Neural Body: Implicit Neural Representations with Structured Latent Codes for Novel View Synthesis of Dynamic Humans

Sida Peng[1] Yuanqing Zhang[1] Yinghao Xu[2] Qianqian Wang[3]
Qing Shuai[1] Hujun Bao[1] Xiaowei Zhou[1*]
[1]Zhejiang University [2]The Chinese University of Hong Kong [3]Cornell University

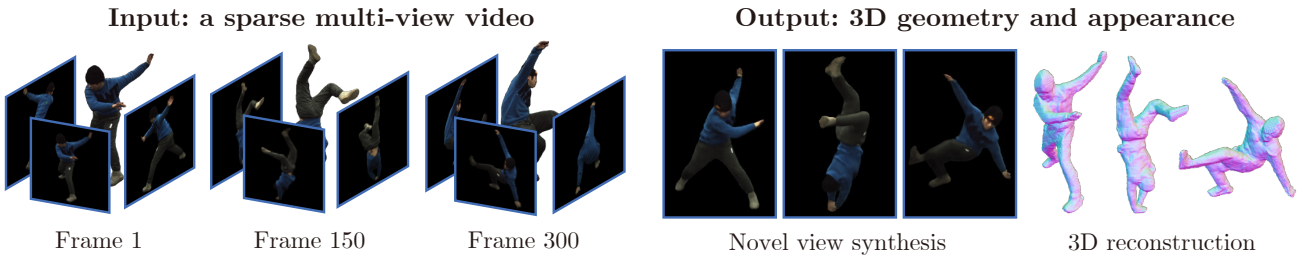


Figure 1: **Novel view synthesis of a performer from a sparse multi-view video.** Neural Body captures the 3D geometry and appearance of the performer, which can be used for 3D reconstruction and novel view synthesis. The code and supplementary materials are available at https://zju3dv.github.io/neuralbody/.

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. To evaluate our approach, we create a multi-view dataset named ZJU-MoCap that captures performers with complex motions. Experiments on ZJU-MoCap show that our approach outperforms prior works by a large margin in terms of novel view synthesis quality. We also demonstrate the capability of our approach to reconstruct a moving person from a monocular video on the People-Snapshot dataset.

The authors from Zhejiang University are affiliated with the State Key Lab of CAD&CG. *Corresponding author: Xiaowei Zhou.

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 [20, 23] or require depth sensors for high-quality 3D reconstruction [8, 14] 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 1. 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 [20, 12] mostly require dense input views and cannot be applied here. For reconstruction-based methods [54, 22], 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 [58, 47, 44] have investigated the potential of implicit neural representations on novel view synthesis. NeRF [44] shows that photorealistic view synthesis can

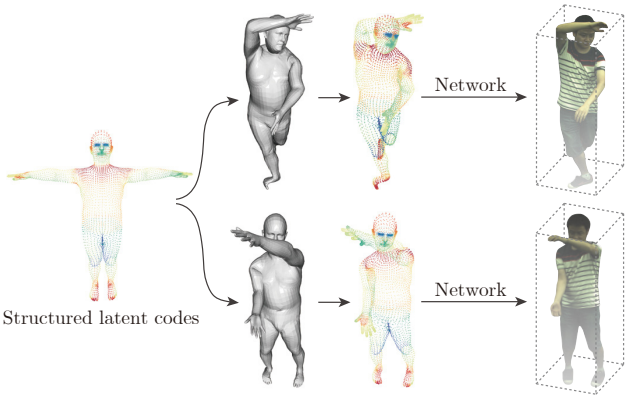


Figure 2: **The basic idea of Neural Body.** Neural Body generates implicit 3D representations of a human body at different video frames from the same set of latent codes, which are anchored to the vertices of a deformable mesh. For each frame, we transform the spatial locations of codes based on the 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.

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 [44] degrades dramatically, as shown by our experimental results in Section 4.1. 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 observations over different video frames. Lombardi et al. [37] 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 2. 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 [38] 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 [3, 13, 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 [36] 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 create a multi-view dataset called ZJU-MoCap that captures dynamic humans in complex motions. Across all captured videos, our approach exhibits state-of-the-art performances on novel view synthesis. We also demonstrate the capability of our approach to capture moving humans from monocular RGB videos on the People-Snapshot dataset [2]. Furthermore, our approach can be used for 3D reconstruction of the performers.

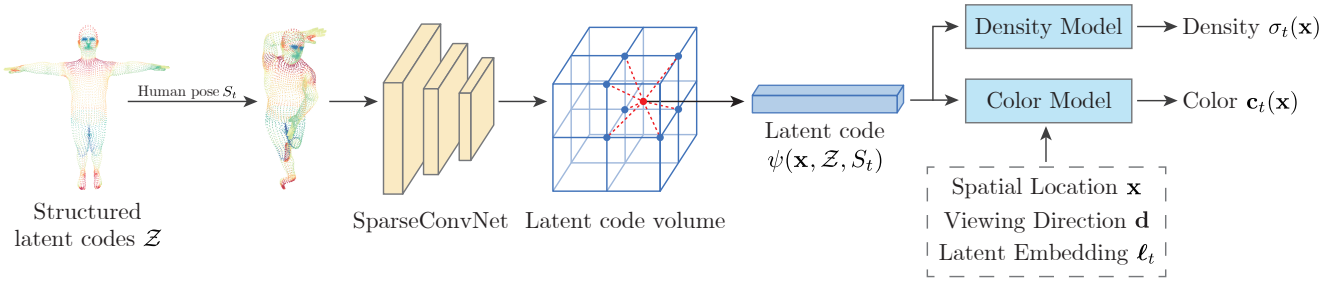
In summary, this work has the following contributions:

- We present a new approach capable of synthesizing photorealistic novel views of a performer in complex motions from a sparse multi-view video.
- We propose Neural Body, a novel implicit neural representation for a dynamic human, which enables us to effectively incorporate observations over video frames.
- We demonstrate significant performance improvements of our approach compared to prior work.

2. Related work

Image-based rendering. These methods aim to synthesize novel views without recovering detailed 3D geometry. Given densely sampled images, some works [20, 9] 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, [6, 50] 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. [28, 23, 7, 62, 31, 30, 63] replace hand-crafted parts of the image-based rendering pipeline with learnable counterparts to improve the robustness.

Human performance capture. Most methods [46, 8, 14, 22] adopt the traditional modeling and rendering pipeline to synthesize novel views of humans. They rely on either depth sensors [8, 14, 60] or a dense array of cameras [11, 22] to achieve the high fidelity reconstruction. [40, 43, 64] 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 [4, 10, 17, 59] assume that there are pre-scanned human models. They reconstruct dynamic humans by deforming the template shapes to fit the



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

Figure 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.

input images. However, the deformed geometries tend to be unrealistic, and pre-scanned human shapes are unavailable in most cases. Recently, [45, 52, 66, 53] 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 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 [57, 37], point clouds [64, 1], textured meshes [61, 34, 32], multi-plane images [67, 16], and implicit functions [58, 35, 47, 44, 33]. As a pioneer, SRN [58] 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 [44] 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.

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 [19] 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). The structured latent codes and neural networks are jointly learned by minimizing the difference between the rendered images and input images (Section 3.5).

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 [36] 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) [38]. 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 [26]. 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 [25, 5, 18], the latent codes are used with a neural network to represent the local geometry and appearance of a human. Anchoring

	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.

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 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 [65, 56, 49], we choose the SparseConvNet [21] 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 [56], 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. Details of network architectures are described in the supplementary material.

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 [37, 44], 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 [48], 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 [51, 44], 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.

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 [27] that accumulates volume densities and colors along the corresponding camera ray. In practice, the integral is approximated using numerical quadrature [41, 44]. Given a pixel, we first compute its camera ray $\mathbf{r}$ using the camera parameters and sample N_k points $\{\mathbf{x}_k\}_{k=1}^{N_k}$ along camera ray $\mathbf{r}$ between near and far bounds. The scene bounds are estimated based on the SMPL model. 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:

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

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

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. Training

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

$$\underset{\{\boldsymbol{\ell}_t\}_{t=1}^{N_t}, \mathcal{Z}, \Theta}{\text{minimize}} \sum_{t=1}^{N_t} \sum_{c=1}^{N_c} L(\mathcal{I}_t^c, P^c; \boldsymbol{\ell}_t, \mathcal{Z}, \Theta), \quad (5)$$

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:

$$L = \sum_{\mathbf{r} \in \mathcal{R}} \left\| \tilde{C}(\mathbf{r}) - C(\mathbf{r}) \right\|_2^2, \quad (6)$$

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 [54, 44], our method optimizes the model using all images in the video and has more information to recover the 3D structures.

We adopt the Adam optimizer [29] 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).

3.6. Applications

The trained Neural Body can be used for novel view synthesis and 3D reconstruction of the performer. 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 supplementary material. 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 [39].

	PSNR				SSIM			
	NV [37]	NT [61]	NHR [64]	OURS	NV [37]	NT [61]	NHR [64]	OURS
Twirl	22.09	25.78	26.68	**30.56**	0.831	0.929	0.935	**0.971**
Taichi	18.57	19.44	19.81	**27.24**	0.824	0.869	0.874	**0.962**
Swing1	22.88	24.96	24.73	**29.44**	0.726	0.905	0.902	**0.946**
Swing2	22.08	24.84	25.01	**28.44**	0.843	0.903	0.906	**0.940**
Swing3	21.29	23.50	23.47	**27.58**	0.842	0.896	0.894	**0.939**
Warmup	21.15	23.74	23.79	**27.64**	0.842	0.917	0.918	**0.951**
Punch1	23.21	24.93	25.02	**28.60**	0.820	0.877	0.879	**0.931**
Punch2	20.74	22.44	22.88	**25.79**	0.838	0.888	0.891	**0.928**
Kick	22.49	24.33	23.72	**27.59**	0.825	0.881	0.873	**0.926**
average	21.39	23.77	23.90	**28.10**	0.821	0.896	0.897	**0.944**

Table 2: **Results on the ZJU-MoCap dataset in terms of PSNR and SSIM (higher is better).** "NV" means Neural Volumes, and "NT" means Neural Textures.

4. Experiments

4.1. Results on the ZJU-MoCap dataset

We create a multi-view dataset called ZJU-Mocap for evaluating our approach. This dataset captures 9 dynamic human videos using a multi-camera system that has 21 synchronized cameras. We select four uniformly distributed cameras for training and use the remaining cameras for test. All sequences have a length between 60 to 300 frames. The humans perform complex motions, including twirling, Taichi, arm swings, warmup, punching, and kicking.

Metrics. For novel view synthesis, we follow [44] to evaluate our method using two standard metrics: peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). For 3D reconstruction, we only provide qualitative results, as there is no ground-truth human geometry.

Performance on novel view synthesis. We compare our method with state-of-the-art view synthesis methods [37, 61, 64] that handle dynamic scenes. All methods train a separate network for each scene. 1) Neural Volumes [37] encodes multi-view images at each frame into a latent vector and decodes it into a discretized RGBα voxel grid. 2) Neural Textures [61] proposes latent texture maps to render a coarse mesh into 2D images. Since [61] is not open-sourced, we reimplement it and take the SMPL mesh as the input mesh. 3) NHR [64] uses networks to render input point clouds to images. Here we take SMPL vertices as input point clouds.

Table 2 shows the comparison of our method with [37, 61, 64] in terms of the PSNR metric and the SSIM metric, respectively. 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 4.20 in terms of PSNR and 0.047 in terms of SSIM.

In contrast to learning the 3D representations from individual latent vectors [37], Neural Body generates implicit fields at different frames from the same set of latent codes.

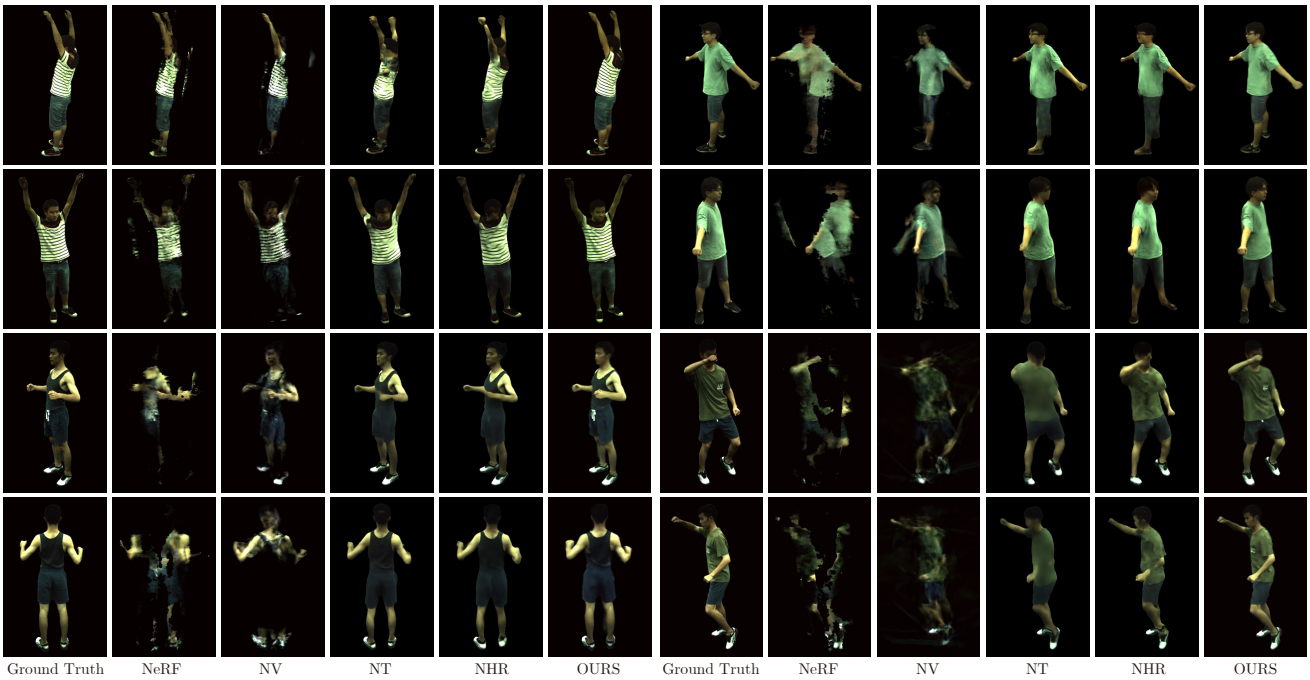


Figure 4: **Novel view synthesis on the ZJU-MoCap dataset.** "NV" means Neural Volumes [37], and "NT" means Neural Textures [61]. The input video is captured by four cameras. We select two novel views for qualitative comparison. Our method significantly outperforms [37, 44]. Furthermore, compared with image-to-image translation methods [61, 64], we can produce temporally consistent free-viewpoint videos, which are presented in the supplementary material.

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 [37, 61, 64, 44]. Here NeRF [44] trains a separate network for each video frame. The rendering results of [44, 37] indicate that they don't accurately capture the 3D human geometry and appearance. The results of NeRF [44] don't appear reasonable shapes, which show that NeRF fails to learn correct 3D human representations. Neural Volumes [37] gives blurry results. As image-to-image translation methods, [61, 64] have difficulty in controlling the rendering viewpoints. In contrast, our method gives photorealistic novel views. Furthermore, our method can generate inter-frame and inter-view consistent free-viewpoint videos, which are presented in the supplementary material.

Performance on 3D reconstruction. We test state-of-the-art multi-view methods COLMAP [54, 55] and DVR [47] on the ZJU-MoCap dataset. COLMAP [54, 55] is a well-developed multi-view stereo algorithm, and DVR [47] learns occupancy fields [42] with a differentiable renderer. We find that they fail to recover reasonable 3D human shapes from only four input views.

For comparison, we choose a learning-based approach, PIFuHD [53], as the baseline method. PIFuHD trains a single-view reconstruction network on 450 high-resolution photogrammetry scans. We use its released code and pre-trained model for inference. The first view is taken as the input of PIFuHD. To improve its performance, we remove the background of the input image. [24, 52] propose multi-view reconstruction networks, but they don't release the pre-trained model, so we don't compare with them.

Figure 5 presents the qualitative comparison between our method and PIFuHD. Neural Body generates accurate geometries for humans in complex motions. Since our method learns the 3D human representations from multi-view images, the 3D human poses of reconstructed human models are highly consistent with the observations. The reconstruction results of PIFuHD indicate that it doesn't generalize well on our data. For persons with complex human poses, PIFuHD fails to recover correct human shapes. Moreover, its reconstructed models are not consistent with the human poses observed from multi-view images.

4.2. Results on monocular videos

We demonstrate that our approach is able to reconstruct dynamic humans from monocular videos on the People-Snapshot dataset [2]. This dataset captures performers that rotate while holding an A-pose. Since the poses of moving humans are not complex, the SMPL parameters can be accurately estimated from the monocular videos. We compare Neural Body with the approach proposed in [2], which

Figure 5: **3D reconstruction on the ZJU-MoCap dataset.** Neural Body achieves high-quality reconstruction. Our method is able to recover the clothing, such as the hoodie of the third person. PIFuHD [53] does not generalize well on the dataset.

Figure 6: **Novel view synthesis on monocular videos.** Our method renders more appearance details than People-Snapshot [2], such as the blouse of the first person and the pants of the second person. Zoom in for details.

deforms vertices of the SMPL model to fit the 2D human silhouettes over the video sequence. Following [2], we report the qualitative results on the People-Snapshot dataset.

Performance on novel view synthesis. Figure 6 shows the qualitative comparison on novel view synthesis. Our method renders more appearance details than [2], especially for the performers wearing the loose clothing. For example, Neural Body accurately renders the blouse for the first person, while the blouse rendered by [2] attaches closely to the human body. Some of scenes are captured in the outdoor environment, which exhibit strong illumination variations. The photorealistic rendering results indicate that Neural Body can handle complex lighting conditions.

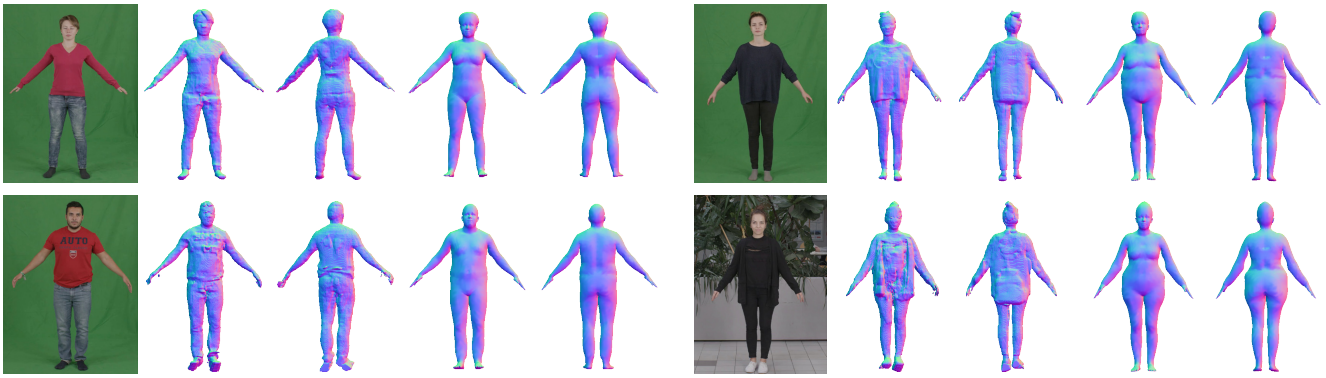


Figure 7: **3D reconstruction on monocular videos.** Compared with the approach in People-Snapshot [2], Neural Body generates more detailed geometries and can handle persons wearing loose clothing.

	1 view	2 views	4 views	6 views
PSNR	25.08	25.49	30.54	**32.73**
SSIM	0.912	0.928	0.971	**0.979**

Table 3: **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.

Frames	1	60	300	600	1200
PSNR	25.64	30.14	**30.66**	30.59	29.97
SSIM	0.940	0.970	**0.971**	0.970	0.970

Table 4: **Results of models trained with different numbers of training frames.** We train models on 1, 60, 300, 600, and 1200 frames and test on the first frame of "Twirl".

Performance on 3D reconstruction. The qualitative results of our method and [2] are presented in Figure 7. Neural Body recovers more geometric details than [2]. For example, the hair shapes are highly consistent with the RGB observations. The results of the last column indicate that our method can handle persons wearing loose clothing, while [2] does not recover correct shapes for such data.

4.3. 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.

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 30.56 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 3 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 [37] 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. The results are evaluated on the first frame of the video "Twirl". Table 4 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 as the network has difficulty in fitting very long videos.

5. 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 evaluate our approach, we created a multi-view dataset called ZJU-MoCap that captures dynamic humans in complex motions. We demonstrated superior view synthesis quality compared to prior work on the newly collected dataset and the People-Snapshot dataset.

Acknowledgements: The authors from Zhejiang University would like to acknowledge the support from the National Key Research and Development Program of China (No. 2020AAA0108901) and NSFC (No. 61806176).

References

[1] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *ECCV*, 2020. 3

[2] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *CVPR*, 2018. 2, 6, 7, 8

[3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016. 2

[4] Joel Carranza, Christian Theobalt, Marcus A Magnor, and Hans-Peter Seidel. Free-viewpoint video of human actors. *ACM TOG*, 2003. 2

[5] Rohan Chabra, Jan Eric Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *ECCV*, 2020. 3

[6] Gaurav Chaurasia, Sylvain Duchene, Olga Sorkine-Hornung, and George Drettakis. Depth synthesis and local warps for plausible image-based navigation. *ACM TOG*, 2013. 2

[7] Inchang Choi, Orazio Gallo, Alejandro Troccoli, Min H Kim, and Jan Kautz. Extreme view synthesis. In *ICCV*, 2019. 2

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

[9] Abe Davis, Marc Levoy, and Fredo Durand. Unstructured light fields. In *Eurographics*, 2012. 2

[10] Edilson De Aguiar, Carsten Stoll, Christian Theobalt, Naveed Ahmed, Hans-Peter Seidel, and Sebastian Thrun. Performance capture from sparse multi-view video. In *SIGGRAPH*, 2008. 2

[11] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH*, 2000. 2

[12] Paul E Debevec, Camillo J Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *SIGGRAPH*, 1996. 1

[13] Junting Dong, Qing Shuai, Yuanqing Zhang, Xian Liu, Xiaowei Zhou, and Hujun Bao. Motion capture from internet videos. In *ECCV*, 2020. 2

[14] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *ACM TOG*, 2016. 1, 2

[15] Qi Fang, Qing Shuai, Junting Dong, Hujun Bao, and Xiaowei Zhou. Reconstructing 3d human pose by watching humans in the mirror. In *CVPR*, 2021. 2

[16] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *CVPR*, 2019. 3

[17] Juergen Gall, Carsten Stoll, Edilson De Aguiar, Christian Theobalt, Bodo Rosenhahn, and Hans-Peter Seidel. Motion capture using joint skeleton tracking and surface estimation. In *CVPR*, 2009. 2

[18] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *CVPR*, 2020. 3

[19] Ke Gong, Xiaodan Liang, Yicheng Li, Yimin Chen, Ming Yang, and Liang Lin. Instance-level human parsing via part grouping network. In *ECCV*, 2018. 3

[20] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. The lumigraph. In *SIGGRAPH*, 1996. 1, 2

[21] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 4

[22] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM TOG*, 2019. 1, 2

[23] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM TOG*, 2018. 1, 2

[24] Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Linjie Luo, Chongyang Ma, and Hao Li. Deep volumetric video from very sparse multi-view performance capture. In *ECCV*, 2018. 6

[25] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. Local implicit grid representations for 3d scenes. In *CVPR*, 2020. 3

[26] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *CVPR*, 2018. 3

[27] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. In *SIGGRAPH*, 1984. 4

[28] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM TOG*, 2016. 2

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[30] Youngjoong Kwon, Stefano Petrangeli, Dahun Kim, Haoliang Wang, Henry Fuchs, and Viswanathan Swaminathan. Rotationally-consistent novel view synthesis for humans. In *ACMMM*, 2020. 2

[31] YoungJoong Kwon, Stefano Petrangeli, Dahun Kim, Haoliang Wang, Eunbyung Park, Viswanathan Swaminathan, and Henry Fuchs. Rotationally-temporally consistent novel view synthesis of human performance video. In *ECCV*, 2020. 2

[32] Yiyi Liao, Katja Schwarz, Lars Mescheder, and Andreas Geiger. Towards unsupervised learning of generative models for 3d controllable image synthesis. In *CVPR*, 2020. 3

[33] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *NeurIPS*, 2020. 3

[34] Lingjie Liu, Weipeng Xu, Michael Zollhoefer, Hyeongwoo Kim, Florian Bernard, Marc Habermann, Wenping Wang, and Christian Theobalt. Neural rendering and reenactment of human actor videos. *ACM TOG*, 2019. 3

[35] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *CVPR*, 2020. 3

[36] John C Loehlin. *Latent variable models*. 1987. 2, 3

[37] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. In *SIGGRAPH*, 2019. 2, 3, 4, 5, 6, 8

[38] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM TOG*, 2015. 2, 3

[39] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, 1987. 5

[40] Ricardo Martin-Brualla, Rohit Pandey, Shuoran Yang, Pavel Pidlypenskyi, Jonathan Taylor, Julien Valentin, Sameh Khamis, Philip Davidson, Anastasia Tkach, Peter Lincoln, et al. Lookingood: Enhancing performance capture with real-time neural re-rendering. In *SIGGRAPH Asia*, 2018. 2

[41] Nelson Max. Optical models for direct volume rendering. *TVCG*, 1995. 4

[42] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*, 2019. 6

[43] Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. Neural rerendering in the wild. In *CVPR*, 2019. 2

[44] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1, 2, 3, 4, 5, 6

[45] Ryota Natsume, Shunsuke Saito, Zeng Huang, Weikai Chen, Chongyang Ma, Hao Li, and Shigeo Morishima. Siclope: Silhouette-based clothed people. In *CVPR*, 2019. 3

[46] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *CVPR*, 2015. 2

[47] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, 2020. 1, 3, 6

[48] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 4

[49] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *ECCV*, 2020. 4

[50] Eric Penner and Li Zhang. Soft 3d reconstruction for view synthesis. *ACM TOG*, 2017. 2

[51] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *ICML*, 2019. 4

[52] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, 2019. 3, 6

[53] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *CVPR*, 2020. 3, 6, 7

[54] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 5, 6

[55] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 6

[56] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *CVPR*, 2020. 4

[57] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. Deepvoxels: Learning persistent 3d feature embeddings. In *CVPR*, 2019. 3

[58] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *NeurIPS*, 2019. 1, 3

[59] Carsten Stoll, Juergen Gall, Edilson De Aguiar, Sebastian Thrun, and Christian Theobalt. Video-based reconstruction of animatable human characters. *ACM TOG*, 2010. 2

[60] Zhuo Su, Lan Xu, Zerong Zheng, Tao Yu, Yebin Liu, et al. Robustfusion: Human volumetric capture with data-driven visual cues using a rgbd camera. In *ECCV*, 2020. 2

[61] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM TOG*, 2019. 3, 5, 6

[62] Justus Thies, Michael Zollhöfer, Christian Theobalt, Marc Stamminger, and Matthias Nießner. Ignor: image-guided neural object rendering. In *ICLR*, 2020. 2

[63] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 2021. 2

[64] Minye Wu, Yuehao Wang, Qiang Hu, and Jingyi Yu. Multi-view neural human rendering. In *CVPR*, 2020. 2, 3, 5, 6

[65] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 2018. 4

[66] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *ICCV*, 2019. 3

[67] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018. 3