Efficient Integration of Neural Representations for Dynamic Humans

Wensheng Li, Lingzhe Zeng, Chengying Gao, Ning Liu

***Abstract*—While numerous studies have explored NeRF-based novel view synthesis for dynamic humans, they often require training that exceeds several hours, limiting their practicality. Efforts to improve training efficiency have also encountered challenges because it is hard to optimize non-rigid transformations, thus leading to coarse renderings. In this work, we introduce an innovative approach for efficiently learning and integrating neural human representations. To achieve this, we propose a comprehensive utilization of the features stored in both canonical and observational spaces, facilitated through a collaborative refinement process that integrates canonical representations with observational details. Specifically, we initially propose decomposing high-dimensional multi-space feature volume into several feature planes, subsequently utilizing matrix multiplication to explicitly establish the correlations between different planes. This enables the simultaneous optimization of their counterparts across all dimensions by optimizing interpolated features, efficiently integrating associated details, and accelerating the rate of convergence. Additionally, we use the proposed collaborative refinement process to iteratively enhance the canonical representation. By integrating multi-space representations, we further facilitate the co-optimization of multiple frames' time-dependent observations. Experiments demonstrate that our method can achieve high-quality free-viewpoint renderings within nearly 5 minutes of optimization. Compared to state-of-the-art approaches, our results show more realistic rendering details, marking a significant advancement in both performance and efficiency.**

***Index Terms*—Dynamic Human Reconstruction, Neural Rendering, Efficient Optimization.**

I. Introduction

FREE-VIEWPOINT rendering of dynamic humans holds a pivotal role in advancing the fields of AR/VR, video games, and filmmaking industries. Numerous studies [1], [2], [3], [4], [5], [6] have shown the potential for realistic rendering employing implicit Neural Radiance Fields (NeRF [7]). Nonetheless, these methods require over ten hours of training, limiting their widespread applicability.

Recent studies have explored the optimization process of NeRF, investigating the utilization of explicit structures [8], [9], [10], [11], [12]. Expanding on these findings, several studies have delved into the development of efficient representations tailored specifically for dynamic human subjects [13], [14], [15], [16], [17]. Remarkably, Instant-Nvr [14] has emerged as a significant breakthrough, accomplishing optimization within approximately 5 minutes. However, despite these advancements, most methods attempt to simulate clothing details through the prediction of non-rigid transformations, a process that is challenging to optimize and often yields coarse rendering results. Furthermore, these methods primarily focus on refining the human representations along with time-specific details within a singular frame in canonical space during each training iteration, thereby overlooking the essential inter-frame and inter-space associations necessary for comprehensive co-optimization. The aforementioned issues not only lead to longer training periods but also cause insufficient rendering details, thereby impeding their broader application.

In this paper, our primary objective is to achieve efficient reconstruction of dynamic human performance within minutes, along with the realization of realistic free-viewpoint renderings. Contrary to the aforementioned methods explicitly predict non-rigid transformations[2], [3], [14], we pioneer the integration of multiple neural representations of human subjects, with a particular emphasis on incorporating multi-space representations. To fully utilize the features stored in canonical and observational spaces, we propose a collaborative refinement process that integrates canonical representations with observational details. This innovative strategy not only leverages the benefits of optimization with shared canonical-space representation but also preserves observational time-dependent details for each frame. Moreover, optimizing for time-dependent details is shown to be more tractable and less computationally onerous compared to addressing non-rigid optimization.

Inspired by Kplanes [18], we propose a factorization of multi-space high-dimension feature volumes into several feature planes. Specifically, we use three spatial feature planes to represent 3D volume in canonical space. And we utilize six feature planes to represent the 4D volume in observation space, where the initial three store spatial information and the subsequent three capture space-time changes. During this stage, we do not explicitly construct spatial feature planes for observation space. Instead, we construct them by aggregating space-time feature planes through Matrix Multiplication along the time dimension.

More generally, as the aforementioned multi-space feature planes inherently capture 2D coordinate features but overlook features along the third dimension and correlations among different planes, we advance the application of Matrix Multiplication between spatial feature planes. As shown in Fig 1, we can delineate and reinforce the inter-plane correlations with Matrix Multiplication, which facilitates a more cohesive integration of features across varying dimensions within each respective space. And optimizing the interpolated feature in this manner explicitly augments the optimization of their counterparts across each dimension, effectively integrating as-

W. Li, L. Zeng, C. Gao, N. Liu are with School of Computer Science and Engineering, Sun Yat-sen University.

sociated details and consequently accelerating the convergence speed.

Subsequently, we refine the canonical representation by using observation-space details through a collaborative refinement process. The objective here is to co-optimize multiple frames of time-dependent features in observation space and facilitate the integration of multi-space representations. This is achieved by blending the features from the current frame with those from corresponding positions in selected alternate frames. We further employ features from observation-space spatial feature planes, which enables the exploitation of residual features at identical positions across all frames, thereby amplifying the observation details. Finally, we build an iterative optimization framework and embed the refinement process to progressively acquire the refined canonical representation.

In summary, our key contributions include:

- Proposing the integration of canonical and observation space features for dynamic human reconstruction, accomplished by factorizing multi-space, high-dimensional feature volumes into a series of feature planes.
- Proposing a novel integration of feature planes via Matrix Multiplication, which explicitly integrates associated details between different planes, leading to a substantial acceleration of the optimization process.
- Efficiently integrate multi-space representations through a collaborative refinement process, augmenting the details of canonical representation from integrated multi-frame observations during iterative optimization.
- Experiments show that our method are capable of rendering free-viewpoint results with fine details in approximately 5 minutes, manifesting a substantial advancement in both performance and efficiency relative to the SOTA.

II. Related Work

A. Mesh-based Human Modeling

Early studies on human modeling rely on scanning equipment [19], [20]. These methods often involve extended optimization time, spanning from several hours to even days. In contrast, recent mesh-based methods [21], [22], [23] predefined parametric models [24], [25] for human modeling. Although these approaches can accurately capture pose and shape [26], [27], their optimization process demands considerable time and a sufficient number of 3D labeled annotations [28]. Additionally, due to topology and vertex count constraints, they face difficulty in reconstructing fine cloth details [29], [30], [31]. Furthermore, they strongly depend on the traditional rendering pipeline [32] for free-viewpoint synthesis, which presents challenges in optimizing rendered results in an end-to-end manner at the image level due to its non-differentiability.

B. NeRF for Human Reconstruction

Recent progress in NeRF have extended its capabilities by integrating the SMPL model [24], enabling free-viewpoint rendering of dynamic humans. Some optimize the latent codes on the human mesh surface [1], [33], [34], while others try to transform the sampled points to canonical space [35], [3], [36], [4], [37], [38], enabling the utilization of observations across all frames. However, these methods require a substantial amount of time for optimizing the geometric transformation.

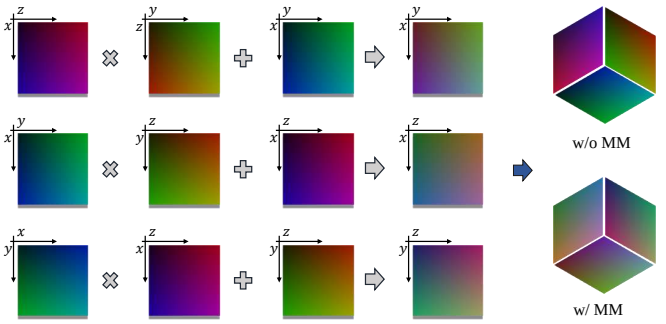


Fig. 1. **Illustration of our proposed integration strategy for feature planes.** We obtain associated features by integrating distinct planes and subsequently augmenting them to the original planes to enhance features along the third axis. The utilization of Matrix Multiplication (MM) can strengthens the correlation between the three feature planes.

To simulate details such as cloth folds, researchers incorporate non-rigid residual transformations [2], [3], [39] or dynamic embeddings [33] into canonical representations. Some methods [40], [41], [42] use time as an extra condition in NeRF, building a 4D function for frame-specific details. But they lack a stable canonical representation and demand extensive optimization time, limiting their efficacy in leveraging shared information. Other approaches try to mitigate artifacts by solving deformation functions [43], [44], [45]. However, the aforementioned methods face challenges due to the significant time required for optimization.

C. Acceleration for NeRF-based Human Reconstruction

NeRF employs a fully implicit approach to represent 3D human, resulting in low optimization efficiency due to the complexity of fitting MLPs. To address this challenge, some approaches [9], [10], [8], [46] propose incorporating explicit geometric representations like multiresolution hash encoding or voxel grids, offering the potential to reduce training time from hours to minutes. But the above representations exhibit significant space complexity growth with increasing dimensionality in dynamic scenarios. To address this issue, TensoRF [12] factorizes the 4D tensor into multiple lowrank 2D matrix and 1D vectors components. HexPlane [47] further extends TensoRF to dynamic scenarios. On the other hand, EG3D [11] proposed to decompose the feature volume into three planes to enhance memory efficiency, whose values are added together to represent the volume. Building upon these, [18], [15] extend it to a 4D scenes with additional 2D feature planes corresponding to space-time coordinates.

To enhance the efficiency of dynamic human modeling, recent works have delved these technologies into human representations. NeRFBlendShape [13] efficiently generates facial NeRF with expression bases in just 20 minutes. However, it is unsuitable for dynamic humans due to the higher complexity of human poses and the need to capture more surface details. IntrinsicNGP [16] suggests employing UVD

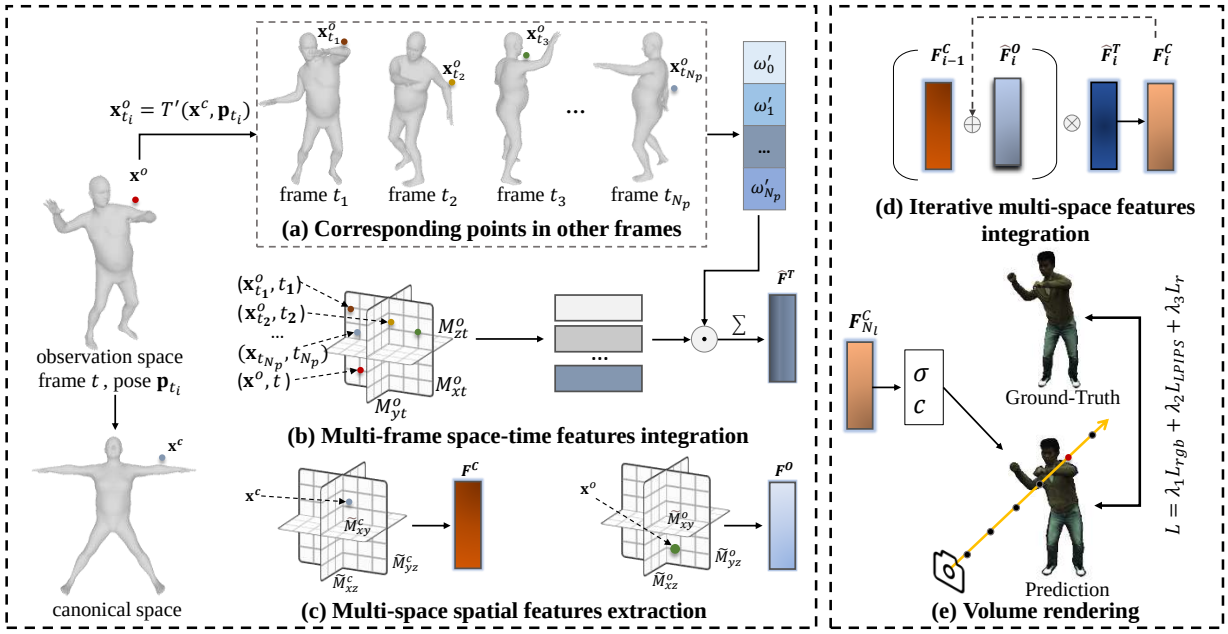


Fig. 2. **Overview of the proposed method**. Given a query point $\mathbf{x}^o$ at frame t, we find its corresponding points at the selected N_p frames, and then integrate the interpolated space-time features with the predicted correlation weights. Subsequently, spatial features are extracted from multi-space. Expanding upon the canonical representation, we propose to refine the canonical features by observational details through the designed iteratively optimization framework. More details about the optimization process can be found in Fig. 3.

coordinates to aggregate inter-frame information for dynamic human representation. However, this approach necessitates the construction of additional offset fields to capture fine geometry details, resulting in relative longer training times. HumanRF [48] lifts Instant-NGP [8] to the temporal domain with four 3D hash grids and four 1D dense grids, leading to a relatively higher spatial cost. To enhance the cross-plane communication, Rodin [49] uses the 3D-aware convolutions, which necessitate an extended training duration. In contrast, Instant-Nvr [14] introduces a part-based representation to more efficiently allocate network capacity among different body parts. Additionally, they utilize a novel 2D motion parameterization scheme to accelerate deformation field learning. However, as observed in the results, although the representation can accelerate reconstruction, obtaining fine details in a short time remains challenging.

On the other hand, [50], [51], [52], [53] use images to pre-train a neural field for a generic subject and then fine-tune it quickly for specific subjects, improving training efficiency. However, the fine-tuning process still takes hours, and these methods haven't achieved the same level of performance as those designed for specific subjects.

III. METHOD

Given captured videos of a human subject, the objective of this paper is to efficiently optimize the 3D human reconstruction process and generate realistic free-viewpoint renderings. We assume a prior knowledge of human poses and foreground segmentation across all frames.

In NeRF-based dynamic human representations, we begin by transforming observation-space sampled point $\mathbf{x}^o = (x^o, y^o, z^o)$ at frame t to canonical space, resulting in $\mathbf{x}^c = T^{-1}(\mathbf{x}^o, \mathbf{p}_t)$, where T^{-1} is the inverse Linear Blend Skinning (LBS) transformation [2], [54], $\mathbf{p}_t$ denotes the human pose at frame t, and $\mathbf{x}^c = (x^c, y^c, z^c)$.

In order to fully exploit the features from both canonical and observational spaces, we propose a collaborative refinement process. This process integrates canonical representations with diverse observational details. To fulfill this, we decompose feature volumes in both canonical and observational spaces into 9 feature planes to facilitate efficient feature storage and optimization. These planes are then integrated through Matrix Multiplication to reinforce the inter-plane correlations along each axis (Sec. III-A). Subsequently, we refine the canonical representation by iteratively integrating observation-space features, yielding a detailed feature representation(Sec. III-B). Finally, we use the detailed feature for predicting both color and density. For color prediction, we propose integrating color features and lighting features to further improve rendering quality(Sec. III-C).

A. Feature planes integration

In order to yield low memory usage and fast training, we propose a multi-space factorization strategy. Specifically, we utilize the Tri-planes factorization [11] for the feature volume in canonical space, resulting in three feature planes denoted as $\mathbf{M}^c_{xy}$, $\mathbf{M}^c_{xz}$, and $\mathbf{M}^c_{yz}$. Meanwhile, for the observation-space feature volume, we employ the Hex-planes factorization [18], resulting in six feature planes. We denote the spatial planes as $\mathbf{M}^o_{xy}$, $\mathbf{M}^o_{xz}$, and $\mathbf{M}^o_{yz}$, and the space-time planes as $\mathbf{M}^o_{xt}$, $\mathbf{M}^o_{yt}$, and $\mathbf{M}^o_{zt}$. Among them, the feature planes of canonical space record the static features of the canonical human, while the feature planes of observation space capture the dynamic details.

Different from Kplanes [18], we do not explicitly construct observation-space spatial feature planes. We further decompose them along the time dimension:

$$\begin{aligned}\mathbf{M}^o_{xy} &= \mathbf{M}^o_{xt}(\mathbf{M}^o_{yt})^{\mathbf{T}} \\ \mathbf{M}^o_{xz} &= \mathbf{M}^o_{xt}(\mathbf{M}^o_{zt})^{\mathbf{T}} \\ \mathbf{M}^o_{yz} &= \mathbf{M}^o_{yt}(\mathbf{M}^o_{zt})^{\mathbf{T}}\end{aligned} \tag{1}$$

where $(\mathbf{M})^{\mathbf{T}}$ means the transpose of the feature plane $\mathbf{M}$. In this way, $\mathbf{M}^o_{xy}$ aggregates dynamic features along the X-axis and along the Y-axis over all times.

Matrix multiplication can linearly establish a correlation between two feature planes, and this operation can be further extended to integrate any two spatial feature planes within the same space. Specifically, we decompose the feature plane into two associated matrix factors located within distinct quadrants:

$$\begin{aligned}\widetilde{\mathbf{M}}^i_{xy} &= \mathbf{M}^i_{xz}(\mathbf{M}^i_{yz})^{\mathbf{T}} + \mathbf{M}^i_{xy} \\ \widetilde{\mathbf{M}}^i_{xz} &= \mathbf{M}^i_{xy}\mathbf{M}^i_{yz} + \mathbf{M}^i_{xz} \\ \widetilde{\mathbf{M}}^i_{yz} &= (\mathbf{M}^i_{xy})^{\mathbf{T}}\mathbf{M}^i_{xz} + \mathbf{M}^i_{yz}\end{aligned} \tag{2}$$

where $i \in \{c, o\}$. This method involves decomposing the feature plane in the XY quadrant into the XZ and YZ quadrants, and then combining them using Matrix Multiplication, resulting in the feature plane in the XY quadrant containing Z-axis features.

Drawing upon the processed feature planes, we obtain the canonical-space spatial features F^C, observation-space spatial features F^O and observation-space space-time feature F^T:

$$\begin{aligned}F^C(\mathbf{x}^c) &= \sum_{i\in C}\mathcal{B}(\widetilde{\mathbf{M}}^c_i, \pi_i(\mathbf{x}^c)),\ C = \{xy, xz, yz\} \\ F^O(\mathbf{x}^o) &= \sum_{i\in O}\mathcal{B}(\widetilde{\mathbf{M}}^o_i, \pi_i(\mathbf{x}^o)),\ O = \{xy, xz, yz\} \\ F^T(\mathbf{x}^o, t) &= \sum_{i\in T}\mathcal{B}(\mathbf{M}^o_i, \pi_i(\mathbf{x}^o)),\ T = \{xt, yt, zt\}\end{aligned} \tag{3}$$

where π_i projects the point to the ith feature plane and $\mathcal{B}$ is bilinear interpolation. Note that we actually build 4-layer feature planes with multi-resolution. Throughout the interpolation process, we obtain the features of each layer and subsequently concatenate them for future utilization.

Discussion: In this section, we introduce a novel approach that decomposes feature planes into two associated matrix factors within distinct quadrants. For example, the XY feature plane is decomposed into XZ and YZ quadrants, which are subsequently integrated using Matrix Multiplication to incorporate Z-axis features into the XY plane. Diverging from conventional tensor decomposition techniques, our matrix multiplication method performs an inverse operation, starting from existing low-dimensional feature planes to construct new matrices with higher-dimensional features. In essence, our approach is aligned with the goal of decomposition-based techniques—to capture and represent multidimensional data efficiently—while placing a greater emphasis on the interrelations among features across dimensions.

More specifically, unlike decomposition-based methods like TensoRF [12], which only combine the XY plane with the Z-axis feature vector but overlook the direct relationship between the Z-axis vector and the XZ or YZ planes, our method emphasizes the association with features along each axis. Hexplane [47] realizes reconstruction through the combination of spatial feature planes and space-time feature planes in the observation space, whereas we use both canonical and observation space planes for spatial features. And we don't construct observational spatial feature planes directly but decompose them into space-time feature planes, establishing correlations between spatial and space-time features.

B. Iterative multi-space features integration

Using coordinate (x^c, y^c, z^c) and its feature $F^C(\mathbf{x}^c)$ can characterize the sampled point but ignores non-rigid transformations caused by pose changes. To tackle this issue, many approaches attempt to predict non-rigid transformations based on (x^c, y^c, z^c) and pose $\mathbf{p}_t$ [35], [2], [3], but optimizing this prediction in a few minutes is challenging. Therefore, we use observation-space features $F^O(\mathbf{x}^o)$ and $F^T(\mathbf{x}^o, t)$ instead of offset prediction.

As shown in Fig. 2, we incorporate time-dependent details into the canonical representation through observation features. In addition to utilizing $F^T_t(\mathbf{x}^o, t)$ of the current frame, we propose leveraging the observations from additional N_p candidate frames to enhance the prediction. Specifically, we select these candidate frames based on the human pose of each frame. To achieve it, we apply the KMeans algorithm for clustering with the pose parameters, and partition pose parameters across N frames into N_p clusters, subsequently obtaining the center for each cluster and resulting in N_p frames with the most distinct poses. These N_p frames correspond to times $t_1, t_2, \ldots, t_{N_p}$, with corresponding poses $\mathbf{p}_{t_1}, \mathbf{p}_{t_2}, \ldots, \mathbf{p}_{t_{N_p}}$. We subsequently apply the forward LBS transformation T to $\mathbf{x}^c$ and obtain $\mathbf{x}^o_{t_i}$ at the t_i-th selected frame:

$$\mathbf{x}^o_{t_i} = T(\mathbf{x}^c, \mathbf{p}_{t_i}),\ i = 1, 2, , \ldots, N_p. \tag{4}$$

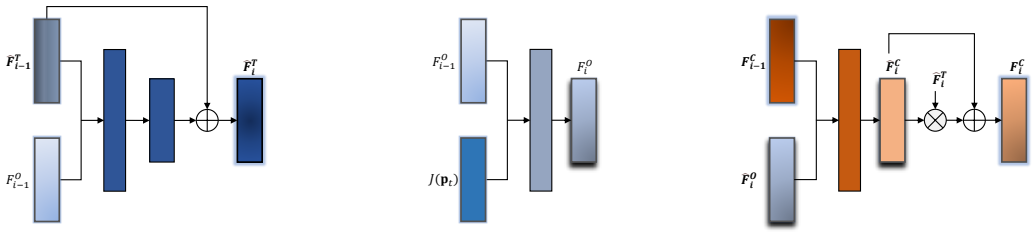
To integrate the time-dependent features of the corresponding positions from the selected N_p frames with the feature of the current frame, we propose utilizing the pose parameters from these $N_p + 1$ frames to obtain the respective fusion weights. Specifically, we initially concatenate pose vectors and create $\mathbf{P} = [\mathbf{p}_t, \mathbf{p}_{t_1}, \ldots, \mathbf{p}_{t_{N_p}}]$. Subsequently, we derive the query $\mathbf{Q} = \mathbf{W}_q\mathbf{P}$ and key $\mathbf{K} = \mathbf{W}_k\mathbf{P}$ using projection matrices $\mathbf{W}_q$ and $\mathbf{W}_k$. This allows us to compute the self-attention weights [55] for the $N_p + 1$ features:

$$\omega_i = \text{softmax}\left(\frac{Q_i \cdot K_i^T}{\sqrt{d_k}}\right),\ i = 0, 1, \ldots, N_p, \tag{5}$$

where $\sqrt{d_k}$ is the scaling factor. To preserve more point features of the current frame, we set the weight of $F^T(\mathbf{x}^o, t)$ to 1 and update $\omega'_i = \omega_i/\omega_0$. As a result, we obtain the integrated time-dependent feature:

$$\widehat{F}^T(\mathbf{x}^o, t) = F^T(\mathbf{x}^o, t) + \sum_{i=1}^{N_p}\omega'_i F^T(\mathbf{x}^o_{t_i}, t_i). \tag{6}$$

Since we build the spatial feature planes in the observation space using the space-time planes, $F^O(\mathbf{x}^o)$aggregates the features along the time dimension at $\mathbf{x}^o$, enabling us to leverage the residual information across all frames at this position.



(a) observation-space features integration (b) spatial and pose features integration (c) Multi-space features integration

Fig. 3. **Each stage of our iterative optimization module is structured as follows**: (a) Architecture for the first stage of iteratively optimization framework. In this stage, we concatenate and integrate the observation-space time-dependent feature $\widehat{F}^T$ with the spatial feature F^O. Residual connection is used to save the space-time feature generated in the previous iteration. (b) Architecture for the second stage of iteratively optimization framework. In this stage, we concatenate and integrate the observation-space spatial feature F^O with the pose-dependent feature $J(\mathbf{p}_t)$. $J(\mathbf{p}_t)$ is generated from the pose vector $\mathbf{p}_t$ with a tiny MLP. (c) Architecture for the third stage of iteratively optimization framework.In this stage, we begin by concatenating and integrating the observation-space spatial feature F^O with the canonical feature F^C. Subsequently, we enhance the details of $\widehat{F}_i^C$ by considering $\widehat{F}^T$ as the attention weight for the current frame.

Subsequently, we build an iterative optimization framework that progressively refine F^C by incorporating observation details. The optimization process comprises three stages, as delineated below, with i representing the i-th iteration, ranging from 1 to N_l. F_0^C, F_0^O, and F_0^T correspond to F^C, F^O, and F^T, respectively.

Stage 1: Iterative observation-space features integration. Fig. 3(a) depicts the integration process of time-dependent observation-space feature $\widehat{F}^T$ with spatial feature F^O via a single-layer MLP E_a in this stage. Throughout the iterative process, we refine the observation details by combining information from both spatial and temporal dimensions:

$$\widehat{F}_i^T = E_a(\widehat{F}_{i-1}^T, F_{i-1}^O) \tag{7}$$

Stage 2: Iterative spatial and pose features integration. Although F^O aggregates observational details from all frames, for the utilization of current pose-related features, we propose integrating F^O with the pose-dependent feature $J(\mathbf{p}_t)$. This is illustrated in Fig. 3(b), where we achieve the integration by employing E_b at this stage, progressively enriching pose-specific details:

$$F_i^O = E_b(F_{i-1}^O, J(\mathbf{p}_t)) \tag{8}$$

Stage 3: Iterative multi-space features integration. As shown in Fig. 3(c), we iteratively refine canonical feature F^C by integrating observation-space details. Within this stage, we initially integrate spatial features from both canonical and observational space with E_c:

$$\widehat{F}_i^C = E_c(F_i^O, F_{i-1}^C) \tag{9}$$

Subsequently, we add more details for $\widehat{F}_i^C$ by regarding $\widehat{F}^T$ as the attention weight for the current frame:

$$F_i^C = \widehat{F}_i^C \times \text{Sigmoid}(\text{AvgPool}(\widehat{F}_i^T)) + \widehat{F}_i^C \tag{10}$$

In the above process, MLPs (referred to as E_a, E_b and E_c) use identical weights throughout the N_l iterations to achieve progressively refinement. And finally we can get the refined canonical representation by combining features from multi-space and co-optimizing the multiple frames of time-dependent observations.

Discussion: In this section, we introduce a method for progressively refining canonical feature by integrating it with features from the observation space. Herein, the spatial feature from the observation space is derived from integrating space-time features across all frames. And we further integrate current-time feature with an additional selected set of N_p frames' time-dependent features, achieving co-optimization across multiple frames. An iterative optimization framework is constructed for the progressive refinement of canonical features using observations. This enables the comprehensive use of canonical features for shared optimization across all frames, improving efficiency while detailing human dynamics via observation-space features.

C. Volume rendering

Once we get $F_{N_l}^C$ for the query point $\mathbf{x}^o$, we feed the feature to a small MLP E_d to predict the density value σ:

$$(\sigma, \mathbf{f}) = E_d(F_{N_l}^C(T(\mathbf{x}^o, \mathbf{p}_i))), \tag{11}$$

where $\mathbf{f}$ is the geometric feature. For color value prediction, we first extract RGB feature F^{rgb} from $\mathbf{f}$ and $F_{N_l}^C$, then generate lighting feature F^l for the ray $\mathbf{r}$ using the following equations:

$$F^{rgb} = E_r(\mathbf{f}, F_{N_l}^c), \;\; F^l = E_l(\gamma(\mathbf{r}), \ell_t). \tag{12}$$

Here, $\gamma(\mathbf{r})$ represents the positional encoding of the ray $\mathbf{r}$, while ℓ_t serves as the latent embedding for frame t, and feature extraction is performed using MLPs denoted as E_r and E_l, respectively. Then, on this basis, we predict the color value $\mathbf{c}$ using MLP E_d:

$$F^{color} = F^{rgb} \times \text{Sigmoid}(F^l), \;\; \mathbf{c} = E_d(F^{color}) \tag{13}$$

Finally, the expected value $C(\mathbf{r})$ of ray $\mathbf{r}$ with D samples can be expressed as:

$$C(\mathbf{r}) = \sum_{i=1}^{D} \left(\prod_{j=1}^{i-1} (1 - \alpha_j) \right) \alpha_i \mathbf{c} \tag{14}$$

where $\alpha_i = 1 - \exp(-\sigma \Delta\delta_i)$ and $\Delta\delta_i$ represents the interval between samples i and $i+1$.

TABLE I
QUANTITATIVE COMPARISON ON THE ZJU-MOCAP DATASET FOR BOTH NOVEL VIEW SYNTHESIS AND NOVEL POSE SYNTHESIS TASKS. WE PRESENT THE AVERAGE METRIC VALUES ACROSS ALL 6 SUBJECTS, HIGHLIGHTING THE BEST PERFORMANCE IN **BOLD TEXT** AND UNDERLINE FOR THE SECOND PERFORMANCE. NOTE THAT WE ONLY PRESENT THE SSIM VALUE ROUNDED TO THREE DECIMAL PLACES, AND THE ACTUAL COMPARISON INVOLVES A GREATER NUMBER OF DIGITS.

	Params(M)	Training Time	Novel View Synthesis			Novel Pose Synthesis		
			PSNR ↑	SSIM ↑	LPIPS* ↓	PSNR ↑	SSIM ↑	LPIPS* ↓
NB [1]	4.34	$\sim$ **10** hours	29.31	0.963	52.08	29.52	0.964	52.64
AN [35]	1.42	$\sim$ **10** hours	30.12	0.967	46.18	29.70	0.967	44.03
AS [2]	1.39	$\sim$ **10** hours	30.62	0.972	32.88	30.00	0.971	33.79
HumanNeRF [3]	64.42	$\sim$ **12** hours	30.53	0.970	31.31	30.29	0.969	32.40
SANeRF [4]	1.18	$\sim$ **15** hours	31.15	**0.972**	37.64	30.85	**0.971**	37.22
Instant-Nvr [14]	285.99	$\sim$ **5** min	31.02	0.971	38.84	30.98	0.971	37.91
Ours	16.75	$\sim$ **5** min	**31.37**	0.972	**30.62**	**31.26**	0.971	**31.67**

TABLE II
QUANTITATIVE COMPARISON ON THE MONOCAP DATASET FOR BOTH NOVEL VIEW SYNTHESIS AND NOVEL POSE SYNTHESIS TASKS. WE PRESENT THE AVERAGE METRIC VALUES ACROSS ALL 4 SUBJECTS, HIGHLIGHTING THE BEST PERFORMANCE IN **BOLD TEXT** AND UNDERLINE FOR THE SECOND PERFORMANCE.

	Novel View Synthesis			Novel Pose Synthesis		
	PSNR ↑	SSIM ↑	LPIPS* ↓	PSNR ↑	SSIM ↑	LPIPS* ↓
NB [1]	32.42	0.986	19.87	31.99	0.986	20.75
AN [35]	32.06	0.986	17.34	31.67	0.986	17.02
AS [2]	32.44	**0.988**	**14.29**	32.17	**0.987**	**14.77**
HumanNeRF [3]	**32.87**	0.987	15.36	32.34	0.986	16.00
SANeRF [4]	31.75	0.987	17.14	31.41	0.986	18.04
Instant-Nvr [14]	32.45	0.987	15.79	32.39	0.987	15.73
Ours	32.72	0.986	14.89	**32.51**	0.986	15.14

D. Training

We optimize the values in the feature planes and the weights of tiny MLPs by minimizing the disparities between the predicted and observed images:

$$\begin{aligned} L_{rgb} &= ||\widehat{I}_p - I_p||_2, \\ L_{LPIPS} &= ||F_{vgg}(\widehat{I}_p) - F_{vgg}(I_p)||_2, \end{aligned} \tag{15}$$

where $\widehat{I}_p$ represents the predicted rendering image patch, and I_p represents the ground truth image patch. F_{vgg} represents a pretrained VGG network used for feature extraction when calculating the LPIPS loss [3], [14].

To enforce smoothness and maintain consistent values across adjacent locations within the feature planes, we apply regularization [15], [18] to all feature planes $\mathbf{M}$:

$$L_r = \sum_M \sum_{i,j} \sqrt{(\mathbf{M}_{i+1,j} - \mathbf{M}_{i,j})^2 + (\mathbf{M}_{i,j+1} - \mathbf{M}_{i,j})^2}. \tag{16}$$

where $\mathbf{M}_{i,j}$ donates the interpolated value of $\mathbf{M}$ at (i, j).

The overall loss function L combines the above losses with weights λ_1, λ_2, and λ_3:

$$L = \lambda_1 L_{rgb} + \lambda_2 L_{LPIPS} + \lambda_3 L_r. \tag{17}$$

IV. EXPERIMENTS

A. Experiment settings

1) Datasets:

In order to evaluate the reconstruction and rendering quality of our proposed method, we use the following datasets:

ZJU-MoCap dataset [1] comprises 23 synchronized cameras for each subject. We use the 4th camera for training and the remaining cameras for testing. Following [14], our experiments focus on 6 selected human subjects (377, 386, 387, 392, 393, 394), collecting training data for each subject by choosing 1 frame every 5 frames, resulting in 100 frames. For the novel pose synthesis task, we select one frame every five frames from the remaining data after the 500th frame.

MonoCap dataset contains 4 subjects collected from the DeepCap dataset [56] and the DynaCap dataset [57] by [2]. We request access to the raw data and adhere to the data processing guidelines provided by [14]. We use one camera for training and ten uniformly distributed cameras for testing. For each subject, the initial 500 frames of the processed data are used for the novel view synthesis task, while the remaining frames beyond the 500th frame are used for the novel pose synthesis task. The frame sampling strategies are same with those described in the preceding paragraph.

DNA-Rendering dataset [58] contains several video sequences of dynamic humans with 60 camera views. We intend to conduct multi-view reconstruction experiments on this dataset. Specifically, We conduct experiments on 4 sequences of this dataset, with 4 camera views for training and 10 camera views for testing. For each sequence, we only collect 100 frames for novel view synthesis task to demonstrate the performance of our model with multi-view input.

2) Implementation details:

We implement the overall approach using the pure PyTorch framework and conduct all experiments on an NVIDIA RTX 3090 graphics card. We train our model within approximately five minutes using the ADAM optimizer with an initial learning rate of 5e-5, which gradually decays throughout training. During implementation, we calculate the resolution of the canonical planes based on the canonical human mesh and determine the size of the observation-space planes by

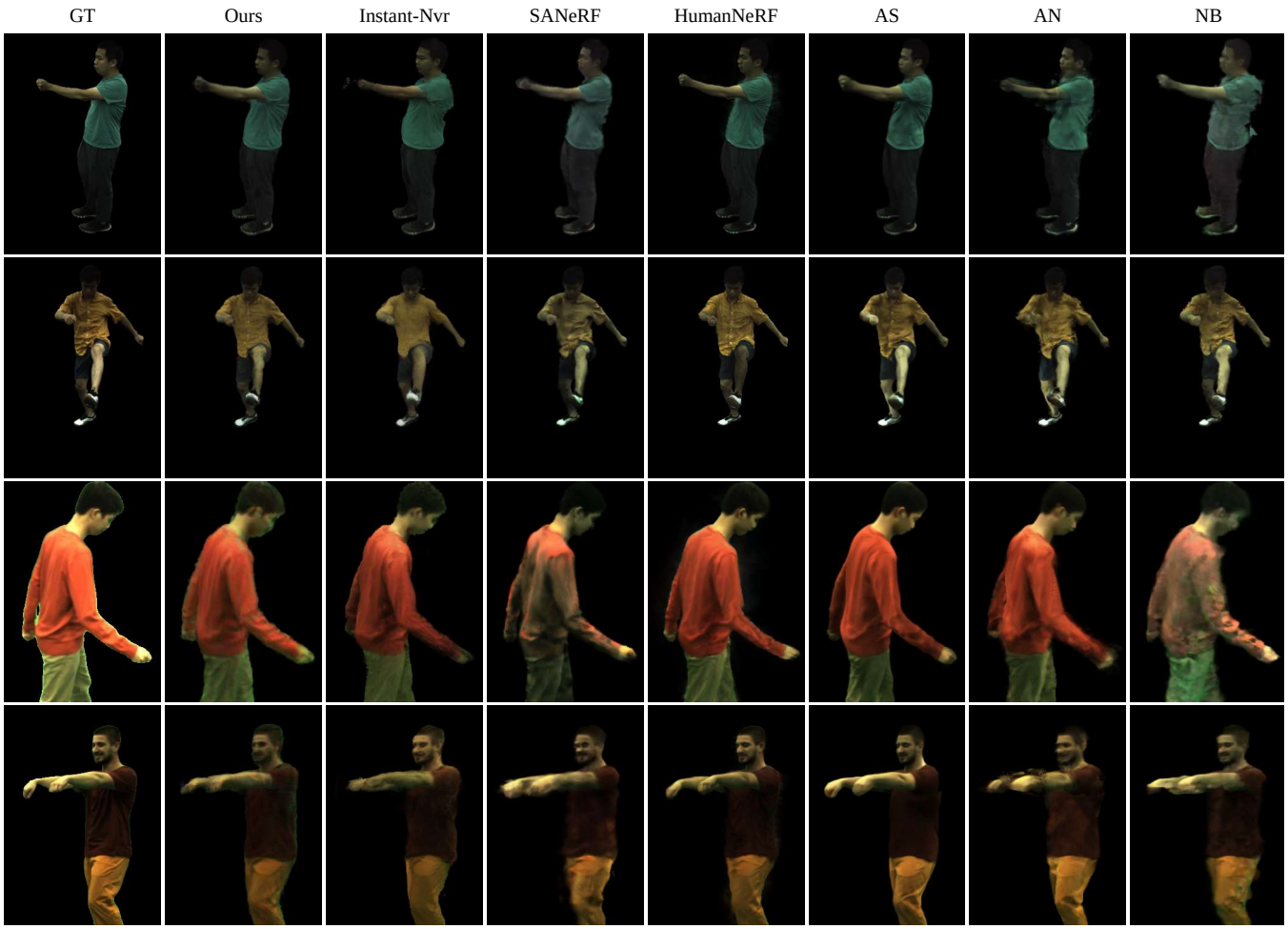


Fig. 4. **Qualitative comparison of novel view synthesis**. The first two rows displaying the results on the ZJU-MoCap dataset and the last two rows showcasing the results on the MonoCap dataset.

referencing the corresponding human mesh across all frames. To compute $\widehat{F}^T$, we configure N_p to 4, implying that we select 4 frames with distinct poses during training. When generating $F^C_{N_l}$, we set the number of iterations N_l to 3.

3) Metrics:

In the evaluation phase, we compute differences between the predicted images in their entirety and the corresponding ground-truth counterparts. We use three evaluation metrics: PSNR, SSIM [59], and LPIPS* (LPIPS*$= 10^3 \times$LPIPS). We also provide visual comparisons of the geometric reconstruction results.

B. Comparison

1) Baseline:

We systematically compare our approach with several baseline methods for both novel view synthesis and novel pose synthesis tasks. NB [1] represents canonical humans using a shared set of latent codes attached to the human model [24] surface. AN [35] uses neural blend weight fields with human skeletons for deformation fields, enabling the conversion between canonical and observation representations. Extending AN's work, AS [2] further improves deformation with a pose-dependent displacement field and use a signed distance field to better capture geometric detail. HumanNeRF [3] optimizes a canonical volumetric representation with a motion field. SANeRF [4] introduces a surface-aligned neural scene representation for controlling deformation. To improve optimization efficiency, Instant-Nvr [14] proposes a part-based human representation and a 2D motion parameterization scheme for effective modeling of canonical humans and deformation fields.

2) Results for Monocular Videos:

As shown in Tab. I, results on the ZJU-MoCap dataset indicate that our approach outperforms most baseline methods by a significant margin. Tab. II shows the comparison on the MonoCap dataset, results indicate that only a few metrics are slightly lower than those of HumanNeRF. This is attributed to HumanNeRF optimizes pose parameters and blend weights concurrently with the training phase. Optimizing these parameters makes their training less efficient, requiring dozens of hours to complete. While Instant-Nvr can achieve impressive predictions within minutes, it comes at the cost of having the largest number of model parameters. Additionally, it falls short of some methods [2], [3], [4] in terms of LPIPS, suggesting potential for further improvement in capturing fine-grained texture details.

Fig. 4 offers an additional qualitative comparison. Results

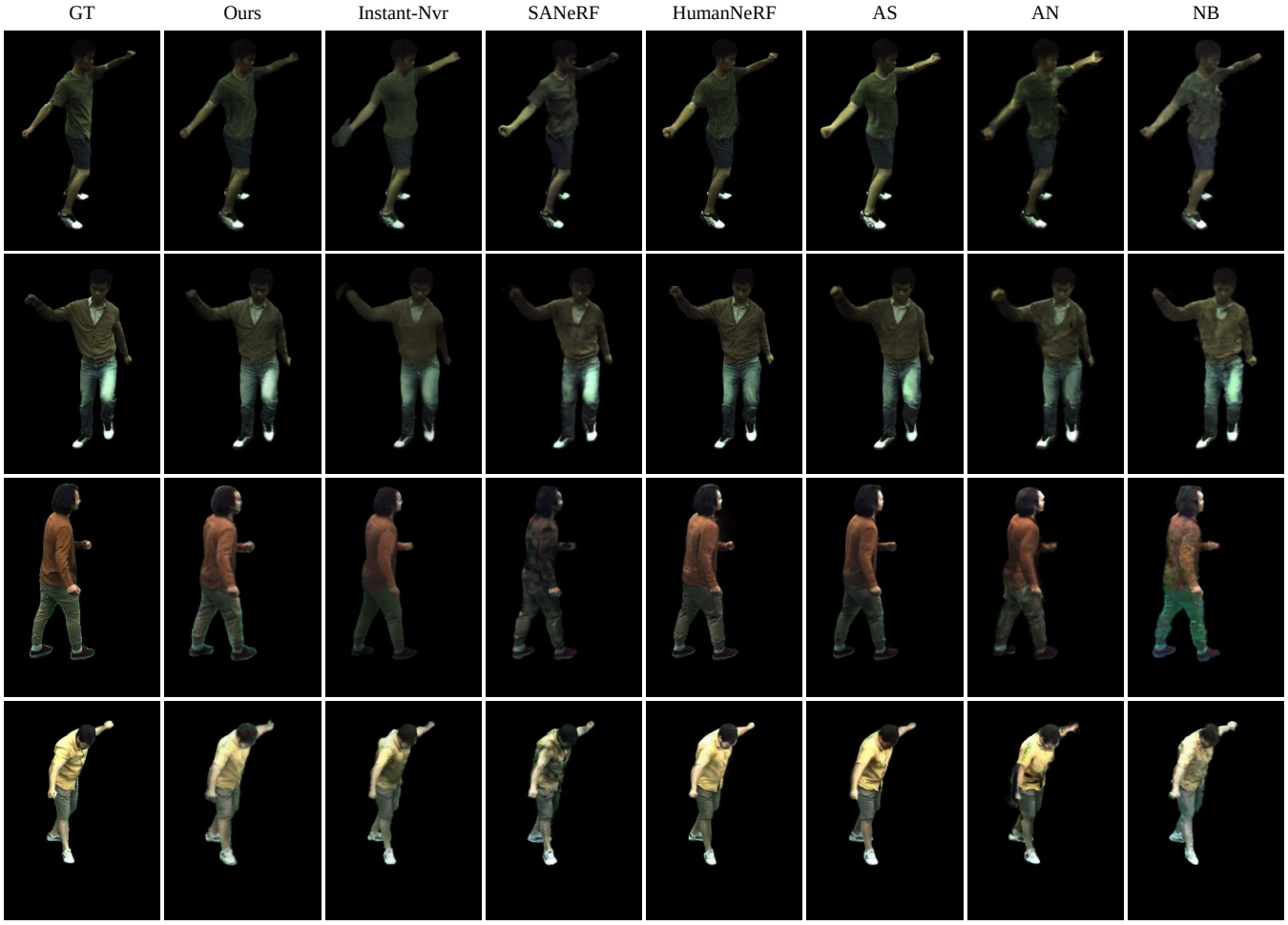


Fig. 5. **Qualitative comparison of novel pose synthesis**. The first two rows displaying the results on the ZJU-MoCap dataset and the last two rows showcasing the results on the MonoCap dataset.

indicate that our method generates renderings with more clothing details compared to Instant-Nvr. Compared to methods such as SANeRF, AN, and NB, our method not only exhibits superior training efficiency but also outperforms these methods in rendering quality. The rendering results of HumanNeRF and AS are comparable to ours, yet their training times are significantly longer than ours.

When conduct novel pose synthesis, we refrain from any fine-tuning of the model trained with input videos, and directly use it for the novel pose synthesis tasks. As illustrated in Fig. 5, despite the substantial reliance of our method on temporal information, we have attained exemplary outcomes in the novel pose synthesis task by efficiently integrating features from multiple reference frames during prediction.

3) Results for Multi-view Videos:

We perform multi-view experiments on the DNA-Rendering dataset [58]. We compare it with NB [1], AS [2], Tensor4D [15] and Instant-Nvr [14]. NB, AS and Tensor4D are originally designed for multi-view inputs, while Instant-Nvr and our model exhibit comparable training efficiency.

Tab. III presents the quantitative comparison. AS [2] struggles to capture complex geometric structures, especially subtle finger movements and hand details, leading to artifacts in their predictions. Although Instant-NVR [14] build an explicit deformation field, predicting non-rigid deformation remains a challenging task within this complex dataset. It is difficult for them to achieve ideal results, especially for hand and clothing folds. NB [1] outperforms AS and Instant-NVR, and its performance in terms of PSNR and SSIM metrics is comparable to ours, attributed to its utilization of ground-truth vertices as a geometric prior. However, NB faces challenges in achieving precise reconstruction from monocular video inputs, and its performance in capturing non-rigid details is inferior to ours. Due to the lack of human model priors, Tensor4D requires a longer optimization time and its capability for human reconstruction is inferior to our method. Furthermore, results

TABLE III
QUANTITATIVE COMPARISON ON THE DNA-RENDERING DATASET WITH MULTI-VIEW INPUT.

	PSNR ↑	SSIM ↑	LPIPS* ↓
NB [1]	31.51	0.977	45.83
AS [2]	30.58	0.974	45.73
Tensor4D [15]	28.31	0.950	71.42
Instant-Nvr [14]	29.14	0.968	47.60
Ours	**31.68**	**0.978**	**27.45**

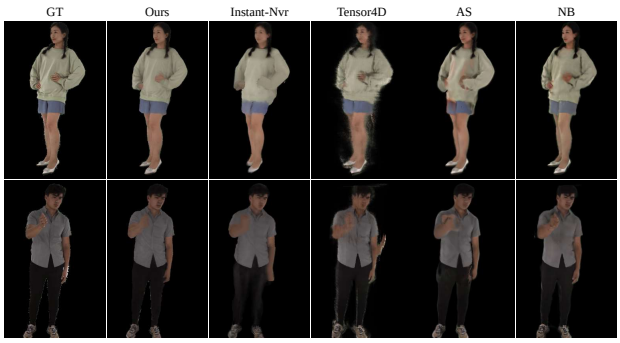


Fig. 6. **Qualitative comparison on the DNA-Rendering for multi-view inputs.** Our method can build more non-rigid details than Instant-Nvr within the same time budget, and demonstrates superior training efficiency compared to AS and NB.

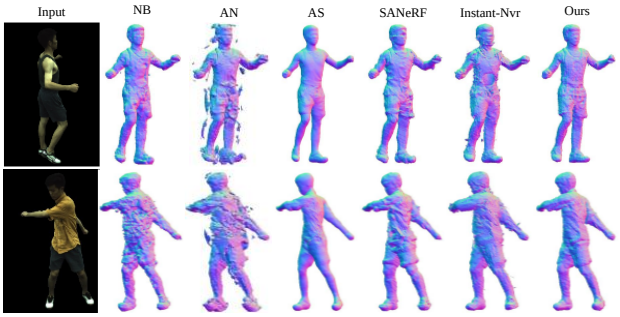


Fig. 7. **3D reconstruction on the ZJU-MoCap dataset.** Compared to methods explicitly predicting non-rigid offsets, our approach exhibits significantly fewer noisy points.

demonstrate that Tensor4D struggles to manage prolonged sequences involving complex motion. As shown in Fig. 6, our model can achieve more detailed reconstruction, result in more realistic renderings, so we outperform other methods in LPIPS by a large margin.

4) Results for geometric reconstruction:

To evaluate the quality of geometric reconstruction, we employ the Marching Cubes algorithm [60] to extract the intricate underlying human geometry from the neural field, thus enabling a detailed analysis of the reconstruction accuracy. In comparison to Instant-Nvr [14], which displays reconstruction results with noticeable holes, our model surpasses in assimilating relatively complete geometric information. Additionally, compared to [1], [35], [4], our reconstruction results exhibit fewer noisy points. AS [2] shows close to the best reconstruction results due to the use of signed distance field. However, our results encompass more clothing details, such as the tank top in the first group of sub-images.

C. Ablation studies

1) Ablation studies on Matrix Multiplication strategy:

To more effectively illustrate the advancements in our strategies for multi-space volume factorization and feature plane integration, we contrast the effects of employing the Vector-Matrix (VM) decomposition [12] with our Matrix Multiplication (MM) strategy. As shown in Tab IV, "Tri-planes" denote the spatial feature planes in canonical space, whereas

TABLE IV
ABLATION STUDIES ON PROPOSED COMPONENTS CONDUCTED ON THE 377 SEQUENCE.

	PSNR ↑	SSIM ↑	LPIPS* ↓
Tri-planes	31.66	0.978	25.27
Tri-planes+VM	31.69	0.978	23.36
Tri-planes+MM	31.74	0.979	22.93
Hex-planes+VM	27.19	0.950	64.81
Ours w/o MM	31.77	0.979	21.91
Ours w/o Other	31.64	0.978	23.00
Ours w/o Stage1	31.97	0.980	20.14
Ours w/o Stage2	32.07	0.980	19.85
Ours w/o Stage3	31.74	0.979	22.93
Ours w/o Iter	31.90	0.980	20.44
Ours w/ deeper MLP	31.93	0.979	20.64
Ours w/ non-rigid offsets	31.12	0.980	19.89
Ours w/o Light	32.05	0.980	20.59
Ours w/o LPIPS	31.76	0.978	28.98
Ours w/o TV	32.15	0.981	19.97
Ours(full model)	**32.25**	**0.981**	**19.84**

TABLE V
COMPARISON WITH TENSORF [12] ON FOUR STATIC OBJECTS, I.E., LEGO, HOTDOG, MATERIALS, MIC, FROM THE SYNTHETIC-NERF [7] DATASET.

	PSNR ↑	SSIM ↑	LPIPS* ↓
TensoRF [12]	34.65	0.976	30.84
Ours	**34.78**	**0.977**	**29.75**

"Hex-planes" denote the feature planes in observation space. Compare with "Tri-planes+VM", employing Matrix Multiplication emphasizes the association with features along each axis, thereby outperforming the VM decomposition across various metrics. "Hex-planes+VM" indicates that the optimization efficiency is compromised in the absence of canonical representation. "Ours w/o MM" signifies the exclusion of Matrix Multiplication when utilizing multi-space features. We further demonstrate the impact of implementing Matrix Multiplication in Fig. 8, highlighting more human geometry features, which demonstrates the capability of MM to achieve efficient optimization.

As shown in Tab. V, we additionally conduct experiments that apply Matrix Multiplication to TensorRF [12] on the static object datasets to further prove the effect of MM strategy.

2) Ablation studies on multi-space features:

In Tab. IV, "Ours w/o Other" means that we only use current-frame observations. The results are comparable to those obtained by employing only canonical features, indicating that capturing fine-grained details with only current-frame observations is insufficient. We also conduct ablation studies on our iterative optimization framework, which elucidate the importance of each stage. In "Ours w/o Stage1", we do not integrate time-dependent feature with spatial feature in observation space. This indicates that only the time-dependent feature is used as the attention weight for the canonical feature in stage 3, and the result shows the importance of feature integration in observation space. "Ours w/o Stage2" means that we do not integrate pose feature with observation-space spatial feature. In Stage3, the features obtained by Stage1

TABLE VI
ABLATION STUDIES ON DIFFERENT SIZES OF N_p AND N_l CONDUCTED ON THE 377 SEQUENCE.

	N_p						N_l			
	0	1	2	3	4	5	1	2	3	4
PSNR	31.64	32.13	32.17	32.16	**32.25**	32.20	31.90	32.10	**32.25**	32.12
SSIM	0.978	0.980	0.980	0.980	**0.981**	0.980	0.980	0.980	**0.981**	0.980
LPIPS*	23.00	20.07	20.04	20.08	**19.84**	20.08	20.44	19.98	**19.84**	20.19

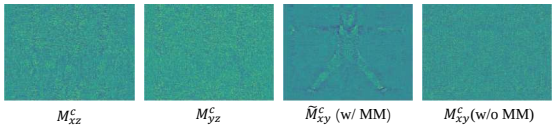


Fig. 8. The visualization of the effects of matrix multiplication, taking the XY plane as an example.

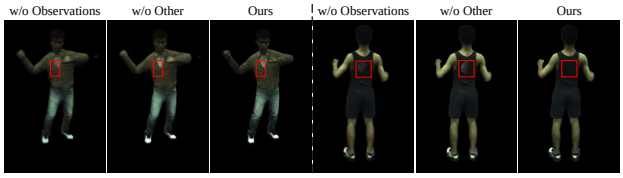


Fig. 9. Visualization of the effects of the observation features. When the observation feature is not used, the renderings are significantly rougher and contain many artifacts. And using observation features from other selected frames can further solve the problem of incomplete predictions.

and Stage2 are integrated with canonical feature. "Ours w/o Stage3" means that we only use canonical feature from Tri-planes for prediction, and the result compared with our full model proves the importance of the feature refinement process. Fig. 9 proves the effect of using observations. Result shows that when only using canonical features from Tri-planes, implicit human reconstruction and novel-view rendering can be already achieved. However, the renderings are coarse and lacking of non-rigid details.

3) Ablation studies on N_p and N_l:

Tab. VI shows the ablation studies on different size of N_p and N_l, in which $N_p = 0$ means that $\widehat{F}^o_t(\mathbf{x}^o, t) = F^o_t(\mathbf{x}^o, t)$. Results demonstrate that the current-frame predictions can be significantly enhanced by exploiting the features from other frames. It's worth noting that when N_p exceeds 6, further improvements become marginal. Therefore, to strike a balance between optimization efficiency and performance, we set the value of N_p to 4. When N_l set to 1, it means that we do not use iterative optimization. Results show that the most favorable outcome is achieved when the number of iterations reaches 3, with further increases resulting in diminishing returns, indicating that continued improvement becomes challenging.

We further illustrate the advancement of iterative optimization by comparing it with the utilization of a deeper MLP with skip connections. In Tab. IV, "Ours w/ deeper MLP" suggests that employing a deeper MLP can achieve a comparable effect to that of a shallow MLP without iterative optimization. This observation implies that while our proposed feature integration strategy is effective in enhancing performance, its combination with iterative optimization techniques can further intensify the integration between different features.

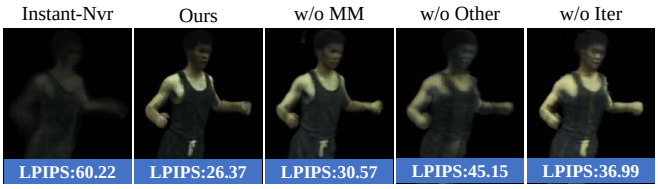


Fig. 10. **Ablation studies** of rendering results after 200 training iterations (approximately 30 seconds).

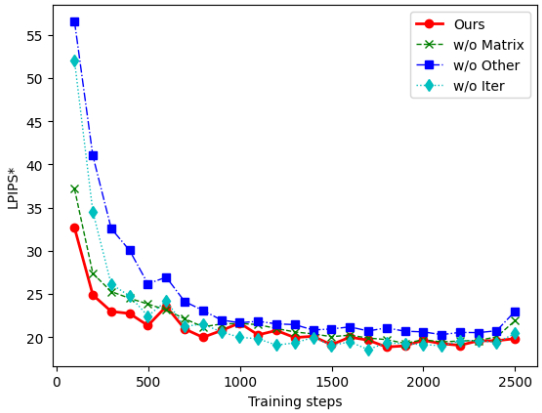


Fig. 11. Comparison of convergence LPIPS* and training steps needed of the proposed components.

4) Ablation studies on non-rigid offset:

As shown in Tab. IV, We conduct an additional set of experiments, incorporating the non-rigid prediction module used in AN to predict non-rigid deformation after the LBS transformation("Ours w/ non-rigid offsets"). Although theoretically using non-rigid deformation may achieve precise point alignment, the results indicate that incorporating extra non-rigid predictions does not significantly impact our prediction outcomes. This is due to inaccurately non-rigid deformation predictions will lead to degrade the performance. And the non-rigid prediction network incurs increased computational overhead and reduces training efficiency.

5) Ablation studies on training efficiency:

We provide a visual comparison of rendering efficiency in Fig. 10, demonstrating the effects of various settings under the same training iterations. It's worth emphasizing that the performance is notably less favorable when features from other frames are excluded, which indicates that more training time needed under this setting. While using feature planes in canonical space can quickly learn canonical human representation, the integration of different feature planes by

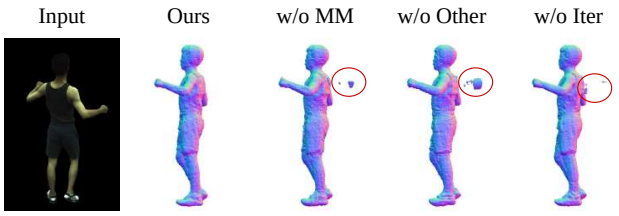


Fig. 12. **Ablation studies** of 3D reconstruction under various settings.

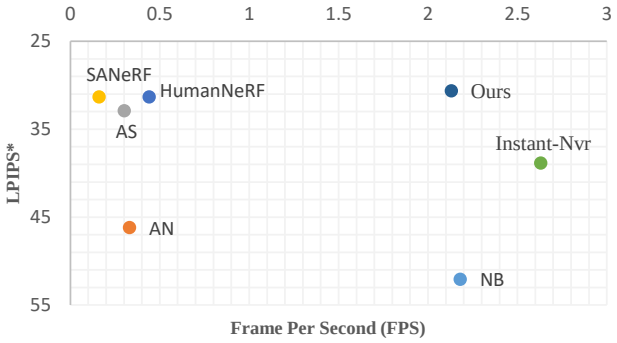


Fig. 13. **FPS Comparison between baseline methods.** Compared to baseline methods such as HumanNeRF and AS that demonstrate superior performance, our method not only boasts higher training efficiency but also faster rendering speed.

TABLE VII
QUANTITATIVE COMPARISON ON THE ZJU-MOCAP DATASET WITH 3D GAUSSIAN-BASED METHODS.

	PSNR ↑	SSIM ↑	LPIPS* ↓
GART [62]	30.89	**0.972**	32.43
HUGS [63]	30.56	0.970	30.89
Ours	**31.37**	**0.972**	**30.63**

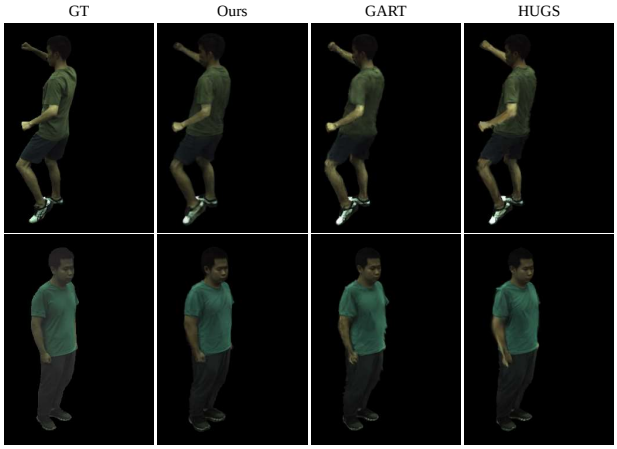


Fig. 14. **Comparison with 3DGS-based methods.** Our method yields renderings with fewer floaters compared to GART and HUGS.

Matrix Multiplication can further accelerates the learning of details. Additionally, using iterative optimization can further promote the refinement process, making the rendering result contains more observational details. We additionally present a convergence comparison, as shown in Fig. 11, to illustrate the effect of the proposed components on the training speed. Results show that our full model can quickly converge to the ideal outcomes.

6) Ablation studies on implicit 3D reconstruction:

We show the 3D reconstruction results under different settings in Fig. 12. Employing Matrix Multiplication and incorporating observations from other frames can significantly reduce noise points, and employing iterative optimization can further refine the results.

V. DISCUSSION

1) Limitation:

While our method effectively optimizes 3D human representations and achieves realistic renderings, it faces challenges in rendering speed. We provide an additional comparison of rendering speeds between our method and baselines in Fig. 13. Compared to HumanNeRF and AS that demonstrate superior performance, our method not only exhibits higher training efficiency but also faster rendering speed. However, with the rapid growth of 3D Gaussian Splatting(3DGS) [61], there are some methods that combine 3DGS with human representations and achieve high rendering quality with high time efficiency. Tab. VII and Fig. 14 provide additional comparison between GART [62] and HUGS [63] on the ZJU-Mocap dataset. Results show that although the 3D Gaussian-based approaches enable fast rendering, the multi-view inconsistent nature of 3D Gaussians leads to floaters during rendering. In contrast, our method achieves relatively accurate reconstruction within a short period of time. We believe that the strategies proposed in our paper holds the potential to inspire advancements in 3D Gaussian-based human reconstruction methods.

2) Conclusion:

We present a novel approach for efficiently modeling dynamic humans and achieving realistic renderings by integrating neural human representations. Our method proposes to decompose the multi-space, high-dimensional feature volume of a dynamic human subject into several feature planes. We further propose to use matrix multiplication to integrate these feature planes, explicitly incorporating correlations among different planes and fully utilizing associated details. Moreover, we incorporate multi-space representations through a collaborative process within the designed iterative optimization framework, allowing for progressive refinement of the canonical representation and facilitating the co-optimization of time-dependent features. Experiments demonstrate the effectiveness of our approach in learning more details within training time constraints, and we believe that our work can provide inspiration for future research.

ACKNOWLEDGMENTS

This work was supported by the Guangdong Basic and Applied Basic Research Foundation(No.2022A1515011425, No.2023A1515110075).

REFERENCES

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

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 9054–9063.

[2] S. Peng, Z. Xu, J. Dong, Q. Wang, S. Zhang, Q. Shuai, H. Bao, and X. Zhou, "Animatable implicit neural representations for creating realistic avatars from videos," *arXiv preprint arXiv:2203.08133*, 2022.

[3] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. 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*, 2022, pp. 16 210–16 220.

[4] T. Xu, Y. Fujita, and E. Matsumoto, "Surface-aligned neural radiance fields for controllable 3d human synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 883–15 892.

[5] Z. Yu, W. Cheng, X. Liu, W. Wu, and K.-Y. Lin, "Monohuman: Animatable human neural field from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 943–16 953.

[6] V. Jayasundara, A. Agrawal, N. Heron, A. Shrivastava, and L. 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*, 2023, pp. 21 118–21 127.

[7] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*, vol. 65, no. 1. Springer, 2020, pp. 405–421.

[8] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics*, 2022.

[9] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5501–5510.

[10] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5459–5469.

[11] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis *et al.*, "Efficient geometry-aware 3d generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 123–16 133.

[12] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision*. Springer, 2022, pp. 333–350.

[13] X. Gao, C. Zhong, J. Xiang, Y. Hong, Y. Guo, and J. Zhang, "Reconstructing personalized semantic facial nerf models from monocular video," *ACM Transactions on Graphics*, 2022.

[14] C. Geng, S. Peng, Z. Xu, H. Bao, and X. Zhou, "Learning neural volumetric representations of dynamic humans in minutes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8759–8770.

[15] R. Shao, Z. Zheng, H. Tu, B. Liu, H. Zhang, and Y. Liu, "Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 632–16 642.

[16] B. Peng, J. Hu, J. Zhou, X. Gao, and J. Zhang, "Intrinsicngp: Intrinsic coordinate based hash encoding for human nerf," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–13, 2023.

[17] T. Jiang, X. Chen, J. Song, and O. Hilliges, "Instantavatar: Learning avatars from monocular video in 60 seconds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2023, pp. 16 922–16 932.

[18] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, "K-planes: Explicit radiance fields in space, time, and appearance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 479–12 488.

[19] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan, "High-quality streamable free-viewpoint video," *ACM Transactions on Graphics*, 2015.

[20] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor *et al.*, "Fusion4d: Real-time performance capture of challenging scenes," *ACM Transactions on Graphics*, 2016.

[21] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis, "Learning to estimate 3d human pose and shape from a single color image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 459–468.

[22] M. Kocabas, N. Athanasiou, and M. J. Black, "Vibe: Video inference for human body pose and shape estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5253–5263.

[23] Y. Rong, T. Shiratori, and H. Joo, "Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1749–1759.

[24] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: a skinned multi-person linear model," *ACM Transactions on Graphics*, 2015.

[25] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: shape completion and animation of people," in *SIGGRAPH*, 2005.

[26] H. Zhang, Y. Tian, Y. Zhang, M. Li, L. An, Z. Sun, and Y. Liu, "Pymaf-x: Towards well-aligned full-body model regression from monocular images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[27] K. Shen, C. Guo, M. Kaufmann, J. J. Zarate, J. Valentin, J. Song, and O. Hilliges, "X-avatar: Expressive human avatars," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 911–16 921.

[28] G. Moon, H. Choi, and K. M. Lee, "Neuralannot: Neural annotator for 3d human mesh training sets," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, June 2022, pp. 2299–2307.

[29] E. Corona, A. Pumarola, G. Alenya, G. Pons-Moll, and F. Moreno-Noguer, "Smplicit: Topology-aware generative model for clothed people," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 875–11 885.

[30] S. Saito, J. Yang, Q. Ma, and M. J. Black, "Scanimate: Weakly supervised learning of skinned clothed avatar networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2886–2897.

[31] Y. Feng, J. Yang, M. Pollefeys, M. J. Black, and T. Bolkart, "Capturing and animation of body and clothing from monocular video," in *SIGGRAPH Asia*, 2022.

[32] A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "Avatarme: Realistically renderable 3d facial reconstruction" in-the-wild"," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 760–769.

[33] Z. Zheng, H. Huang, T. Yu, H. Zhang, Y. Guo, and Y. Liu, "Structured local radiance fields for human avatar modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 893–15 903.

[34] L. Liu, M. Habermann, V. Rudnev, K. Sarkar, J. Gu, and C. Theobalt, "Neural actor: Neural free-view synthesis of human actors with pose control," *ACM Transactions on Graphics*, 2021.

[35] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 314–14 323.

[36] W. Jiang, K. M. Yi, G. Samei, O. Tuzel, and A. Ranjan, "Neuman: Neural human radiance field from a single video," in *European Conference on Computer Vision*, vol. 13692. Springer, 2022, pp. 402–418.

[37] R. Zhang, J. Chen, and Q. Wang, "Explicifying neural implicit fields for efficient dynamic human avatar modeling via a neural explicit surface," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 1955–1963.

[38] R. Zhang and J. Chen, "Ndf: Neural deformable fields for dynamic human modelling," in *European Conference on Computer Vision*. Springer, 2022, pp. 37–52.

[39] B. Jiang, Y. Hong, H. Bao, and J. Zhang, "Selfrecon: Self reconstruction your digital avatar from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2022, pp. 5605–5615.

[40] W. Xian, J.-B. Huang, J. Kopf, and C. Kim, "Space-time neural irradiance fields for free-viewpoint video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9421–9431.

[41] Y. Du, Y. Zhang, H.-X. Yu, J. B. Tenenbaum, and J. Wu, "Neural radiance flow for 4d view synthesis and video processing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 304–14 314.

[42] X. Guo, J. Sun, Y. Dai, G. Chen, X. Ye, X. Tan, E. Ding, Y. Zhang, and J. Wang, "Forward flow for novel view synthesis of dynamic scenes," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, October 2023, pp. 16 022–16 033.
[43] A. Noguchi, X. Sun, S. Lin, and T. Harada, "Neural articulated radiance field," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5762–5772.
[44] X. Chen, Y. Zheng, M. J. Black, O. Hilliges, and A. Geiger, "Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 594–11 604.
[45] R. Li, J. Tanke, M. Vo, M. Zollhöfer, J. Gall, A. Kanazawa, and C. Lassner, "Tava: Template-free animatable volumetric actors," in *European Conference on Computer Vision*. Springer, 2022, pp. 419–436.
[46] J. Fang, T. Yi, X. Wang, L. Xie, X. Zhang, W. Liu, M. Nießner, and Q. Tian, "Fast dynamic radiance fields with time-aware neural voxels," in *SIGGRAPH Asia*, 2022.
[47] A. Cao and J. Johnson, "Hexplane: A fast representation for dynamic scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 130–141.
[48] M. Işık, M. Rünz, M. Georgopoulos, T. Khakhulin, J. Starck, L. Agapito, and M. Nießner, "Humanrf: High-fidelity neural radiance fields for humans in motion," *ACM Transactions on Graphics*.
[49] T. Wang, B. Zhang, T. Zhang, S. Gu, J. Bao, T. Baltrusaitis, J. Shen, D. Chen, F. Wen, Q. Chen *et al.*, "Rodin: A generative model for sculpting 3d digital avatars using diffusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4563–4573.
[50] Y. Kwon, D. Kim, D. Ceylan, and H. Fuchs, "Neural human performer: Learning generalizable radiance fields for human performance rendering," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 741–24 752, 2021.
[51] C.-Y. Weng, P. P. Srinivasan, B. Curless, and I. Kemelmacher-Shlizerman, "Personnerf: Personalized reconstruction from photo collections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 524–533.
[52] X. Gao, J. Yang, J. Kim, S. Peng, Z. Liu, and X. Tong, "Mps-nerf: Generalizable 3d human rendering from multiview images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
[53] S. Hu, F. Hong, L. Pan, H. Mei, L. Yang, and Z. Liu, "Sherf: Generalizable human nerf from a single image," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9352–9364.
[54] J. P. Lewis, M. Cordner, and N. Fong, "Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation," in *SIGGRAPH*, 2000.
[55] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.
[56] M. Habermann, W. Xu, M. Zollhofer, G. Pons-Moll, and C. Theobalt, "Deepcap: Monocular human performance capture using weak supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5052–5063.
[57] M. Habermann, L. Liu, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt, "Real-time deep dynamic characters," *ACM Transactions on Graphics*, 2021.
[58] W. Cheng, R. Chen, S. Fan, W. Yin, K. Chen, Z. Cai, J. Wang, Y. Gao, Z. Yu, Z. Lin *et al.*, "Dna-rendering: A diverse neural actor repository for high-fidelity human-centric rendering," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 982–19 993.
[59] Z. Wang, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
[60] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *SIGGRAPH*, 1987.
[61] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–14, 2023.
[62] J. Lei, Y. Wang, G. Pavlakos, L. Liu, and K. Daniilidis, "Gart: Gaussian articulated template models," *arXiv preprint arXiv:2311.16099*, 2023.
[63] M. Kocabas, J.-H. R. Chang, J. Gabriel, O. Tuzel, and A. Ranjan, "Hugs: Human gaussian splats," *arXiv preprint arXiv:2311.17910*, 2023.

Wensheng Li is a Ph.D. student at the School of Computer Science and Engineering, Sun Yat-sen University. He received his master's degree in Software Engineering from School of Computer Science and Engineering, Sun Yat-sen University in 2021. His research interests include computer vision and computer graphics.

Lingzhe Zeng is currently working toward the master degree in Sun Yat-sen University. He received the BE degree in computer science and technology from Sun Yat-sen University, in 2023. His research interests include 3D reconstruction and inverse rendering.

Chengying Gao is an associate professor in the School of Computer Science and Engineering, Sun Yat-sen University. She received her Ph.D. degree in computer science from the School of Information Science and Technology, Sun Yat-sen University in 2003. Her research interests include computer graphics and image processing.

Ning Liu is currently a professor in the School of Computer Science and Engineering, Sun Yat-sen University. He received his Ph.D. degree in computer science from School of Information Science and Technology, Sun Yat-sen University in 2004. His current research interests include computer vision, indoor localization, and deep learning.