HumanSplat: Closing the Loop Between Human Mesh Recovery and Gaussian Splatting Avatars

Yeheng Zong* Pou-chun Kung* Yike Pan Seth Isaacson Yizhou Chen
Ram Vasudevan Katherine A. Skinner
Department of Robotics, University of Michigan
{yehengz, pckung, yikepan, sethgi, yizhouch, ramv, kskin} @umich.edu

Abstract

*Accurately recovering human pose and appearance from video is an essential component of scene reconstruction, with applications to motion capture, motion prediction, virtual reality, and digital twinning. Despite significant interest in building realistic human avatars from video, this paper demonstrates that existing methods do not accurately recover the 3D geometry of humans. ViT-based approaches are not consistently reliable and can overfit to 2D views, while NeRF- and Gaussian Splatting–based avatars treat pose and appearance separately, limiting rendering generalization to new poses. To resolve these shortcomings, this paper proposes **HumanSplat**, a joint optimization framework that refines 3D human poses while simultaneously learning a high-fidelity avatar for novel-view and novel-pose synthesis. Our key insight is to close the loop between geometric pose estimation and differentiable rendering. Unlike prior human avatar methods that rely on accurate human pose obtained through motion capture systems or offline refinement, which are impractical in in-the-wild scenarios, our approach uses only human mesh estimates from a state-of-the-art human pose estimator to better reflect real-world conditions. Therefore, instead of using the human pose only as a deformation prior, HumanSplat back-propagates photometric, segmentation, and depth losses through a differentiable renderer to the pose parameters and global position. This coupling refines the global 3D pose over time, improving accuracy and alignment while producing better renderings from novel views. Experiments show consistent improvements over pose recovery baselines that omit image-level refinement and avatar baselines that decouple pose estimation from avatar reconstruction.*

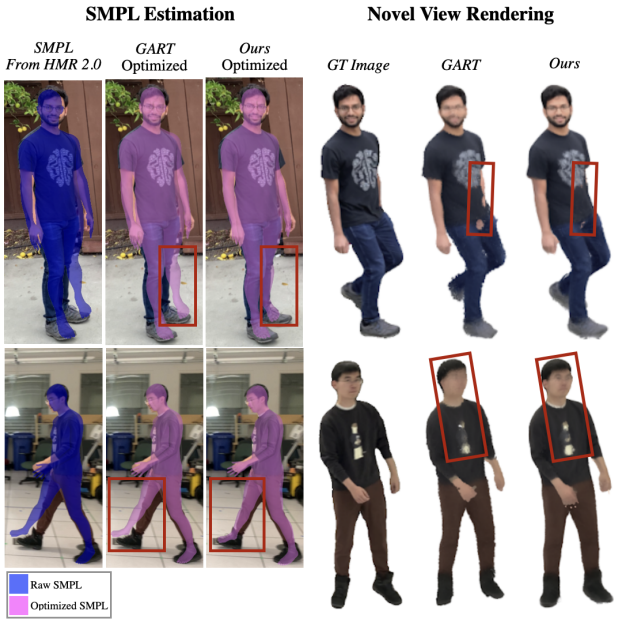

Figure 1. HumanSplat advances both SMPL estimation and novel-view rendering by enabling joint optimization of the human mesh and Gaussian splats, taking SMPL estimates from HMR 2.0 [4] as input. In contrast, prior works, such as GART [21], rely on accurate SMPL from motion capture or refined SMPL to deform Gaussians for avatar reconstruction. These methods also deliberately decouple SMPL from Gaussian splats to improve rendering quality, but this prevents refinement of human pose estimation and ultimately results in sub-optimal novel-view performance. HumanSplat addresses this limitation and allows human pose and Gaussian splats to mutually refine each other. As a result, it achieves more accurate SMPL estimation and consistently higher rendering quality.

1. Introduction

Photorealistic human rendering and human mesh reconstruction are two fundamental and complementary goals in computer vision and graphics, with wide-ranging applications in AR/VR, gaming, simulation, virtual try-on, and cinematic production [37]. While photorealistic rendering enables immersive visual experiences, mesh reconstruction provides explicit and animatable geometry that sup-

ports editing, motion retargeting, physics-based simulation, and physical interaction. Despite their importance, reconstructing realistic avatars and accurate human meshes from monocular video remains highly challenging. Monocular human mesh reconstruction methods often struggle to produce meshes that align well with the image. Photorealistic avatar approaches typically use human mesh reconstruction as a motion prior for novel-view and novel-pose rendering. However, these methods generally overlook errors in mesh reconstruction and over-rely on the estimated meshes, which can lead to suboptimal avatar quality and degraded realism.

Recent studies of Human Mesh Recovery (HMR) center on the Skinned Multi-Person Linear (SMPL) [1] model, which provides a parametric representation of the human body with a compact set of shape parameters [13]. Recent end-to-end HMR methods directly regress SMPL parameters either from a single image [3, 4, 13] or from video sequences with temporal cues [17], achieving promising results but still prone to imperfect alignment and overfitting to 2D images. Based on SMPL estimation methods, human avatar modeling has advanced through the application of radiance field rendering models, such as neural radiance fields (NeRF) [23] or 3D Gaussian Splatting (3DGS [15]). However, there exists a trade-off in existing methods between accurate SMPL estimation and high fidelity rendering quality. Methods that optimize SMPL with photorealistic representation tightly-coupled to SMPL achieve accurate human geometry, but sacrifice rendering fidelity [24]. Conversely, approaches that allow photorealistic representation to detach from SMPL through learnable skinning weights or deformation field deliver high-quality renderings, yet degrade SMPL accuracy [6, 18, 21, 28]. This trade-off between geometry and appearance remains unresolved.

In this work, we propose HumanSplat, a unified framework that bridges these gaps, as shown in Figure 1. Firstly, our method combines end-to-end HMR with a rendering-based iterative refinement stage. In particular, HumanSplat leverages Gaussian representations supervised by photometric, depth, and segmentation cues instead of 2D keypoints. As a result, experiments demonstrate that HumanSplat supports global pose estimation, reduces overfitting to 2D views, and improves alignment of the estimated mesh and pose to the ground-truth. Further, to mitigate the trade-off between geometry and appearance in human avatar modeling, we introduce a joint optimization between the Gaussian representation and human poses with simple-yet-effective Cloth-Aware Mesh-Embedded Loss (CAMEL) that improves both human pose estimation and novel view rendering. Together, our method yields consistent improvements in both SMPL accuracy and photorealistic rendering, enabling the reconstruction of high-fidelity human avatars from video.

2. Related Work

2.1. Human Mesh Recovery

Human mesh recovery is commonly performed using the SMPL model [1], whose parameters can represent different human poses and shapes. Many early approaches estimated 3D human pose and shape through iterative optimization. The first fully automatic method, SMPLify [35], proposed to instead fit SMPL to 2D keypoint detections obtained from an off-the-shelf keypoint detector, guided by strong priors to stabilize optimization. However, this method is slow and only uses sparse supervision at the keypoint locations.

Subsequent research introduced direct prediction of SMPL parameters using pre-trained models. Kanazawa et. al. [13] introduced the first end-to-end learning-based model for human pose recovery. SPIN [19] extended this idea by incorporating an "in-the-loop" optimization that embeds SMPLify within HMR training using a CNN-based model. To further address temporal consistency, many approaches add sequential modeling mechanisms, such as a temporal encoder that fuses per-frame features [14] or recurrent temporal encoders to smooth pose predictions [2, 17]. Beyond single-image human recovery, PHALP [36] extends HMR frameworks to human tracking, exploiting 3D reconstruction for robust identity association over time. Still, the performance is limited by single-frame estimation.

Most recently, HMR approaches have been extended to ViT-based methods such as HMR2.0 [4], TokenHMR [3], and HSMR [33], which have shown promising performance by improving speed and accuracy. However, end-to-end HMR methods often suffer from sub-optimal 3D alignment and overfitting to 2D views.

2.2. Human Avatar

Recent advances in NeRF [23] and Gaussian Splatting [15] have significantly accelerated progress in human avatar modeling, enabling photorealistic rendering of humans. These renderings can be formed both from novel camera views and novel human joint configurations. NeRF-based human avatar methods typically leverage a skeleton or the SMPL model to construct a deformation field, allowing reconstruction from monocular video and articulation of the human body after training [11, 25, 32, 34].

Gaussian Splatting–based avatar methods [6, 18, 21, 28, 31] adopt a similar idea by deforming Gaussians according to the SMPL mesh using classical linear blend skinning (LBS) [10]. However, relying exclusively on fixed SMPL estimates to deform Gaussians propagates errors and leads to degraded performance when SMPL predictions fail, while also limiting the ability to capture non-rigid clothing. To address this limitation, recent human Gaussian Splatting works enable joint optimization of SMPL and Gaussian splats by incorporating a rendering-based photometric loss.

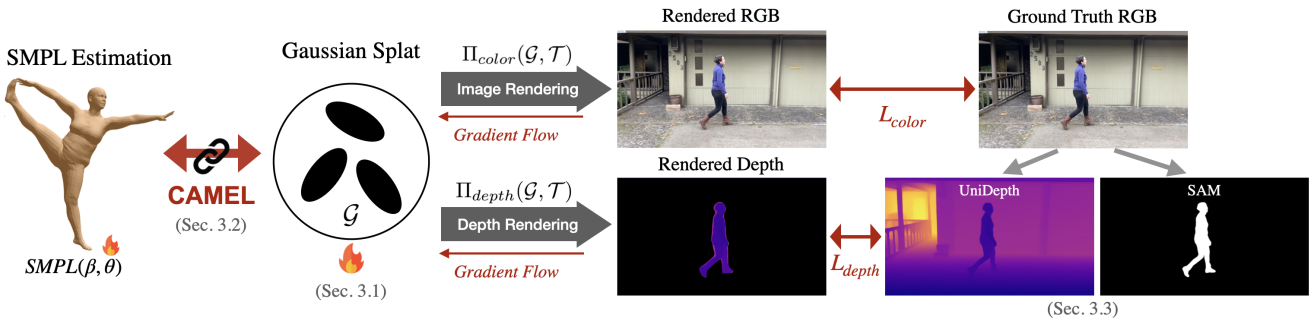

Figure 2. Method Overview. HumanSplat takes SMPL estimation as input and combines SMPL with Gaussians with the proposed CAMEL. This enables both SMPL and Gaussian representation optimization using rendered color and depth images.

Most of the works demonstrated improved rendering quality by using learnable skinning weights [6, 18, 21, 28] or non-rigid deformations [31]. We compare against GART [21] as the primary baseline in our evaluations, chosen as a representative method that performs well. However, this added flexibility reduces SMPL accuracy: once the Gaussians are allowed to decouple from the mesh, they are no longer constrained to follow SMPL geometry. As a result, the rendering may overfit to training views while the SMPL pose remains wrong, as shown in Figure 3. While this leads to strong evaluations when the test set remains close to the training set, the models do not generalize to views or poses too different from those seen during training. Conversely, iHuman [24] tightly couples Gaussians with the SMPL mesh using mesh-embedded Gaussians. This enables SMPL refinement and improves pose estimation, but the strict coupling reduces flexibility, leading to worse novel view rendering.

In HumanSplat, we introduce a Cloth-Aware Mesh-Embedded Loss (CAMEL) that establishes a loose but consistent coupling between the Gaussians and the SMPL model. This formulation simultaneously guides Gaussian placement with mesh-aware priors and allows local flexibility for clothing and non-rigid deformations. As a result, CAMEL not only facilitates accurate refinement of SMPL parameters but also enhances rendering fidelity, yielding superior novel-view synthesis and robust human pose generation.

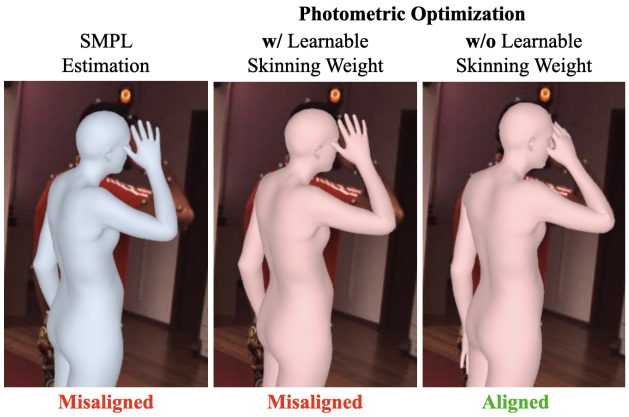

Figure 3. Raw SMPL estimation and photometric optimized SMPL with and without learnable skinning weights. Enabling learnable deformation like learnable skinning weights allows Gaussians to detach from the SMPL mesh, which causes sub-optimal SMPL refinement.

3. Method

In this section, we first review the preliminaries and existing approaches that combine Gaussian Splatting with the SMPL model. We then introduce a novel loss function that loosely couples Gaussians with the SMPL mesh. Finally, we present the full loss function used for optimization. The method overview is shown in Figure 2.

3.1. Preliminaries

3.1.1. Gaussian Splatting

3D Gaussian Splatting [15] models the 3D scene using a set of Gaussian functions. Each 3D Gaussian is parameterized by its position μ, rotation quaternion q, scaling vector S, opacity α, and spherical harmonic (SH) coefficients sh.

$$\mathcal{G} = \{G_i := (\mu_i, q_i, S_i, \alpha_i, sh_i)\}_{i=1}^{N} \tag{1}$$

The covariance of each Gaussian is computed according to

$$\Sigma = RSS^T R^T, \tag{2}$$

where $S \in \mathbb{R}^3$ is a 3D scale vector with the singular values of Σ, and $R \in \mathrm{SO}(3)$ is rotation matrix computed from quaternion q. The color and depth image can be rendered from camera view $\mathcal{T}$ following [15]. We denote the rendering color and depth rendering equation as $\hat{\mathbf{I}} = \Pi_{color}(\mathcal{G}, \mathcal{T})$ and $\hat{\mathbf{D}} = \Pi_{depth}(\mathcal{G}, \mathcal{T})$, respectively.

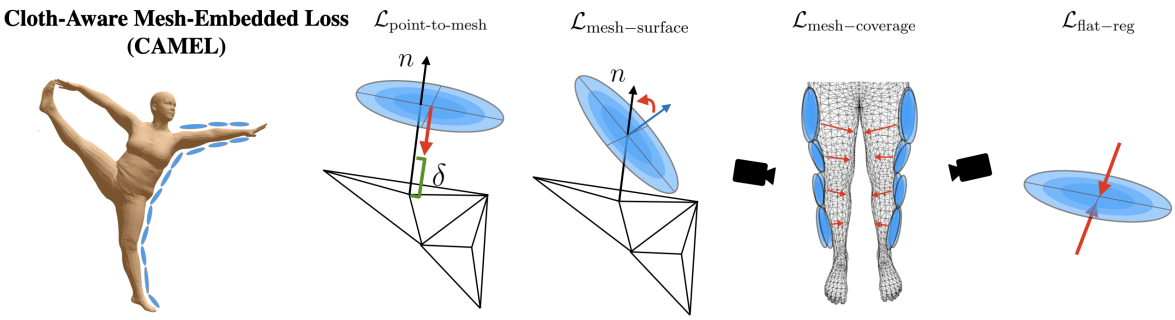

Figure 4. Cloth-Aware Mesh-Embedded Loss (CAMEL) illustration. CAMEL loosely couples the SMPL mesh with the Gaussian representation. The key motivation is to better model clothing by allowing local non-rigid deformations. CAMEL constrains Gaussians to remain close to the human mesh while enforcing surface alignment and ensuring full mesh coverage. n is the normal of the mesh vertex and δ represents the tolerance margin between the cloth and the body.

3.1.2. SMPL Model and Template

The SMPL model [1] provides a compact parametric representation of the human body. It takes as input pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{24\times3\times3}$ and shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$, and generates a mesh with vertices $\mathbf{V} \in \mathbb{R}^{3\times N}$, where $N = 6890$. The pose vector $\boldsymbol{\theta}$ is composed of the body pose $\boldsymbol{\theta}_b \in \mathbb{R}^{23\times3\times3}$ together with a global orientation term $\boldsymbol{\theta}_g \in \mathbb{R}^{3\times3}$. The vertices of an SMPL mesh in canonical space can be obtained by $\mathbf{V}_c = \mathbf{T}(\boldsymbol{\beta})$, where $\mathbf{T}$ is the shape template. The mesh can be deformed using linear blend skinning (LBS) based on articulated pose $\boldsymbol{\theta}$:

$$\mathbf{V}(\boldsymbol{\beta},\boldsymbol{\theta}) = \left(\sum_{k=1}^{n_b} w_k(\mathbf{V}_c(\boldsymbol{\beta}))\, B_k(\boldsymbol{\theta})\right)\mathbf{V}_c(\boldsymbol{\beta}), \quad (3)$$

where B is the rigid body bones based on pose $\boldsymbol{\theta}$, n_b is the number of bones, and $w_k(x)$ is the skinning weight describing how each vertex is influenced by bones. Note that skinning weight is precomputed and fixed for the SMPL template mesh.

In this work, we leverage detected SMPL models $(\boldsymbol{\beta},\boldsymbol{\theta})$ from HMR2.0 [4], then optimize pose parameters $\boldsymbol{\theta}$ through joint optimization of human pose and Gaussian parameters.

3.1.3. Human Gaussian Splat Representation

To combine Gaussian splatting with the SMPL model, Gaussians can also be deformed based on pose $\boldsymbol{\theta}$ using LBS:

$$\boldsymbol{\mu}'(\boldsymbol{\beta},\boldsymbol{\theta}) = \left(\sum_{k=1}^{n_b} w_k(\boldsymbol{\mu})\, B_k(\boldsymbol{\theta})\right)\boldsymbol{\mu}, \quad (4)$$

where $\boldsymbol{\mu}'$ denotes Gaussian center deformed from $\boldsymbol{\mu}$. To enable more photorealistic rendering and non-rigid cloth-modeling, most prior Gaussian avatar works [6, 18, 21, 28] propose learnable skinning $\Delta\mathbf{w}$ to allow non-rigid motion:

$$\widehat{w}(\boldsymbol{\mu}) = w(\boldsymbol{\mu}) + \Delta\mathbf{w}, \quad (5)$$

or apply learnable non-rigid deformation fields [31].

However, the LBS skinning weight of an SMPL template is pretrained using high-quality and expensive 3D human model scans. Re-enabling skinning weight optimization can improve rendering fidelity by letting the projected mesh overfit the 2D image, but it hurts both novel view or pose rendering and SMPL pose estimation. First, the learnable skinning allows Gaussians to detach from the human mesh, which makes the Gaussian splat overfit to 2D training views. This can severely degrade rendering quality when rendering from views far from training views. Secondly, due to the disconnect between SMPL meshes and Gaussian splats, the mesh refinement can degrade SMPL estimation, as shown in Figure 1.

3.2. Cloth-Aware Mesh-Embedded Loss

To achieve high-fidelity rendering without the overfitting and mesh-disentanglement induced by learnable skinning weights, we propose the Cloth-Aware Mesh-Embedded Loss (CAMEL). CAMEL enables high-quality avatar generation with 3D Gaussian Splatting while maintaining the geometric correlation between Gaussians and the SMPL model. The loss is designed to keep Gaussians close to the human mesh within a cloth margin, ensure full coverage of the human body, and encourage alignment of Gaussians with the SMPL surface. As a result, CAMEL ensures both geometric and rendering consistency. The illustration is shown in Figure 4.

Cloth-Aware Point-to-Mesh Surface Loss. Rather than forcing the Gaussians to stay strictly on the mesh surface, we propose a point-to-plane loss that keeps Gaussians within a thin band of thickness δ around the SMPL mesh. In particular, we compute this loss as

$$\boldsymbol{d}_{p2mesh} = \langle \boldsymbol{\mu}_i - \mathbf{v}_{\text{NN}(\boldsymbol{\mu}_i)},\, \mathbf{n}_{\text{NN}(\boldsymbol{\mu}_i)} \rangle \quad (6)$$

$$\mathcal{L}_{\text{p2mesh}} = \frac{1}{N}\sum_{i=1}^{N} \max\Big(\big|\boldsymbol{d}_{p2mesh}\big| - \delta,\, 0 \Big) \quad (7)$$

where $\mathbf{n}$ denotes the surface normal of the SMPL vertices and $\mathbf{v}_{NN(\boldsymbol{\mu}_i)}$ and $\mathbf{n}_{NN(\boldsymbol{\mu}_i)}$ denotes the position and normal of nearest neighbor SMPL vertex of $\boldsymbol{\mu}_i$.

Mesh Coverage Loss. To ensure full-body Gaussian coverage, we require each mesh vertex $\mathbf{v}$ to be close to at least one Gaussian:

$$\mathcal{L}_{\text{coverage}} = \mathbb{E}_{\mathbf{v}}\Big[\max\big(\|\mathbf{v} - \boldsymbol{\mu}_{\text{NN}(\mathbf{v})}\| - \delta,\, 0\big)\Big]. \tag{8}$$

Gaussian Flatness Regularization. Inspired by recent works [5, 7, 16, 20, 22] demonstrating that flattened, surface-aligned Gaussians improve novel-view rendering quality, we introduce a regularization loss that encourages Gaussian shapes to remain flat and aligned with the mesh surface. For each Gaussian scale $\mathbf{S}_i = [s_0, s_1, s_2]$ with $s_0 \leq s_1 \leq s_2$, we take $s_n = s_0$ as the normal-axis scale and $\{s_{t1}, s_{t2}\} = \{s_1, s_2\}$ as the tangential-axis scales. We encourage the tangential axes to exceed the normal axis by a ratio τ:

$$\mathcal{L}_{\text{flatness}} = \big|\log \tfrac{s_{t1}}{s_n} - \log\tau\big| + \big|\log \tfrac{s_{t2}}{s_n} - \log\tau\big|. \tag{9}$$

Surface Alignment Loss. Complementary to the flatness regularization, we add a loss encouraging the smallest axis of the Gaussian to align with the SMPL surface normal:

$$\mathcal{L}_{\text{surface}} = \frac{1}{N}\sum_{i=1}^{N}(1 - \big|\langle \mathbf{R}_i\mathbf{e}_n, \mathbf{n}_{NN(\boldsymbol{\mu}_i)}\rangle\big|), \tag{10}$$

where $\mathbf{e}_n$ denotes the canonical axis associated with s_n and $\mathbf{R}$ is the rotation matrix of a Gaussian.

The total CAMEL loss is:

$$\mathcal{L}_{\text{CAMEL}} = \lambda_1\mathcal{L}_{\text{p2mesh}} + \lambda_2\mathcal{L}_{\text{coverage}} + \lambda_3\mathcal{L}_{\text{flatness}} + \lambda_4\mathcal{L}_{\text{suface}}, \tag{11}$$

where λ_i denotes the weights associated with the respective loss terms.

3.3. Color and Depth Supervision

In addition to the geometric losses proposed in Section 3.2, rendering-based losses are applied to directly supervise the rendering process.

3.3.1. Photometric Loss

Given the rendered image $\hat{\mathbf{I}}$ and the ground-truth frame $\mathbf{I}$, the L1 loss $\mathcal{L}_{L1} = |\hat{\mathbf{I}} - \mathbf{I}|$ and SSIM loss $\mathcal{L}_{SSIM}$ are computed using the image similarity metric [30]. We then minimize photometric error using:

$$\mathcal{L}_{color} = \mathcal{L}_{L1} + \lambda_{SSIM}\mathcal{L}_{SSIM} \tag{12}$$

3.3.2. Depth Loss

Unlike previous works that only produce a SMPL mesh to fit the 2D view without considering global 3D scale and pose, we aim to estimate the SMPL mesh with global scale and pose. We leverage monocular depth estimation output from UniDepthV2 [26] to supervise training, thereby improving global 3D pose estimation.

Rendered Depth Loss. Let $\hat{\mathbf{D}}$ be the rendered depth and $\mathbf{D}$ the ground-truth metric depth. We supervise only pixels with positive ground-truth depth values and with a human mask generated by Segment Anything Model 2 (SAMv2) [27].

$$\mathcal{L}_{\text{depth-2D}} = \Big|\mathbb{1}_{\{\mathbf{D}>0,\ \text{SAM}>0\}} \odot (\hat{\mathbf{D}} - \mathbf{D})\Big|, \tag{13}$$

Gaussian Center Loss. From the current frame, we extract the set of visible Gaussian centers μ and construct a depth point cloud $\mathbf{p} = \pi^{-1}(\mathbf{D}, K)$ by inverse projection, where D is the depth image and K is the camera intrinsic. For each Gaussian center $g \in \mu$, we compute its squared Euclidean distance to the closest point in the depth point cloud. We minimize these nearest-neighbor distances using the Huber penalty [8]:

$$\mathcal{L}_{\text{depth}-3\text{d}} = \frac{1}{2|\mu|}\sum_{\mathbf{g}\in\mu}\rho_\beta\Big(\min_{p\in\mathbf{p}}\|g-p\|^2\Big), \tag{14}$$

where $\rho_\beta(u) = \sqrt{u+\epsilon^2} - \epsilon$ and ϵ is the scale parameter of the Huber penalty and u is Gaussian mean. This anchors the learned geometry to metric depth while tolerating noise and outliers.

In summary, the depth supervision consists of two components:

$$\mathcal{L}_{depth} = \mathcal{L}_{depth-2D} + \mathcal{L}_{depth-3D} \tag{15}$$

3.3.3. Full Loss

The full objective combines image and geometry terms:

$$\mathcal{L} = \mathcal{L}_{color} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{CAMEL}}\mathcal{L}_{\text{CAMEL}} \tag{16}$$

, where each λ denotes the weight associated with the respective loss terms.

4. Experiments

This section details the experiments used to evaluate the performance of HumanSplat.

Method	H36M		3DPW		
	mpjpe↓	pa-mpjpe↓	mpjpe↓	pa-mpjpe↓	v2v ↓
HMR2.0	80.58	**41.13**	75.64	56.26	239.27
GART	80.58	**41.13**	75.64	56.27	239.27
Ours	**80.53**	**41.13**	**72.63**	**55.10**	**196.95**

Table 1. Evaluation of SMPL pose accuracy on H36M and 3DPW datasets. HumanSplat consistently achieves the lowest pose error.

4.1. Implementation Details

We use Pytorch for the implementation of our joint optimization framework that refines the 3D human poses when reconstructing an expressive human avatar. All tests were run using the Adam optimizer. We conducted all experiments on a single NVIDIA Tesla V100 SXM2 16GB GPU. We used SMPL as the initial mesh template with its blend skinning weight. We initialized Gaussians on the vertices of the default mesh with $V = 6890$ and $F = 13776$. During training, we take a monocular RGB video as input, accompanied by SMPL parameters, binary mask, metric depth map, and camera parameters at each frame. To best mimic the real-world deployment where most likely only RGB video is available, we use HMR2.0 [4] for SMPL parameters [1], SAMv2 [27] for binary masks, and UniDepthv2 [26] for metric depth maps and camera parameters. Our method runs at 1000 iterations per second during training with $>$ 120 fps during inference on NVIDIA V100. During test-time optimization, we follow the design in InstantAvatar [11] and GART [21] to pass the gradient from Gaussian parameters to optimize SMPL parameters.

4.2. Dataset

We evaluate our method on four different datasets, including controlled studio capture and in-the-wild scenes, and select sequences that contain only one human subject.

Human3.6M (H36M) [9]. H36M is a large-scale indoor motion-capture dataset with calibrated multi-camera studio recordings of multiple subjects performing diverse actions. It is widely used to train human pose and mesh recovery networks for accurate 3D joint keypoint annotations and camera calibration. We use H36M primarily to assess the SMPL pose accuracy.

NeuMan [12]. The NeuMan dataset consists of 6 videos lasting between 10 to 20 seconds and features a single human subject captured using a mobile phone. It offers calibrated sequences with optimized SMPL poses as reference ground truth. We use NeuMan to assess both the rendering quality of novel views and pose synthesis and SMPL pose accuracy.

3D Poses in the Wild (3DPW) [29]. 3DPW is an in-the-wild dataset with handheld videos and challenging poses, clothing, and lighting. It includes accurate ground-truth 3D poses obtained from wearable sensors, enabling robust evaluation under real-world conditions. We use 3DPW primarily to assess the SMPL pose accuracy.

4.3. Baselines

4.3.1. SMPL Baseline

We compare against three complementary SMPL-based methods. **HMR2.0** [4] is a fully transformer-based human-mesh-recovery backbone that performs per-image SMPL mesh/pose recovery with strong single-frame accuracy.

4.3.2. Human-Avatar Baseline

We also benchmark against an animatable avatar method tailored for monocular video. With a SMPL template prior, **GART** [21] builds Gaussian splatting avatars by jointly optimizing 3D Gaussian representations and SMPL representation with learnable skinning weights enabled as proposed.

Note that, unlike prior works that rely on accurate SMPL parameters from motion capture systems or offline refinement, we use SMPL estimates from HMR2.0 [4] as input for all baselines to simulate in-the-wild conditions.

4.4. SMPL Evaluation

We report mean per-joint position error (MPJPE) [9] and Procrustes-aligned MPJPE (PA-MPJPE) [13] on Human3.6M (H36M) and 3DPW in Table 1. Additionally, NeuMan results are reported in Table 2. H36M is a relatively simple dataset, so HumanSplat only attains slightly improved joint accuracy. In contrast, on 3DPW, our joint mesh–pose refinement consistently yields the lowest error, indicating larger gains when initial poses are noisier, which demonstrates our capability to handle noisier initial SMPL estimation. On the NeuMan dataset, we also demonstrate overall better performance compared to other baselines, as shown in Figure 5.

4.5. Rendering Evaluation

We evaluate rendering quality on six challenging in-the-wild sequences from the NeuMan dataset under three training setups to cover different train/test ratios and view differences. First, we train *HumanSplat* on 80% of the frames and evaluate on the held-out 20% by uniformly withholding every fifth frame. Next, we train on the first 80% of the frames and test on the final 20%. Finally, we uniformly sample only 20 camera views for training and use all remaining views for testing. Novel view synthesis results for these different train/test schemes are reported in Table 3. In all setups, *HumanSplat* consistently outperforms GART. Qualitative results are shown in Figure 6.

4.6. Ablation Studies

The ablation study in Table 4 shows that the full method achieves the best balance of rendering quality and SMPL

Method	Bike			Citron			Jogging			Lab			ParkingLot			Seattle		
	mpjpe↓	pa-mpjpe↓	v2v↓	mpjpe↓	pa-mpjpe↓	v2v↓	mpjpe↓	pa-mpjpe↓	v2v↓	mpjpe↓	pa-mpjpe↓	v2v↓	mpjpe↓	pa-mpjpe↓	v2v↓	mpjpe↓	pa-mpjpe↓	v2v↓
HMR2.0	77.21	58.23	70.82	89.29	52.81	77.37	99.36	69.90	90.79	94.83	58.08	75.42	**69.90**	51.33	56.87	95.79	55.73	74.80
GART	76.12	57.52	70.57	**87.59**	**51.96**	74.33	98.25	69.50	89.82	93.57	57.51	73.07	69.95	**51.25**	**55.72**	93.36	54.91	72.65
Ours	**73.62**	**50.49**	**56.75**	88.44	52.19	**66.54**	**93.25**	**67.25**	**76.83**	**89.36**	**55.78**	**67.11**	74.73	53.77	58.97	**84.55**	**51.53**	**67.02**

Table 2. SMPL accuracy evaluation on NeuMan dataset. Our method achieves overall the best SMPL estimation with lowest error. In this experiment, we deploy uniform 80% train-split strategy, witholding every fifth frame.

		Bike			Citron			Jogging			Lab			Parkinglot			Seattle		
	Method	psnr↑	ssim↑	lpips↓	psnr↑	ssim↑	lpips↓	psnr↑	ssim↑	lpips↓	psnr↑	ssim↑	lpips↓	psnr↑	ssim↑	lpips↓	psnr↑	ssim↑	lpips↓
Uniform 80%	GART	23.57	0.958	0.044	24.76	0.964	0.031	24.76	0.963	0.038	23.16	0.963	0.041	27.60	0.974	0.031	27.94	0.975	0.021
	Ours	**25.53**	**0.964**	**0.033**	**27.18**	**0.971**	**0.022**	**26.58**	**0.968**	**0.029**	**25.27**	**0.971**	**0.030**	**28.54**	**0.976**	**0.024**	**29.85**	**0.980**	**0.016**
20 Views	GART	22.89	0.954	0.045	24.31	0.960	0.033	24.21	0.959	0.040	22.29	0.959	0.044	26.29	0.966	0.037	27.43	0.973	0.022
	Ours	**23.98**	**0.955**	**0.036**	**26.10**	**0.965**	**0.025**	**25.10**	**0.961**	**0.033**	**23.87**	**0.964**	**0.034**	**27.11**	**0.970**	**0.029**	**29.17**	**0.978**	**0.017**
First 80%	GART	20.97	0.958	0.047	23.28	0.952	0.039	24.40	0.963	0.036	21.67	0.955	0.051	24.06	0.955	0.047	27.01	0.971	0.024
	Ours	**24.22**	**0.965**	**0.038**	**24.53**	**0.959**	**0.033**	**25.79**	**0.965**	**0.032**	**24.04**	**0.963**	**0.037**	**25.22**	**0.959**	**0.037**	**28.84**	**0.976**	**0.018**

Table 3. Novel view rendering quality using multiple train-test splits. 'Uniform 80%' indicates witholding every fifth frame, as is standard in the baselines. '20 Views' indicates training on 20 randomly-sampled views. The 'First 80%' split witholds the final 20% of frames from training, resulting in train and test views that are more dissimilar than alternate train-test schemes, showing the benefits of HumanSplat for rendering configurations far from the training frames.

accuracy. Disabling SMPL optimization or depth supervision leads to noticeable performance degradation. Notably, enabling learnable skinning weights produces rendering quality comparable to the full method but reduces SMPL accuracy, since Gaussians can detach from the mesh and overfit the training views without improving pose estimation, as illustrated in Figure 3.

We also perform ablation studies on different strategies for binding Gaussians to SMPL. Without any constraint between Gaussians and the SMPL mesh, both rendering quality and SMPL accuracy degrade significantly, as expected. Using learnable skinning weights (LSW), following most existing works [6, 18, 21, 28, 31], yields the best rendering quality but poorer SMPL estimation due to the weak connection between Gaussians and the mesh. In contrast, enabling Gaussian-on-Mesh (GoM) embedding produces more accurate human geometry but suboptimal rendering quality, consistent with prior findings [24]. Finally, our proposed CAMEL achieves the best overall trade-off, delivering strong rendering quality and accurate SMPL estimation by loosely coupling Gaussians with the SMPL mesh.

	Rendering Quality			SMPL Estimation		
Ablation	psnr↑	ssim↑	lpips↓	mpjpe↓	pa-mpjpe↓	v2v↓
w/o SMPL Opt.	23.52	0.960	0.0400	93.11	59.59	76.25
w/o Depth Loss	25.12	0.966	0.0337	85.58	56.79	70.49
w/ LSW	26.02	**0.970**	0.0309	87.36	57.28	68.46
Full (Ours)	**26.24**	**0.970**	**0.0306**	**82.88**	**56.31**	**66.06**
Ablation on Gaussian-to-SMPL Binding						
No Constraint	25.67	0.967	0.0314	84.88	57.45	67.59
LSW	26.16	0.968	**0.0303**	86.80	58.29	69.15
GoM	26.05	0.968	0.0314	**81.83**	57.20	**65.69**
CAMEL	**26.24**	**0.970**	0.0306	82.88	**56.31**	66.06

Table 4. Ablation of key components. Left: novel view rendering quality of 3D Gaussian splats. Right: SMPL reconstruction accuracy. We evaluate novel-view/pose synthesis and SMPL accuracy by disabling key components. Results are averaged on 6 subjects of NeuMan. 'LSW' denotes Learnable Skinning Weights

5. Conclusion

We introduced HumanSplat, a framework that enables Gaussian–SMPL pose joint optimization through our proposed Cloth-Aware Mesh-Embedded Loss (CAMEL). Unlike prior works that face a trade-off between novel-view rendering and mesh reconstruction accuracy, CAMEL supports joint optimization and achieves improvements in both photorealistic human avatar rendering and mesh recovery. Extensive experiments on three public datasets demonstrate consistent gains over three mesh recovery and three human avatar baselines. Our pipeline is flexible and can be integrated with different end-to-end mesh recovery models to further refine SMPL estimation. While our current method focuses on single-human reconstruction, future work will extend HumanSplat to handle multi-human tracking and reconstruction.

Figure 5. SMPL estimation comparison on NeuMan dataset. HumanSplat shows better SMPL refinement results than GART.

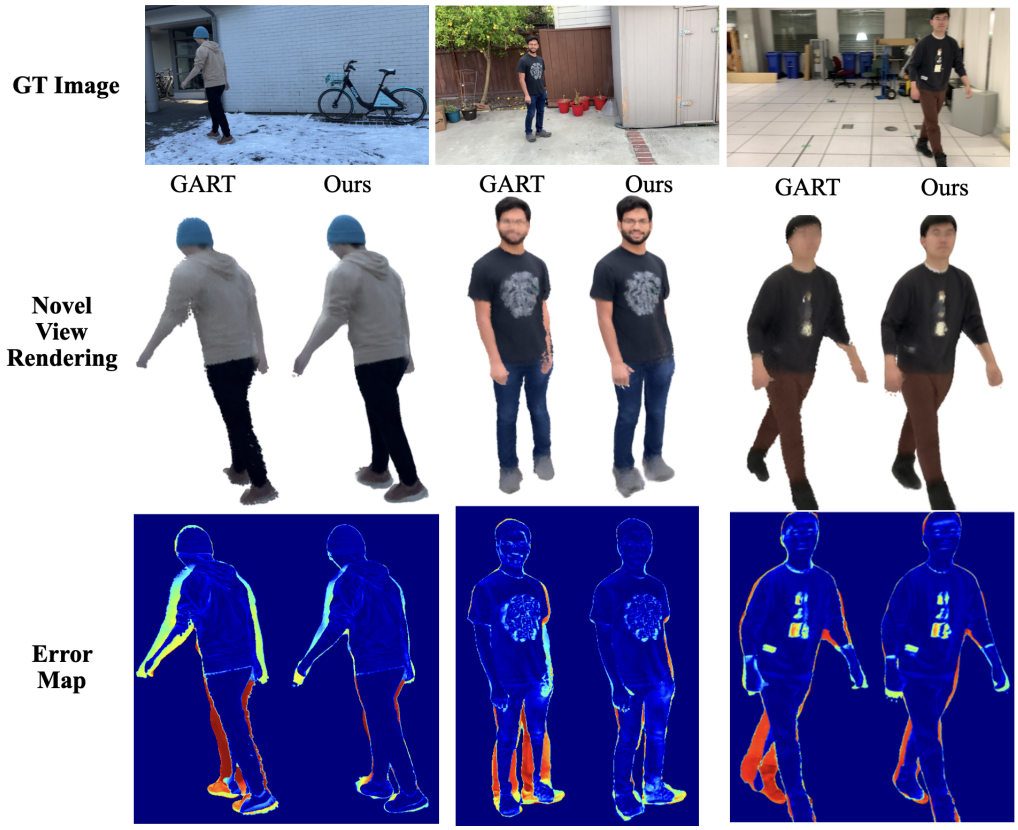

Figure 6. Novel view rendering evaluation. HumanSplat shows better rendering quality compared to GART [21]. While GART enables learnable skinning weights to handle SMPL pose error from HMR2.0 [4] in training views, the novel view/pose rendering can still be bad without correctly refining SMPL poses. In contrast, the proposed CAMEL enables joint optimization of both SMPL pose and Gaussian splats, thus outperforming previous works.

References

[1] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European conference on computer vision*, pages 561–578. Springer, 2016. 2, 4, 6

[2] Kellie Corona, Katie Osterdahl, Roderic Collins, and Anthony Hoogs. Meva: A large-scale multiview, multimodal video dataset for activity detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1060–1068, 2021. 2

[3] Sai Kumar Dwivedi, Yu Sun, Priyanka Patel, Yao Feng, and Michael J Black. Tokenhmr: Advancing human mesh recovery with a tokenized pose representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1323–1333, 2024. 2

[4] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14783–14794, 2023. 1, 2, 4, 6, 8

[5] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2024. 5

[6] 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. 2, 3, 4, 7

[7] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pages 1–11, 2024. 5

[8] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer, 1992. 5

[9] 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, 2013. 6

[10] Alec Jacobson, Ilya Baran, Ladislav Kavan, Jovan Popović, and Olga Sorkine. Fast automatic skinning transformations. *ACM Transactions on Graphics (ToG)*, 31(4):1–10, 2012. 2

[11] Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16922–16932, 2023. 2, 6

[12] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video. In *European Conference on Computer Vision*, pages 402–418. Springer, 2022. 6

[13] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018. 2, 6

[14] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5614–5623, 2019. 2

[15] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 3

[16] Mustafa Khan, Hamidreza Fazlali, Dhruv Sharma, Tongtong Cao, Dongfeng Bai, Yuan Ren, and Bingbing Liu. Autosplat: Constrained gaussian splatting for autonomous driving scene reconstruction. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8315–8321. IEEE, 2025. 5

[17] Muhammed Kocabas, Nikos Athanasiou, and Michael 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*, pages 5253–5263, 2020. 2

[18] Muhammed Kocabas, Jen-Hao Rick Chang, James Gabriel, Oncel Tuzel, and Anurag Ranjan. Hugs: Human gaussian splats. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 505–515, 2024. 2, 3, 4, 7

[19] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2252–2261, 2019. 2

[20] Pou-Chun Kung, Seth Isaacson, Ram Vasudevan, and Katherine A Skinner. Sad-gs: Shape-aligned depth-supervised gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2842–2851, 2024. 5

[21] Jiahui Lei, Yufu Wang, Georgios Pavlakos, Lingjie Liu, and Kostas Daniilidis. Gart: Gaussian articulated template models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19876–19887, 2024. 1, 2, 3, 4, 6, 7, 8

[22] Jiahao Luo, Jing Liu, and James Davis. Splatface: Gaussian splat face reconstruction leveraging an optimizable surface. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 774–783. IEEE, 2025. 5

[23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[24] Pramish Paudel, Anubhav Khanal, Danda Pani Paudel, Jyoti Tandukar, and Ajad Chhatkuli. ihuman: Instant animatable digital humans from monocular videos. In *European Conference on Computer Vision*, pages 304–323. Springer, 2024. 2, 3, 7

[25] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body:

Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9054–9063, 2021. 2

[26] Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unidepthv2: Universal monocular metric depth estimation made simpler. *arXiv preprint arXiv:2502.20110*, 2025. 5, 6

[27] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5, 6

[28] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1606–1616, 2024. 2, 3, 4, 7

[29] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 601–617, 2018. 6

[30] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 5

[31] 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. 2, 3, 4, 7

[32] Chung-Yi Weng, Brian Curless, Pratul P Srinivasan, Jonathan T Barron, and Ira Kemelmacher-Shlizerman. Humannerf: Free-viewpoint rendering of moving people from monocular video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern Recognition*, pages 16210–16220, 2022. 2

[33] Yan Xia, Xiaowei Zhou, Etienne Vouga, Qixing Huang, and Georgios Pavlakos. Reconstructing humans with a biomechanically accurate skeleton. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5355–5365, 2025. 2

[34] Menglei Yang, Yuhang Han, Shenhao Zhang, and Xiaohui Zhang. Animatable nerf dynamic detail enhancement based on residual deformation field with progressive training. In *2025 5th International Conference on Computer Graphics, Image and Virtualization (ICCGIV)*, pages 161–165. IEEE, 2025. 2

[35] Yuchen Yang, Linfeng Dong, Wei Wang, Zhihang Zhong, and Xiao Sun. Learnable smplify: A neural solution for optimization-free human pose inverse kinematics. *arXiv preprint arXiv:2508.13562*, 2025. 2

[36] Vickie Ye, Georgios Pavlakos, Jitendra Malik, and Angjoo Kanazawa. Decoupling human and camera motion from videos in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21222–21232, 2023. 2

[37] Ning Zhang and Belei Pu. Film and television animation production technology based on expression transfer and virtual digital human. *Scalable Computing: Practice and Experience*, 25(6):5560–5567, 2024. 1