Implicit Neural Representations with Structured Latent Codes for Human Body Modeling

Sida Peng, Chen Geng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang,
Qing Shuai, Xiaowei Zhou, Hujun Bao

Abstract—This paper addresses the challenge of novel view synthesis for a human performer from a very sparse set of camera views. Some recent works have shown that learning implicit neural representations of 3D scenes achieves remarkable view synthesis quality given dense input views. However, the representation learning will be ill-posed if the views are highly sparse. To solve this ill-posed problem, our key idea is to integrate observations over video frames. To this end, we propose Neural Body, a new human body representation which assumes that the learned neural representations at different frames share the same set of latent codes anchored to a deformable mesh, so that the observations across frames can be naturally integrated. The deformable mesh also provides geometric guidance for the network to learn 3D representations more efficiently. Furthermore, we combine Neural Body with implicit surface models to improve the learned geometry. To evaluate our approach, we perform experiments on both synthetic and real-world data, which show that our approach outperforms prior works by a large margin on novel view synthesis and 3D reconstruction. We also demonstrate the capability of our approach to reconstruct a moving person from a monocular video on the People-Snapshot dataset. The code and data are available at https://zju3dv.github.io/neuralbody/.

Index Terms—Novel View Synthesis, Human Reconstruction, Differentiable Rendering

1 Introduction

Free-viewpoint videos of human performers have a variety of applications such as movie production, sports broadcasting, and telepresence. Previous free-viewpoint video systems either rely on a dense array of cameras for image-based novel view synthesis [1], [2] or require multiple depth sensors for high-quality 3D reconstruction [3], [4] to produce realistic rendering. The complicated hardware makes free-viewpoint video systems expensive and only applicable in constrained environments.

This work focuses on the problem of novel view synthesis for a human performer from a sparse multi-view video captured by a very limited number of cameras, as illustrated in Figure 2. This setting significantly decreases the cost of free-viewpoint systems and makes the systems more widely applicable. However, this problem is extremely challenging. Traditional image-based rendering methods [1], [5] mostly require dense input views and cannot be applied here. For reconstruction-based methods [6], [7], the wide baselines between cameras make dense stereo matching intractable. Moreover, part of the human body may be invisible due to self-occlusion in sparse views. As a result, these methods tend to give noisy and incomplete reconstructions, resulting in heavy rendering artifacts.

Recent works [8], [9], [10] have investigated the potential of implicit neural representations on novel view synthesis.

- *S. Peng, C. Geng, Y. Zhang, Q. Shuai, H.Bao and X. Zhou are affiliated with the State Key Lab of CAD&CG, the College of Computer Science, Zhejiang University, China.*
- *Y. Xu is affiliated with the Department of Information Engineering, The Chinese University of Hong Kong.*
- *Q. Wang is affiliated with the Department of Computer Science, Cornell University.*
- *Corresponding author: Hujun Bao.*

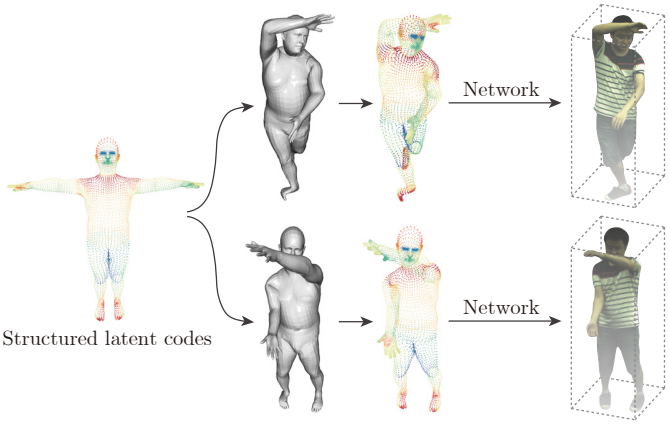


Fig. 1: **The basic idea of Neural Body.** Neural Body generates implicit 3D representations of a human body at different video frames from the same set of latent codes, which are anchored to the vertices of a deformable mesh. For each frame, we transform the spatial locations of codes based on the human pose, and use a network to regress the density and color for any 3D location based on the structured latent codes. Then, images at any viewpoints can be synthesized by the volume rendering.

NeRF [10] shows that photorealistic view synthesis can be achieved by representing 3D scenes as implicit fields of density and color, which are learned from images with a differentiable renderer. However, when the input views are highly sparse, the performance of [10] degrades dramatically, as shown by our experimental results in Section 4.3. The reason is that it is ill-posed to learn the neural representations with very sparse observations. We argue that the key to solving this ill-posed problem is to aggregate all

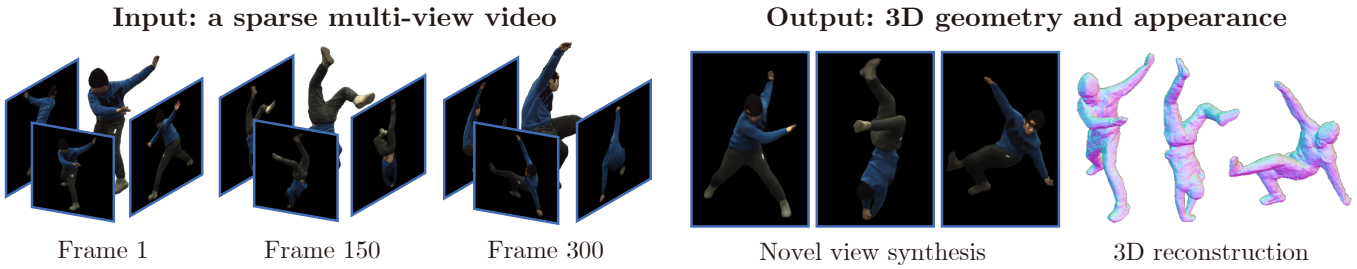


Fig. 2: **Novel view synthesis of a performer from a sparse multi-view video.** Neural Body captures the 3D geometry and appearance of the performer, which can be used for 3D reconstruction and novel view synthesis.

observations over different video frames. Lombardi et al. [11] implement this idea by regressing the 3D representation for each frame using the same network with different latent codes as input. Since the latent codes are independently obtained for each frame, it lacks sufficient constraints to effectively fuse observations across frames.

In this paper, we introduce a novel implicit neural representation for dynamic humans, named Neural Body, to solve the challenge of novel view synthesis from sparse views. The basic idea is illustrated in Figure 1. For the implicit fields at different frames, instead of learning them separately, Neural Body generates them from the same set of latent codes. Specifically, we anchor a set of latent codes to the vertices of a deformable human model (SMPL [12] in this work), namely that their spatial locations vary with the human pose. To obtain the 3D representation at a frame, we first transform the code locations based on the human pose, which can be reliably estimated from sparse camera views [13], [14], [15]. Then, a network is designed to regress the density and color for any 3D point based on these latent codes. Both the latent codes and the network are jointly learned from images of all video frames during the reconstruction process. This model is inspired by the latent variable model [16] in statistics, which enables us to effectively integrate observations at different frames. Another advantage of the proposed method is that the deformable model provides a geometric prior (rough surface location) to enable more efficient learning of implicit fields.

To evaluate our approach, we perform experiments on ZJU-MoCap [17] and Human 3.6M [18] datasets, which captures dynamic humans in complex motions with multiple synchronized cameras. Across all captured videos, our approach exhibits state-of-the-art performances on novel view synthesis. We also collect a synthetic dataset that contains high-quality 3D human models to evaluate our performance on 3D reconstruction. Furthermore, we also demonstrate the capability of our approach to capture moving humans from monocular RGB videos on the People-Snapshot dataset [19].

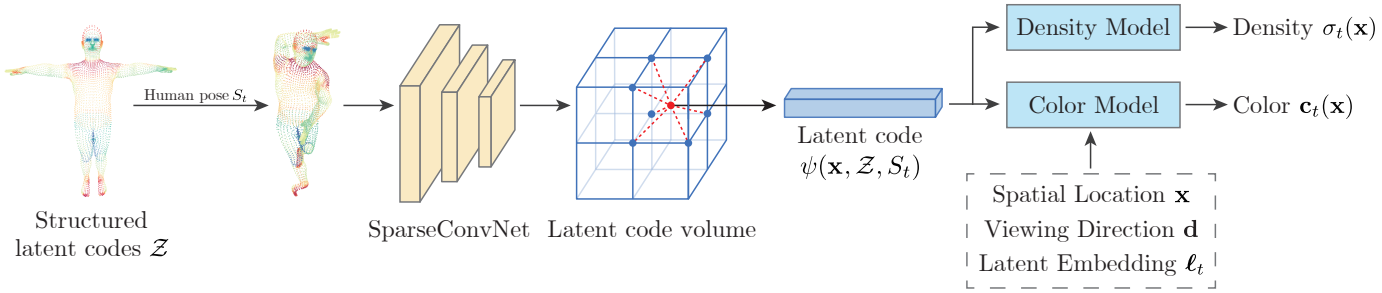
In the light of previous work, this work has the following contributions: i) We present a new approach capable of synthesizing photorealistic novel views of a performer in complex motions from a sparse multi-view video. ii) We propose Neural Body, a novel implicit neural representation for a dynamic human, which enables us to effectively incorporate observations over video frames. iii) We demonstrate significant performance improvements of our approach compared to prior work.

A preliminary version of this work appeared in CVPR 2021 [17]. Here, the work is extended in the following ways. First, inspired by [20], we integrate an implicit surface model into Neural Body, which effectively constrains the learned geometry during training and helps the geometry extraction. Second, we perform two additional ablation studies to analyze our approach. Third, additional experiments on ZJU-MoCap [17] and Human 3.6M [18] datasets are conducted to evaluate our approach and investigate the potential of Neural Body to synthesize performers under unseen human poses. We also add comparisons with two recent methods [21], [22]. Moreover, for quantitative comparison on 3D reconstruction, we create a synthetic dataset that has several multi-view videos and corresponding ground-truth human models, and validate the effectiveness of the adaptive sampling strategy.

2 Related Work

Image-based rendering. These methods aim to synthesize novel views without recovering detailed 3D geometry. Given densely sampled images, some works [1], [23] apply light field interpolation to obtain novel views. Although their rendering results are impressive, the range of renderable viewpoints is limited. To extend the range, [24], [25] infer depth maps from input images as proxy geometries. They utilize the depth to warp observed images into the novel view and perform image blending. However, these methods are sensitive to the quality of reconstructed proxy geometries. [2], [26], [27], [28], [29], [30], [31] replace handcrafted parts of the image-based rendering pipeline with learnable counterparts to improve the robustness.

Human performance capture. Most methods [3], [4], [7], [32] adopt the traditional modeling and rendering pipeline to synthesize novel views of humans. They rely on either depth sensors [3], [4], [33] or a dense array of cameras [7], [34] to achieve the high fidelity reconstruction. [35], [36], [37] improve the rendering pipeline with neural networks, which can be trained to compensate for the geometric artifacts. To capture human models in the highly sparse multi-view setting, template-based methods [38], [39], [40], [41] assume that there are pre-scanned human models. They reconstruct dynamic humans by deforming the template shapes to fit the input images. However, the deformed geometries tend to be unrealistic, and pre-scanned human shapes are unavailable in most cases. Recently, [42], [43], [44], [45] capture the human prior from training data using networks, which enables them to recover 3D human geometry and texture from a single image. However, it is difficult



(a) Code diffusion (b) Density and color regression

Fig. 3: **Implicit neural representation with structured latent codes.** (a) The structured latent codes are input into a SparseConvNet which outputs a latent code volume. This process diffuses the input codes defined on the surface to nearby 3D space. (b) For any 3D point, its latent code is obtained using trilinear interpolation from its neighboring vertices in the latent code volume and passed into MLP networks for density and color regression.

for them to achieve photo-realistic view synthesis or deal with people under complex human poses that are unseen during training.

Neural representation-based methods. In these works, deep neural networks are employed to learn scene representations from 2D images with differentiable renderers, such as voxels [11], [46], point clouds [37], [47], textured meshes [48], [49], [50], multi-plane images [51], [52], and implicit functions [8], [9], [10], [53], [54]. As a pioneer, SRN [8] proposes an implicit neural representation that maps xyz coordinates to feature vectors, and uses a differentiable ray marching algorithm to render 2D feature maps, which are then interpreted into images with a pixel generator. NeRF [10] represents scenes with implicit fields of density and color, which are well-suited for the differentiable rendering and achieve photorealistic view synthesis results. Instead of learning the scene with a single implicit function, our approach introduces a set of latent codes, which are used with a network to encode the local geometry and appearance. Furthermore, anchoring these codes to vertices of a deformable model enables us to represent a dynamic scene.

More recently, some works attempt to improve NeRF in various aspects. [31], [55], [56], [57], [58] add image features to the input of NeRF networks and train networks on a large amount of data, enabling them to infer complete 3D scenes from very sparse views. [21], [22], [59] establish dense correspondences across video frames by learning deformation fields. This explicitly integrates temporal information and enables them to work on monocular videos. To improve the reconstruction quality, [20], [60], [61] combine implicit surface models with volume rendering techniques.

3 Neural Body

Given a sparse multi-view video of a performer, our task is to generate a free-viewpoint video of the performer. We denote the video as $\{\mathcal{I}_t^c | c = 1, ..., N_c, t = 1, ..., N_t\}$, where c is the camera index, N_c is the number of cameras, t is the frame index, and N_t is the number of frames. The cameras are pre-calibrated. For each image, we apply [62] to obtain the foreground human mask and set the values of the background image pixels as zero.

The overview of the proposed model is illustrated in Figure 3. Neural Body starts from a set of structured latent codes attached to the surface of a deformable human model (Section 3.1). The latent code at any location around the surface can be obtained with a code diffusion process (Section 3.2) and then decoded to density and color values by neural networks (Section 3.3). The image from any viewpoint can be generated by volume rendering (Section 3.4). Considering that neural radiance fields tend to give noisy geometries [20], [60], [61], we replace radiance fields with implicit surface models [20] to improve the reconstruction performance of Neural Body (Section 3.6). The structured latent codes and neural networks are jointly learned by minimizing the difference between the rendered images and input images (Section 3.6).

Neural Body generates the human geometry and appearance at each frame from the same set of latent codes. From a statistical perspective, this is a type of latent variable model [16] that relates the observed variables at each frame to a set of latent variables. With such a latent variable model, we effectively integrate observations in the video.

3.1 Structured latent codes

To control the spatial locations of latent codes with the human pose, we anchor these latent codes to a deformable human body model (SMPL) [12]. SMPL is a skinned vertex-based model, which is defined as a function of shape parameters, pose parameters, and a rigid transformation relative to the SMPL coordinate system. The function outputs a posed 3D mesh with 6890 vertices. Specifically, we define a set of latent codes $\mathcal{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_{6890}\}$ on vertices of the SMPL model. For the frame t, SMPL parameters S_t are estimated from the multi-view images $\{\mathcal{I}_t^c | c = 1, ..., N_c\}$ using [63]. The spatial locations of the latent codes are then transformed based on the human pose S_t for the density and color regression. Figure 3 shows an example. The dimension of latent code $\boldsymbol{z}$ is set to 16 in our experiments.

Similar to the local implicit representations [64], [65], [66], the latent codes are used with a neural network to represent the local geometry and appearance of a human. Anchoring these codes to a deformable model enables us to represent a dynamic human. With the dynamic human representation, we establish a latent variable model that maps the same set of latent codes to the implicit fields

	Layer Description	Output Dim.
	Input volume	D×H×W×16
1-2	$(3 \times 3 \times 3$ conv, 16 features, stride 1$) \times 2$	D×H×W×16
3	$3 \times 3 \times 3$ conv, 32 features, stride 2	½D×½H×½W×32
4-5	$(3 \times 3 \times 3$ conv, 32 features, stride 1$) \times 2$	½D×½H×½W×32
6	$3 \times 3 \times 3$ conv, 64 features, stride 2	¼D×¼H×¼W×64
7-9	$(3 \times 3 \times 3$ conv, 64 features, stride 1$) \times 3$	¼D×¼H×¼W×64
10	$3 \times 3 \times 3$ conv, 128 features, stride 2	⅛D×⅛H×⅛W×128
11-13	$(3 \times 3 \times 3$ conv, 128 features, stride 1$) \times 3$	⅛D×⅛H×⅛W×128
14	$3 \times 3 \times 3$ conv, 128 features, stride 2	1/16D×1/16H×1/16W×128
15-17	$(3 \times 3 \times 3$ conv, 128 features, stride 1$) \times 3$	1/16D×1/16H×1/16W×128

TABLE 1: **Architecture of SparseConvNet.** Each layer consists of sparse convolution, batch normalization and ReLU.

of density and color at different frames, which naturally integrates observations.

3.2 Code diffusion

Figure 3(a) shows the process of code diffusion. The implicit fields assign the density and color to each point in the 3D space, which requires us to query the latent codes at continuous 3D locations. This can be achieved with the trilinear interpolation. However, since the structured latent codes are relatively sparse in the 3D space, directly interpolating the latent codes leads to zero vectors at most 3D points. To solve this problem, we diffuse the latent codes defined on the surface to nearby 3D space.

Inspired by [67], [68], [69], we choose the SparseConvNet [70] to efficiently process the structured latent codes, whose architecture is described in Table 1. Specifically, based on the SMPL parameters, we compute the 3D bounding box of the human and divide the box into small voxels with voxel size of $5mm \times 5mm \times 5mm$. The latent code of a non-empty voxel is the mean of latent codes of SMPL vertices inside this voxel. SparseConvNet utilizes 3D sparse convolutions to process the input volume and output latent code volumes with $2\times, 4\times, 8\times, 16\times$ downsampled sizes. With the convolution and downsampling, the input codes are diffused to nearby space. Following [68], for any point in 3D space, we interpolate the latent codes from multi-scale code volumes of network layers $5, 9, 13, 17$, and concatenate them into the final latent code. Since the code diffusion should not be affected by the human position and orientation in the world coordinate system, we transform the code locations to the SMPL coordinate system.

For any point $\mathbf{x}$ in 3D space, we query its latent code from the latent code volume. Specifically, the point $\mathbf{x}$ is first transformed to the SMPL coordinate system, which aligns the point and the latent code volume in 3D space. Then, the latent code is computed using the trilinear interpolation. For the SMPL parameters S_t, we denote the latent code at point $\mathbf{x}$ as $\psi(\mathbf{x}, \mathcal{Z}, S_t)$. The code vector is passed into MLP networks to predict the density and color for point $\mathbf{x}$.

3.3 Density and color regression

Figure 3(b) overviews the regression of density and color for any point in 3D space. The density and color fields are represented by MLP networks. The details of network architectures can be found in the released code at https://github.com/zju3dv/neuralbody/.

Density model. For the frame t, the volume density at point $\mathbf{x}$ is predicted as a function of only the latent code $\psi(\mathbf{x}, \mathcal{Z}, S_t)$, which is defined as:

$$\sigma_t(\mathbf{x}) = M_\sigma(\psi(\mathbf{x}, \mathcal{Z}, S_t)), \tag{1}$$

where M_σ represents an MLP network with four layers.

Color model. Similar to [10], [11], we take both the latent code $\psi(\mathbf{x}, \mathcal{Z}, S_t)$ and the viewing direction $\mathbf{d}$ as input for the color regression. To model the location-dependent incident light, the color model also takes the spatial location $\mathbf{x}$ as input. We observe that temporally-varying factors affect the human appearance, such as secondary lighting and self-shadowing. Inspired by the auto-decoder [71], we assign a latent embedding $\boldsymbol{\ell}_t$ for each video frame t to encode the temporally-varying factors.

Specifically, for the frame t, the color at $\mathbf{x}$ is predicted as a function of the latent code $\psi(\mathbf{x}, \mathcal{Z}, S_t)$, the viewing direction $\mathbf{d}$, the spatial location $\mathbf{x}$, and the latent embedding $\boldsymbol{\ell}_t$. Following [10], [72], we apply the positional encoding to both the viewing direction $\mathbf{d}$ and the spatial location $\mathbf{x}$, which enables better learning of high frequency functions. The color model at frame t is defined as:

$$\mathbf{c}_t(\mathbf{x}) = M_{\mathbf{c}}(\psi(\mathbf{x}, \mathcal{Z}, S_t), \gamma_{\mathbf{d}}(\mathbf{d}), \gamma_{\mathbf{x}}(\mathbf{x}), \boldsymbol{\ell}_t), \tag{2}$$

where $M_{\mathbf{c}}$ represents an MLP network with two layers, and $\gamma_{\mathbf{d}}$ and $\gamma_{\mathbf{x}}$ are positional encoding functions for viewing direction and spatial location, respectively. We set the dimension of $\boldsymbol{\ell}_t$ to 128 in experiments. During rendering images of novel human poses, we simply fix $\boldsymbol{\ell}_t$ as the latent embedding of the last training frame.

3.4 Volume rendering

Given a viewpoint, we utilize the classical volume rendering techniques to render the Neural Body into a 2D image. The pixel colors are estimated via the volume rendering integral equation [73] that accumulates volume densities and colors along the corresponding camera ray. In practice, the integral is approximated using numerical quadrature [10], [74]. Given a pixel, we first compute its camera ray $\mathbf{r}$ using the camera parameters. Then, the camera ray $\mathbf{r}$ intersects with the bounding box of the SMPL model, which gives near and far intersection points. Based on the near and far bounds, we use a stratified sampling approach [10] to sample N_k points $\{\mathbf{x}_k\}_{k=1}^{N_k}$ along camera ray $\mathbf{r}$. Then, Neural Body predicts volume densities and colors at these points. For the video frame t, the rendered color $\tilde{C}_t(\mathbf{r})$ of the corresponding pixel is given by:

$$\tilde{C}_t(\mathbf{r}) = \sum_{k=1}^{N_k} T_k(1 - \exp(-\sigma_t(\mathbf{x}_k)\delta_k))\mathbf{c}_t(\mathbf{x}_k), \tag{3}$$

$$\text{where} \quad T_k = \exp(-\sum_{j=1}^{k-1} \sigma_t(\mathbf{x}_j)\delta_j), \tag{4}$$

where $\delta_k = ||\mathbf{x}_{k+1} - \mathbf{x}_k||_2$ is the distance between adjacent sampled points. We set N_k as 64 in all experiments. With volume rendering, our model is optimized by comparing the rendered and observed images.

3.5 Neural Body with implicit surface models

Neural radiance fields with volume rendering achieves impressive performance on novel view synthesis. However, the volume rendering does not constrain the learned geometry, resulting in that radiance fields tend to produce noisy geometries. Moreover, NeRF does not model the geometry

as a particular level set of the density function. As a result, we need to carefully select the density threshold to extract the geometry. To overcome these problems, UNISURF [20] proposes to replace neural radiance fields with occupancy fields and improves the rendering process with the surface guidance. We introduce these strategies to Neural Body to improve the performance on 3D reconstruction.

Specifically, we first revise the density model to output a value $\sigma_t^o(\mathbf{x})$ between 0 and 1 at the point $\mathbf{x}$, where $\sigma_t^o(\mathbf{x}) = 0$ means free space and $\sigma_t^o(\mathbf{x}) = 1$ means occupied space. With the occupancy field, the volume rendering equation of solid objects like humans can be rewritten as

$$\tilde{C}_t(\mathbf{r}) = \sum_{k=1}^{N_k} \sigma_t^o(\mathbf{x}_k) \prod_{j=1}^{k-1} (1 - \sigma_t^o(\mathbf{x}_j)) \mathbf{c}_t(\mathbf{x}_k), \tag{5}$$

as shown in UNISURF [20]. We also add the normal $\mathbf{n}(\mathbf{x})$ at point $\mathbf{x}$ as an input to the color network:

$$\mathbf{c}_t(\mathbf{x}) = M_{\mathbf{c}}(\psi(\mathbf{x}, \mathcal{Z}, S_t), \mathbf{n}(\mathbf{x}), \gamma_{\mathbf{d}}(\mathbf{d}), \gamma_{\mathbf{x}}(\mathbf{x}), \boldsymbol{\ell}_t). \tag{6}$$

Surface-guided sampling. In original NeRF, it takes the stratified sampling approach to sample points along camera rays during the volume rendering. To encourage surface points to contribute more in the volume rendering, we adopt the surface-guided sampling strategy as in UNISURF [20]. Specifically, given the occupancy field, we first utilize [9] to obtain the surface point of the camera ray $\mathbf{r}$. Denote the depth of surface point as D. Then, the stratified sampling strategy is applied within the depth interval $[D-\Delta, D+\Delta]$. During training, the sampling interval Δ monotonically decreases with the iteration number, which is defined as:

$$\Delta_k = \max(\Delta_{\text{max}} \exp(-i\beta), \Delta_{\text{min}}), \tag{7}$$

where i is the iteration number, and β, Δ_{min} and Δ_{max} are hyperparameters of the interval function. We follow UNISURF [20] to set β as $1.5e-5$, Δ_{min} as 0.05, and Δ_{max} as 1.0. For camera rays that do no have intersection points with the surface, we use the sampling strategy described in Section 3.4. Experimental results show that combining implicit surface models with Neural Body improves the performance on 3D reconstruction.

3.6 Training

Through the volume rendering techniques, we optimize the Neural Body to minimize the rendering error of observed images $\{\mathcal{I}_t^c | c = 1, ..., N_c, \, t = 1, ..., N_t\}$:

$$\underset{\{\boldsymbol{\ell}_t\}_{t=1}^{N_t}, \mathcal{Z}, \Theta}{\text{minimize}} \sum_{t=1}^{N_t} \sum_{c=1}^{N_c} L(\mathcal{I}_t^c, P^c; \boldsymbol{\ell}_t, \mathcal{Z}, \Theta), \tag{8}$$

where Θ means the network parameters, P^c is the camera parameters, and L is the total squared error that measures the difference between the rendered and observed images. The corresponding loss function is defined as:

$$L = \sum_{\mathbf{r} \in \mathcal{R}} ||\tilde{C}(\mathbf{r}) - C(\mathbf{r})||_2, \tag{9}$$

where $\mathcal{R}$ is the set of camera rays passing through image pixels, and $C(\mathbf{r})$ means the ground-truth pixel color. In contrast to frame-wise reconstruction methods [6], [10], our method optimizes the model using all images in the video and has more information to recover the 3D structures.

subject	Twirl	Taichi	Swing1	Swing2	Swing3	Warmup	Punch1	Punch2	Kick
training	60	400	300	300	300	300	300	300	300
test	1000	1000	356	559	358	317	346	354	700

TABLE 2: The number of training frames and test frames for each subject in the ZJU-MoCap dataset.

We adopt the Adam optimizer [75] for training the Neural Body. The learning rate starts from $5e^{-4}$ and decays exponentially to $5e^{-5}$ along the optimization. We conduct the training on four 2080 Ti GPUs. The training on a four-view video of 300 frames typically takes around 200k iterations to converge (about 14 hours).

3.7 Applications

The trained Neural Body can be used for novel view synthesis, novel pose synthesis and 3D reconstruction of the performer. 1) The view synthesis is achieved through the volume rendering. Novel view synthesis on dynamic humans results in free-viewpoint videos, which give the viewers the freedom to watch human performers from arbitrary viewpoints. Our experimental results show that the generated videos exhibit high inter-frame and inter-view consistency, which are presented in the project page. 2) For novel pose synthesis, given a human pose, we transform the spatial locations of latent codes based on the SMPL deformation framework. Then, the structured latent codes are input into the network to obtain the 3D representation of target human, which can be used to render images. 3) For 3D reconstruction, we first discretize the scene with a voxel size of $5mm \times 5mm \times 5mm$. Then, we evaluate the volume densities for all voxels and extract the human mesh with the Marching Cubes algorithm [76].

4 EXPERIMENTS

4.1 Datasets and metrics

ZJU-MoCap [17] is a multi-view dataset that captures 9 dynamic human videos using a multi-camera system that has 21 synchronized cameras. The humans perform complex motions, including twirling, Taichi, arm swings, warmup, punching, and kicking. The split of training and test frames is listed in Table 2. We select four cameras for training and use the remaining cameras for testing. Since the inter-frame human images are very similar, we calculate the metrics every 30 frames on this dataset.

Human3.6M [18] records multi-view videos with 4 cameras and collects human poses using the marker-based motion capture system. It includes multiple subjects performing complex actions. Its videos have a length between 200 to 500 frames. We exactly follow the experiment setting of Ani-NeRF [22] on this dataset.

RenderPeople is a multi-view dataset that contains ground-truth 3D human shapes for each video frame. We collect 5 high-quality animated human models from RenderPeople [78] and Mixamo [79]. Then, each dynamic human model is rendered into 10 camera views with BlenderProc [80], [81]. We select four uniformly distributed cameras for training. Considering that the rendered images lack of complicated illumination conditions, we only use this dataset to evaluate

	PSNR ↑						SSIM ↑					
	NV [11]	NT [48]	NHR [37]	Ani-NeRF [22]	OURS	OURS*	NV [11]	NT [48]	NHR [37]	Ani-NeRF [22]	OURS	OURS*
Twirl	20.09	25.78	26.68	29.27	**30.54**	30.43	0.831	0.929	0.935	0.962	0.969	**0.970**
Taichi	18.57	19.44	19.81	24.22	**27.42**	26.22	0.824	0.869	0.874	0.922	**0.963**	0.957
Swing1	22.88	24.96	24.73	27.79	**29.69**	29.04	0.726	0.905	0.902	0.928	0.948	**0.949**
Swing2	22.08	24.84	25.01	26.06	**28.64**	28.60	0.843	0.903	0.906	0.916	0.939	**0.942**
Swing3	21.29	23.50	23.47	27.53	**27.66**	27.22	0.842	0.896	0.894	0.925	0.938	**0.939**
Warmup	21.15	23.74	23.79	26.63	**27.89**	27.73	0.842	0.917	0.918	0.941	0.955	**0.957**
Punch1	23.21	24.93	25.02	26.78	28.39	**28.99**	0.820	0.877	0.879	0.891	0.928	**0.937**
Punch2	20.74	22.44	22.88	24.75	25.83	**26.33**	0.838	0.888	0.891	0.913	0.928	**0.935**
Kick	22.49	24.33	23.72	26.19	27.32	**27.95**	0.825	0.881	0.873	0.915	0.924	**0.933**
average	21.39	23.77	23.90	26.58	**28.15**	28.05	0.821	0.896	0.897	0.924	0.943	**0.947**

TABLE 3: **Results of novel view synthesis on the ZJU-MoCap dataset in terms of PSNR and SSIM metrics.** "Ours*" means Neural Body with implicit surface model introduced in Section 3.5, "NV" means Neural Volumes, and "NT" means Neural Textures. Results of Ani-NeRF were obtained from [77]. Note that we re-trained Neural Body, so the results of "OURS" are slightly different from the conference version [17].

	PSNR ↑					SSIM ↑				
	NT	NHR	Ani-NeRF	OURS	OURS*	NT	NHR	Ani-NeRF	OURS	OURS*
Twirl	22.56	23.05	23.61	23.24	**24.08**	0.889	0.893	0.908	0.897	**0.911**
Taichi	18.38	18.88	19.45	19.87	**20.65**	0.841	0.844	0.854	0.863	**0.883**
Swing1	24.08	23.66	24.15	25.70	**25.95**	0.900	0.893	0.900	0.914	**0.924**
Swing2	22.67	22.87	23.97	24.64	**25.11**	0.871	0.874	0.899	0.893	**0.907**
Swing3	22.45	22.27	**24.29**	23.84	24.22	0.888	0.885	0.893	0.903	**0.913**
Warmup	22.07	21.94	25.03	24.96	**25.18**	0.886	0.885	0.927	0.922	**0.931**
Punch1	23.70	23.70	25.14	26.61	**27.29**	0.851	0.853	0.878	0.895	**0.915**
Punch2	20.64	20.97	22.94	24.50	**23.64**	0.862	0.866	0.892	0.886	**0.906**
Kick	22.90	22.65	24.51	23.08	**25.18**	0.864	0.858	0.889	0.889	**0.903**
average	22.16	22.22	23.68	24.05	**24.59**	0.872	0.872	0.893	0.896	**0.910**

TABLE 4: **Results of novel pose synthesis on the ZJU-MoCap dataset in terms of PSNR and SSIM.** "Ours*" means Neural Body with implict surface model. Results of Ani-NeRF were obtained from [77].

the reconstruction performance. Since the inter-frame human models are very similar, we calculate the reconstruction metric every 10 frames on this dataset.

People-Snapshot [19] captures monocular videos with a fixed camera. Its performers rotate while holding an A-pose, whose human poses are relatively easy and enables us to recover accurate SMPL parameters. This dataset exhibits many challenges for differentiable rendering-based methods, including single view, loose clothing, and varying lighting conditions. Following [19], we only present qualitative results on this dataset.

Metrics. For image synthesis, we follow [10] to evaluate our method using two standard metrics: peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). The performances on novel view synthesis and novel pose synthesis are evaluated on the training and test frames, respectively. Note that both of them are evaluated on the test camera views. For 3D reconstruction, we follow [43] to adopt two metrics: point-to-surface Euclidean distance (P2S) and Chamfer distance. Units for the two metrics are in cm.

4.2 Baseline methods

Image synthesis. We compare our method with state-of-the-art view synthesis methods [11], [37], [48] that handle dynamic scenes. All methods train a separate network for each scene. 1) Neural Volumes (NV) [11] encodes multi-view images at each frame into a latent vector and decodes it into a discretized RGBα voxel grid. 2) Neural Textures (NT) [48] proposes latent texture maps to render a coarse mesh into 2D images. Since [48] is not open-sourced, we reimplement it and take the SMPL mesh as the input mesh. 3) NHR [37] uses networks to render input point clouds to images. Here we take SMPL vertices as input point clouds. 4) D-NeRF [21] encodes a video as a canonical NeRF and a set of deformation fields that establish correspondences between the canonical space and observation spaces. It uses translational vector fields to represent deformation fields. 5) Ani-NeRF [22] also use a canonical NeRF and deformation fields to represent the video. It adopts the linear blend skinning model to calculate deformation fields.

3D reconstruction. We compare our method with recent methods [22], [82]. 1) IDR [82] represents 3D scenes as signed distance fields and learns it from images with differentiable sphere tracing. Note that it can only handle static scenes. 2) Ani-NeRF [22] can be used for reconstruction using Marching Cubes [76], similar to Neural Body.

4.3 Results on the ZJU-MoCap dataset

Performance on novel view synthesis. Table 3 compares our method with [11], [22], [37], [48] in terms of the PSNR and SSIM metrics. For both metrics, our model achieves the best performances among all methods. In particular, our method outperforms previous works by a margin of at least 1.57 in terms of PSNR and 0.019 in terms of SSIM.

In contrast to learning the 3D representations from individual latent vectors [11], Neural Body generates implicit fields at different frames from the same set of latent codes. The results indicate that our method better integrates observations of the target performer across video frames.

Figure 4 shows the qualitative results of our method and other methods [11], [37], [48]. The rendering results of [11] indicate that they don't accurately capture the 3D human geometry and appearance. Neural Volumes [11] gives blurry results. As image-to-image translation methods, [37], [48] have difficulty in controlling the rendering viewpoints. In contrast, our method gives photorealistic novel views.

Performance on novel pose synthesis. Table 4 shows the comparison of our method with [11], [22], [37], [48] in terms of the PSNR and SSIM metrics. Since Neural Volumes [11] is designed for over-fitting a video sequence, we do not compare with it. Our method is comparable with Ani-NeRF [22] and outperforms other methods. Qualitative results are presented in Figure 4.

Results on novel pose synthesis indicate that Neural Body can generalize to novel human poses. The reason may

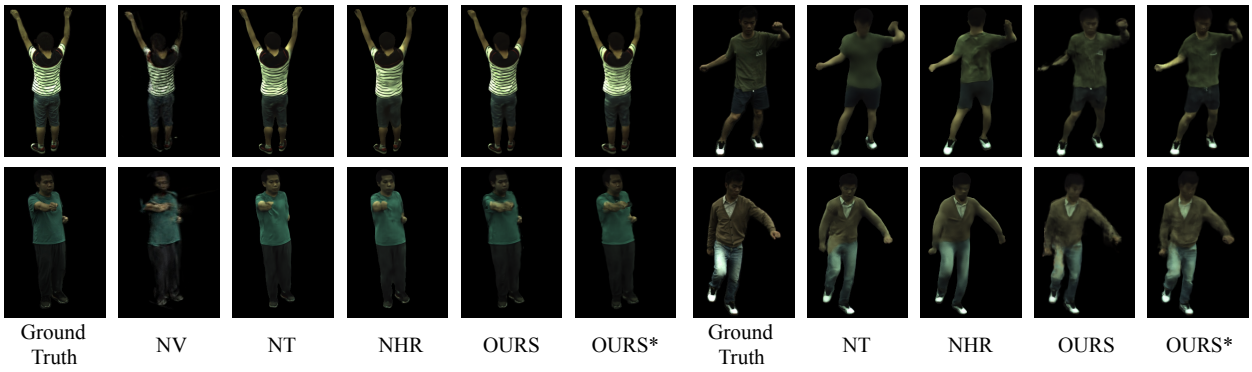


Fig. 4: **Qualitative comparison on the ZJU-MoCap dataset.** The results of two subjects on the left are novel views of training human poses, and the results of two subjects on the right are renderings of novel human poses. "NV" means Neural Volumes [11], "NT" means Neural Textures [48], and "Ours*" means Neural Body with implicit surface model.

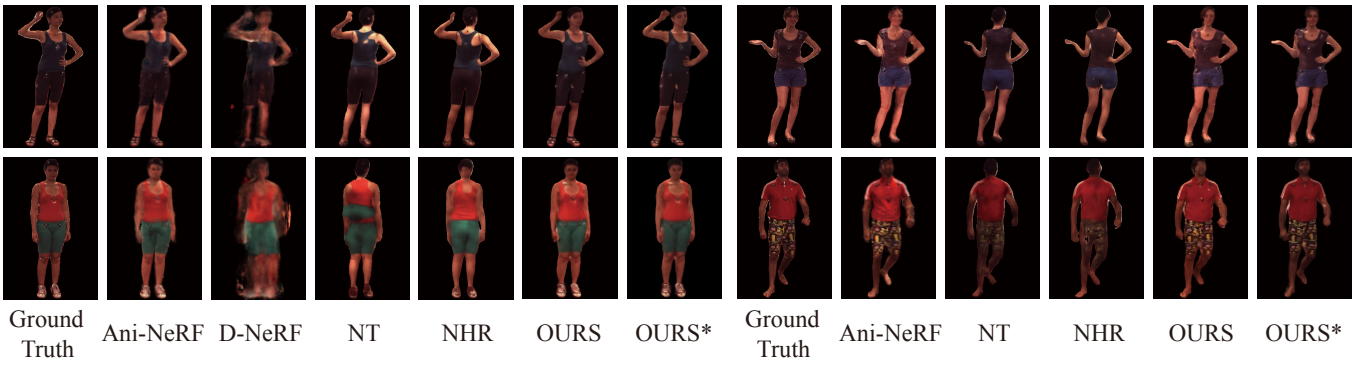


Fig. 5: **Qualitative comparison on the Human3.6M dataset.** We present novel views of training human poses of two subjects on the left, and show rendering results of novel human poses of two subjects on the right. "Ours*" means Neural Body with implicit surface model.

	PSNR ↑						SSIM ↑					
	NT [48]	NHR [37]	D-NeRF [21]	Ani-NeRF [22]	OURS	OURS*	NT [48]	NHR [37]	D-NeRF [21]	Ani-NeRF [22]	OURS	OURS*
S1	20.98	21.08	19.63	22.05	22.87	**23.83**	0.860	0.872	0.838	0.888	0.897	**0.905**
S5	19.87	20.64	20.92	23.27	24.60	**24.61**	0.855	0.872	0.807	0.892	**0.917**	**0.917**
S6	20.18	20.40	20.64	21.13	22.82	**24.53**	0.816	0.830	0.811	0.854	0.888	**0.901**
S7	20.47	20.29	17.90	22.50	23.17	**24.46**	0.856	0.868	0.722	0.890	0.914	**0.915**
S8	16.77	19.13	20.81	22.75	21.72	**23.48**	0.837	0.871	0.845	0.898	0.894	**0.913**
S9	22.96	23.04	23.79	24.72	24.28	**26.04**	0.873	0.879	0.889	0.908	0.910	**0.919**
S11	21.71	21.91	17.23	**24.55**	23.70	24.14	0.859	0.871	0.737	**0.902**	0.896	**0.902**
average	20.42	20.93	20.13	23.00	23.31	**24.44**	0.851	0.866	0.807	0.890	0.903	**0.910**

TABLE 5: **Results of novel view synthesis on Human3.6M**. "Ours*" means Neural Body with implicit surface model.

be that, after training on seen human poses, SparseConvNet is able to extract meaningful feature volumes from structured latent codes under novel poses. Such generalization ability of SparseConvNet has also been demonstrated in 3D segmentation [70] and detection [68] tasks, where SparseConvNet needs to processes various point clouds. However, Neural Body does not generalize well to human poses that are very different from training poses, which are discussed in Section 5. Experimental results also show that the implicit surface model introduced in Section 3.5 contributes a lot to performance of novel pose synthesis, as it can produce more accurate geometry information.

4.4 Results on the Human3.6M dataset

To extensively evaluate our model, we perform experiments on the Human3.6M [18] dataset that collects accurate SMPL parameters with a marker-based motion capture system.

	PSNR ↑					SSIM ↑				
	NT [48]	NHR [37]	Ani-NeRF [22]	OURS	OURS*	NT [48]	NHR [37]	Ani-NeRF [22]	OURS	OURS*
S1	20.09	20.48	21.37	22.11	**23.30**	0.837	0.853	0.868	0.879	**0.890**
S5	20.03	20.72	22.29	**23.51**	23.34	0.843	0.860	0.875	**0.897**	0.891
S6	20.42	20.47	22.59	23.52	**24.41**	0.844	0.856	0.884	0.889	**0.896**
S7	20.03	19.66	22.22	22.33	**23.15**	0.838	0.852	0.878	**0.889**	0.883
S8	16.69	18.83	21.78	20.94	**22.53**	0.824	0.855	0.882	0.876	**0.893**
S9	22.20	22.18	23.72	23.04	**24.48**	0.851	0.860	0.886	0.884	**0.892**
S11	21.72	22.12	**23.91**	23.72	23.90	0.854	0.867	**0.889**	0.884	0.884
average	20.17	20.64	22.55	22.74	**23.59**	0.842	0.858	0.880	0.885	**0.890**

TABLE 6: **Results of novel pose synthesis on Human3.6M**. "Ours*" means Neural Body with implicit surface model.

Only three camera views are used during training, which makes the novel view synthesis more challenging.

Performance on novel view synthesis. We compare our method with [21], [22], [37], [48] in Table 5. For both PSNR and SSIM metrics, our approaches achieves the best performances among all methods.

Performance on novel pose synthesis. Table 6 shows the comparison of our method with [22], [37], [48] in terms of

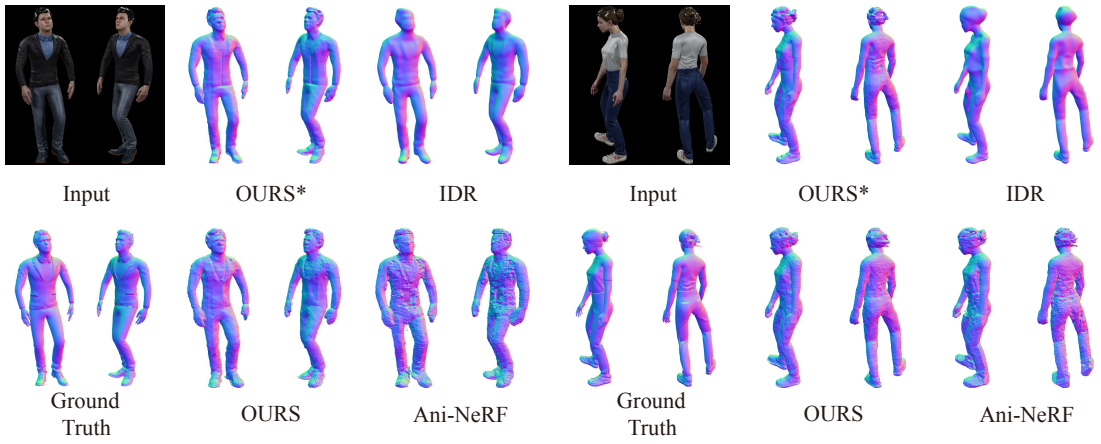


Fig. 6: **3D reconstruction on the RenderPeople dataset.** IDR is trained separately per frame, while other methods are trained on the whole video. Our method with implicit surface model has better reconstruction results than other methods.

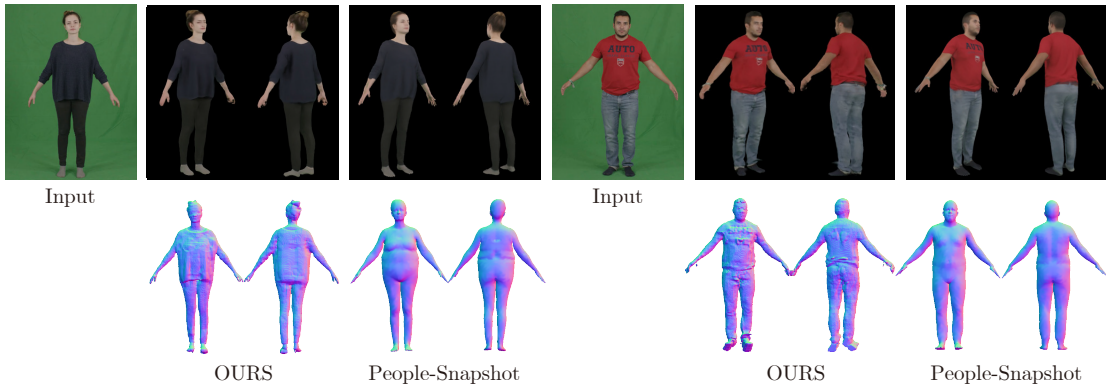


Fig. 7: **Qualitative results on monocular videos.** Our method renders more appearance details and reconstruct more geometric details than People-Snapshot [19]. "Ours" indicates Neural Body without implicit surface model.

	Chamfer Distance ↓			P2S ↓		
	Ani-NeRF [22]	Ours	Ours*	Ani-NeRF [22]	Ours	Ours*
manuel	1.77	0.82	**0.74**	2.29	0.84	**0.81**
megan	1.45	1.11	**0.49**	1.55	1.00	**0.36**
lenonard	1.66	1.18	**0.61**	1.88	1.14	**0.55**
josh	1.73	1.31	**0.67**	1.90	1.17	**0.47**
jody	1.89	1.46	**0.71**	2.18	1.32	**0.57**
average	1.70	1.17	**0.64**	1.96	1.09	**0.55**

TABLE 7: **Results of 3D reconstruction on the RenderPeople dataset in terms of Chamfer distance and P2S (lower is better).** "Ours*" means Neural Body with implicit surface model. The testing is performed on some sampled frames with an interval of 10 frames.

	Chamfer Distance ↓				P2S ↓			
	Ani-NeRF [22]	IDR [82]	Ours	Ours*	Ani-NeRF [22]	IDR [82]	Ours	Ours*
manuel	1.68	1.15	0.78	**0.73**	2.29	1.05	0.80	**0.73**
megan	1.46	0.90	1.14	**0.48**	1.58	0.73	1.02	**0.33**
lenonard	1.73	1.29	1.22	**0.63**	1.96	1.10	1.18	**0.62**
josh	1.78	0.83	1.31	**0.66**	1.89	0.50	1.16	**0.45**
jody	1.93	1.11	1.50	**0.73**	2.21	0.81	1.40	**0.61**
average	1.71	1.06	1.19	**0.65**	1.98	0.84	1.11	**0.55**

TABLE 8: **Results of single frame reconstruction on RenderPeople dataset in terms of Chamfer distance and P2S (lower is better).** The testing is performed on the first frame. IDR [82] is trained on the first frame, and other methods are trained on the whole video.

the PSNR metric and the SSIM metric, respectively. Since D-NeRF [21] is not designed for animation, we do not compare with it. For both metrics, our approach gives the best results among all methods. Figure 5 presents the qualitative results of our method and other baselines on image synthesis, which indicates that our method has a higher rendering quality than other methods.

4.5 Results on the RenderPeople dataset

We use the RenderPeople dataset to demonstrate that our method can reconstruct accurate geometries from sparse input videos. Only four camera views are used for training, which makes this task challenging.

We compare our method with the latest dynamic human reconstruction method, Ani-NeRF [22], and the method that handles static scenes, IDR [82]. As shown in Table 7, under the task of dynamic reconstruction, our method significantly outperform Ani-NeRF in both P2S and Chamfer distance. The reason behind this phenomenon can be inferred from qualitative result Figure 6, in which we can observe that Ani-NeRF [22] produces noisy reconstruction results due to the lack of surface constraints on the density fields.

Furthermore, it can also be concluded from Table 7 that the implicit surface model introduced in Section 3.5 greatly improves the reconstruction accuracy. The mesh generated from Neural Body without the assistance of implicit surface

	1 view	2 views	4 views	6 views
PSNR	25.08	25.49	30.54	**32.73**
SSIM	0.912	0.928	0.969	**0.979**

TABLE 9: **Results of models trained with different numbers of camera views** on the video "Twirl" of the ZJU-MoCap dataset. We select six camera views for ablation studies and use the remaining views for test.

model is much less smooth, because the density field does not sufficiently regularize the underlying 3D geometry.

To show the effectiveness of leveraging temporal information, we compare our reconstruction results in the first frame of each test series with IDR [82], which is trained separately on each frame. Our method gives the best results among all methods, which shows that aggregating information across video frames benefits the reconstruction.

4.6 Results on the People-Snapshot dataset

We demonstrate that our approach is able to reconstruct dynamic humans from monocular videos on the People-Snapshot dataset [19]. We compare Neural Body with the approach proposed in [19], which deforms vertices of the SMPL model to fit the 2D human silhouettes over the video sequence. Following [19], only the qualitative results are reported on the People-Snapshot dataset.

Figure 7 shows the qualitative comparison on novel view synthesis and 3D reconstruction. Our method reconstructs more appearance and geometric details than [19]. For example, the hair shapes are highly consistent with the RGB observations. The results of the first column indicate that our method can handle persons wearing loose clothing, while [19] does not recover correct shapes for such data.

4.7 Ablation studies on the ZJU-Mocap dataset

We conduct ablation studies on the video "Twirl". We first analyze the effects of per-frame latent embedding. Then we explore the performances of our models trained with different numbers of video frames and input views. Considering that Neural Body is based on the SMPL model, we also train Neural Body on a human wearing a dress to see if it can work on humans with loose clothes. Finally, we compare different ways to diffuse the structured latent codes.

Impact of per-frame latent embedding. We train a model without latent embeddings $\{\boldsymbol{\ell}_t\}_{t=1}^{N_t}$ that are proposed in Section 3.3, which gives 30.03 PSNR, lower than 30.54 PSNR of the complete model. This comparison indicates that the latent embeddings yield 0.53 PSNR improvement.

Impact of the number of camera views. Table 9 compares our models trained with different numbers of camera views. The results show that the number of training views improves the performance on novel view synthesis. Neural Body trained on single view still outperforms [11] trained on four views, which gives 23.12 PSNR and 0.875 SSIM on test views of the ablation study.

Impact of the video length. We train our model with 1, 60, 300, 600, and 1200 frames, respectively. Table 10 shows the quantitative results, which indicate that training on the video improves the view synthesis performance, but training on too many frames may decrease the performance

	Frames	1	60	300	600	1200
Training frame	PSNR	25.64	30.14	**30.66**	30.59	29.97
	SSIM	0.940	0.970	**0.971**	0.970	0.970
Test frame	PSNR	21.39	22.70	23.00	22.71	**24.45**
	SSIM	0.832	0.882	0.894	0.889	**0.910**

TABLE 10: **Results of models trained with different numbers of training frames.** We train models on 1, 60, 300, 600, and 1200 frames of "Twirl". On novel view synthesis of training frame, the first training frame is selected for test, and the images from 1201-st frame to 1400-th frame are used for evaluating novel pose synthesis.

	PSNR	SSIM	Iteration time
Ball query	18.23	0.797	**0.1045**
PointNet++	26.05	0.931	0.8555
SparseConvNet	**30.54**	**0.969**	0.1748

TABLE 11: **Results of models with different diffusion methods** on the video "Twirl" of the ZJU-MoCap dataset. The iteration time means the time each iteration takes during the training. The unit for the iteration time is in second.

on novel view synthesis of training frames as the network has difficulty in fitting very long videos. In addition, results in Table 10 indicate that more training frames enable Neural Body to generalize better to novel human poses.

Impact of code diffusion method. Section 3.2 proposes to use SparseConvNet to diffuse the latent codes on the surface to nearby 3D space. To validate its effectiveness, we here compare it with two other fusion ways: 1) Ball query [83]. For any point, we find all SMPL vertices that within a radius to it (at most K vertices) and apply a PointNet layer [84] to the latent codes of these vertices to obtain the point feature. 2) PointNet++ [83]. We first perform a hierarchical feature learning on the structured latent codes with PointNet++. Then, for any 3D point, we use the multi-scale grouping to extract features at different scales, which are concatenated to form a multi-scale feature.

Table 11 compares the three fusion methods in terms of the quality of iamge synthesis and the running time during training. The SparseConvNet significantly outperforms the two other fusion methods in terms of the PSNR and SSIM metrics. And SparseConvNet is much faster than PointNet++ in term of the training time.

Impact of image perceptual loss. As described in Section 3.6, we simply adopt the MSE loss for training Neural Body. It is interesting to investigate the impact of using a more complex loss function, such as image perceptual loss in [37], on the performance of the model. To this end, we train a model that is supervised with both the image perceptual loss and the MSE loss. We found that this model exhibits similar rendering performance to the original model (PSNR: 30.70 vs. 30.54, SSIM: 0.970 vs. 0.969).

5 Discussion

Limitations. Although Neural Body has produced high-quality free-viewpoint synthesis, it still has some limitations. (1) Since Neural Body is built on naked parametric model [12], it struggles to give photorealistic rendering results for people wearing loose clothes. To illustrate this problem, we select the video sequence "Magdalena"

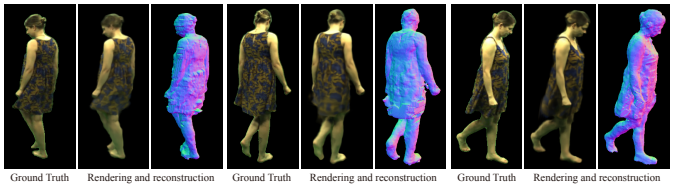


Fig. 8: **Novel view synthesis on the video "Magdalena".** The rendered dress tends to be blurry. The reason may be that the dress's hem deforms non-rigidly along with the human movement, and latent codes around the hem will correspond to different human geometry and appearance at different video frames. Consequently, different content will be encoded into the same latent code, averaging the information of the latent code and degrading the performance of Neural Body. The visualization also indicates that Neural Body does not correctly reconstruct the geometry of hem. Note that this model is without implicit surface model.

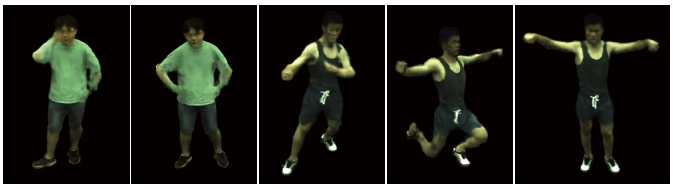


Fig. 9: **Results of human poses that are very different from training poses.** We animates human subjects in ZJU-MoCap dataset with human poses from Human3.6M dataset. The rendered faces and bodies are distorted and blurry, indicating that Neural Body has limited generalization ability on novel pose synthesis.

from the DeepCap dataset [85], which captures that a performer wearing a loose dress walks around. Figure 8 shows the qualitative results. Although rendered images appear natural-looking, the rendering results of the loose dress are blurry and have some spatial misalignments from the observed images. Replacing SMPL model with a clothed parametric model [86] is a possible solution for this limitation. (2) Neural Body has difficulty in representing animatable avatars. Its performance on novel pose synthesis seems reasonable when test poses are similar to training poses, as indicated by experiments. However, when the test pose is quite different from training poses, Neural Body fails to render high-quality images. Figure 9 presents some examples. (3) Neural Body is sensitive to inaccurate SMPL poses, which can cause misalignment between structured latent codes and observed images. It may be addressed by optimizing input SMPL poses along with Neural Body.

Future perspectives. In this paper, we have reached a stage that we can create free-viewpoint videos of dynamic human performers from sparse multi-view videos. Nevertheless, there is still room for future advancement. (1) Relightable human avatar is desirable, as many applications require placing reconstructed avatars into scenes under different lighting conditions. (2) To faithfully describe digital humans, we need to animate avatars with not only body postures but also facial expressions and hand poses. (3) For the richness of user interaction, digital avatars should be editable, such as changing human clothes, modifying haircuts and putting on accessories.

6 Conclusion

We introduced a novel implicit neural representation, named Neural Body, for novel view synthesis of dynamic humans from sparse multi-view videos. Neural Body defines a set of latent codes, which encode local geometry and appearance with a neural network. We anchored these latent codes to vertices of a deformable human model to represent a dynamic human. This enables us to establish a latent variable model that generates implicit fields at different video frames from the same set of latent codes, which effectively incorporates observations of the performer across video frames. We learned Neural Body over the video with volume rendering. To improve the reconstruction quality, we combined Neural Body with an implicit surface model that efficiently regularizes the learned geometry. To evaluate our approach, we created a multi-view dataset called ZJU-MoCap that captures dynamic humans in complex motions, and a synthetic dataset called RenderPeople that contains ground-truth human shapes. We demonstrated superior view synthesis quality and better reconstruction performance compared to prior work on the ZJU-MoCap, Human3.6M, RenderPeople, and People-Snapshot datasets.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0108901, in part by NSFC under Grant 62172364, and in part by the Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

References

[1] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *SIGGRAPH*, 1996.
[2] P. Hedman, J. Philip, T. Price, J.-M. Frahm, G. Drettakis, and G. Brostow, "Deep blending for free-viewpoint image-based rendering," *ACM TOG*, 2018.
[3] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan, "High-quality streamable free-viewpoint video," *ACM TOG*, 2015.
[4] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor *et al.*, "Fusion4d: Real-time performance capture of challenging scenes," *ACM TOG*, 2016.
[5] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach," in *SIGGRAPH*, 1996.
[6] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.
[7] K. Guo, P. Lincoln, P. Davidson, J. Busch, X. Yu, M. Whalen, G. Harvey, S. Orts-Escolano, R. Pandey, J. Dourgarian *et al.*, "The relightables: Volumetric performance capture of humans with realistic relighting," *ACM TOG*, 2019.
[8] V. Sitzmann, M. Zollhöfer, and G. Wetzstein, "Scene representation networks: Continuous 3d-structure-aware neural scene representations," in *NeurIPS*, 2019.
[9] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *CVPR*, 2020.
[10] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.
[11] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh, "Neural volumes: Learning dynamic renderable volumes from images," in *SIGGRAPH*, 2019.
[12] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM TOG*, 2015.

[13] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *ECCV*, 2016.

[14] J. Dong, Q. Shuai, Y. Zhang, X. Liu, X. Zhou, and H. Bao, "Motion capture from internet videos," in *ECCV*, 2020.

[15] Q. Fang, Q. Shuai, J. Dong, H. Bao, and X. Zhou, "Reconstructing 3d human pose by watching humans in the mirror," in *CVPR*, 2021.

[16] J. C. Loehlin, *Latent variable models*, 1987.

[17] S. Peng, Y. Zhang, Y. Xu, Q. Wang, Q. Shuai, H. Bao, and X. Zhou, "Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans," in *CVPR*, 2021.

[18] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *PAMI*, 2013.

[19] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, "Video based reconstruction of 3d people models," in *CVPR*, 2018.

[20] M. Oechsle, S. Peng, and A. Geiger, "Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction," in *ICCV*, 2021.

[21] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *CVPR*, 2021.

[22] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *ICCV*, 2021.

[23] A. Davis, M. Levoy, and F. Durand, "Unstructured light fields," in *Eurographics*, 2012.

[24] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis, "Depth synthesis and local warps for plausible image-based navigation," *ACM TOG*, 2013.

[25] E. Penner and L. Zhang, "Soft 3d reconstruction for view synthesis," *ACM TOG*, 2017.

[26] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, 2016.

[27] I. Choi, O. Gallo, A. Troccoli, M. H. Kim, and J. Kautz, "Extreme view synthesis," in *ICCV*, 2019.

[28] J. Thies, M. Zollhöfer, C. Theobalt, M. Stamminger, and M. Nießner, "Ignor: image-guided neural object rendering," in *ICLR*, 2020.

[29] Y. Kwon, S. Petrangeli, D. Kim, H. Wang, E. Park, V. Swaminathan, and H. Fuchs, "Rotationally-temporally consistent novel view synthesis of human performance video," in *ECCV*, 2020.

[30] Y. Kwon, S. Petrangeli, D. Kim, H. Wang, H. Fuchs, and V. Swaminathan, "Rotationally-consistent novel view synthesis for humans," in *ACMMM*, 2020.

[31] Q. Wang, Z. Wang, K. Genova, P. Srinivasan, H. Zhou, J. T. Barron, R. Martin-Brualla, N. Snavely, and T. Funkhouser, "Ibrnet: Learning multi-view image-based rendering," in *CVPR*, 2021.

[32] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *CVPR*, 2015.

[33] Z. Su, L. Xu, Z. Zheng, T. Yu, Y. Liu *et al.*, "Robustfusion: Human volumetric capture with data-driven visual cues using a rgbd camera," in *ECCV*, 2020.

[34] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *SIGGRAPH*, 2000.

[35] R. Martin-Brualla, R. Pandey, S. Yang, P. Pidlypenskyi, J. Taylor, J. Valentin, S. Khamis, P. Davidson, A. Tkach, P. Lincoln *et al.*, "Lookingood: Enhancing performance capture with real-time neural re-rendering," in *SIGGRAPH Asia*, 2018.

[36] M. Meshry, D. B. Goldman, S. Khamis, H. Hoppe, R. Pandey, N. Snavely, and R. Martin-Brualla, "Neural rerendering in the wild," in *CVPR*, 2019.

[37] M. Wu, Y. Wang, Q. Hu, and J. Yu, "Multi-view neural human rendering," in *CVPR*, 2020.

[38] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel, "Free-viewpoint video of human actors," *ACM TOG*, 2003.

[39] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, "Performance capture from sparse multi-view video," in *SIGGRAPH*, 2008.

[40] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel, "Motion capture using joint skeleton tracking and surface estimation," in *CVPR*, 2009.

[41] C. Stoll, J. Gall, E. De Aguiar, S. Thrun, and C. Theobalt, "Video-based reconstruction of animatable human characters," *ACM TOG*, 2010.

[42] R. Natsume, S. Saito, Z. Huang, W. Chen, C. Ma, H. Li, and S. Morishima, "Siclope: Silhouette-based clothed people," in *CVPR*, 2019.

[43] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li, "Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization," in *ICCV*, 2019.

[44] Z. Zheng, T. Yu, Y. Wei, Q. Dai, and Y. Liu, "Deephuman: 3d human reconstruction from a single image," in *ICCV*, 2019.

[45] S. Saito, T. Simon, J. Saragih, and H. Joo, "Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization," in *CVPR*, 2020.

[46] V. Sitzmann, J. Thies, F. Heide, M. Nießner, G. Wetzstein, and M. Zollhofer, "Deepvoxels: Learning persistent 3d feature embeddings," in *CVPR*, 2019.

[47] K.-A. Aliev, A. Sevastopolsky, M. Kolos, D. Ulyanov, and V. Lempitsky, "Neural point-based graphics," in *ECCV*, 2020.

[48] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM TOG*, 2019.

[49] L. Liu, W. Xu, M. Zollhoefer, H. Kim, F. Bernard, M. Habermann, W. Wang, and C. Theobalt, "Neural rendering and reenactment of human actor videos," *ACM TOG*, 2019.

[50] Y. Liao, K. Schwarz, L. Mescheder, and A. Geiger, "Towards unsupervised learning of generative models for 3d controllable image synthesis," in *CVPR*, 2020.

[51] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification: Learning view synthesis using multiplane images," in *SIGGRAPH*, 2018.

[52] J. Flynn, M. Broxton, P. Debevec, M. DuVall, G. Fyffe, R. Overbeck, N. Snavely, and R. Tucker, "Deepview: View synthesis with learned gradient descent," in *CVPR*, 2019.

[53] S. Liu, Y. Zhang, S. Peng, B. Shi, M. Pollefeys, and Z. Cui, "Dist: Rendering deep implicit signed distance function with differentiable sphere tracing," in *CVPR*, 2020.

[54] L. Liu, J. Gu, K. Z. Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," in *NeurIPS*, 2020.

[55] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021.

[56] A. Chen, Z. Xu, F. Zhao, X. Zhang, F. Xiang, J. Yu, and H. Su, "Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo," in *ICCV*, 2021.

[57] J. Chibane, A. Bansal, V. Lazova, and G. Pons-Moll, "Stereo radiance fields (srf): Learning view synthesis from sparse views of novel scenes," in *CVPR*, 2021.

[58] A. Trevithick and B. Yang, "Grf: Learning a general radiance field for 3d scene representation and rendering," *arXiv preprint arXiv:2010.04595*, 2020.

[59] Z. Li, S. Niklaus, N. Snavely, and O. Wang, "Neural scene flow fields for space-time view synthesis of dynamic scenes," in *CVPR*, 2021.

[60] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, "Volume rendering of neural implicit surfaces," *arXiv preprint arXiv:2106.12052*, 2021.

[61] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *arXiv preprint arXiv:2106.10689*, 2021.

[62] K. Gong, X. Liang, Y. Li, Y. Chen, M. Yang, and L. Lin, "Instance-level human parsing via part grouping network," in *ECCV*, 2018.

[63] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies," in *CVPR*, 2018.

[64] C. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, and T. Funkhouser, "Local implicit grid representations for 3d scenes," in *CVPR*, 2020.

[65] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe, "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction," in *ECCV*, 2020.

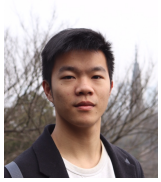
[66] K. Genova, F. Cole, A. Sud, A. Sarna, and T. Funkhouser, "Local deep implicit functions for 3d shape," in *CVPR*, 2020.

[67] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, 2018.

[68] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *CVPR*, 2020.

[69] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, "Convolutional occupancy networks," in *ECCV*, 2020.

[70] B. Graham, M. Engelcke, and L. Van Der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *CVPR*, 2018.
[71] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *CVPR*, 2019.
[72] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," in *ICML*, 2019.
[73] J. T. Kajiya and B. P. Von Herzen, "Ray tracing volume densities," in *SIGGRAPH*, 1984.
[74] N. Max, "Optical models for direct volume rendering," *TVCG*, 1995.
[75] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.
[76] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *SIGGRAPH*, 1987.
[77] R. Zhang and J. Chen, "Ndf: Neural deformable fields for dynamic human modelling," in *ECCV*, 2022.
[78] "Renderpeople," https://renderpeople.com/.
[79] "Mixamo," https://www.mixamo.com/.
[80] "Blender," https://www.blender.org/.
[81] M. Denninger, M. Sundermeyer, D. Winkelbauer, Y. Zidan, D. Olefir, M. Elbadrawy, A. Lodhi, and H. Katam, "Blenderproc," *arXiv preprint arXiv:1911.01911*, 2019.
[82] Y. Lior, K. Yoni, M. Dror, G. Meirav, A. Matan, B. Ronen, and L. Yaron, "Multiview neural surface reconstruction by disentangling geometry and appearance," in *NeurIPS*, 2020.
[83] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *NeurIPS*, 2017.
[84] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *CVPR*, 2017.
[85] M. Habermann, W. Xu, M. Zollhofer, G. Pons-Moll, and C. Theobalt, "Deepcap: Monocular human performance capture using weak supervision," in *CVPR*, 2020.
[86] C. Patel, Z. Liao, and G. Pons-Moll, "Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style," in *CVPR*, 2020.



Sida Peng is currently a Ph.D. student of computer science at Zhejiang University, advised by Dr. Xiaowei Zhou. He received B.E. degree in information engineering from Zhejiang University in 2018. His research interests include 3D reconstruction and object pose estimation.

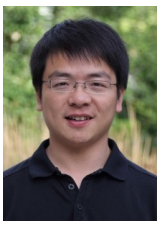
Chen Geng is currently a senior undergraduate student majoring in Computer Science at School of Computer Science and Chu Kochen Honors College, Zhejiang University. His research lies at the intersection between Graphics, 3D Vision, and Machine Learning. He is especially interested in scene understanding through inverse graphics and digital human reconstruction.

Yuanqing Zhang is currently a master student at Zhejiang University, advised by Dr. Xiaowei Zhou. She received B.E. degree in digital media technology from Zhejiang University in 2020. Her research interests include 3D reconstruction and inverse rendering.

Yinghao Xu is a fourth-year Ph.D student at Multimedia Lab (MMLab), Department of Information Engineering in The Chinese University of Hong Kong. He graduated from Zhejiang University in 2019. His research interests include video understanding, generative models.

Qianqian Wang is a Ph.D. candidate in computer science at Cornell Tech, Cornell University. Prior to that she received her B.E. from Zhejiang University. Her research interests include 3D computer vision and graphics. She is a recipient of Google PhD fellowship in 2022.

Qing Shuai is currently a Ph.D. student of computer science at Zhejiang University, advised by Dr. Xiaowei Zhou. He received a B.E. degree from Zhejiang University in 2019. His research interests include 3D human pose estimation and reconstruction, novel view synthesis, and differentiable rendering problems.



Xiaowei Zhou is a Research Professor of Computer Science at Zhejiang University, China. He obtained his PhD degree from The Hong Kong University and Science and Technology, after which he was a postdoctoral researcher at the GRASP Lab, University of Pennsylvania. His research interests include 3D reconstruction and scene understanding.

Hujun Bao is currently a professor in the State Key Laboratory of CAD&CG and College of Computer Science and Technology, Zhejiang University. He graduated from Zhejiang University in 1987 with a B.Sc. degree in mathematics, and obtained his Ph.D. degree in applied mathematics from the same university in 1993.