FlyHuman: On-the-fly Feature Aggregation for Real-time Free-viewpoint Dynamic Human Video from Sparse Views

Wenbin Lin, Ruchang Yao, Jiaqi Li, Junhai Yong, Feng Xu

Abstract

Generating free-viewpoint videos of dynamic humans is a challenging yet crucial problem with numerous applications in immersive telepresence, film production, and sports analytics. Existing methods either require dense camera setups or struggle with high computation costs and temporal incoherence. In this paper, we present FlyHuman, a novel approach that achieves real-time, temporal consistent free-viewpoint rendering of dynamic humans from as few as three input views, and even from a single view. We model the dynamic human body using deformable 3D Gaussian Splatting, enabling efficient rendering. As the videos stream in, we track the motion and update the color of each 3D Gaussian frame-by-frame. To ensure temporally consistent geometry, we propose a graph-based deformation method that deforms the 3D Gaussians on-the-fly. For accurate appearance modeling with insufficient observations from sparse views, we introduce a spatial-temporal feature aggregation (STFA) module that gradually refines the features of the Gaussians. Extensive experiments demonstrate that our approach outperforms previous work in terms of rendering quality, temporal consistency, and efficiency. Code and models will be publicly available.

1. Introduction

Generating free-viewpoint videos of dynamic humans has a wide range of applications, including immersive telepresence, film production, and sports analytics. These applications necessitate both real-time efficiency and photorealistic rendering to ensure a seamless and engaging experience. Traditional methods [8, 34, 52] achieved this goal by employing dense camera settings to record the subjects, which, however, limits their widespread adoption among end-users due to the high cost. Given only sparse-view videos, as the recorded information is quite insufficient, rendering high-fidelity images of dynamic humans in real-time remains a significant challenge.

In recent years, Neural Radiance Fields (NeRF) [33] have achieved significant success in novel view synthesis. Numerous works [37, 38, 51] have integrated NeRF with parametric human models [29] to enable novel view synthesis of dynamic humans. However, these approaches are computationally expensive due to the time-consuming ray marching inherent in NeRF. More recently, 3D Gaussian Splatting (3DGS) [19] has demonstrated substantial advances in training and rendering efficiency, and has been utilized for rendering of dynamic humans [14, 16, 22, 26, 36, 40, 41, 50]. Nevertheless, these methods still require per-scene optimization, making them unsuitable for applications such as real-time telepresence. On the other hand, some methods [27, 31, 35, 61] have proposed using a generalizable feed-forward approach to achieve rendering of unseen human bodies. However, for continuous dynamic videos, these methods do not consider any temporal information, instead rendering the subject frame-by-frame.

In this paper, we introduce FlyHuman, a novel method for real-time rendering of dynamic humans from sparse viewpoints, enabling the generation of free-viewpoint videos from as few as three input views, and even monocular video.To achieve efficient rendering, we employ 3DGS as the representation of the dynamic human body. The first challenge is to maintain a dynamic 3D body geometry with consistency over time. To handle this problem, we construct the 3D Gaussians in the canonical space and then drive them to each observation space. The dynamic body motion is decomposed into rigid bone transformations and non-rigid deformation driven by a deformation graph. The deformation graph is optimized frame-by-frame to align the reconstructed geometry with input images while ensuring a temporally consistent geometry.

Given the tracked geometry, we then estimate the color of each Gaussian. Similar to existing image-based rendering methods [6, 47, 57], we compute the color of each Gaussian based on the pixel-aligned features. However, the insufficient observation from sparse viewpoints presents another challenge in estimating the appearance of occluded regions. We propose a spatial-temporal feature aggregation (STFA) module to enhance the input image features with spatial and temporal information. Specifically, we first project each 3D Gaussian onto 2D input image planes to obtain the image features of visible regions. Then, we use a graph convolutional network (GCN) to aggregate the image features with spatial information through the graph, obtaining completed features for all regions. Next, we further enhance the features with the accumulated historical features of each Gaussian. The feature on each Gaussian will be updated using a visibility-aware fusion method, and the updated feature will be used in future frames. Finally, with the aggregated feature, we employ a multi-layer perceptron (MLP) to predict the color of each Gaussian. With the estimated color, we can render photorealistic and temporally coherent free-viewpoint videos.

We conducted extensive experiments on the ZJU-MoCap[38, 39], Human3.6M[17] and THUman4.0[62] datasets. The proposed FlyHuman outperforms existing approaches in terms of rendering quality, temporal consistency, and efficiency. Moreover, our method achieves real-time speed of 25 frames per second on an NVIDIA RTX 4090 GPU, which is approximately 50 times faster than previous state-of-the-art methods.

In summary, this work makes the following contributions:

- We introduce a novel method for rendering free-viewpoint videos of dynamic human bodies in real-time from sparse views, achieving state-of-the-art performance in both quality and efficiency, as demonstrated by our experiments.
- We propose a spatial-temporal feature aggregation (STFA) method that effectively incorporates image features from different body regions and time steps, thereby ensuring consistent rendering of occluded regions.
- We combine graph-based deformation and linear blend skinning for real-time on-the-fly dynamic 3D Gaussian tracking, further enhancing the accuracy and geometry consistency of our approach.

2. Related Work

2.1. Novel View Synthesis

Novel view synthesis has been an active area of research for a long time, with various approaches proposed, such as light fields [1, 5, 12, 18, 23, 45], image-based rendering [2–4, 10, 44, 54, 63] and multi-plane images [32, 64]. In recent years, neural representations have been extensively utilized in 3D scene representation, demonstrating their effectiveness in modeling complex scenes and leading to a new trend for novel view synthesis. Neural Radiance Fields (NeRF) [33] has achieved significant success in photorealistic rendering of novel views. However, NeRF requires time-consuming per-scene optimization and cannot easily generalize to novel scenes. To address the generalization limitation, several approaches have been proposed. PixelNeRF [57] and IBRNet [47] condition NeRF with pixel-aligned image features from nearby views and learn scene priors from large datasets, enabling NeRF to generalize to novel scenes. MVSNeRF [6] and ENeRF [27] further involve cost volumes to investigate the correlation between multi-view features, enhancing the robustness of NeRF.

2.2. Neural Radiance Fields for Human Body

Although Neural Radiance Fields (NeRF) [33] enables photorealistic rendering of static scenes, it does not inherently support dynamic human bodies. To address this limitation, numerous efforts [25, 37, 38, 48, 51] have integrated NeRF with parametric human models, such as SMPL[29], to model the motion of dynamic humans. These methods achieve novel view synthesis of dynamic human using sparse videos, but they typically require per-subject optimization and cannot generalize to unseen subjects. To further improve the rendering quality, recent works [28, 42, 55] leverage captured images to provide more details, but these methods require dense input views, and pre-scaned models [42] or offline geometry fitting [28, 55], making them also not generalizable to unseen subjects.

For generalizable novel view synthesis of human bodies, various methods have been developed. NHP [21], HumanNeRF [60], MPS-NeRF [11] and GM-NeRF [7] project image features onto the SMPL model and diffuse the features for volume rendering. KeypointNeRF[31] leverages embedding based on 3D skeleton points to model the spatial information of human bodies. TransHuman [35] employs a transformer structure to model global relationships of features on the human body. SHERF [15] incorporates hierarchical features to achieve body reconstruction from a single image. However, these NeRF-based approaches typically suffer from inefficient rendering due to the computationally expensive ray marching process, which can lead to significant latency and hinder real-time applications. Moreover, existing methods for generalizable dynamic human rendering often neglect the temporal information inherent in input video sequences, processing each frame in isolation. This oversight can result in suboptimal rendering quality and a lack of temporal coherence. In contrast, our method achieves real-time rendering of unseen subjects while explicitly considering the temporal information present in the input video sequences, thereby enabling the generation of high-fidelity, temporally coherent dynamic human videos.

2.3. 3D Gaussian Splatting

Recently, 3D Gaussian Splatting (3DGS) [19] introduced a differentiable Gaussian ellipsoids splatting algorithm, which achieved notable advancements in accelerating rendering compared to ray marching-based methods like NeRF.

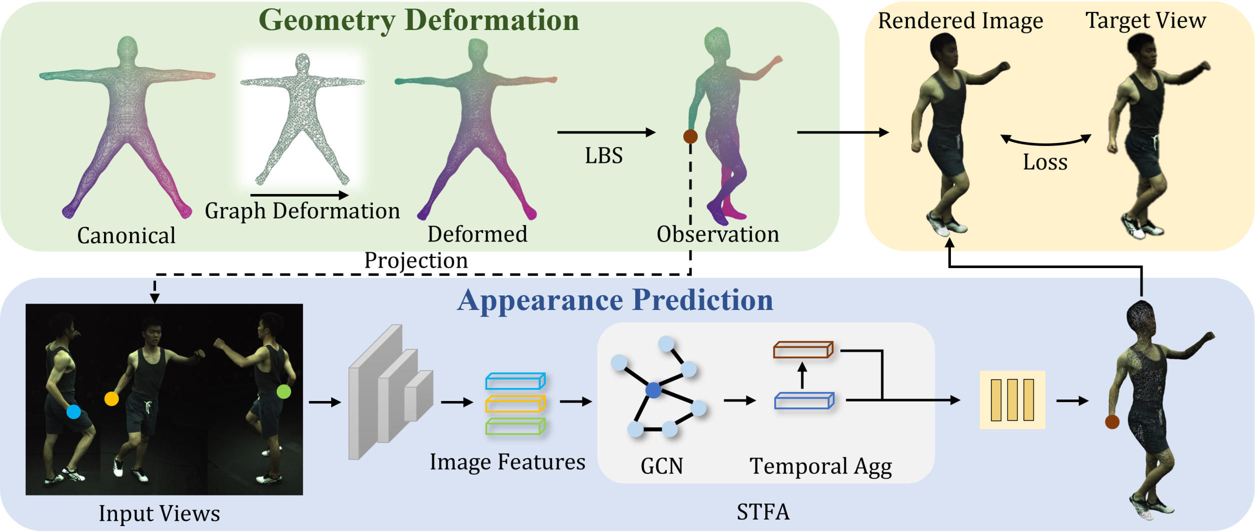

Figure 1. The pipeline of our method. Starting from the 3D Gaussians in the canonical space, we apply a graph-based deformation to model the non-rigid motions, and then deform the Gaussians into the observation space using linear blend skinning (LBS). The deformation is optimized on-the-fly to align with the input data (Sec. 3.1). For appearance modeling, we first project each Gaussian onto the input image planes and obtain the corresponding image features. Subsequently, we employ our Spatial-Temporal Feature Aggregation (STFA) method to enhance the image features. STFA is achieved through a Graph Convolutional Network (GCN) to incorporate spatial information, and a temporal aggregation method to incorporate historical features. Finally, with the aggregated feature, we use a multi-layer perceptron (MLP) to predict the color of each Gaussian, and render the human body from the target viewpoint (Sec. 3.2).

Numerous follow-up works have emerged, extending 3DGS to dynamic scenes[30, 46, 53, 56] and dynamic human bodies[14, 16, 22, 26, 36, 40, 41, 50], but per-subject offline training if still needed. Different from them, GPS-Gaussian[61] leverages pixel-wise 3DGS for generalizable rendering of human bodies based on stereo-matching between nearby views. However, the view interpolation of GPS-Gaussian requires relatively dense input views, limiting its applicability in scenarios with sparse viewpoints. Our method also builds upon 3DGS for efficient rendering and achieves real-time novel view synthesis with only sparse input views.

3. Method

Given sparse videos of a dynamic human, our goal is to generate a free-viewpoint video of the dynamic human on-the-fly as the video streams in real-time. In addition to the videos, our system also utilizes the foreground mask and the estimated body pose as input. The first challenge of this task is to reconstruct a temporally consistent geometry in real-time. To address this, we propose to combine graph-based deformation and linear blend skinning to model the dynamic body motion, and track the body deformation on-the-fly (Sec. 3.1). With the tracked geometry, we then predict the appearance of each 3D Gaussian from the input images. However, the observation from sparse inputs is insufficient to predict a complete and high-quality appearance. To overcome this limitation, we propose a Spatial-Temporal Feature Aggregation (STFA) method (Sec. 3.2) to enhance the input image features. The pipeline of our method is illustrated in Fig. 1.

3.1. On-the-fly Body Deformation Optimization

Body Representation. We employ 3D Gaussians Splatting [19] as the representation for human bodies. In this representation, bodies are explicitly constructed with point primitives, where each 3D Gaussian is parameterized by a 3D mean $\mu \in \mathbb{R}^3$, a 3D rotation $\mathrm{R} \in SO(3)$, a scaling vector $\mathrm{s} \in \mathbb{R}^3_+$ and an opacity factor $\eta \in (0, 1]$. We then project each 3D Gaussian onto the image plane using the elliptical Gaussian projection method proposed in EWA volume splatting [65], and perform point-based alpha-blend for rendering as:

$$C = \sum_{i \in N} c_i \alpha_i \Pi_{j=1}^{i-1} (1 - \alpha_j) \tag{1}$$

where c_i and α_i is the color and density of each point. Besides, we employ filters proposed in Mip-Splatting [58] to avoid alias.

In our approach, we first construct a body mesh in the canonical space and associate a 3D Gaussian with each vertex of the mesh. The mesh comprises approximately 115k points, which are evenly distributed across the body surface. To simplify the problem, we fix the 3D rotation R to an identity matrix, the opacity factor η to 1. The scaling vector s of each Gaussian is set based on its distance d between nearest neighbor as $s = [0.75d, 0.75d, 0.75d]$, where the average distance between neighboring Gaussian is about 4mm. Our method then estimates the position μ and the color c of each Gaussian.

Deformation Representation. For dynamic body deformation, we decompose it into articulated rigid motion driven by bones and non-rigid deformation driven by a deformation graph. Given the body pose, the rigid motion is computed using the linear blend skinning (LBS) algorithm [24] with bone transformations of the SMPL [29] model, which can model body motion at a coarse level. However, rigid LBS alone is insufficient to capture the intricate details of human motion. Therefore, we further apply non-rigid motion in the canonical space. To avoid potential geometric artifacts caused by directly optimizing the position of each 3D Gaussian, we introduce a deformation graph to interpolate the motion to the Gaussians, as illustrated in Fig. 1. The deformation graph is parameterized as $\mathcal{G} = \{\mathrm{p}_i, \mathrm{t}_i\}$, where p_i is the position of the ith graph node, and t_i is the translation of the node. The positions of the graph nodes are acquired by sub-sampling points from the canonical body mesh, and each node is connected to its neighboring nodes by edges. The translation $\mathrm{t_x}$ of a point x in the canonical space can be computed by convex combinations of the translations t of its neighboring graph nodes:

$$\mathrm{t_x} = \frac{\sum_{j=1}^{J} w_j \mathrm{t}_j}{\sum_{j=1}^{J} w_j}, w_j = \exp\left(-\frac{\|\mathrm{x} - \mathrm{p}_j\|_2^2}{\sigma^2}\right) \tag{2}$$

where J is the number of neighboring nodes of point x, the combination weights w_j are computed based on the distance between x and the node position p_j, and σ is a constant set to 0.001. Overall, for a point x in the canonical space, its transformed position in the observation space is given by $\mathrm{LBS}(\mathrm{x} + \mathrm{t_x})$, where $\mathrm{LBS}(\cdot)$ denotes the linear blend skinning operation.

Deformation Optimization. Given the multi-view images, foreground masks, and the body pose at time t, our goal at this stage is to optimize the translation of the graph nodes to align the deformed model with the input images and masks. This is achieved by optimizing the following loss function:

$$L(\mathcal{G}) = \lambda_{\text{color}} L_{\text{color}} + \lambda_{\text{mask}} L_{\text{mask}} + \lambda_{\text{reg}} L_{\text{reg}} \tag{3}$$

where L_{color} and L_{mask} ensure that the rendered images and masks align with the inputs, the L_{reg} is the regularization term.

Specifically, since the optimization is performed frame-by-frame, we use the optimized graph deformation and the estimated Gaussian color from the previous time step $t-1$ to deform and render the 3D Gaussians. With the rendered colors and masks, we employ an L2 loss to construct the L_{color} and L_{mask} terms. To stabilize the optimization process, we employ regularization based on the spatial and temporal smoothness of the body deformation:

$$\begin{aligned} L_{\text{reg}} =& \lambda_{\text{spat}} \sum_{i=1}^{G} \sum_{j=1}^{J} ||\mathrm{t}_i - \mathrm{t}_j||_2^2 + \\ & \lambda_{\text{temp}} \sum_{i=1}^{G} \sum_{j=1}^{J} \left|\left|(\mathrm{t}_i - \mathrm{t}_j) - \left(\mathrm{t}_i^{\text{temp}} - \mathrm{t}_j^{\text{temp}}\right)\right|\right|_2^2 \end{aligned} \tag{4}$$

where G and J are the number of graph nodes and the number of neighbors for each node, and $\mathrm{t}_i^{\text{temp}}$ is a temporally exponentially smoothed translation of the ith node. The first part of the regularization term enforces spatial smoothness between neighboring nodes, while the second part prevents the relative translations of neighboring nodes from drastic changes over time. With the defined loss function $L(\mathcal{G})$, we iteratively optimize the deformation using a gradient descent algorithm, allowing for efficient and accurate tracking of the dynamic human body geometry.

3.2. Appearance Prediction with Spatial-Temporal Feature Aggregation

Overview. With the optimized geometry, we then predict the color of each Gaussian for free-viewpoint rendering. Similar to existing image-based rendering methods [35, 47, 57], for a point x in the observation space, we first project it onto the input image planes and obtain the pixel-aligned multi-view image features encoded by a pretrained CNN. These image features are then leveraged to predict the color of the point. However, since the input views are limited, the point x may not be visible in the input images, leading to incomplete or inaccurate color information. To address this challenge, we propose a spatial-temporal feature aggregation (STFA) method to refine the features, enhancing the color prediction for occluded or poorly observed regions. Finally, we train a multi-layer perceptron (MLP) to predict the color of each Gaussian from the refined features obtained through the STFA module.

Multi-view Feature Aggregation. We aggregate all the visible image features from multiple views by:

$$\mathrm{f}_{\text{vis}} = \sum_{i=1}^{V} w_i \mathrm{f}_{\text{img}}^i(\mathrm{x}) \tag{5}$$

where $\mathrm{f}_{\text{img}}^i(\mathrm{x})$ is the image feature of point x in the ith view, V is the number of input views, and w_i is the weight of the ith view computed based on its visibility. The visibility weight w_i is computed based on the viewing angle and an occlusion check:

$$w_i = w_{\text{angle}} \cdot w_{\text{dist}} \tag{6}$$

$$w_{\text{angle}} = \begin{cases} 1 - |\sin\langle -\mathrm{n}_\mathrm{x}, \mathrm{v}_i(\mathrm{x})\rangle|, & \text{if } \mathrm{n}_\mathrm{x} \cdot \mathrm{v}_i(\mathrm{x}) < 0 \\ 0, & \text{if } \mathrm{n}_\mathrm{x} \cdot \mathrm{v}_i(\mathrm{x}) \geq 0 \end{cases} \tag{7}$$

$$w_{\text{dist}} = \max(1 - k \cdot d_i(\mathrm{x}), 0) \tag{8}$$

where n_x is the normal of point x in the observation space, $\mathrm{v}_i(\mathrm{x})$ is the viewing direction from the ith camera to point x, and $d_i(\mathrm{x})$ is the distance between the projected pixel point and point x. Instead of using the projected depth for distance computation, we render the canonical position map (similar to the observation point cloud in the green part of Fig. 1) and compute the distance in the canonical space to avoid incorrect projections between different body parts. The constant k is set to 20, so if the distance is larger than $0.05m$, the point x is considered occluded. Besides, if $\sum_{i=1}^{V} w_i > 1$, we normalize the weights as $w_i = w_i / \left(\sum_{i=1}^{V} w_i\right)$. We denote the total weight of input views as $w_{\text{vis}} = \sum_{i=1}^{V} w_i$.

GCN-based Spatial Aggregation. However, due to the limited observation, there could be regions invisible to all input views. To handle this problem, we propose to use spatial information to inpaint the occluded regions. We leverage the graph structure used in non-rigid deformation to propagate information on the body surface. For each graph node, the feature is computed using the feature from neighboring Gaussians. In addition to the image features, we also use the positional encoding of the coordinates of Gaussians in the canonical space and the visibility weight w_{vis} as input features. Then, we employ a graph convolutional network (GCN) to propagate spatial information among neighboring graph nodes through the edges. The basic graph convolution block in our GCN is the graph transformer proposed by [43]. Next, we upsample the features from graph nodes back to Gaussians and concatenate the GCN features with the original image features. We then use a linear layer to map the concatenated feature to the same dimension as the input image features, obtaining the inpainted feature f_{gcn}. Finally, we combine the inpainted feature with the visible image feature f_{vis} to obtain the feature for the current frame f_{curr}:

$$\mathrm{f}_{\text{curr}} = (1 - w_{\text{vis}})\mathrm{f}_{\text{gcn}} + \mathrm{f}_{\text{vis}} \tag{9}$$

Visibility-aware Temporal Fusion. Even with the aggregated features, predicting high-quality appearance using only images from a single time step remains challenging. Therefore, we propose a visibility-aware temporal fusion method and leverage historical features to further refine the features. The historical feature is accumulated over time using a visibility-aware fusion method. Given the feature $\mathrm{f}_{\text{curr}}^t$ at time step t and the historical accumulated feature $\mathrm{f}_{\text{acc}}^{t-1}$, the accumulated feature is updated as:

$$\begin{aligned} \mathrm{f}_{\text{acc}}^t &= (w_{\text{acc}}^{t-1}\mathrm{f}_{\text{acc}}^{t-1} + w_{\text{curr}}^t \mathrm{f}_{\text{curr}}^t)/(w_{\text{acc}}^{t-1} + w_{\text{curr}}^t) \\ w_{\text{acc}}^t &= \min(w_{\text{acc}}^{t-1} + w_{\text{curr}}^t, w_{\max}) \end{aligned} \tag{10}$$

where w_{curr}^t is the weight for $\mathrm{f}_{\text{curr}}^t$, w_{acc}^t is the accumulated weight at time t, at the first frame $\mathrm{f}_{\text{acc}}^0 = 0$, and $w_{\max} = 8$ is the maximum accumulated weight. To maintain the quality of the accumulated feature, we only fuse well-observed features into it. Therefore, we use a visibility-aware weight $w_{\text{curr}} = w_{\text{vis}}$, which considers the viewing angle and occlusion.

Appearance Prediction. After the fusion of the accumulated feature, we combine the historical feature with the observation at the current time step to predict the color of each Gaussian. Firstly, when the historical information is insufficient, we rely more on the current observation. Additionally, for well-observed regions, we also place greater emphasis on the current observation, ensuring that the rendered appearance accurately reflects the most recent input. Here, we use the distance $d(\mathrm{x})$ to define the quality of current observation, which is similar to Eq. 8, and we use an exponential function to smoothly combine the features. The combination is computed as:

$$w = (w_{\text{acc}}^t / w_{\max})(1 - \exp(-k \cdot d(\mathrm{x}))) \tag{11}$$

$$\mathrm{f}_{\text{final}} = w\mathrm{f}_{\text{acc}}^t + (1 - w)\mathrm{f}_{\text{curr}}^t \tag{12}$$

With the feature $\mathrm{f}_{\text{final}}$, we employ an MLP to predict the color of each Gaussian for rendering.

Network Training. The appearance prediction relies on three neural networks: the CNN for image feature encoding, the GCN for spatial feature aggregation, and the MLP to predict the color. We train our models on the ZJU-MoCap dataset[38, 39], which contains multi-view videos of dynamic humans. During training, we randomly sample three views as input and a random target view. We supervise the rendered image to be close to the target image using MSE loss and LPIPS loss [59]:

$$L = L_{\text{MSE}} + \lambda L_{\text{LPIPS}} \tag{13}$$

where we set $\lambda = 0.01$. During the training process, we drop the temporal feature fusion block, and the color prediction MLP takes only f_{curr} as input. This simplification allows us to train the networks in a more efficient and stable manner. These three networks are trained end-to-end.

4. Experiments

In this section, we first provide implementation details of the proposed method (Sec. 4.1). Then, we introduce the datasets and evaluation metrics used (Sec. 4.2). Next, we compare our method with existing approaches, both quantitatively and qualitatively (Sec. 4.3). Finally, we perform ablation studies to validate our key designs (Sec. 4.4). For sequence results, please refer to our supplemental video.

Method	ZJU-MoCap (3 views)			ZJU-MoCap (single view)			Human3.6M (3 views)			THUman4.0 (3 views)			FPS
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	
KeypointNeRF	25.82	0.9100	0.1024	24.11	0.8857	0.1359	21.94	0.8590	0.1847	23.57	0.9128	0.1071	0.13
TransHuman	**26.85**	0.9143	0.1078	25.31	0.8938	0.1369	24.44	0.8830	0.1559	24.49	0.9152	0.1110	0.55
Ours	26.68	**0.9148**	**0.0934**	**25.42**	**0.9026**	**0.1125**	**25.02**	**0.9072**	**0.1289**	**25.24**	**0.9313**	**0.0889**	**25**

Table 1. Comparisons with KeypointNeRF[31] and TransHuman[35]. Note that we use the refined version of the ZJU-MoCap dataset in [39] with more accurate annotations, which makes the results slightly different from their original papers.

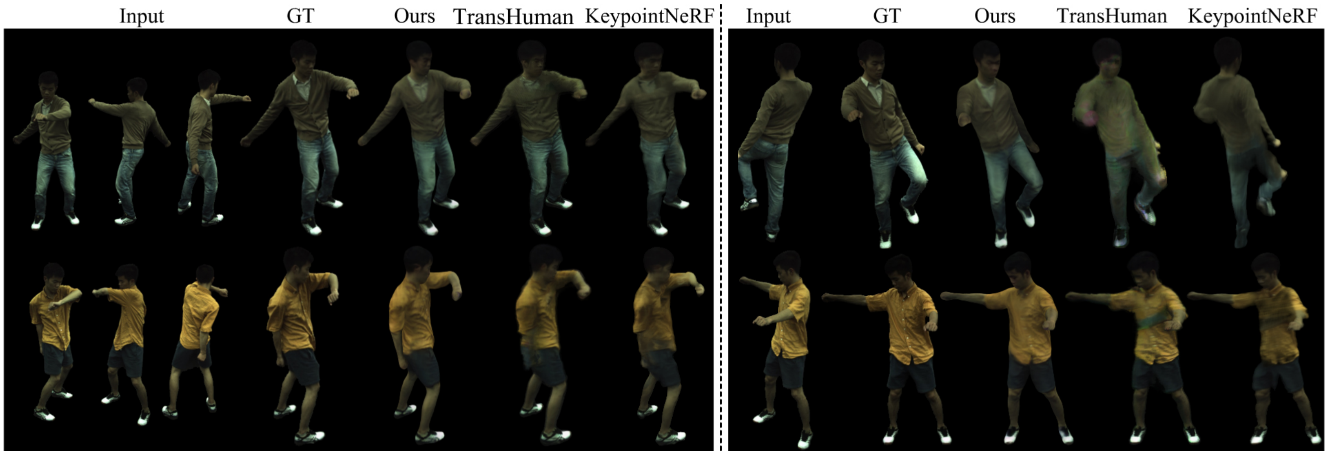

Figure 2. Comparisons with KeypointNeRF[31] and TransHuman[35] on the ZJU-MoCap dataset. For the results in the left part, three input views were used, while for the results in the right part, only a monocular video was used as input.

4.1. Implementation Details

In the graph-based deformation module, the graph contains 1620 nodes, and each node is connected to 4 neighboring nodes. The optimization weights are set as follows: $\lambda_{\text{color}} = \lambda_{\text{mask}} = 2$, $\lambda_{\text{reg}} = 200$, $\lambda_{\text{spat}} = 1$, and $\lambda_{\text{temp}} = 2$. The iterations of gradient descent is 10 for the first 10 frames and 2 for later frames. For the appearance prediction component, the networks are trained using the Adam [20] optimizer with a learning rate of 1e-4. The models are trained for 25,000 iterations. The CNN network is initialized with pretrained weights from ResNet-18[13], and we only use the first three layers for feature extraction. The inference speed of our method is 25 frames per second (FPS) for images of 512×512 resolution with 3 input views on an NVIDIA RTX 4090 GPU. For monocular input, the speed further increases to 53 FPS, since the time for feature extraction and projection is significantly reduced. More details about the network architectures, configurations, and inference speed can be found in the supplemental document.

4.2. Datasets and Metrics

We use the ZJU-MoCap dataset [38, 39] to train our model. The ZJU-MoCap dataset contains videos of 9 different subjects captured by 23 synchronized cameras. We use 6 subjects for training and the remaining 3 for evaluation. During testing, we use either 3 views or a monocular video as input and evaluate on 6 novel views. To further evaluate the cross-dataset generalization ability of our method, we use the Human3.6M[17] and THUman4.0[62] dataset. The Human3.6M dataset contains 7 subjects captured by 4 cameras, with 3 views as inputs and 1 for testing. The THUman4.0 dataset contains 3 subjects captured by 24 cameras, we use 3 views as inputs and another 3 views for testing. More details on the datasets can be found in Sec. 8 of the supplemental document.

To evaluate rendering quality, we employ PSNR, SSIM [49], and LPIPS [59] as metrics. These metrics are computed within the foreground regions determined by the bounding box of the humans in the scene.

4.3. Comparisons

We first compare our method with two generalizable sparse-view based human rendering methods, KeypointNeRF[31] and TransHuman[35]. These two methods process each frame in the videos independently, without considering temporal information like our approach. The comparison is conducted on four different settings: using three or single input views on the ZJU-MoCap dataset, and cross-dataset evaluation on the Human3.6M and THUman4.0 dataset with three input views. We present the numerical results in Tab 1. Overall, our method achieves better performance than KeypointNeRF and TransHuman across all settings.

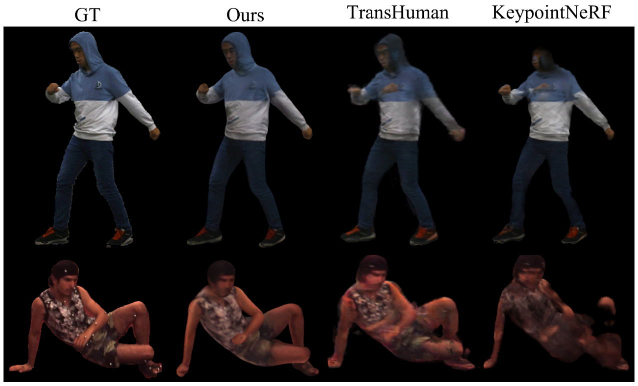

Figure 3. Comparisons with KeypointNeRF[31] and TransHuman[35] on the THUman4.0 (first row) and the Human3.6M dataset (second row).

Moreover, our method achieves real-time frame rates of 25FPS, while the other two methods take seconds to render a single image.

We further show qualitative comparisons on the ZJU-MoCap dataset in Fig. 2. On the ZJU-MoCap dataset with three input views, the results are mostly plausible for all three methods. However, for occluded regions, we can observe incorrect appearance projections in the results of TransHuman and KeypointNeRF. For example, the appearance of the arm and fist is projected onto the chest in the results in the top row of the left half in Fig. 2. Our approach does not suffer from these artifacts since we filter out invisible image features and leverage spatial-temporal information to predict the appearance of occluded regions. Additionally, we can find that the geometry of the arm and body are blended in the results of TransHuman and KeypointNeRF in the bottom row, while our method provides a clear geometry, as the geometry is well-tracked over time with the on-the-fly graph deformation optimization.

When the input is reduced to a single view, novel view synthesis becomes much more challenging, especially when the target view differs significantly from the input view (the top row of the right half of Fig. 2). However, with the accumulated temporal features, our method can still generate plausible results based on historical observations. When the target view is not far from the input view (results in the bottom row), TransHuman and KeypointNeRF still suffer from occlusions, while our approach achieves better results with the spatial-temporal feature aggregation module.

Furthermore, in the cross-dataset evaluation on the Human3.6M and THUman4.0 dataset shown in Fig. 3. In the results of TransHuman and KeypointNeRF, we can observe incorrect appearance caused by occlusion. Besides, floaters appear in the geometry of TransHuman, whereas body corruption appears in the geometry of KeypointNeRF. In contrast, our method achieves best results even for challenging body poses (the second row of Fig. 3), demonstrating its robust capability for generalization across poses and datasets.

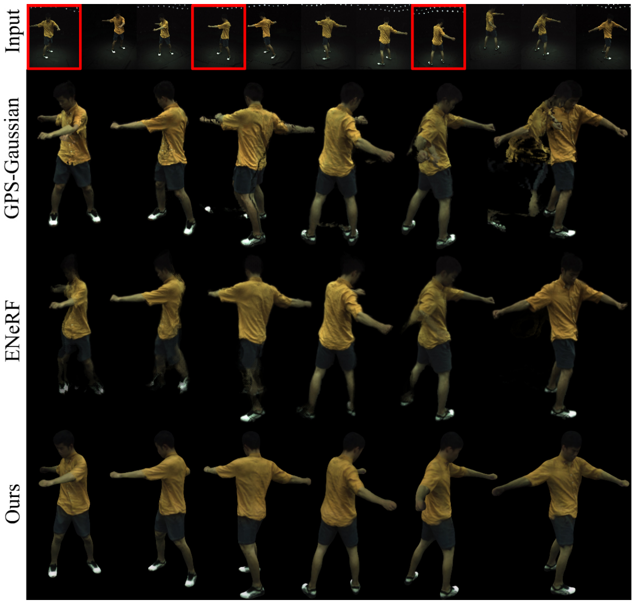

Figure 4. Comparisons with GPS-Gaussian[61] and ENeRF[27]. Note that GPS-Gaussian and ENeRF use 11 input views, while our method only uses 3 input views (in red rectangles).

Next, we compare our method with two real-time generalizable rendering methods, ENeRF[27] and GPS-Gaussian[61]. However, these two methods rely on relatively dense input views, so we use 11 input views for their methods and 3 views for ours for comparison. We render novel view images and show qualitative comparisons in Fig. 4. We can find that our method achieves consistent novel view synthesis results, while ENeRF and GPS-Gaussian exhibit jitters or broken geometry in their results. This is because ENeRF and GPS-Gaussian use nearby views for depth estimation, and when the discrepancy between nearby views becomes large due to the sparsity of input views, it becomes challenging to generate plausible results like ours. Additionally, when shifting between different nearby views, there could be jitters in their results. In contrast, the geometry of our method is view-independent and tracked over time with spatial and temporal smoothness, providing stable results.

For more comparisons, please refer to Sec. 9 of the supplemental document and the supplemental video.

4.4. Ablation Study

We evaluate two key components of our technique: the graph-based deformation optimization and the spatial-temporal feature aggregation module.

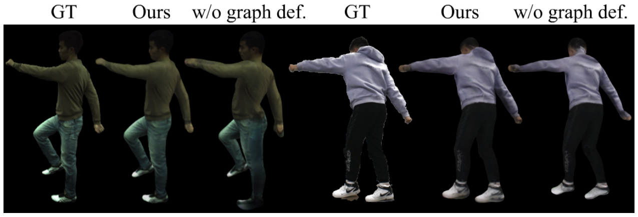

Figure 5. Ablation study on the graph-based deformation. We see that without the graph-based deformation, the body shapes do not match the ground truth.

Method	ZJU-MoCap (3 views)			ZJU-MoCap (single view)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
w/o ST	26.17	0.9095	0.0962	24.18	0.8883	0.1207
w/o T	26.58	0.9145	0.0935	25.18	0.9001	0.1165
Ours	**26.68**	**0.9148**	**0.0934**	**25.42**	**0.9026**	**0.1125**

Table 2. Ablation study on the spatial-temporal feature aggregation. ”w/o ST” means using only original image feature without spatial-temporal aggregation, ”w/o T” means using only spatial aggregation and no temporal information used.

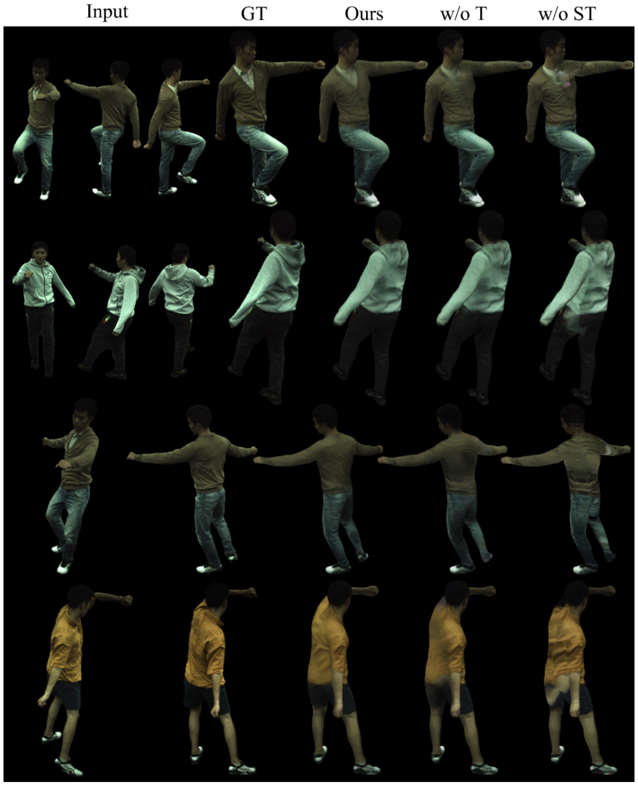

Figure 6. Ablation study on the spatial-temporal feature aggregation. ”w/o ST” means using only original image feature without spatial-temporal aggregation, ”w/o T” means using only spatial aggregation and no temporal information used.

Graph-based Deformation. To evaluate the importance of the on-the-fly geometry optimization, we remove the optimization of graph nodes and use only the initial geometry model and linear blend skinning (LBS) for geometry tracking. The comparative results are shown in Fig. 5. We can observe that the result without graph deformation cannot fit the body shape well and cannot model the shape of hat and shoes in the rightmost column, leading to incorrect appearance projections.

Spatial-Temporal Feature Aggregation. To evaluate the effectiveness of the spatial and temporal feature aggregation module, we build a baseline method that only uses the original image features to predict the appearance. The image features are computed using Eq. 5, and to avoid the features of occluded regions from being zeros, we add a small number to the weight of each view and normalize the weights. Based on this baseline, we incrementally add the spatial and temporal aggregation steps. The quantitative results on the ZJU-MoCap dataset are shown in Tab. 2. We can observe that both spatial and temporal aggregation bring improvements to the results. Moreover, for the single-view scenario, the improvement becomes more significant because the visible information is fewer, and our spatial-temporal feature aggregation plays a more crucial role. Qualitative results are shown in Fig. 6. We can see that with only the original image features, it is challenging to obtain plausible appearance for occluded regions. In the result of ”w/o ST” the appearance of the occluded regions is simply the incorrect projection of visible parts. With our spatial aggregation module involved, the results of ”w/o T” can inpaint the features of occluded regions with spatial priors, providing much better rendering results. However, using only observations from a single time step, it is still difficult to provide high-quality results, especially for monocular input. By further leveraging the accumulated historical information, our method achieves the best results, demonstrating the importance of both spatial and temporal feature aggregation.

5. Conclusion

In this work, we propose FlyHuman, a novel approach for on-the-fly real-time free-viewpoint video synthesis of dynamic humans from sparse viewpoints. Our method combines graph-based deformation with LBS to enable real-time tracking of dynamic body motions with temporal consistency. Furthermore, our method introduces a spatial-temporal feature aggregation module to enhance the limited sparse-view observations with spatial priors and historical information, achieving high-fidelity appearance estimation. In summary, this work paves a step forward in free-viewpoint video, making applications such as real-time telepresence, virtual events possible with low-cost devices.

References

[1] Daniel Wood Daniel Azuma Wyvern Aldinger, B Curless T Duchamp DH Salesin, and W Stuetzle. Surface light fields for 3d photography. In *Computer Graphics, SIGGRAPH 2000 Proc*, pages 287–296, 2000. 2

[2] Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. Unstructured lumigraph rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 425–432, 2001. 2

[3] Gaurav Chaurasia, Olga Sorkine, and George Drettakis. Silhouette-aware warping for image-based rendering. In *Computer Graphics Forum*, pages 1223–1232. Wiley Online Library, 2011.

[4] Gaurav Chaurasia, Sylvain Duchene, Olga Sorkine-Hornung, and George Drettakis. Depth synthesis and local warps for plausible image-based navigation. *ACM transactions on graphics (TOG)*, 32(3):1–12, 2013. 2

[5] Anpei Chen, Minye Wu, Yingliang Zhang, Nianyi Li, Jie Lu, Shenghua Gao, and Jingyi Yu. Deep surface light fields. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 1(1):1–17, 2018. 2

[6] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14124–14133, 2021. 1, 2

[7] Jianchuan Chen, Wentao Yi, Liqian Ma, Xu Jia, and Huchuan Lu. Gm-nerf: Learning generalizable model-based neural radiance fields from multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20648–20658, 2023. 2

[8] Shenchang Eric Chen and Lance Williams. View interpolation for image synthesis. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 279–288, 1993. 1

[9] Wei Cheng, Ruixiang Chen, Siming Fan, Wanqi Yin, Keyu Chen, Zhongang Cai, Jingbo Wang, Yang Gao, Zhengming Yu, Zhengyu 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*, pages 19982–19993, 2023. 2

[10] PE DEBEC. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *Proc. SIGGRAPH'96*, 1996. 2

[11] 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. 2

[12] SJ GORTLER, R GRZESZCZUK, R SZELISKI, and MF COHEN. The lumigraph. In *Computer graphics proceedings, annual conference series*, pages 43–54. Association for Computing Machinery SIGGRAPH, 1996. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6, 1

[14] Liangxiao Hu, Hongwen Zhang, Yuxiang Zhang, Boyao Zhou, Boning Liu, Shengping Zhang, and Liqiang Nie. Gaussianavatar: Towards realistic human avatar modeling from a single video via animatable 3d gaussians. *arXiv preprint arXiv:2312.02134*, 2023. 1, 3

[15] Shoukang Hu, Fangzhou Hong, Liang Pan, Haiyi Mei, Lei Yang, and Ziwei Liu. Sherf: Generalizable human nerf from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9352–9364, 2023. 2

[16] Shoukang Hu, Tao Hu, and Ziwei Liu. Gauhuman: Articulated gaussian splatting from monocular human videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20418–20431, 2024. 1, 3, 2

[17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014. 2, 6, 1

[18] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–10, 2016. 2

[19] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4):1–14, 2023. 1, 2, 3

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR 2015,*, 2015. 6

[21] 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

[22] Jiahui Lei, Yufu Wang, Georgios Pavlakos, Lingjie Liu, and Kostas Daniilidis. Gart: Gaussian articulated template models. *arXiv preprint arXiv:2311.16099*, 2023. 1, 3

[23] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996. 2

[24] John P. Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH 2000*, pages 165–172. ACM, 2000. 4

[25] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhöfer, Jürgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 419–436. Springer, 2022. 2

[26] Zhe Li, Zerong Zheng, Lizhen Wang, and Yebin Liu. Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19711–19722, 2024. 1, 3

[27] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance

fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 1, 2, 7

[28] Haotong Lin, Sida Peng, Zhen Xu, Tao Xie, Xingyi He, Hujun Bao, and Xiaowei Zhou. High-fidelity and real-time novel view synthesis for dynamic scenes. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–9, 2023. 2

[29] 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. 1, 2, 4

[30] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. *arXiv preprint arXiv:2308.09713*, 2023. 3

[31] Marko Mihajlovic, Aayush Bansal, Michael Zollhoefer, Siyu Tang, and Shunsuke Saito. Keypointnerf: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints. In *European conference on computer vision*, pages 179–197. Springer, 2022. 1, 2, 6, 7, 3

[32] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 2

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

[34] Byong Mok Oh, Max Chen, Julie Dorsey, and Frédo Durand. Image-based modeling and photo editing. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 433–442, 2001. 1

[35] Xiao Pan, Zongxin Yang, Jianxin Ma, Chang Zhou, and Yi Yang. Transhuman: A transformer-based human representation for generalizable neural human rendering. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 3544–3555, 2023. 1, 2, 4, 6, 7, 3

[36] Haokai Pang, Heming Zhu, Adam Kortylewski, Christian Theobalt, and Marc Habermann. Ash: Animatable gaussian splats for efficient and photoreal human rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1165–1175, 2024. 1, 3

[37] 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 *ICCV*, pages 14294–14303, 2021. 1, 2

[38] 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 *CVPR*, pages 9054–9063, 2021. 1, 2, 5, 6

[39] Sida Peng, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Representing volumetric videos as dynamic mlp maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4252–4262, 2023. 2, 5, 6, 1

[40] Zhiyin Qian, Shaofei Wang, Marko Mihajlovic, Andreas Geiger, and Siyu Tang. 3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting. *arXiv preprint arXiv:2312.09228*, 2023. 1, 3

[41] Zhijing Shao, Zhaolong Wang, Zhuang Li, Duotun Wang, Xiangru Lin, Yu Zhang, Mingming Fan, and Zeyu Wang. Splattingavatar: Realistic real-time human avatars with mesh-embedded gaussian splatting. *arXiv preprint arXiv:2403.05087*, 2024. 1, 3

[42] Ashwath Shetty, Marc Habermann, Guoxing Sun, Diogo Luvizon, Vladislav Golyanik, and Christian Theobalt. Holoported characters: Real-time free-viewpoint rendering of humans from sparse rgb cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1206–1215, 2024. 2

[43] Yunsheng Shi, Zhengjie Huang, Shikun Feng, Hui Zhong, Wenjin Wang, and Yu Sun. Masked label prediction: Unified message passing model for semi-supervised classification. *arXiv preprint arXiv:2009.03509*, 2020. 5, 1

[44] Sudipta Sinha, Drew Steedly, and Rick Szeliski. Piecewise planar stereo for image-based rendering. In *2009 International Conference on Computer Vision*, pages 1881–1888, 2009. 2

[45] Pratul P Srinivasan, Tongzhou Wang, Ashwin Sreelal, Ravi Ramamoorthi, and Ren Ng. Learning to synthesize a 4d rgbd light field from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2243–2251, 2017. 2

[46] Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 3dgstream: On-the-fly training of 3d gaussians for efficient streaming of photo-realistic free-viewpoint videos. *arXiv preprint arXiv:2403.01444*, 2024. 3

[47] 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. 1, 2, 4

[48] Shaofei Wang, Katja Schwarz, Andreas Geiger, and Siyu Tang. Arah: Animatable volume rendering of articulated human sdfs. In *European Conference on Computer Vision*, 2022. 2

[49] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4): 600–612, 2004. 6

[50] Jing Wen, Xiaoming Zhao, Zhongzheng Ren, Alexander G Schwing, and Shenlong Wang. Gomavatar: Efficient animatable human modeling from monocular video using gaussians-on-mesh. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2059–2069, 2024. 1, 3

[51] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 16189–16199, 2022. 1, 2

[52] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. In *ACM siggraph 2005 papers*, pages 765–776. 2005. 1

[53] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. *arXiv preprint arXiv:2310.08528*, 2023. 3

[54] Zexiang Xu, Sai Bi, Kalyan Sunkavalli, Sunil Hadap, Hao Su, and Ravi Ramamoorthi. Deep view synthesis from sparse photometric images. *ACM Transactions on Graphics (ToG)*, 38(4):1–13, 2019. 2

[55] Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 4k4d: Real-time 4d view synthesis at 4k resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20029–20040, 2024. 2

[56] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023. 3

[57] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 1, 2, 4

[58] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19447–19456, 2024. 3

[59] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR 2018*, pages 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. 5, 6

[60] 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

[61] Shunyuan Zheng, Boyao Zhou, Ruizhi Shao, Boning Liu, Shengping Zhang, Liqiang Nie, and Yebin Liu. Gps-gaussian: Generalizable pixel-wise 3d gaussian splatting for real-time human novel view synthesis. *arXiv preprint arXiv:2312.02155*, 2023. 1, 3, 7, 2

[62] Zerong Zheng, Han Huang, Tao Yu, Hongwen Zhang, Yandong Guo, and Yebin Liu. Structured local radiance fields for human avatar modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15893–15903, 2022. 2, 6, 1

[63] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 286–301. Springer, 2016. 2

[64] 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. 2

[65] Matthias Zwicker, Hanspeter Pfister, Jeroen Van Baar, and Markus Gross. Ewa splatting. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):223–238, 2002. 3