HumanNeRF-SE: A Simple yet Effective Approach to Animate HumanNeRF with Diverse Poses

Caoyuan Ma[1,2], Yu-Lun Liu[3], Zhixiang Wang[4,5], Wu Liu[2], Xinchen Liu[2], Zheng Wang[1†]
[1]Wuhan University [2]JD Explore Academy [3]National Yang Ming Chiao Tung University
[4]The University of Tokyo [5]National Institute of Informatics

Figure 1. **Overview.** We synthesize images of performers in *diverse* poses in a *simple* yet *effective* way. (a) **Novel Pose**: In order to test the ability of synthesizing images of diverse poses, we input a simple action video and animate it with novel poses. (b) **Few Shot**: Furthermore, to test the ability to avoid overfitting, we use only few input to generalize novel pose images. (c) Compared to similar methods, HumanNeRF-SE uses less than 1% of learnable parameters, 1/20 training time, and achieves better results in few-shot experiments. †LPIPS = 1,000×LPIPS. Our code, models, and demos will be available.

Abstract

We present HumanNeRF-SE, which can synthesize diverse novel pose images with simple input. Previous HumanNeRF studies require large neural networks to fit the human appearance and prior knowledge. Subsequent methods build upon this approach with some improvements. Instead, we reconstruct this approach, combining explicit and implicit human representations with both general and specific mapping processes.

Our key insight is that explicit shape can filter the information used to fit implicit representation, and frozen general mapping combined with point-specific mapping can effectively avoid overfitting and improve pose generalization performance. Our explicit and implicit human represent combination architecture is extremely effective. This is reflected in our model's ability to synthesize images under arbitrary poses with few-shot input and increase the speed of synthesizing images by 15 times through a reduction in computational complexity without using any existing acceleration modules. Compared to the state-of-the-art HumanNeRF studies, HumanNeRF-SE achieves better performance with fewer learnable parameters and less training time (see Figure 1).

1. Introduction

The prevalence of human-centric videos recorded on handheld devices has led to a growing need for technologies capable of analyzing, reconstructing, and synthesizing photo-realistic human appearances from monocular video sequences. This need is particularly critical for a wide range of gesture-controlled applications such as human-computer interactions and gaming. However, achieving this goal is challenging due to the limited and patchy representations of human bodies typically present in monocular videos. Although there are methods [11, 16, 17, 53, 59] that can learn high-quality human representations from monocular videos and synthesize images from arbitrary viewpoints, *they cannot perform well in synthesizing novel pose images that are significantly different from the training videos and usually take long time to train.* These limitations restrict the potential application scenarios of these techniques.

To address this limitation, a crucial challenge is to generate human body images in arbitrary zero-shot poses from monocular videos with limited poses. This task is exceptionally challenging. Unlike multi-camera videos that contain more informative data, limited poses provided by monocular videos are often simplistic and cannot encompass all possible human body poses. We need to employ

Figure 2. **Different weights for deformation.** (a) Learning weight volume through neural networks or fine-tuning SMPL weight. When synthesizing novel pose images, points are deformed onto the canonical space incorrectly because of the overfitting weight. This leads to artifacts. (b) Simply use the nearest SMPL vertex's LBS (Linear Blending Skinning) weight through methods such as KNN. This method leads to the sampling space being filled with colors related to the clothes, which we named color bleeding. For the given data, the performer's regular punching causes many transparent points to be deformed onto the T-shirt in the canonical space, leading to convergence in the color bleeding space. For leg movements pose, points in the space also deform onto the other leg in the canonical space. (c) We perform convolution filtering in the voxel volume and then query the nearest SMPL vertex's LBS weight. This architecture is simple yet effective. Our method not only avoids overfitting but also greatly reduces the required size of the neural network and achieves performance improvement.

methods to learn implicit human representations from simple poses and synthesize human body images that are entirely different from the training data. Simultaneously, it is crucial to minimize training time and enhance the efficiency of the methods as much as possible, making it more practical for applications.

Previous works [12, 22, 42, 54, 57, 58] typically employ methods such as point clouds and textured meshes to optimize a tailless three-dimensional structure, which need to be translated into images, limiting their applicability. Subsequently, neural rendering-based methods [33, 51, 52, 55, 60] can directly achieve satisfactory image synthesis results. However, these methods typically assume multi-camera inputs and do not account for object deformation. Deformable methods [35, 36, 40, 50, 56] extend upon this by endowing implicit fields with the capability to express dynamic objects. Nevertheless, these methods perform poorly on humans, due to the complex deformation of motions. Subsequent research aims to address the issue of human body deformation. Some studies [5, 18, 24, 38, 39, 63] leverage NeRF to learn dynamic human representations from multi-camera images and some of them can generalize to novel poses. However, the increasing data and equipment requirements limited their practical applicability. Certain monocular methods [11, 16, 17, 53, 59] eased data constraints by allowing the utilization of monocular videos as input. HumanNeRF [53]showcases remarkable novel view results by decoupling the motion field. It utilizes extensive neural networks to fit various modules, often requiring a considerable amount of time for training. MonoHuman [59] further improves the capability to synthesize novel pose images by using reference images and bidirectional constraints. This results in longer time requirements and poor results when there is a large difference between the training poses and the rendering poses. Moreover, when the input poses become more limited, the aforementioned methods often yield highly unsatisfactory results due to overfitting of their neural weight (Figure 2).

We present HumanNeRF-SE, which can synthesize novel pose images with tens of simple inputs and a few learnable parameters in hours. Our approach distinguishes itself from previous methods by effectively preventing potential overfitting while maintaining a robust correlation between human NeRF representation and prior pose. Furthermore, our technique does not depend on a diverse range of motion samples to improve generalization performance under varying poses. To begin, we want to extract the relation between explicit SMPL vertices and sampling points. To achieve this, we convert the SMPL vertex coordinates into a voxelized form and employ an adaptive unified kernel to diffuse the vertices across the voxel volume. Following this, the spatial-aware feature and occupancy of the sampling point can be easily queried in this volume, and contribution comes from all vertices within the receptive field centered on the sampling point. Throughout this procedure, a significant number of points unrelated to the body are filtered out, leaving behind only those points likely to be related to the human body. These points can be deformed to a unified canonical pose to get the coarse coordinates. We refine the coarse coordinates by performing a specific mapping using the point-level spatial-aware feature obtained in the convoluted voxel volume. Finally, we get the colors and densities of the sampling points through a neural network and render the image through a differentiable renderer.

Overall, we propose a novel HumanNeRF method that can synthesize images of varying poses simply yet effectively. Our approach utilizes explicit SMPL vertices and leverages a general and specific mapping process to model the deformation from observation to canonical space. Our experiments show that our method can generate novel poses with significant differences from the training poses, even when the input is limited to a few shots. To further demonstrate the performance of our method, we captured in-the-wild data, where the input video involves a simple rotation of the user. In summary, this work makes the following contributions:

- We use an adaptive space conv-filter and frozen weight for the general mapping process. This greatly reduces computational complexity and avoids overfitting.
- We design a spatial-aware feature point refine module

as our specific mapping process to reduce the unnatural sharp results caused by rigid deformation.

- Our architecture is extremely simple and effective. Compared to methods with similar performance, HumanNeRF-SE uses less than 1% learnable parameters, 1/20 training time, and increases rendering speed by 15 times without using any existing acceleration modules.

2. Related Work

Human Performance Capture. Recently, Deep neural networks are commonly used to learn scene or human representation from images, with a range of methods now available including voxels [26, 43], point clouds [1, 12, 54, 58], textured meshes [15, 21, 22, 41, 42, 49, 57, 64, 65], multi-plane images [8, 66], and implicit functions [23, 25, 31, 34, 40, 44]. Most of these methods aim to optimize a detailed 3d geometry, and then synthesize results into images or videos for application, which limited their usage. Our method can directly synthesize human body images.

Image-based Rendering. To synthesize novel view images of a static object without recovering detailed 3D geometry, previous neural rendering method [2, 13, 28, 29, 33, 45, 48, 51, 52, 55, 60, 62] represent amazing image synthesis result, but these methods typically assume multi-camera input and usually don't take object's deformation into consider. The deformable methods [9, 19, 35, 36, 40, 50, 56] endow the implicit field with the ability to express dynamic objects but don't perform well in the human because of the complexity of human deformation. Our method employs a simple and effective architecture for human body deformation. The ability to synthesize the movements of specific human bodies in different poses has a wide range of applications. Therefore, it is very meaningful to extend methods to adapt to dynamic human bodies.

Multi-camera HumanNeRF. There has been some research [4–6, 10, 14, 18, 20, 24, 38, 39, 46, 63] on learning dynamic human representation through NeRF from multi-camera images. NeuralBody [39] uses structured pose features generated from SMPL [27] vertices to anchor sampling points in any poses from sparse multi-camera videos, which inspired us to use spatial information of vertices. [46] learn dynamic human bodies by modeling the relationship between sampling points and joints. [10, 14, 20, 24, 38, 63] transform sampling points of dynamic human to a canonical space for NeRF training. Because the information in monocular data is much more limited than in multi-camera data, some of them have the ability to train with only monocular video, but they are not designed for monocular scenes and usually don't perform well in monocular data. Our method only requires few-shot input and also performs well in monocular data.

Monocular HumanNeRF. Since obtaining monocular videos is much easier than obtaining synchronized multi-view videos, it is significant to extend the capabilities of HumanNeRF to monocular videos. Inspired by [35, 36, 40], which maps rays in dynamic scenes to canonical space, [11, 16, 17, 53, 59] introduce priors to regularize the deformation. [11, 16, 37] greatly improve the speed of training and rendering by using more efficient spatial encoding methods [3, 32], while these encoding methods don't perform well in terms of acceleration in our experiments. [17, 38] learns the blending field by extending the deformation weight of the closest correspondence from the SMPL mesh, which leads to a significant increase in computational cost. HumanNeRF [53] demonstrates amazing novel view results by decoupling the motion field, which uses a significant amount of neural networks to fit various modules and often takes a very long time to train. Monohuman [59] further improves the pose generalization performance of HumanNeRF by adding a reference image module and consist loss. Our experiments aimed to explore the fundamental reasons why HumanNeRF performs poorly in pose generalization, and to achieve better results on more challenging data by rebuilding a simple yet efficient architecture with SMPL vertices to combine explicit and implicit human representation.

3. HumanNeRF-SE

We propose a simple yet effective approach for learning the implicit representation of human bodies from limited inputs and being capable of synthesizing diverse novel poses. Compared to other similar methods, each module of our approach is designed to map the explicit and implicit human representation better and generalize to arbitrary novel poses without overfitting. This results in fewer data demands and computational load and improves pose generalization capabilities compared to other approaches. The pipeline is illustrated in Figure 3.

The key to learning the deformable NeRF representation of human bodies lies in canonicalizing the sampling points within the dynamic observation space. Prior methods predominantly depend on neural network fitting or supplementary texture information to precisely anchor the sampling points relative to the human body. These methods also introduce frame-level features to augment the multi-view results, albeit at the expense of substantially diminishing the pose generalization capabilities of the model.

Different from these methods, our method effectively leverages the explicit vertices $\mathbf{v}$ to canonicalize the sampling points $\mathbf{x}$ within the dynamic human observation space. This process involves constructing voxel volume V (Sec.3.1), convoluting voxel volume channel-by-channel

Figure 3. **Framework of HumanNeRF-SE.** (a) We first voxelize the observation space as a voxel volume V. For a voxel containing vertices, the value will be the number of vertices (as one occupancy channel) and the corresponding SMPL weight. (b) We performed channel-by-channel convolution on the volume. All sampling points are queried in the convolutional volume to get their spatial-aware features. Those points with zero occupancy will be filtered out. (c) We query the nearest weight of the remained points in the volume, which is used for rigid deformation. Spatial-aware features are utilized in the neural network to correct the rigid results and obtain the final point coordinates in the canonical space. (d) The sampling points in the canonical space obtain their colors and densities through the NeRF network. The densities of filtered points are forced to be zero.

to obtain spatially-aware features F_s and filtering out useless sampling points $\mathbf{x_o}$ and get human-related points $\mathbf{x_r}$ (Sec.3.2), mapping points to canonical space generally D and specifically P (Sec.3.3). It is important to emphasize that we exclusively utilize point-level features throughout the entire process, abstaining from the use of any frame-level features, thereby ensuring the pose generalization capabilities of our model.

In summary, our method can be represented as:

$$\mathbf{c}, \sigma = M_\sigma(P(F_s, \mathbf{x_r}) + D(\mathbf{x_r}, K(\mathbf{x_r}, V), J)). \quad (1)$$

where J represents the pose and M_σ is the NeRF network which is similar to baselines for better comparison. We use K to query the nearest weight of points $\mathbf{x}$ in the volume V. Our rigid deformation D is a general mapping process, as it is not influenced by the training individual.

3.1. Voxelization

In order to more efficiently handle the relationship between the sampling points and the SMPL model, we first voxelized the SMPL space. However, since the vertices of the SMPL model within the boundary of the human body are still relatively sparse, we used the SpConv Sp [7] to construct our voxel volume V.

We processed data similarly to NeuralBody [39] in this part. For a given set of SMPL vertices $\mathbf{v}$, we first calculate the maximum and minimum values on the coordinate axis to get the bounding box, scale the bounding box to the set voxel size $\mathbf{vs}$, and find the least common multiple of axis and 32, in preparation for subsequent sparse convolution. For each SMPL vertex, we also scale it down according to voxel size after subtracting the minimum value.

$$V = Sp(\mathbf{v}, \mathbf{vs}, W) \quad (2)$$

In the generated voxel volume, each voxel that contains vertices holds two values: the corresponding LBS weight w_j in SMPL weight W and the count of contained vertices n_v as an occupancy indicator. For voxels without any contained vertices, all channels are assigned a value of zero. The value of a certain voxel is:

$$\begin{cases} (n_v, w_1, w_2..., w_{24}), & \text{contain vertices} \\ (0, 0, 0..., 0), & \text{empty voxel} \end{cases} \quad (3)$$

3.2. Conv-Filter

We innovatively use spatial convolution to filter the sampling points and extract features simultaneously. A convolution kernel initialized to one to convolve the value of the voxel volume. To preserve high-frequency information, we use channel-by-channel convolution.

$$F_{s_i} = \sum_{m=h'}^{h} \sum_{n=w'}^{w} \sum_{t=d'}^{d} \mathbf{vo_{m,n,t,i}} \cdot kw_{m,n,t,i}\,, \quad (4)$$

where $h' = h - k$, $w' = w - k$ and $d' = d - k$. The i-th channel of F_s results from convolving i-th channel of voxel's value $\mathbf{vo}$ and kernel's weight kw. If the occupancy of the convolution result is zero, it means that the current convolution center coordinate is not related to the human body (eq. 3). We force these points not to participate in

subsequent calculations and set their density value d to zero.

$$\begin{cases} F_s(x) = 0, & \mathbf{x_o} \text{ filterout}, \quad d(\mathbf{x_o}) = 0 \\ F_s(x) > 0, & \mathbf{x_r} \text{ remain} \end{cases} \tag{5}$$

3.3. Point Canonicalization

We compute the rotation R_j and translation T_j of the current pose relative to the joints in the T-pose of the human body in a similar way of HumanNeRF [53]. Inspired by NeuralBody [39]'s use of convolution to diffuse vertex occupancy, we use a simpler method to query the nearest neighbor SMPL weight $W_{smpl} = K(\mathbf{x}, \mathbf{v})$ in Eq.1

$$\mathbf{x^r_{cnl}} = \sum_{j \in J} w_j((\mathbf{x_r} \cdot R_j) + T_j). \tag{6}$$

The process of $(\mathbf{x} \cdot R_j) + T_j$ is abstracted as $B_j(x)$ in Figure 3. In this formula, J represents a set of joints, w_j represents the deformation weight of the j-th channel of W_{smpl} on the current sampled point, and $\mathbf{x^r_{cnl}}$ represents the coordinates of the sampled point in the canonical space after rigid deformation.

Directly using the nearest weight amounts to giving up non-rigid modeling of the possible clothing deformation caused by human body deformation, which often leads to unsatisfactory results. To enable the network to learn non-rigid deformation from limited images as much as possible, some methods introduce frame-level features to help learning. These features may include frame index or body pose, and for a particular frame, the frame-level features of all sampling points are generally consistent. This method is useful in the task of synthesizing new viewpoints in dynamic scenes, but it is not suitable for animatable human body.

We designed an additional network here to refine the coordinates of the sampled points in the canonical space. We use limited point-level features to learn the offset of the sampling points in canonical space to improve the novelview evaluation metric. The spatial-aware feature has shown significant advantages in this process because it aggregates vertex information within the receptive field of the sampling point in the current pose:

$$\mathbf{x_{cnl}} = \mathbf{x^r_{cnl}} + P(F_s, \mathbf{x_r}) \tag{7}$$

In this formula, $\mathbf{x_{cnl}}$ is the optimized result of the sampled point coordinates in the canonical space, and P is the network module for optimizing the sampled point coordinates in the canonical space.

3.4. Appearance Net and Rendering

In order to better compare with methods such as HumanNeRF [53], which rely on neural networks to fit the deformation weights, we used the same neural radiance field structure. The coordinates of the sampled points in the canonical space were encoded using the same positional encoding as in NeRF, and the MLP was used to output the corresponding colors and densities. Finally, the images were rendered using the differentiable renderer:

$$\mathbf{c}, \sigma = M_\sigma(\mathbf{x_{cnl}}) \tag{8}$$

Following NeRF, we render the neural human by the volume rendering equation [30]. The rendered color $C(\mathbf{r})$ of the corresponding pixel with N_s samples of ray $\mathbf{r}$ can be written as:

$$\tilde{C}_t(\mathbf{r}) = \sum_{k=1}^{N_s} (\prod_{j-1}^{i-1} (1 - \alpha_j)) \alpha_i \mathbf{c}(\mathbf{x_i}) \tag{9}$$

$$\alpha_i = 1 - \exp(-\sigma(\mathbf{x_i}) \triangle t_i) \tag{10}$$

3.5. Training

Given a set of monocular videos, the frame images of the video $\{I_n | n = 1, 2, ..., N\}$. Most of the images are used for training and the rest are used for evaluation. The foreground mask Θ_{fore} is obtained from the density after the network output. For each foreground ray $\mathbf{r} \in \mathbf{r_f}$, the corresponding loss function is defined as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N_{r_f}} \sum_{\mathbf{r} \in \mathbf{r_f}} \left|\left| \tilde{C}(\mathbf{r}) - C(\mathbf{r}) \right|\right|^2, \tag{11}$$

$$\mathcal{L}_{\text{LPIPS}} = lpips(\Theta_{\text{fore}}(\tilde{I}_i), \Theta_{\text{fore}}(I_i)), \tag{12}$$

$$\mathcal{L} = \mathcal{L}_{\text{LPIPS}} + \lambda \mathcal{L}_{\text{MSE}}. \tag{13}$$

4. Experiments

Evaluation metrics. We use three metrics: peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), and learned perceptual image patch similarity (LPIPS). It should be noted that *LPIPS is the most human-perceptually-aligned metric* among these indicators, while PSNR prefers smooth results but may have bad visual quality [61].

Dataset. We use ZJU-MOCAP and our self-captured videos (in-the-wild data) to evaluate our method. We follow [53] [59] to select the same 6 subjects in ZJU-MOCAP for our evaluation. ZJU-MOCAP is a dataset that captures the target human body from 23 different perspectives synchronously in a professional light stage room. We only use the first view captured in each subject.

The previous work did not consider the issue of pose leakage caused by the strong repetition of actions in ZJU-MOCAP. In order to further validate the performance of the model and make the task more applicable, we shot videos using handheld devices. We limited the training videos to a person spinning one round, and used diverse action videos for evaluation, which is a more real-world applicable evaluation method.

Figure 4. **Some image synthesis results in the quantitative research.** Because of limited information used in training, previous methods [38, 53, 59] were unable to learn appropriate human weights. The official code of Ani-NeRF [38] did not produce reasonable results on our data since it is designed for multi-camera input. HumanNeRF [53] exhibits distortion and artifacts. The performance of Monohuman [59] is heavily influenced by the specific data.

Table 1. **Average results of six subjects on ZJU-MOCAP.** Our method (green) exhibits excellent quantitative metrics, especially in terms of LPIPS. This indicates that the results of our method are more in line with human visual perception. Our method demonstrated a better ability to avoid overfitting compared to other methods on our custom data. The official code of Ani-NeRF did not produce reasonable results on our custom data. †LPIPS = 1,000×LPIPS.

Methods	ZJU-MOPCAP						IN-THE-WILD DATA					
	Full			**Few-shot**			**Full**			**Few-shot**		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
Ani-NeRF [38]	21.24	0.8458	68.221	22.18	0.8339	64.839	-	-	-	-	-	-
HumanNeRF [53]	31.15	0.9739	24.822	29.90	0.9683	33.056	28.97	0.9629	48.128	28.82	0.9618	50.240
MonoHuman [59]	30.91	0.9718	31.292	30.10	0.9677	36.494	29.15	0.9639	51.623	29.21	0.9636	56.220
Ours	31.09	0.9740	24.085	30.11	0.9684	32.084	29.23	0.9666	46.308	29.26	0.9669	47.161

Compared Methods. We compared our method in terms of the performance of image synthesis with the most influential method HumanNeRF [53] and the latest state-of-the-art method MonoHuman [59] which improves the pose generalization of HumanNeRF and works better than [17, 39]. We also take Ani-NeRF [38] as one of the baselines because this work presents SMPL-based neural blend weight that can better generalize novel poses.

4.1. Quantitative Evaluation

For any set of data in ZJU-MOCAP, we divided the data into training and testing data in a 4:1 ratio, which follows the setting of previous work. Differently, we uniformly select only about 30 frames from the divided training data as few-shot input. This setting can avoid pose leakage compared to the full input in a certain space. Our experimental results in ZJU-MOCAP with few-shot training images are shown in Figure 4, and all the experimental results are shown in Table 1.

Figure 5. **Some of the synthesis results with Subject-387 in ZJU-MOCAP.** We use all the videos of the performers to train and synthesize images with different pose sequence from Subject-387. The baselines produce noticeable artifacts, while our method maintain high-quality image synthesis.

Table 2. **Comparison of Training and Rendering Time** Our method does not use existing acceleration modules. The simple yet effective architecture greatly reduces the computation required, thereby improving the overall speed.

	Training	Rendering
Ani-NeRF [38]	40h	2.73s/it
HumanNeRF [53]	56h	2.37s/it
MonoHuman [59]	70h	5.96s/it
Ours	2.5h ●	0.16s/it ●

For our custom in-the-wild data, we train with a video of a performer spinning one round and use another video of the same person doing different poses for evaluation. The results are shown in Figure 4.

4.2. Qualitative Evaluation

In previous work, researchers often divide data into a certain ratio as novel pose experiments for quantitative evaluation. However, in the most widely used dataset ZJU-MOCAP, performers repeat the same action in a set of data. This leads to the poses in evaluation data being highly similar to poses used for training. Although we mitigate pose leakage issues through few-shot input, a considerable portion of poses in the evaluation data are still similar, resulting in limited difference in the average results.

A simple way to avoid pose leakage problem is to use completely different action sequences as evaluation. But as there is no ground truth of ZJU-MOCAP, we provided comparison results as Figure 5, and synthesized over 200 zero-shot pose images and converted them into 20 videos, which were subjectively evaluated by six or more evaluators as shown in Figure 6.

(a) HumanNeRF [53] (b) Ours

Figure 6. **Subjective evaluation results.** We use cross-rendering of different subjects in the ZJU-MOCAP to ensure that the evaluation of the novel pose is sufficiently novel. Our results were evaluated to have a significantly higher perfect proportion than HumaneNeRF.

The entire evaluation process is single-blind, meaning that the evaluators do not know which specific method generated the results. We also included some test seeds, which serve as the Ground Truth, and all of these seeds received high scores from the evaluators, indicating that their evaluations were professional and objective. The evaluators were asked to evaluate the video in terms of image distortion, artifacts, details, plausibility, and precision, and to provide a final score that was assigned to one of three different levels.

As a result, almost all the evaluators subjectively think our results have a better performance. Our results were classified as perfect in significantly higher numbers than HumanNeRF, with fewer votes for flaws. However, both the results performed poorly in cases of poor data quality.

(a) With Conv-Filter (b) Without Conv-Filter

Figure 7. Comparison of results with Conv-Filter and without Conv-Filter.

Table 3. Ablation study of our method in ZJU-MOCAP, we conducted a quantitative evaluation of the effectiveness of modules, including canonical points refined with F_s, Conv-Filter, and frozen SMPL weights.

	PSNR↑	SSIM↑	†LPIPS↓
w/o F_s refine	30.93	0.971	24.712
w/o filter	9.22	0.607	318.460
Full	31.09	0.974	24.085

4.3. Ablation Studies

Conv-Filter. In our method, the point filter is essential. Our experiments showed that if the sampled points are not filtered, we cannot learn the correct alpha map (see Figure 7) in our experiments, and it greatly increases the computational complexity, requiring longer training time. We further investigated the reason why the phenomenon of alpha map learning errors occurs due to color diffusion into the surrounding space, and we believe that this is determined by the distribution of skin weight. As shown in Figure 2, the learnable weights tend to give negative values to irrelevant joint weights, but this is unreasonable. It can be explained by the fact that these methods do not require filters to avoid the phenomenon, because we observe similar phenomena when we map the learnable weights to the same distribution as ours through sigmoid.

Point-level Feature Refine. It is a common practice to add offsets to the deformation process using neural networks, but previous methods often used time or the pose of the current frame as control information. This frame-level feature often leads to overfitting, but in previous experiments, this phenomenon was not significant due to the similarity and repetition of actions in ZJU-MOCAP. We extract point-level spatial-aware feature while filtering points in ConvFilter. This not only corrects the unnatural joints caused by rigid deformation (see Figure 8, Table 3) but also avoids overfitting compared to the previous frame-level feature (see Figure 9).

(a) With Point-Refine (b) Without Point-Refine

Figure 8. Comparison of results with Point-Refine and without Point-Refine.

(a) With frame-level feature (b) With point-level feature

Figure 9. Ablation experiments using frame-level features as the input to the non-rigid deformation network showed a significant drop in performance on the novel pose.

5. Disscusions

Our method is state-of-the-art in the task of learning implicit human representations from limited input and synthesizing diverse pose images. What's required is only a monocular video even a few images and easily obtainable SMPL information, without the need for additional calculations of texture information, greatly expanding the method's universality. However, our method still has certain limitations: 1. The effectiveness of our method depends on the accuracy of the estimated SMPL, and when the SMPL accuracy is low, the results may be blurred. Currently, the accuracy of SMPL estimation methods is not always satisfactory. 2. Our method uses coordinate voxelization to assist calculation, which may cause image edge serrations. Fine-tuning can be achieved by adjusting the voxel size and convolution kernel size, which will increase computational cost. 3. Our method uses basic SMPL information for training, so it is difficult to drive hand and facial details.

6. Conclusion

We propose a human neural radiance field model that can train with limited inputs and generalize to diverse zero-shot poses. Unlike previous methods, our approach filters the sampling points and obtains point-level features in a voxel volume of explicit vertices, and subsequently deforms the points to a canonical space using general and specific mapping. Our approach uses less than 1% learnable parameters and achieves state-of-the-art novel pose metrics in our experiments, and maintains the best performance with few-shot input. Our approach only requires estimated SMPL information, which can be easily obtained using existing methods, thereby maintaining usability while being able to generalize to industries such as video production.

References

[1] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 696–712. Springer, 2020. 3

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 3

[3] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 3

[4] Mingfei Chen, Jianfeng Zhang, Xiangyu Xu, Lijuan Liu, Yujun Cai, Jiashi Feng, and Shuicheng Yan. Geometry-guided progressive nerf for generalizable and efficient neural human rendering. In *European Conference on Computer Vision*, pages 222–239. Springer, 2022. 3

[5] Yue Chen, Xuan Wang, Xingyu Chen, Qi Zhang, Xiaoyu Li, Yu Guo, Jue Wang, and Fei Wang. Uv volumes for real-time rendering of editable free-view human performance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16621–16631, 2023. 2

[6] Wei Cheng, Su Xu, Jingtan Piao, Chen Qian, Wayne Wu, Kwan-Yee Lin, and Hongsheng Li. Generalizable neural performer: Learning robust radiance fields for human novel view synthesis. *arXiv preprint arXiv:2204.11798*, 2022. 3

[7] SpConv Contributors. Spconv: Spatially sparse convolution library. https://github.com/traveller59/spconv, 2022. 4

[8] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019. 3

[9] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5712–5721, 2021. 3

[10] Xiangjun Gao, Jiaolong Yang, Jongyoo Kim, Sida Peng, Zicheng Liu, and Xin Tong. Mps-nerf: Generalizable 3d human rendering from multiview images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[11] Chen Geng, Sida Peng, Zhen Xu, Hujun Bao, and Xiaowei Zhou. Learning neural volumetric representations of dynamic humans in minutes. *arXiv preprint arXiv:2302.12237*, 2023. 1, 2, 3

[12] Marc Habermann, Weipeng Xu, Michael Zollhofer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5052–5063, 2020. 2, 3

[13] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall, Jonathan T Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5875–5884, 2021. 3

[14] Vinoj Jayasundara, Amit Agrawal, Nicolas Heron, Abhinav Shrivastava, and Larry S Davis. Flexnerf: Photorealistic free-viewpoint rendering of moving humans from sparse views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21118–21127, 2023. 3

[15] Boyi Jiang, Yang Hong, Hujun Bao, and Juyong Zhang. Selfrecon: Self reconstruction your digital avatar from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5605–5615, 2022. 3

[16] Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16922–16932, 2023. 1, 2, 3

[17] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 402–418. Springer, 2022. 1, 2, 3, 6

[18] Youngjoong Kwon, Dahun Kim, Duygu Ceylan, and Henry Fuchs. Neural human performer: Learning generalizable radiance fields for human performance rendering. *Advances in Neural Information Processing Systems*, 34:24741–24752, 2021. 2, 3

[19] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 3

[20] Zhe Li, Zerong Zheng, Yuxiao Liu, Boyao Zhou, and Yebin Liu. Posevocab: Learning joint-structured pose embeddings for human avatar modeling. *arXiv preprint arXiv:2304.13006*, 2023. 3

[21] Yiyi Liao, Katja Schwarz, Lars Mescheder, and Andreas Geiger. Towards unsupervised learning of generative models for 3d controllable image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5871–5880, 2020. 3

[22] 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 Transactions on Graphics (TOG)*, 38(5):1–14, 2019. 2, 3

[23] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020. 3

[24] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM Transactions on Graphics (TOG)*, 40(6):1–16, 2021. 2, 3

[25] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2019–2028, 2020. 3

[26] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019. 3

[27] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 3

[28] 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. *arXiv preprint arXiv:1811.05029*, 2018. 3

[29] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021. 3

[30] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 5

[31] 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. *Communications of the ACM*, 65(1):99–106, 2021. 3

[32] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 3

[33] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021. 2, 3

[34] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 3

[35] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 2, 3

[36] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 2, 3

[37] Bo Peng, Jun Hu, Jingtao Zhou, Xuan Gao, and Juyong Zhang. Intrinsicngp: Intrinsic coordinate based hash encoding for human nerf. *arXiv preprint arXiv:2302.14683*, 2023. 3

[38] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14314–14323, 2021. 2, 3, 6, 7

[39] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9054–9063, 2021. 2, 3, 4, 5, 6

[40] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2, 3

[41] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2304–2314, 2019. 3

[42] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 84–93, 2020. 2, 3

[43] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2446, 2019. 3

[44] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[45] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7495–7504, 2021. 3

[46] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. *Advances in Neural Information Processing Systems*, 34:12278–12291, 2021. 3

[47] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people.

In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11179–11188, 2021. 2

[48] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 3

[49] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG)*, 38(4):1–12, 2019. 3

[50] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12959–12970, 2021. 2, 3

[51] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. Nerf-sr: High quality neural radiance fields using supersampling. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6445–6454, 2022. 2, 3

[52] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2, 3

[53] Chung-Yi Weng, Brian Curless, Pratul P Srinivasan, Jonathan T Barron, and Ira Kemelmacher-Shlizerman. Humannerf: Free-viewpoint rendering of moving people from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16210–16220, 2022. 1, 2, 3, 5, 6, 7, 4

[54] Minye Wu, Yuehao Wang, Qiang Hu, and Jingyi Yu. Multi-view neural human rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1682–1691, 2020. 2, 3

[55] Zijin Wu, Xingyi Li, Juewen Peng, Hao Lu, Zhiguo Cao, and Weicai Zhong. Dof-nerf: Depth-of-field meets neural radiance fields. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1718–1729, 2022. 2, 3

[56] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9421–9431, 2021. 2, 3

[57] Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J Black. Icon: Implicit clothed humans obtained from normals. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13286–13296. IEEE, 2022. 2, 3

[58] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37 (2):1–15, 2018. 2, 3

[59] Zhengming Yu, Wei Cheng, Xian Liu, Wayne Wu, and Kwan-Yee Lin. Monohuman: Animatable human neural field from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16943–16953, 2023. 1, 2, 3, 5, 6, 7, 4

[60] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 2, 3

[61] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5

[62] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 3

[63] Fuqiang Zhao, Wei Yang, Jiakai Zhang, Pei Lin, Yingliang Zhang, Jingyi Yu, and Lan Xu. Humannerf: Efficiently generated human radiance field from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7743–7753, 2022. 2, 3

[64] Zerong Zheng, Xiaochen Zhao, Hongwen Zhang, Boning Liu, and Yebin Liu. Avatarrex: Real-time expressive full-body avatars. *arXiv preprint arXiv:2305.04789*, 2023. 3

[65] Shihao Zhou, Mengxi Jiang, Shanshan Cai, and Yunqi Lei. Dc-gnet: Deep mesh relation capturing graph convolution network for 3d human shape reconstruction. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 171–180, 2021. 3

[66] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 3

HumanNeRF-SE: A Simple yet Effective Approach to Animate HumanNeRF with Diverse Poses

Supplementary Material

7. Network Architexture

7.1. Conv-Filter

We performed channel-by-channel single-layer convolution on the voxel volume we constructed, with convolution kernel weights initialized to one. Our convolution kernel size is 5, padding is 2, and stride is 1 to keep the volume size. Channel-by-channel convolution can preserve the semantic information of high-frequency details.

7.2. Point Refiene

Figure 10. **Visualization of point refine network.** We use F_s and x_o as inputs to the network to obtain offsets $\Delta(x)$ to refine the canonical coordinates after the general rigid deformation. We initialize the bias of the last layer to zero, and the weight is within the range of $(-1e^{-5}, -1e^{-5})$.

7.3. NeRF Network

Figure 11. **Visualization of appearance network.** Follow the baseline [53], we use an 8-layer MLP with width=256, taking as input positional encoding γ of position x and producing color c and density σ. A skip connection that concatenates $\gamma(x)$ to the fifth layer is applied. We adopt ReLU activation after each fully connected layer, except for the one generating color c where we use sigmoid.

8. Canonicalization

To map points in the observation space to the canonical space, our baselines utilize neural networks to learn backward warping weight fields. This method is easy to implement but suffers from poor generalization to unseen poses, as the backward weight fields attempt to learn a spatial weight fields that deform with pose variations, necessitating memorization of weight fields for different spatial configurations. Generalizing to unseen poses using such pose-dependent weight fields is difficult. Our method directly queries the nearest SMPL's LBS weights in an explicit voxel, which is highly efficient and generalizable. However, this method suffers from unnatural deformations when the deformation angle is too large (see Figure 8). To address this issue, we use an additional neural network to learn a residual for canonical points, which is specific to the data and empowered by point-level features, considering the different clothing of the performers.

The rotation R_j and translation T_j for the rigid deformation are represented as:

$$\begin{bmatrix} R_j & T_j \\ 0 & 1 \end{bmatrix} = \prod_{i \in p(j)} \begin{bmatrix} R(\omega_i^c) & o_i^c \\ 0 & 1 \end{bmatrix} \left\{ \prod_{i \in p(j)} \begin{bmatrix} R(\omega_i) & o_i \\ 0 & 1 \end{bmatrix} \right\}^{-1} \tag{14}$$

where $p(j)$ is the ordered set of parents of joint j in the kinematic tree, ω_i defines local joint rotations using axis-angle representations, $R(\omega_i) \in \mathbb{R}^{3\times3}$ is the converted rotation matrix of ω_i via the Rodrigues formula, and o_i is the i-th joint center.

9. Ablation Study

Voxel Size. In the Conv-Filter, we voxelized all coordinates, which resulted in some loss of information compared to a dense space. However, this greatly facilitated our subsequent processing and computations. We conducted ablation experiments on the voxel size used in the voxelization process, as shown in Table 4. A smaller voxel size did not result in higher accuracy because the prior does not perfectly match the actual human body, and clothing is also outside the prior.

Weight Distribution. In the main text, we discussed the importance of filtering operations. It is worth noting that previous methods did not require similar operations. Our experiments suggest that this is because the learned neural weight field assigns negative weights to irrelevant joints in

Table 4. Ablation experiment on the voxel size of our method in ZJU-MOCAP. For different subjects, there is a different optimal voxel size. We use 0.02 as voxel size in the experiment section because it has the best overall performance.

	PSNR↑	SSIM↑	†LPIPS↓
voxel=0.01	30.49	0.971	26.386
voxel=0.015	30.69	0.972	25.354
voxel=0.02	31.09	0.974	24.085
voxel=0.025	31.07	0.974	24.705
voxel=0.03	30.72	0.973	25.013

Figure 12. By mapping the learned weight field of HumanNeRF [53] to the same distribution as SMPL weight using softmax, the same phenomenon occurred.

the data. When we simply use $softmax$ to map the neural weight field to a distribution that is the same as the SMPL weight, $\sum_{j \in J} w_j = 1, w_j > 0$, similar phenomena occur, see Figure 12.

10. More Results

Quantitative experiments. Due to the space limitation of the main text, the Figure 4 provided in the main text is the few-shot image synthesis result and Table 1 is a overall table.

Table 6, Table 7 and Table 8 are detailed data for Table 1. For ZJU-MOCAP, we use videos from the same six subjects as in previous work. For our IN-THE-WILD DATA, each subject is composed of multiple pose sequences to enable more convincing novel pose experiments. The videos are captured with a single camera, and SMPL are estimated with ROMP [47].

In order to better compare the performance of different methods, we synthesized the results using full input images. Figure 14 shows the synthesis results on ZJU-MOCAP, and Figure 13 shows the image synthesis results on the IN-THE-WILD DATA. Figure 15 directly compares the novel pose image synthesis ability of the three methods under two input conditions.

Table 5. Pose similarity of test poses and train poses in previous novel pose experiments. The previous setting of novel pose is not novel enough because of the high similarity.

	Min	Max	Average
377	0.876	0.997	0.919
386	0.939	0.995	0.964
387	0.969	0.996	0.985
392	0.845	0.957	0.909
393	0.834	0.998	0.904
394	0.760	0.993	0.842

Compared with Figure 4, the defects of the baselines have been significantly improved on ZJU-MOCAP, but there are still obvious defects in these methods on IN-THE-WILD DATA. This is highly related to the data distribution.

Pose similarity. We found that the novel pose experimental setting in previous studies is unreasonable because the test poses and training poses are very similar, as the characters in ZJU-MOCAP perform similar movements repeatedly. This also encourages us to use custom data and subjective research of results on poses without ground truth images. Novel poses synthesized by HumanNeRF should be sufficiently novel and easy to obtain, rather than being limited to professional laboratories.

To quantify the similarity of poses, we calculate the highest cosine similarity with all training poses for each pose in the test data. As shown in Table 5, simply dividing each subject into training and testing data in a 4:1 ratio may result in highly similar poses in the testing data being present in the training data.

In our experiments, high pose similarity does not always result in a decrease in baseline performance, because during the training process, similar poses and viewpoints are limited. For example, the pose similarity of Subject 386 is very high, but the corresponding pose only appears at the beginning and can only see the performer's right side, so when we synthesize this highly similar pose, the performance of the baselines is not good.

Qualitative results. In the qualitative experiments, we drive the model to generate new pose images by using pose sequences of different performers. We make a series of image results into a video to help the human eyes better distinguish the performance of different methods. The video results are included in the supplementary materials in the form of a compressed file.

Table 6. **Comparison of full input results on ZJU-MOCAP.** Our method shows a leading advantage in LPIPS, which is aligned with human perception. Each data in the table represents the average value of all test frames' metrics in the corresponding video, which is consistent with the previous studies.

Methods	Subject377			Subject386			Subject387		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
Ani-NeRF [38]	22.12	0.8790	51.796	23.92	0.8545	55.697	16.26	0.7850	89.790
HumanNeRF [53]	30.74	0.9795	17.387	33.46	0.9716	36.326	30.30	0.9768	20.010
MonoHuman [59]	31.82	0.9822	17.561	30.10	0.9561	69.107	30.43	0.9755	23.954
Ours	31.34	0.9826	15.129	33.80	0.9741	32.990	29.36	0.9742	21.462

Methods	Subject392			Subject393			Subject394		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
Ani-NeRF [38]	22.78	0.8610	67.744	20.37	0.8417	75.381	22.05	0.8537	69.188
HumanNeRF [53]	31.80	0.9743	26.811	29.80	0.9708	25.615	30.85	0.9702	22.783
MonoHuman [59]	32.06	0.9749	27.043	29.69	0.9701	26.570	31.37	0.9720	23.521
Ours	31.75	0.9740	25.537	29.50	0.9642	25.462	30.81	0.9697	23.929

Table 7. **Comparison of few-shot input results on ZJU-MOCAP.** On the most important metric LPIPS, our method demonstrates the best results. our method exhibits less performance degradation with few-shot input indicates that our method does not overly rely on data to fit the model, unlike previous methods.

Methods	Subject377			Subject386			Subject387		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
Ani-NeRF [38]	21.77	0.8313	66.532	24.48	0.8386	74.342	20.74	0.8279	65.747
HumanNeRF [53]	30.33	0.9799	18.510	31.47	0.9618	48.410	27.92	0.9610	39.941
MonoHuman [59]	31.02	0.9774	22.562	31.26	0.9601	56.916	28.5	0.9619	43.147
Ours	31.07	0.9806	18.484	32.44	0.9644	43.876	28.05	0.9612	39.733

Methods	Subject392			Subject393			Subject394		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
Ani-NeRF [38]	22.49	0.8433	61.917	21.96	0.8384	59.169	21.65	0.8240	61.328
HumanNeRF [53]	31.55	0.9747	28.043	29.45	0.9694	26.930	28.66	0.9629	36.507
MonoHuman [59]	31.48	0.9739	30.090	29.45	0.9691	30.113	28.86	0.9636	36.139
Ours	31.36	0.9734	27.604	29.25	0.9681	27.328	28.51	0.9626	35.479

Table 8. **Comparison of results on the IN-THE-WILD DATA.** Since IN-THE-WILD DATA is not captured in laboratory conditions and only contains monocular information, the accuracy of SMPL estimation is lower compared to ZJU-MOCAP, which leads to a decrease in overall performance metrics. However, this is more in line with real-world application scenarios. Our method demonstrates the best performance, especially in the LPIPS metric, which reflects image quality the most. Even with reduced input data, our method maintains excellent performance, while other methods experience a greater degree of decline.

Full Input	S1			S2			S3		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
HumanNeRF [53]	32.94	0.9737	43.259	26.65	0.9650	41.505	27.32	0.9500	59.620
MonoHuman [59]	32.99	0.9725	46.272	26.74	0.9683	42.198	27.72	0.9509	66.400
Ours	33.72	0.9764	42.343	26.43	0.9709	39.174	27.54	0.9524	57.408

Few-shot Input	S1			S2			S3		
	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓	PSNR↑	SSIM↑	†LPIPS↓
HumanNeRF [53]	33.37	0.9761	45.012	25.82	0.9599	43.326	27.27	0.9495	62.384
MonoHuman [59]	33.58	0.9744	50.561	26.57	0.9640	45.955	27.48	0.9522	72.143
Ours	33.70	0.9768	41.954	26.45	0.9712	39.894	27.62	0.9525	59.636

Figure 13. **Image synthesis results with full input on IN-THE-WILD DATA.** Our method maintains good performance at joint junctions, but the results of the baselines are blurry and have unnatural distortions.

Figure 14. **Image synthesis results with full input on ZJU-MOCAP.** For Subject 386 (line 1), the baselines still have very poor image synthesis results. For Subject 392 and 393, there are still irregular deformations and artifacts in the image synthesis results, whereas our method achieves the best performance in visual comparison.

Figure 15. **Comparison between full input and few-shot input.** HumanNeRF [53] and MonoHuman [59] exhibit more artifacts and unnatural deformations with less data. MonoHuman [59] even produces hand missing. Our method maintains almost unchanged performance under the same testing conditions.