feature encoder with an MLP decoder for clothed human reconstruction. Specifically, as in Figure 2 (b), we sample a set of regular volume grid $\{\mathbb{X}_{whd}\}$ in 3D space, where $\{w, h, d\}$ are the grid indices, and associate each grid vertex with the image features at its projected image positions. We then apply 5 layers of 3D convolutions with filter size $3 \times 3 \times 3$ to the feature volume, and use an MLP to decode the TSDF value at each sampled grid vertex from its feature. In this way, we can compute the TSDF values at all the grid

	64F5M	128F5M	256F5M	256F5M*	256F1M*	256F'5M
Chamfer	0.574	0.459	0.406	0.404	0.432	0.592
P2S	0.524	0.395	0.332	0.358	0.365	0.538

Table 2. Results of the toy network using various network settings. In each column, the F-number represents the volume resolution, while the M-number denotes the depth of the MLP decoder. M* signifies a discrete MLP which evaluates TSDF values only on the grid vertices, and F' indicates convolution is not applied to the 3D features.

vertices $\{\mathbb{X}_{whd}\}$ as,

$$\mathbb{D}(\mathbb{X}_{whd}) = f(\mathbb{X}_{whd}, \mathbf{F}^{3D}(\mathbb{X}_{whd})) = s, \quad s \in [-1, 1]. \quad (3)$$

Note that, Equation 3 is a discretized version of Equation 2 and only computes TSDF values for the pre-sampled volume grids. As we discuss in the supplementary file, we empirically find this discrete approach is close to the original continuous version with high-resolution feature volumes. The MLP $f(\cdot)$ can also be heavily simplified to just one layer in our experiments.

Alternatively, as shown in Figure 2 (a), we might use the 2D image features directly as input to the MLP to decode the TSDF values at grid vertices $\{\mathbb{X}_{whd}\}$. To ensure a fair comparison, we employ additional 3×3 2D convolutions in the image space to make the number of learnable parameters similar.

We train these two toy networks using pre-sampled volume grids at 256 resolution on the Twindom [Twindom [n. d.]] dataset, and evaluate them on the 160 testing human models. Table 1 shows the Chamfer distance and point-2-surface errors of both methods. It becomes evident that learning a 3D feature volume with 3D convolutions leads to more accurate reconstructions¹ since the 3D CNNs can better leverage context information.

3.2 Network Settings

Our formulation in Equation 3 includes a convolutional encoder and an MLP decoder, similar to the hybrid representation in [Chibane et al. 2020; Peng et al. 2020]. In this subsection, we explore variations in the network settings, including feature volume resolution, discrete versus continuous MLP, and the depth of MLP, to understand their impact on shape reconstruction results. We first test our toy network with 3D features using different volume resolutions. In these experiments, we choose to learn a continuous surface represented by the MLP $f(\cdot)$, as in PIFu [Saito et al. 2019]. Specifically, we randomly sample 3D points and trilinearly interpolate features

¹Note that PIFu [Saito et al. 2019] reconstructs a continuous surface which is not limited to the pre-sampled grid vertices $\{\mathbb{X}_{whd}\}$. In the same experiment, its Chamfer distance error is 0.592 and P2S error is 0.538, which are slightly better than our discretized version with 2D features, but inferior to the version employing 3D features.

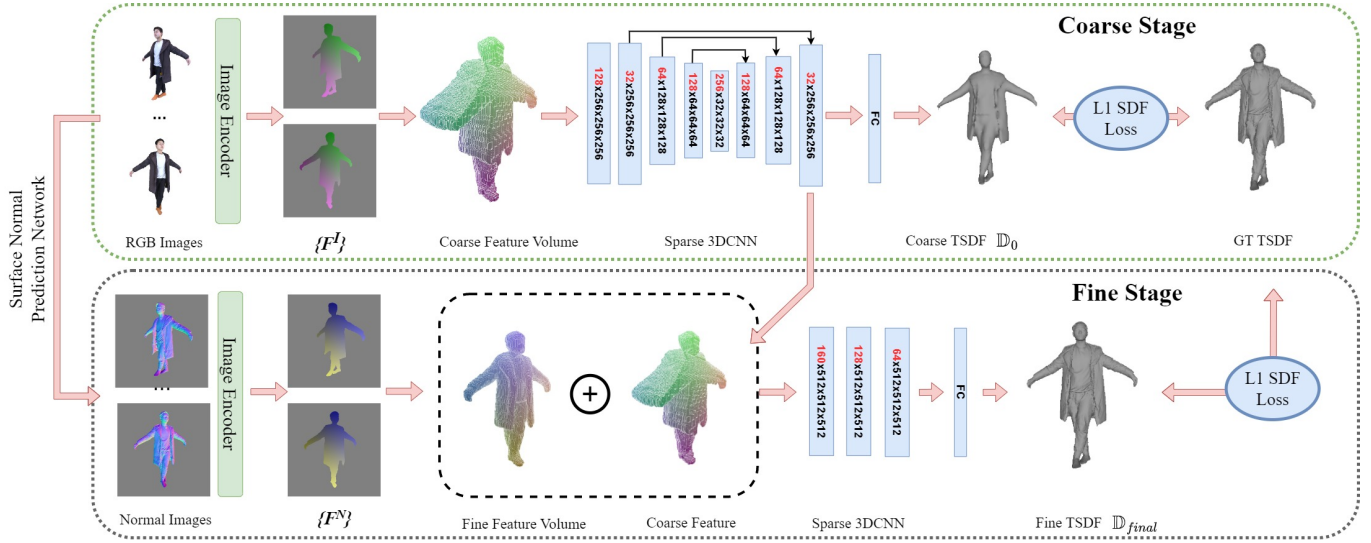


Fig. 4. Pipeline of our shape reconstruction: given sparse multi-view images, our method works in a coarse to fine manner to compute the TSDF volume of the clothed human. During the coarse stage, we gather features from the input images for voxels within the visual hull and compute a coarse TSDF \mathbb{D}_0 by a sparse 3D CNN. During the fine stage, we discretize a narrow band nearby the surface computed at the coarse level, and collect normal features to compute a fine TSDF \mathbb{D}_{final} using another sparse 3D CNN.

at these sampled points from the discrete grid vertices $\{\mathbb{X}_{whd}\}$, and then use the MLP $f(\cdot)$ to compute the TSDF value. Table 2 summarizes the results of various settings, where the F-number and M-number in each column represent the volume resolution and the MLP depth, respectively. From the left three columns, it is evident that increasing the volume resolution from 64 to 256 can significantly reduce reconstruction errors by about 30%, indicating that a high-resolution feature volume is crucial for precise results.

We further test other network settings. The two columns with an M* indicate results with a discrete MLP, which computes TSDF results only on the grid vertices $\{\mathbb{X}_{whd}\}$. From these two columns, it is apparent that the discrete MLP only slightly compromises result quality, and even a 1-layer discrete MLP can achieve satisfactory results. The rightmost column with an F* represents results where convolution is not applied to the 3D feature volume. In this scenario, even a high-resolution 3D feature volume produces a substantial error, which clearly highlights the effectiveness of 3D convolution.

3.3 Quantization Error

The formulation in Equation 3 involves discretization, which is often undesirable and motivates implicit function representation [Chen and Zhang 2019; Mescheder et al. 2019; Park et al. 2019]. In the following, we analyze the quantization error of the TSDF \mathbb{D} with different volume resolutions. Surprisingly, we find that with relatively high volume resolution, e.g. 512 or higher, the quantization error is no longer a limiting factor for the reconstruction accuracy of clothed humans. Specifically, we compute ground truth TSDFs according to the ground truth mesh models in the THuman2.0 [Zheng et al. 2021] dataset. We then discretize those TSDFs into volumes of different resolutions, ranging from $32 \times 32 \times 32$ to $1024 \times 1024 \times 1024$. The quantization error is measured by the average error in TSDF values

on the ground truth mesh surface. As shown in Figure 3, the quantization error drops quickly with higher volume resolution. When the volume resolution is 512 or higher, the quantization error is less than 0.05 cm, much smaller than the reconstruction error of SOTA methods [Saito et al. 2019; Zheng et al. 2021, 2022], which typically exceeds 0.5 cm. This indicates that discrete TSDF representation is not a limiting factor for clothed human reconstruction with a volume resolution of 512 or higher.

4 METHOD

As discussed in Section 3, a convolutional encoder with 3D feature volume can boost shape reconstruction accuracy. While discretization causes additional quantization errors, a high-resolution volume can effectively mitigate this issue. Therefore, it is important to design an efficient method to overcome the memory and computation costs associated with high-resolution volumes for improved results. For this purpose, we design a sophisticated voxel culling process, implement a coarse-to-fine strategy, and employ the efficient sub-manifold sparse convolutional networks [Graham et al. 2018].

Figure 4 shows our system pipeline for shape reconstruction. Our method works in two stages from coarse to fine. In the coarse stage, we adopt a $256 \times 256 \times 256$ resolution volume and employ the visual hull of the foreground object to eliminate irrelevant voxels. The remaining voxel grids are associated with image features at their projected positions. We then apply the efficient subspace sparse 3D CNN [Graham et al. 2018] to compute an initial TSDF, \mathbb{D}_0 . Unlike conventional 3D CNN, sparse 3D CNN builds a hash-table for indexing non-zero elements and the convolution operator only applies to those non-zero elements, which makes the convolution computational and memory efficient when the input tensor is sparse. In the fine stage, we focus on the narrow band nearby the zero-level set of \mathbb{D}_0 and discretize that narrow band into smaller voxels of

512 × 512 × 512 resolution. Each voxel grid is then associated with features from normal maps computed from the input images by the method [Newell et al. 2016]. We further fuse a coarse geometry feature from the coarse level, and apply the sparse 3D CNN again on the fine volume to compute the final TSDF, $\mathbb{D}_{\text{final}}$. The visual hull culling and narrow-band culling substantially reduce the sampling grids, making the feature volume sparse enough for efficient sparse convolution.

After reconstructing the 3D shape, we proceed to estimate the surface texture. Texture maps need even higher resolution to capture appearance details. To address this problem, instead of naïvely applying our TSDF regression network to compute a color volume that evaluates color as a function of coordinates, we choose to solve a field of blending weights. At each 3D point, the surface color is the weighted average of the colors at its image projections. We only solve the blending weights for a narrow band nearby the final surface $\mathbb{D}_{\text{final}}$ for better efficiency. Our pipeline to solve this blending weight volume is shown in Figure 6.

4.1 Feature Volume Construction

To construct the feature volume, we initial a cubic volume grid \mathbb{V} with an edge length of 256cm and a resolution of 256 × 256 × 256. We then project each voxel grid point \mathbb{V}_{whd} on the input mask images $\{M_i\}$ to discard points outside of the visual hull. Pruning by the visual hull significantly reduce the number of ‘active’ voxel vertices in the volume. Typically, over 95% of the voxel grids are culled away by the visual hull, leaving around 200–300K voxels remaining. Given the set of remaining voxel grids $\{\mathbb{V}_{whd}\}$, we project them onto feature maps to compute a feature on each grid vertex as follows,

$$\mathbb{F}_{whd}^I = \text{Mean}(F_i^I(\Pi_i(\mathbb{V}_{whd}))). \quad (4)$$

Here, Π_i is the perspective projection of the input image I_i . We sample the 2D feature maps F_i^I using bi-linear interpolation at the projected positions, and average the sampled features from all views to compute the feature volume \mathbb{F}^I . The image feature F_i^I is computed from the input image I_i by a single stacked hourglass network [Newell et al. 2016], which has 128 feature channels and at resolution of 256 × 256.

4.2 Coarse to Fine Reconstruction

To ensure the memory consumption and inference speed, we take a coarse-to-fine architecture to compute the TSDF \mathbb{D} .

Coarse stage. We use a 3D U-Net with skip layers to encode the topology context. As shown in Figure 4 the 3D U-Net consists of three conv and deconv blocks with skip connections. The initial TSDF \mathbb{D}_0 at each volume grid is computed by an FC layer, i.e. the MLP $f(\cdot)$ in Equation 3. More network details are provided in the supplementary file. We have experimented with more FC layers and empirically found that adding more layers does not help, thanks to the 3D convolutional feature encoder with proper local information encoding. This network outputs a coarse TSDF \mathbb{D}_0 with the same resolution as the feature volume \mathbb{F}^I . During training, we calculate the ground truth TSDF for each clothed human model from the ground truth mesh model. We further truncate the TSDF value within [-5cm,5cm]. The training loss function for the coarse stage

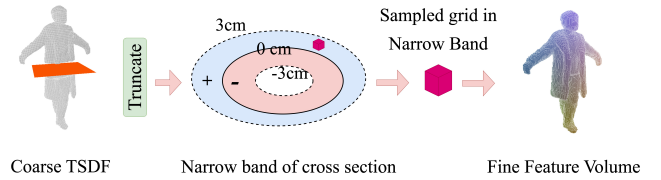


Fig. 5. At the fine stage, we focus on a narrow band nearby the coarse stage result and discretize it to voxels. We then project each voxel vertex to the normal feature map to form the fine feature volume \mathbb{F}^f .

is then defined as:

$$L_c = \sum_{(w,h,d) \in \mathcal{V}_{\text{hull}}} \left\| (\mathbb{D}_{whd}^{\text{pred}} + \text{bias}_c) - \mathbb{D}_{whd}^{\text{gt}} \right\|_{L_1} \quad (5)$$

Here, \mathbb{D}^{pred} and \mathbb{D}^{gt} are the predicted and ground truth TSDF volume respectively, and $\mathcal{V}_{\text{hull}}$ is the set of remaining voxel grids after visual hull culling. The training loss is the L1 distance between the predicted and ground truth TSDF values. Note that the predicted TSDF values of voxel grids outside the visual hull will be zero according to the submanifold sparse convolution (SSC). Hence, we add a constant bias $\text{bias}_c = 0.05\text{m}$ which is the truncation distance to the TSDF.

Fine stage. After getting the coarse TSDF \mathbb{D}_0 , we use another branch to further refine geometry details. At this stage, we down-sample the volume to denser voxels and associate each voxel vertex with high frequency shape information encoded in normal maps like [Saito et al. 2020; Zheng et al. 2021]. To facilitate computation, we focus on a narrow band nearby the zero-level set of \mathbb{D}_0 . Specifically, we tri-linearly interpolate the coarse TSDF volume \mathbb{D}_0 by 2 times to a 512 × 512 × 512 voxel grid. We only preserve all the voxel vertices satisfying: $|\mathbb{D}_0| < 0.03\text{m}$, which are within a narrow band of 6cm width around the zero-level set surface of \mathbb{D}_0 . Figure 5 shows the narrow band for re-sampling. This narrow band removes irrelevant voxel vertices based on the initial result, helping to further improve the storage and computation efficiency of our method.

We compute a feature at each voxel vertex in the fine stage from the normal feature maps as follows,

$$\mathbb{F}_{whd}^N = \text{Mean}(F_i^N(\Pi_i(\mathbb{V}_{whd}))). \quad (6)$$

We use the same normal estimator proposed in [Zheng et al. 2021] to estimate the normal image N_i for each input view I_i , and the normal feature map F_i^N is computed from the input normal image N_i by the same hourglass network as F_i^I . Furthermore, this feature is concatenated with the down-sampled coarse level features \mathbb{F}_{whd}^I at the last two convolution layers, which have large receptive fields and encode strong shape information. Since the fine stage mainly focuses on local shape details, we use a shallow sparse 3D CNN which has 3 Conv blocks followed by an FC layer to regress the final TSDF volume. More details of network architecture are in the supplementary file.

We define the training loss for the fine stage as follows,

$$L_f = \sum_{(w,h,d) \in \mathcal{V}_{\text{band}}} \left\| (\mathbb{D}_{whd}^{\text{pred}} + \text{bias}_f) - \mathbb{D}_{whd}^{\text{gt}} \right\|_{L_1}. \quad (7)$$

Here, $\mathcal{V}_{\text{band}}$ is the set of voxel grids within the narrow band. We also add a constant bias $bias_f = 0.03m$ to deal with vertices outside of the narrow band.

4.3 Texture Prediction

With the shape reconstructed, we then estimate the color at each surface point. Instead of solving a color field encoded by an implicit function like the earlier works [Saito et al. 2019; Shao et al. 2022a], we exploit high-resolution input images for rich appearance details. Specifically, we estimate a blending weight vector \mathbb{W} at each surface point \mathbf{X} . The color at \mathbf{X} is then computed as a weighted average of the colors at its image projections,

$$c(\mathbf{X}) = \sum_i \mathbb{W}_i I_i(\mathbf{x}_i), \quad (8)$$

where $\mathbf{x}_i = \Pi_i(\mathbf{X})$ is the projection of \mathbf{X} in image I_i . Similar to shape reconstruction, we also sample a volume grid \mathbb{W} and estimate the blending weights at the grid vertices \mathbb{W}_{whd} . In this way, our method essentially estimates a blended texture map over the surface, instead of computing a color field which tends to be limited by the 3D sampling rate. With our method, the predicted texture map carries sharp appearance details inherited from the input images.

The network architecture of our texture weight estimation is shown in Figure 6. Thanks to the precisely reconstructed TSDF $\mathbb{D}_{\text{final}}$ from the shape branch, we only consider a 2cm width narrow-band nearby its zero-level set, which is defined as $|\mathbb{D}_{\text{final}}| < 0.01m$. For each input image, we compute its texture feature \mathbb{F}_i^C and construct a volume \mathbb{F}_i^C by projecting them back to the discretized narrow-band. The texture feature maps \mathbb{F}_i^C are also computed by an hour-glass [Newell et al. 2016] network from the input image I_i with a smaller network to extract a 32-channel feature of size 256×256 . We further compute the truncated PSDF [Yu et al. 2021b], which is a view-dependent function indicating if the surface is viewed from a slanted direction, and concatenate it to the texture feature.

The attention model [Vaswani et al. 2017] is used here to handle visibility by re-weighting features across different views. Ideally, if a 3D point is not visible from a particular view, the projected color from that view should have less influence on the final blended color. Therefore, the attention module is used to adjust the contribution of the projected texture features by re-weighting them. Following this, we apply a sparse 3D CNN on these re-weighted feature volumes individually to regress the blending weight volume of each view. We then normalize these blending weight volumes across views by a soft-max to obtain the normalized blending weights \mathbb{W} .

The training loss for the texture blending weight estimation is defined as:

$$L_c = \sum_{(w,h,d) \in \mathcal{W}_{\text{band}}} \left\| \mathbb{C}_{whd}^{\text{pred}} - \mathbb{C}_{whd}^{\text{gt}} \right\|_{L_1}. \quad (9)$$

Here, $\mathbb{C}_{whd}^{\text{pred}}$ is the surface color volume computed by applying the blending weights \mathbb{W}_{whd} . The ground truth color volume $\mathbb{C}_{whd}^{\text{gt}}$ is generated from the ground truth textured mesh with nearest search. The set $\mathcal{W}_{\text{band}}$ includes all voxel grids within the narrow band for color estimation.

After solving the blending volume \mathbb{W} , we use the Blender [Community [n. d.]] to generate a texture atlas map of the reconstructed mesh model. For each pixel in the atlas map, we use barycentric

interpolation to compute its 3D location from the mesh vertices, and then determine its color according to Equation 8. Figure 7 shows the atlas map computed by our method. To capture rich details in the input images, a high-resolution texture atlas map, such as 2K resolution, can be chosen. In this way, our method generates high-quality textured results.

5 EXPERIMENTS

5.1 Implementation Details

We experimented with three commonly used datasets, Twindom [Twindom [n. d.]], THuman2.0 [Zheng et al. 2021], and MultiHuman [Zheng et al. 2021]. These datasets consist of high-quality scanned 3D models of clothed humans with varying poses and body shapes. We followed [Saito et al. 2019] to generate multi-view images under spherical harmonic lighting to train our network. Before training our shape networks, we pre-computed the ground truth TSDF \mathbb{D}^{gt} for each human model in our training set. For training the texture network, we computed the ground truth color volume \mathbb{C}^{gt} by finding the nearest mesh vertex for each volume grid. In the experiment, we used 1,000 models from Twindom [Twindom [n. d.]] and THuman2.0 [Zheng et al. 2021] for training. Another 200 models from Twindom and 30 models from MultiHuman were used for testing. We employed the Adam optimizer with a learning rate of $1e-4$. The network was trained in an end-to-end fashion for 30 epochs, and the training of our pipeline took approximately 8 hours using 8 NVIDIA A100 GPUs.

At testing time, for each model, we used 1/2/4/6 input images from different viewpoints to reconstruct the clothed human model. To estimate normal maps, we employed the pre-trained model from [Zheng et al. 2021]. Our testing experiments were performed with an NVIDIA 3090 GPU. The breakdown of the running time for our method with 6 input images and 512 volume resolution is provided in Table 5 and Table 6 for shape and texture estimation respectively. The most time-consuming step of our shape reconstruction is to extract feature maps $\{\mathbb{F}_i^F\}$ and $\{\mathbb{F}_i^N\}$. We can use buffer swapping techniques with two GPUs to achieve $\times 2$ speedup.

In terms of memory consumption, our method takes 18G and 45G of GPU memory during training for the 256 and 512 volume resolutions respectively. The testing time memory consumption is 12G and 18G.

5.2 Quantitative Results on Synthetic Data

In this subsection, we compare our methods against other reconstruction methods, including Multi-view PIFu [Saito et al. 2019], Multi-view PIFuHD [Saito et al. 2020], DeepMultiCap [Zheng et al. 2021], DoubleField [Shao et al. 2022a], and DiffuStereo [Shao et al. 2022b]. We test the robustness of these methods with different numbers of input images. To ensure a fair comparison, we implemented MultiView PIFuHD [Saito et al. 2020] and MultiView PIFu [Saito et al. 2019] based on the public code of their single-view versions. We used the same training and testing data for MultiView PIFu, MultiView PIFuHD, and our method. The authors of DeepMultiCap [Zheng et al. 2021] and DoubleField [Shao et al. 2022a] kindly provided us with their results. According to their paper, these two methods were trained on a larger set of data than our method. To

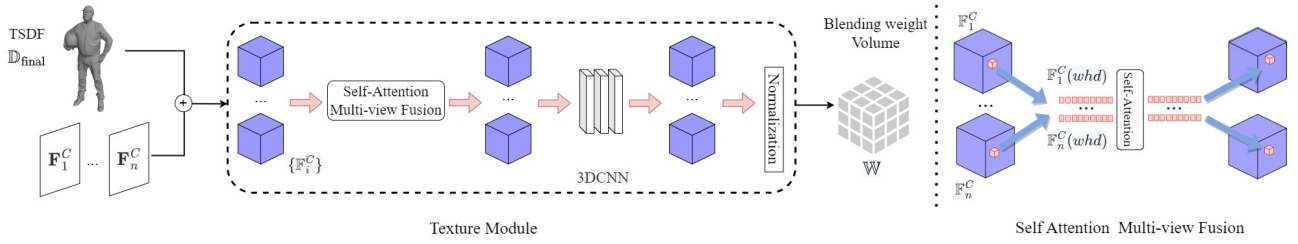


Fig. 6. The texture module predicts a blending weight volume using the input texture feature maps F_i^C and the estimated TSDF \mathcal{D}_{final} . The final texture map is computed by interpolating pixel values from the input images according to the blending weights.

	1 view		2 views		4 views		6 views	
	Chamfer	P2S	Chamfer	P2S	Chamfer	P2S	Chamfer	P2S
PIFu	2.528/1.612	2.421/1.587	1.626/1.200	1.507/1.170	0.929/0.823	0.783/0.773	0.776/0.678	0.725/0.625
PIFuHD	2.814/1.725	2.793/1.704	1.369/1.162	1.223/1.126	0.821/0.765	0.719/0.645	0.720/0.698	0.705/0.501
DeepMultiCap	–	–	1.529/1.159	1.496/1.117	1.150/0.969	1.115/1.001	1.062/0.890	1.024/0.944
Doublefield	–	–	–	–	0.836/0.905	0.822/0.869	0.711/0.779	0.690/0.740
Ours(256)	2.457/1.563	2.374/1.537	1.221/0.899	1.080/0.860	0.810/0.550	0.629/0.500	0.668/0.459	0.470/0.402
Ours(512)	2.398/1.565	2.363/1.539	1.110/0.889	1.052/0.837	0.514/0.447	0.429/0.389	0.390/0.314	0.287/0.242

Table 3. Mean Chamfer and point-2-surface (P2S) errors of the reconstructed mesh on the Twindom dataset. In each entry, we report two error metrics as x/y , where x represents recall and y stands for precision.

	1 view		2 views		4 views		6 views	
	Chamfer	P2S	Chamfer	P2S	Chamfer	P2S	Chamfer	P2S
PIFu	1.975/1.540	1.872/1.511	1.370/1.216	1.249/1.169	0.893/0.653	0.742/0.604	0.660/0.523	0.472/0.462
PIFuHD	2.142/1.936	2.121/1.916	1.140/0.915	0.998/0.873	0.739/0.589	0.564/0.523	0.630/0.492	0.438/0.433
DeepMultiCap	–	–	1.114/0.928	1.077/0.914	0.932/0.781	0.891/0.777	0.681/0.678	0.678/0.676
Doublefield	–	–	–	–	0.743/0.788	0.621/0.748	0.652/0.664	0.579/0.621
Ours(256)	1.922/1.516	1.824/1.485	0.995/0.735	0.847/0.691	0.644/0.412	0.445/0.345	0.570/0.341	0.356/0.275
Ours(512)	1.852/1.515	1.819/1.480	0.822/0.689	0.763/0.602	0.453/0.360	0.372/0.296	0.348/0.271	0.252/0.195

Table 4. Mean Chamfer and point-2-surface (P2S) errors of the reconstructed mesh on the Multihuman dataset. In each entry, we report two error metrics as x/y , where x represents recall and y stands for precision.



Fig. 7. The estimated shape, texture atlas map, and a rendering of the textured model.

make the comparison fair, we sample a $512 \times 512 \times 512$ volume to compute the final shape using the marching cube algorithm in all the compared methods.

	Coarse Stage			Fine Stage			Total
	$\{F_i^I\}$	VH	3D CNN	$\{F_i^N\}$	NB	3D CNN	
Time (ms)	112	85	52	112	11	93	465

Table 5. Shape reconstruction time cost. Here, columns $\{F_i^I\}$ and $\{F_i^N\}$ are the time on computing these feature maps. 'VH' and 'NB' are the time on visual hull culling and narrow band culling.

	$\{F_i^C\}$	Attention	3D CNN	UV Atlas	Total
	Time (ms)	28	18	150	57

Table 6. Texture reconstruction time cost. Here, the column $\{F_i^C\}$ indicates the time for computing the feature maps.

In the case of a single input image, we determine the visual hull by truncating a cone defined by the camera center and the image silhouette with depth thresholds of -0.5m and 0.5m . We retrain the public code of PIFu [Saito et al. 2019] on our dataset. PIFuHD [Saito et al. 2020] only releases testing code, so we test its pre-trained model on our dataset.

Table 3 and Table 4 provide a quantitative comparison of different methods with 1–6 input images. Note that Deepmulticap [Zheng

	DiffStereo	Ours
Chamfer/P2S	0.120/0.126	0.158/0.103

Table 7. The shape precision of our method and the DiffuStereo [Shao et al. 2022b] on the 8-view setting.

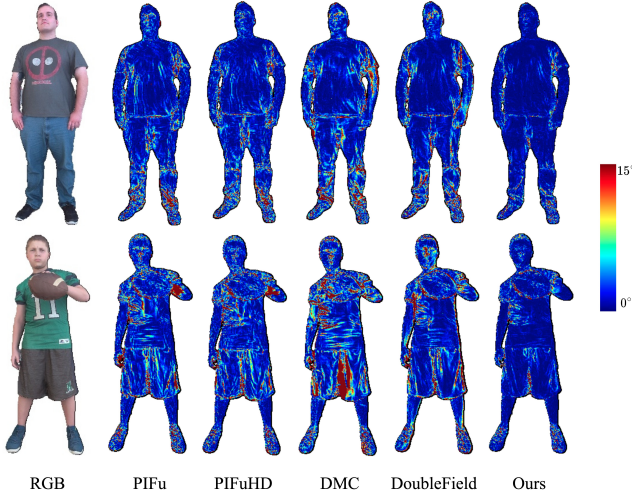


Fig. 8. The normal errors of different methods are visualized as heat maps.

et al. 2021] cannot work with a single input image, while DoubleField [Shao et al. 2022a] cannot work with a single or two input images. Thus, their results are absent for those settings. We report our results with two different volume resolutions, 256 and 512 for the fine stage (with corresponding coarse stage resolutions of 128 and 256, respectively). To better evaluate these methods, we report both recall and precision errors for all methods. Precision is computed as the average distance between each vertex on the predicted mesh and its nearest correspondence on the ground truth mesh. Conversely, recall is the average distance between each vertex on the ground truth mesh and its nearest correspondence on the predicted mesh. Our method, with a 512 volume resolution, has the lowest errors among all methods for different numbers of input images. On the Twindown dataset (shown in Table 3), our method reduces the mean point-2-surface (P2S) precision to 0.24cm with 6 input images, a remarkable error reduction of 51% over DoubleField [Shao et al. 2022a] and Multi-view PIFu [Saito et al. 2019, 2020]. Furthermore, it achieves a mean P2S precision of 0.39cm using just 4 input images, with over a 39% error reduction. Even with only 2 input images, our method can achieve a 0.84cm mean P2S precision, surpassing DeepMultiCap [Zheng et al. 2021] with 6 input views. This significant improvement over SOTA methods is also apparent on the MultiHuman dataset (shown in Table 4) and with the Chamfer error metric. When using a single input image, the performance improvement by our method is smaller. This is reasonable since the main source of error here is the single view depth ambiguity, which cannot be solved by our 3D convolution without additional input images.

DiffuStereo [Shao et al. 2022b] requires image pairs with a smaller baseline for their diffusion-based stereo matching, which does not

	PIFu	PIFuHD	DMC	DoubleField	Ours
Twindom	9.76	9.69	13.02	11.18	6.94
Multihuman	10.61	10.46	11.92	11.66	7.25

Table 8. The normal errors of different methods measured by the mean angular error (in degrees). Here DMC stands for the ‘DeepMultiCap’ method.

	PIFu	PixelNerf	DoubleField	Ours
PSNR	20.66	21.85	23.56	26.31
SSIM	0.807	0.813	0.857	0.863

Table 9. PSNR and SSIM of the re-rendered mesh on the synthetic dataset. Our method produces results most consistent with the ground truth.

	Twindom		MultiHuman	
	Chamfer	P2S	Chamfer	P2S
AB1 (G)	0.351	0.286	0.303	0.233
AB2 (N)	0.339	0.275	0.280	0.207
Proposed method	0.314	0.242	0.271	0.195

Table 10. Results of different ablation settings in shape reconstruction. AB1 and AB2 use different input features at the fine stage.

work on our sparse view setting in Table 3 and Table 4. Table 7 compares our method with DiffuStereo using the same 8-view setting as [Shao et al. 2022b], where each view has an adjacent view facilitating stereo reconstruction. We employ our model trained for the 6-view input, which has not been trained or finetuned on the 8-view setting. Our P2S precision is 18% smaller than that of DiffuStereo, which demonstrates the generalization capability of our method. In this case, we follow [Shao et al. 2022b] to normalize the height of all human subjects to 1 meter when evaluating the error metrics, resulting in smaller error metrics than those in Table 3 and Table 4.

Table 8 and Figure 8 show the surface normal error of different methods with 6 input views to evaluate their capability in capturing fine-scale shape details. To compute the surface normal error, we re-render the reconstructed surface into normal maps and compare them with the ground truth results. Our method achieves the smallest mean angular error on both datasets, demonstrating our capability of reconstructing shape details. As shown in Figure 8, other methods often produce larger errors at concave regions, such as the pants in the second row.

Table 9 assesses the rendering quality of our textured models using PSNR and SSIM metrics. All results are obtained under the 6-view setting. We cite the results of PixelNerf [Yu et al. 2021a] and DoubleField [Shao et al. 2022a] from DoubleField [Shao et al. 2022a]. Our method achieves the highest score on both metrics.

To better understand the quantitative comparison, we visualize some of the results in Figure 9, where (a)–(b), (c)–(d), and (e)–(f) are results reconstructed with 6, 4, and 2 input images, respectively. From left to right, the shown figures are input images, results from Multi-view PIFu, Multi-view PIFuHD, DeepMultiCap, DoubleField, our method, and ground truth, respectively. It is evident that our method generates more shape details and is more robust to loose garments and rare poses. From examples (e, f), we can observe that Multi-View PIFu and Multi-View PIFuHD often generate broken arms when the number of input images is small, while our method

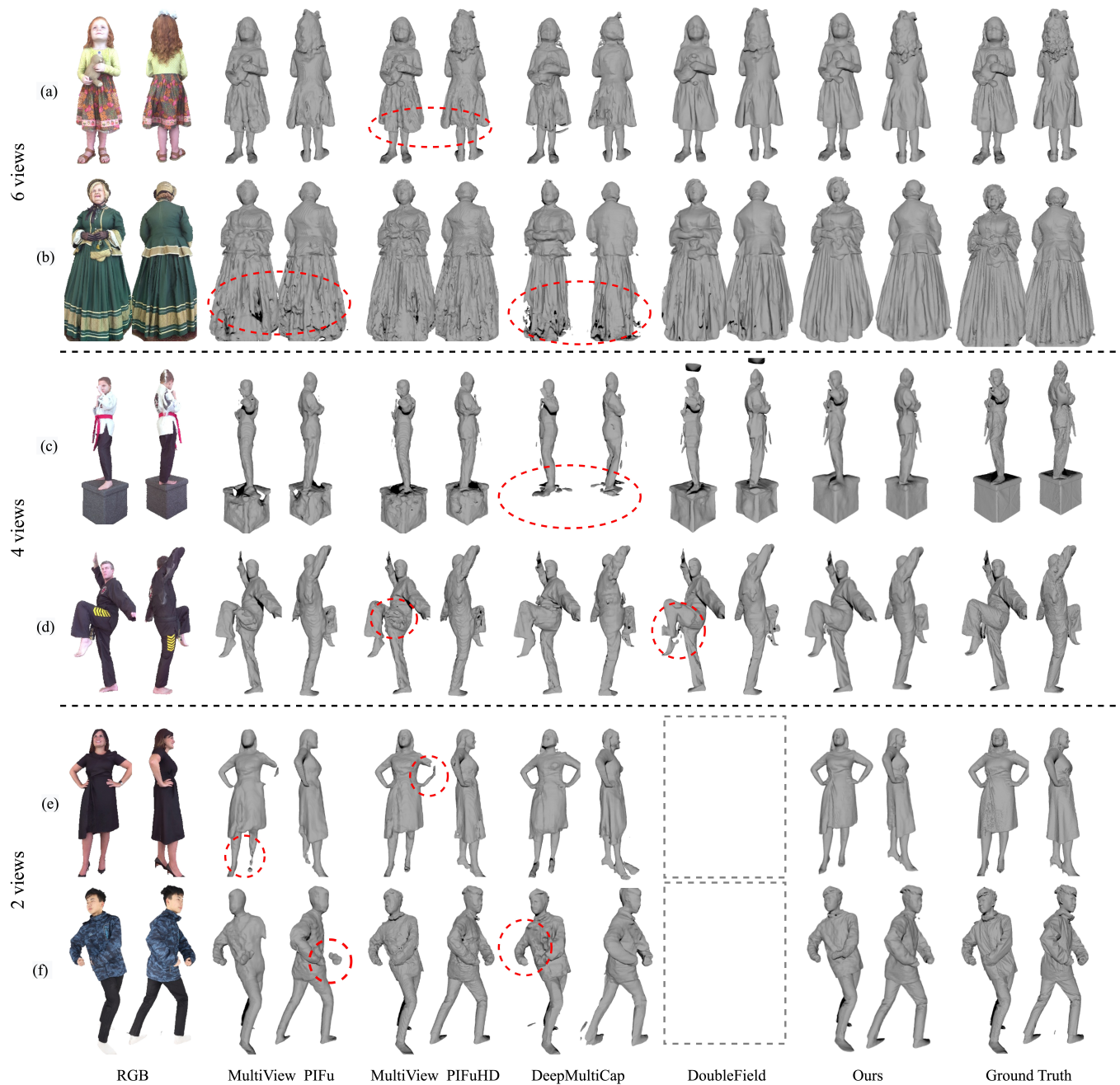


Fig. 9. Visual results on the synthetic dataset. From left to right, shown are one of the input images, results from Multiview-PIFu [Saito et al. 2019], Multiview-PIFuHD [Saito et al. 2020], DeepMultiCap [Zheng et al. 2021], DoubleField [Shao et al. 2022a], our method, and ground truth, respectively.

does not suffer from this problem. Examples (a, b) highlight our strength in handling loose garments, where other methods often generate noisy reconstructions. Examples (c, d) showcase our capability in addressing rare poses and unusual objects.

Figure 10 visualizes the recovered normal maps and blending weight maps for some examples. We visualize the blending weights of three input images in the respective RGB channels. The smooth

transition of these weights generates seamless textured models with vivid texture details, as shown in the zoomed-in regions.

Figure 11 shows some challenging examples, such as occlusion and multiple persons. Our method can still recover faithful shape details and poses in these situations. Note that we use the ground truth foreground segmentation, which includes the luggage and all persons together. Instance segmentation, as employed in DeepMultiCap [Zheng et al. 2021], is not used here. Surprisingly, our method

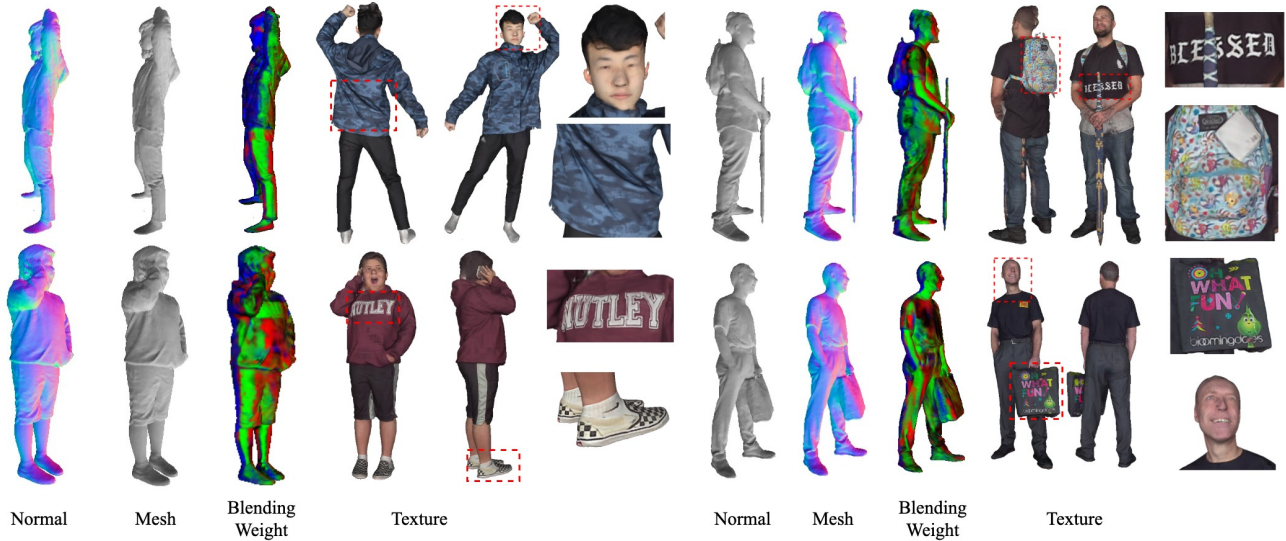


Fig. 10. Texturing results. The blending weight map shows the weights of three input images in the RGB channels. The final texture map captures appearance details, such as facial expressions and cloth patterns, as shown in the zoomed-in regions.



Fig. 11. Some challenging cases on synthetic data. Our method is robust to multiple persons (without instance segmentation), occlusion, and rare poses.

can even reconstruct backpacks and luggage quite well, even though it has never been trained on these types of objects. We believe our 3D feature volume helps to learn local implicit functions, like those in [Jiang et al. 2020], which generalize well across object categories. This is because, locally, backpacks and luggage have similar shapes as clothed humans. More animated examples are provided in the supplementary video.

5.3 Qualitative Results on Real Data

We also experiment with our own real data, which is captured using 6 calibrated and synchronized Kinect cameras (only RGB images are used) surrounding the subject. We employ [Sengupta et al. 2020] to generate the image masks. Figure 12 shows some of the results. As demonstrated in these examples, our method does not rely on SMPL

model estimation, enabling it to handle challenging cases with loose garments and unusual poses. All these examples are reconstructed using 6 input images. Video results and additional examples can be found in the supplementary files.

5.4 Ablation Study

We conduct ablation studies to examine the effectiveness of our various design choices. To justify our system design, we also test a naïve implementation using a 3D UNet, which has the same architecture as the coarse stage with conventional 3D convolutions. We report the system performance and GPU memory consumption for different volume resolutions in Figure 13. It is evident that higher volume resolution can significantly reduce shape errors, especially recall errors. However, GPU memory consumption also increases



Fig. 12. Results on real captured images. Our method generalizes well to various garments and poses and recovers high-quality shape details.

substantially, from 6G to 69G for volume resolution of 32 and 256, respectively. It is not feasible to scale this naïve implementation to a volume resolution of 512 on an A100.

Ablation I: Shape ablation. To test the effectiveness of the normal feature and coarse level feature at the fine stage shape reconstruction, we conduct experiments using: AB1. only the coarse level feature; AB2. only the normal feature; and the proposed method, which uses both normal and coarse level features together.

We summarize the mean Chamfer and P2S errors of these different settings in Table 10. From AB1 and AB2, we can see that the normal features and the coarse level feature complement each other and should both be included when computing the final TSDF.

Ablation II: Texture ablation. To test the effectiveness of our texture estimation, we conduct experiments using: AB3. the network to directly compute a color field as in previous methods[Saito et al. 2019; Zheng et al. 2021]; AB4 & AB5. after solving our blending weight field (with input images of 512 resolution by optimizing Equation 9), we use 1K & 2K images to compute the texture atlas map, respectively.

Table 11 summarizes the PSNR and SSIM of the different settings. Firstly, AB3 has much poorer results than the other two settings

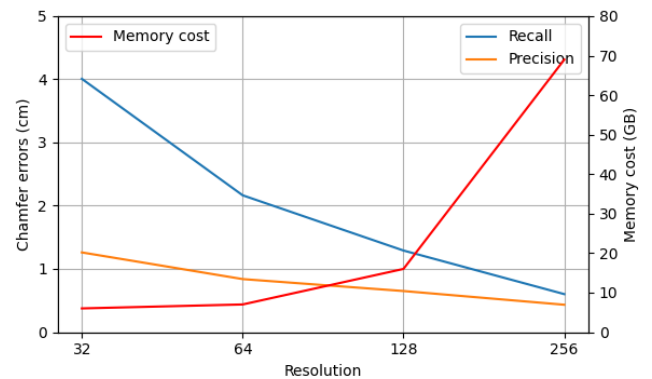


Fig. 13. The performance and memory consumption with different volume resolutions when using a naïve 3D UNet.

that involve blending weight estimation. It is evident that our blending weight strategy is crucial for generating sharp texture. Visualization in Figure 14 reveals that our strategy can produce sharp high-frequency texture details, while color field regression causes blurriness. AB4 and AB5 further demonstrate the scalability of our method. Both settings share the same blending weight volume \mathbb{W} , computed from input images of 512 resolution. We apply \mathbb{W} to 1K



Fig. 14. Results of different ablation settings for texture prediction. From left to right, they are the results of color field estimation, and our blending weight field estimation with 1K and 2K images, respectively.

	Twindom		MultiHuman	
	PSNR	SSIM	PSNR	SSIM
AB3	23.776	0.854	24.027	0.860
AB4	26.309	0.862	26.033	0.866
AB5	26.656	0.864	26.544	0.867

Table 11. Results of different ablation settings in texture prediction. Please refer to text for more details.

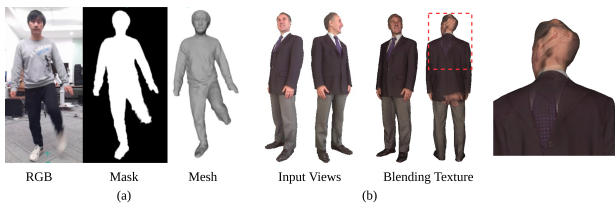


Fig. 15. Failure cases of our method. (a) Due to the incomplete visual hull caused by poor matting, our method is unable to reconstruct the correct shape. (b) Unobserved regions have incorrect texture.

and 2K images to compute the texture atlas map. Both settings generate high-quality results, while AB5 is slightly better. These two experiments demonstrate that our method can capture more texture details by re-evaluating the texture atlas map without the need for retraining.

5.5 Limitations and Future Work

Our method has difficulties when addressing cases where the segmentation module fails. Figure 15 (a) shows such an example, where inaccurate segmentation due to motion blur results in an incomplete feature volume, consequently leading to poor final results. Inaccurate camera calibration may also contribute to poor feature volume construction and, subsequently, inferior shape results. As for texture prediction, our method computes texture maps by blending input images, which makes it difficult to handle unobserved regions, as shown in Figure 15 (b). In the future, we might consider employing an implicit function to address this problem.

6 CONCLUSION

We re-examine volumetric reconstruction for clothed humans and demonstrate that, with proper system design, it can generate superior results than recent deep implicit methods. We find that a high volume resolution, such as 512 or above, effectively reduces the notorious quantization error and capitalizes on the advantages of 3D CNNs for enhanced exploration of local context information. To address the memory and computational challenges associated with high-resolution volumes, our method takes a coarse-to-fine

approach, integrating sparse 3D CNN and voxel culling through visual hulls and narrow bands. Finally, it employs an image-based rendering approach to compute the texture atlas map by blending input images with learned weights. Extensive experiments demonstrate that our method significantly improves shape accuracy over SOTA techniques and captures vivid appearance details.

ACKNOWLEDGMENTS

We would like to thank DGene company and Prof. Yebing Liu (Tsinghua University) for kindly providing human body datasets for our experiments.

REFERENCES

- Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. 2019. Tex2Shape: Detailed Full Human Body Geometry From a Single Image. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2293–2303. <https://doi.org/10.1109/ICCV.2019.00238>
- Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. 2022. Photorealistic Monocular 3D Reconstruction of Humans Wearing Clothing. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1496–1505. <https://doi.org/10.1109/CVPR52688.2022.00156>
- Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. 2005. SCAPE: Shape Completion and Animation of People. *ACM Trans. Graph.* 24, 3 (jul 2005), 408–416. <https://doi.org/10.1145/1073204.1073207>
- Bharat Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. 2019. Multi-Garment Net: Learning to Dress 3D People From Images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 5419–5429. <https://doi.org/10.1109/ICCV.2019.00552>
- Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. 2016. Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *Computer Vision – ECCV 2016*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer International Publishing, Cham, 561–578.
- Zhiqin Chen and Hao Zhang. 2019. Learning Implicit Fields for Generative Shape Modeling. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5932–5941. <https://doi.org/10.1109/CVPR.2019.00609>
- Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. 2020. Implicit Functions in Feature Space for 3D Shape Reconstruction and Completion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6968–6979. <https://doi.org/10.1109/CVPR42600.2020.00700>
- Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. 2015. High-Quality Streamable Free-Viewpoint Video. *ACM Trans. Graph.* 34, 4, Article 69 (jul 2015), 13 pages. <https://doi.org/10.1145/2766945>
- Blender Online Community. [n. d.]. *Blender*. <https://www.blender.org/>
- Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, Pushmeet Kohli, Vladimir Tankovich, and Shahram Izadi. 2016. Fusion4D: Real-Time Performance Capture of Challenging Scenes. *ACM Trans. Graph.* 35, 4, Article 114 (jul 2016), 13 pages. <https://doi.org/10.1145/2897824.2925969>
- Liuhaohao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. 2017. 3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5679–5688. <https://doi.org/10.1109/CVPR.2017.602>
- Andrew Gilbert, Marco Volino, John Collomosse, and Adrian Hilton. 2018. Volumetric Performance Capture from Minimal Camera Viewpoints. In *Computer Vision – ECCV 2018*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.). Springer International Publishing, Cham, 591–607.
- Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 2018. 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9224–9232. <https://doi.org/10.1109/CVPR.2018.00961>
- Kaichen 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. 2019. The Relightables: Volumetric Performance Capture of Humans with Realistic Relighting. *ACM Trans. Graph.* 38, 6, Article 217 (nov 2019), 19 pages. <https://doi.org/10.1145/3355089.3356571>
- Riza Alp Güler and Iasonas Kokkinos. 2019. HoloPose: Holistic 3D Human Reconstruction In-The-Wild. In *2019 IEEE/CVF Conference on Computer Vision and Pattern*

- Recognition (CVPR). 10876–10886. <https://doi.org/10.1109/CVPR.2019.01114>
- Tong He, Yuanlu Xu, Shunsuke Saito, Stefano Soatto, and Tony Tung. 2021. ARCH++: Animation-Ready Clothed Human Reconstruction Revisited. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 11026–11036. <https://doi.org/10.1109/ICCV48922.2021.01086>
- Yang Hong, Juyong Zhang, Boyi Jiang, Yudong Guo, Ligang Liu, and Hujun Bao. 2021. StereoPIFu: Depth Aware Clothed Human Digitization via Stereo Vision. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 535–545. <https://doi.org/10.1109/CVPR46437.2021.00060>
- Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V. Gehler, Javier Romero, Ijaz Akhter, and Michael J. Black. 2017. Towards Accurate Marker-Less Human Shape and Pose Estimation over Time. In *2017 International Conference on 3D Vision (3DV)*. 421–430. <https://doi.org/10.1109/3DV.2017.00055>
- Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Chongyang Ma, Linjie Luo, and Hao Li. 2018. Deep Volumetric Video From Very Sparse Multi-View Performance Capture. In *European Conference on Computer Vision (ECCV)*. https://doi.org/10.1007/978-3-030-01270-0_21
- Zeng Huang, Yuanlu Xu, Christoph Lassner, Hao Li, and Tony Tung. 2020. ARCH: Animatable Reconstruction of Clothed Humans. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3090–3099. <https://doi.org/10.1109/CVPR42600.2020.00316>
- Aaron S. Jackson, Chris Manafas, and Georgios Tzimiropoulos. 2018. 3D Human Body Reconstruction from a Single Image via Volumetric Regression. *ArXiv abs/1809.03770* (2018).
- Chiyu Jiang, Avneesh Sud, Ameer Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. 2020. Local Implicit Grid Representations for 3D Scenes. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6000–6009. <https://doi.org/10.1109/CVPR42600.2020.00604>
- Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. 2019. Panoptic Studio: A Massively Multiview System for Social Interaction Capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 1 (2019), 190–204. <https://doi.org/10.1109/TPAMI.2017.2782743>
- Hanbyul Joo, Tomas Simon, and Yaser Sheikh. 2018. Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8320–8329. <https://doi.org/10.1109/CVPR.2018.00868>
- Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. 2018. End-to-End Recovery of Human Shape and Pose. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7122–7131. <https://doi.org/10.1109/CVPR.2018.00744>
- Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. 2020. VIBE: Video Inference for Human Body Pose and Shape Estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5252–5262. <https://doi.org/10.1109/CVPR42600.2020.00530>
- Nikos Kolotouros, Georgios Pavlakos, Michael Black, and Kostas Daniilidis. 2019. Learning to Reconstruct 3D Human Pose and Shape via Model-Fitting in the Loop. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2252–2261. <https://doi.org/10.1109/ICCV.2019.00234>
- Ruilong Li, Yuliang Xiu, Shunsuke Saito, Zeng Huang, Kyle Olszewski, and Hao Li. 2020. Monocular Real-Time Volumetric Performance Capture. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 49–67.
- Junbang Liang and Ming Lin. 2019. Shape-Aware Human Pose and Shape Reconstruction Using Multi-View Images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 4351–4361. <https://doi.org/10.1109/ICCV.2019.00445>
- Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graph.* 34, 6, Article 248 (nov 2015), 16 pages. <https://doi.org/10.1145/2816795.2818013>
- William E. Lorensen and Harvey E. Cline. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *SIGGRAPH Comput. Graph.* 21, 4 (aug 1987), 163–169. <https://doi.org/10.1145/37402.37422>
- Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy Networks: Learning 3D Reconstruction in Function Space. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4455–4465. <https://doi.org/10.1109/CVPR.2019.00459>
- Ben Mildenhall, Pratul P. Srinivasan, Matthew Tanicli, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 405–421.
- Richard A. Newcombe, Dieter Fox, and Steven M. Seitz. 2015. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 343–352. <https://doi.org/10.1109/CVPR.2015.7298631>
- Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneux, David Kim, Andrew J. Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*. 127–136. <https://doi.org/10.1109/ISMAR.2011.6092378>
- Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked Hourglass Networks for Human Pose Estimation. In *Computer Vision – ECCV 2016*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer International Publishing, Cham, 483–499.
- Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. 2018. Neural Body Fitting: Unifying Deep Learning and Model Based Human Pose and Shape Estimation. In *2018 International Conference on 3D Vision (3DV)*. 484–494. <https://doi.org/10.1109/3DV.2018.00062>
- Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 165–174. <https://doi.org/10.1109/CVPR.2019.00025>
- Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. Osman, Dimitrios Tzionas, and Michael J. Black. 2019. Expressive Body Capture: 3D Hands, Face, and Body From a Single Image. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10967–10977. <https://doi.org/10.1109/CVPR.2019.01123>
- Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. 2018. Learning to Estimate 3D Human Pose and Shape from a Single Color Image. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 459–468. <https://doi.org/10.1109/CVPR.2018.00055>
- Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. 2020. Convolutional Occupancy Networks. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 523–540.
- Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. 2021. Neural Body: Implicit Neural Representations with Structured Latent Codes for Novel View Synthesis of Dynamic Humans. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9050–9059. <https://doi.org/10.1109/CVPR46437.2021.00894>
- Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Hao Li, and Angjoo Kanazawa. 2019. PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2304–2314. <https://doi.org/10.1109/ICCV.2019.00239>
- Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. 2020. PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 81–90. <https://doi.org/10.1109/CVPR42600.2020.00016>
- Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. 2020. Background Matting: The World Is Your Green Screen. In *CVPR*.
- Ruizhi Shao, Hongwen Zhang, He Zhang, Mingjia Chen, Yan-Pei Cao, Tao Yu, and Yebin Liu. 2022a. DoubleField: Bridging the Neural Surface and Radiance Fields for High-fidelity Human Reconstruction and Rendering. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15851–15861. <https://doi.org/10.1109/CVPR52688.2022.01541>
- Ruizhi Shao, Zerong Zheng, Hongwen Zhang, Jingxiang Sun, and Yebin Liu. 2022b. DiffuStereo: High Quality Human Reconstruction via Diffusion-based Stereo Using Sparse Cameras. In *ECCV*.
- Twindom. [n. d.]. Human 3D Body Model Datasets. <https://web.twindom.com/>.
- Gül Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. 2018. BodyNet: Volumetric Inference of 3D Human Body Shapes. In *Computer Vision – ECCV 2018*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.). Springer International Publishing, Cham, 20–38.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (Long Beach, California, USA) (NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.
- Daniel Vlasic, Pieter Peers, Ilya Baran, Paul Debevec, Jovan Popović, Szymon Rusinkiewicz, and Wojciech Matusik. 2009. Dynamic Shape Capture Using Multi-View Photometric Stereo. *ACM Trans. Graph.* 28, 5 (dec 2009), 1–11. <https://doi.org/10.1145/1618452.1618520>
- Michael Waechter, Nils Moehle, and Michael Goesele. 2014. Let There Be Color! Large-Scale Texturing of 3D Reconstructions. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 836–850.
- Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J. Black. 2022. ICON: Implicit Clothed humans Obtained from Normals. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13286–13296. <https://doi.org/10.1109/CVPR52688.2022.01294>
- Yuanlu Xu, Song-Chun Zhu, and Tony Tung. 2019. DenseRaC: Joint 3D Pose and Shape Estimation by Dense Render-and-Compare. In *2019 IEEE/CVF International*

- Conference on Computer Vision (ICCV)*. 7759–7769. <https://doi.org/10.1109/ICCV.2019.00785>
- Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021a. pixelNeRF: Neural Radiance Fields from One or Few Images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4576–4585. <https://doi.org/10.1109/CVPR46437.2021.00455>
- Tao Yu, Kaiwen Guo, Feng Xu, Yuan Dong, Zhaoqi Su, Jianhui Zhao, Jianguo Li, Qionghai Dai, and Yebin Liu. 2017. BodyFusion: Real-Time Capture of Human Motion and Surface Geometry Using a Single Depth Camera. In *2017 IEEE International Conference on Computer Vision (ICCV)*. 910–919. <https://doi.org/10.1109/ICCV.2017.104>
- Tao Yu, Jianhui Zhao, Zerong Zheng, Kaiwen Guo, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. 2020. DoubleFusion: Real-Time Capture of Human Performances with Inner Body Shapes from a Single Depth Sensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 10 (2020), 2523–2539. <https://doi.org/10.1109/TPAMI.2019.2928296>
- Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. 2021b. Function4D: Real-time Human Volumetric Capture from Very Sparse Consumer RGBD Sensors. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5742–5752. <https://doi.org/10.1109/CVPR46437.2021.00569>
- Yang Zheng, Ruizhi Shao, Yuxiang Zhang, Tao Yu, Zerong Zheng, Qionghai Dai, and Yebin Liu. 2021. DeepMultiCap: Performance Capture of Multiple Characters Using Sparse Multiview Cameras. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 6219–6229. <https://doi.org/10.1109/ICCV48922.2021.00618>
- Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. 2022. PaMIR: Parametric Model-Conditioned Implicit Representation for Image-Based Human Reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 6 (2022), 3170–3184. <https://doi.org/10.1109/TPAMI.2021.3050505>
- Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. 2019. DeepHuman: 3D Human Reconstruction From a Single Image. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 7738–7748. <https://doi.org/10.1109/ICCV.2019.00783>

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009