Expressive Gaussian Human Avatars from Monocular RGB Video

Hezhen Hu[1] **Zhiwen Fan**[1] **Tianhao Wu**[2] **Yihan Xi**[1] **Seoyoung Lee**[1]
Georgios Pavlakos[1] **Zhangyang Wang**[1]
[1] University of Texas at Austin [2] University of Cambridge

Abstract

Nuanced expressiveness, especially through detailed hand and facial expressions, is pivotal for enhancing the realism and vitality of digital human representations. In this work, we aim to learn expressive human avatars from a monocular RGB video; a setting that introduces new challenges in capturing and animating fine-grained details. To this end, we introduce EVA, a drivable human model that can recover fine details based on 3D Gaussians and an expressive parametric human model, SMPL-X. Focused on enhancing expressiveness, our work makes three key contributions. First, we highlight the importance of aligning the SMPL-X model with the video frames for effective avatar learning. Recognizing the limitations of current methods for estimating SMPL-X parameters from in-the-wild videos, we introduce a reconstruction module that significantly improves the image-model alignment. Second, we propose a context-aware adaptive density control strategy, which is adaptively adjusting the gradient thresholds to accommodate the varied granularity across body parts. Third, we develop a feedback mechanism that predicts per-pixel confidence to better guide the optimization of 3D Gaussians. Extensive experiments on two benchmarks demonstrate the superiority of our approach both quantitatively and qualitatively, especially on the fine-grained hand and facial details. We make our code available at the project website: https://evahuman.github.io.

1 Introduction

High-quality digital avatar modeling has a wide range of applications, including AR/VR, movie production, sign language, and more. For digital human representation, capturing nuanced expressions is essential for enhancing fidelity and vitality. This is particularly evident in the detailed portrayal of hands and facial expressions, which add emotional depth and interactive expression capabilities to human avatars. In this work, we investigate expressiveness when building human avatars from monocular video. The task involves taking as input a monocular human video and learning an animated human avatar which enables multiple capabilities, such as novel view and pose synthesis.

Reconstructing expressive human avatars is challenging, particularly due to the subtle and complex movements of the hands and face. Compared to the body, hands and faces occupy smaller spatial areas and have distinct characteristics. For example, the hand has many degrees of freedom, intricate textures, and frequent self-occlusions. To achieve accurate avatar modeling, it is crucial to capture these fine textures from RGB video and ensure effective animation.

Current studies [49, 31, 15, 22, 12] mainly focus on learning human avatar on the body region and have made remarkable progress. Early works [49, 31, 15] mainly utilize NeRF as an implicit representation but usually have the drawback of low training/inference speed. Recently, more and more works build on top of 3D Gaussian Splatting [19] for its effectiveness and efficiency, which could further speed up rendering to over 100fps. GART [22] utilizes a mixture of moving 3D

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

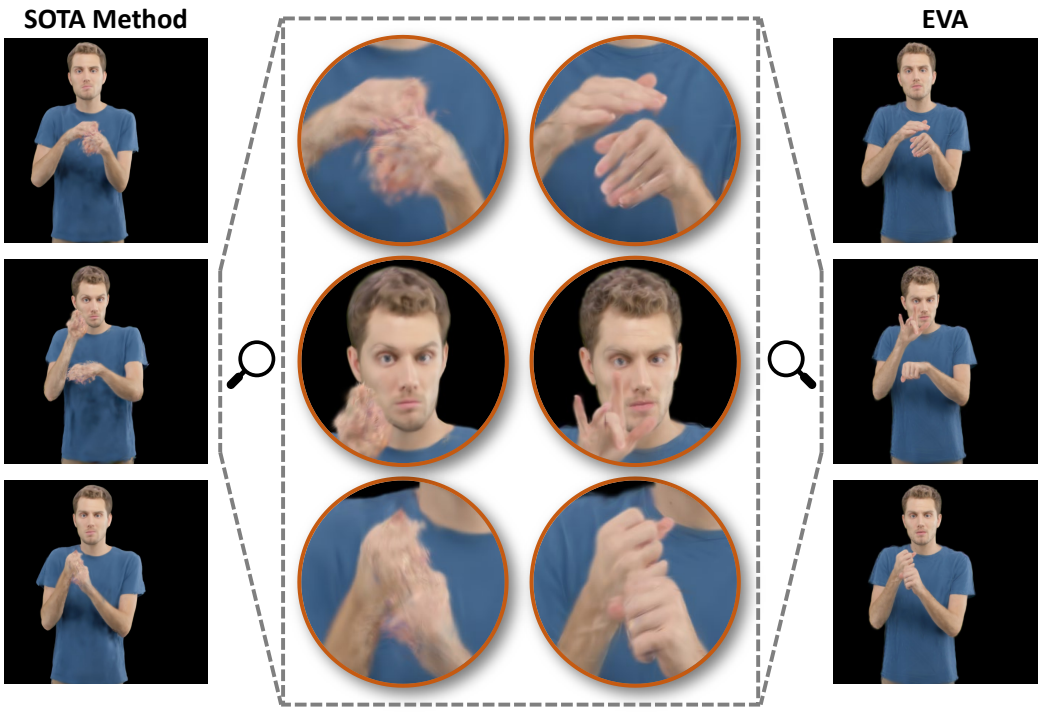

Figure 1: **Learning expressive Gaussian human avatars from RGB video.** Given a monocular video as input, our proposed approach, EVA, learns an expressive 3D Gaussian avatar. The results outperform the SOTA method [22] on novel pose synthesis, especially for the hand and facial details.

Gaussians to approximate human geometry and appearance and enhances fine details with learnable forward skinning and latent bones. GauHuman [12] proposes a new density control strategy, *e.g.* split and clone with KL divergence and a new merge operation. However, these methods do not consider the fine-grained details of hand and face, which cannot meet the requirement of expressiveness.

In this work, we introduce EVA, a drivable human model that can represent expressive details using 3D Gaussians and a human parametric model. Given a monocular RGB video, we extract the pose and mask information corresponding to each frame, which allows us to map each frame's human observation to a canonical space. Once the avatar is constructed, it can be animated using linear blend skinning given a new pose, followed by rendering to the 2D human image. To tackle new challenges introduced by expressiveness, we start by solving the misalignment issue between the SMPL-X model and the RGB frames via a reconstruction module. By employing a fitting-based optimization, this module can produce more reliable 3D SMPL-X reconstruction, providing a more robust foundation for digital avatar modeling.

Considering the granularity differences across different body parts, we propose a context-aware adaptive density control strategy for 3D Gaussian optimization. It leverages attributes specific to different parts and historical gradient information, to adaptively control Gaussian density. Furthermore, to improve the Gaussian optimization, we design a feedback mechanism, which adaptively predicts confidence scores based on the rendering result, thereby ensuring the supervision signal effectively transferring to the corresponding Gaussian. For evaluation on expressiveness, we build comparison baselines from related body avatar methods with a few modifications and collect a new benchmark called UPB containing in-the-wild upper body videos. The image quality is evaluated with multiple metrics, separately on the full body, as well as the hand and face regions. Our approach, as shown in Figure 1, largely outperforms previous state of the art, with more fidelity in fine-grained details.

Our contributions are summarized as follows:

- We introduce EVA, an approach that can build expressive human avatars based on 3D Gaussians, given as input a monocular RGB video. Through extensive experiments on two datasets, we demonstrate the superiority of EVA, particularly on the hand and facial details.

- To enhance expressiveness, we propose a context-aware adaptive density control strategy to accommodate the granularity differences across human parts, followed by a feedback mechanism to better guide the 3D Gaussian optimization.
- To handle challenging in-the-wild videos, we propose a reconstruction module, which critically improves the SMPL-X alignment to the RGB frames compared to off-the-shelf methods. We demonstrate the importance of this improvement in recovering accurate avatars.

2 Related Work

2.1 Human Avatar Modeling from Monocular RGB Video

The problem of building a human avatar from a monocular RGB video is challenging due to the dynamic nature of the capture and the partial observations for each frame. Early works [31, 2, 37, 38, 3, 15, 49, 46, 39, 42, 6, 21, 41] mainly resort to the combination of implicit neural representations (*e.g.,* NeRF [27]) and parametric models to represent a human avatar with high fidelity and flexibility. To address the slow computation in implicit models, previous work has proposed various techniques to reduce training [10, 14] or inference time [33, 25, 5]. With the introduction of 3D Gaussian Splatting [19] which can achieve both fast and high-quality rendering, an increasing number of works [22, 24, 11, 12, 20, 23, 32, 13, 18] utilize it as their base representation, jointly with parametric human models like SMPL [26]. More specifically, GaussianAvatar [11] enhances 3D Guassians via two key components for final photorealistic quality, *i.e.,* dynamic properties and joint optimization of motion and appearance. GauHuman [12] leverages human prior and KL divergence to propose a new density control strategy. In this work, we systematically explore expressiveness while building human avatars from a monocular RGB video and tackle the new challenges brought by expressiveness.

2.2 Expressive Human Representations

Expressiveness plays a vital role in non-verbal communication, which, besides the body, particularly involves the hand and face regions [44, 48, 51, 8, 9, 28, 16, 7, 35]. To represent expressive humans, various parametric models have been proposed, including SMPL-X [28], Adam [17], GHUM [43], and more. These models usually have predefined topology and can provide a compact mapping from the low-dimensional embedding to the 3D mesh. However, the predefined topology limits their capability of depicting fine-grained texture. Recent works build human avatars via neural 3D representations while leveraging the parametric model to provide human shape priors and animation signals. For example, X-Avatar [35] combines NeRF and SMPL-X to build the avatar. It further proposes part-aware sampling and initialization strategies to ensure efficient learning from high-quality 3D scans or RGB-D data. GVA [24] integrates the SMPL-X model to improve the rendering quality. AvatarRex [50] proposes a compositional avatar representation to separately model body, hand and faces from multi-view RGB video data. Different from them, we aim to relax the input requirements, by building expressive human avatars from a real-world monocular RGB video.

3 Technical Approach

In this section, we first introduce the preliminary knowledge on the SMPL-X human model [28] and 3D Gaussian Splatting [19] and present the general framework design on articulated human modeling. Then, we elaborate on the key technical contributions of our approach.

3.1 Preliminaries

SMPL-X body model. SMPL-X [28] is a parametric human body model, which extends the SMPL body model [26] by modeling both hand articulation and facial expressions. SMPL-X can be defined as a mapping function $M(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\psi}) : \mathbb{R}^{|\boldsymbol{\theta}|} \times \mathbb{R}^{|\boldsymbol{\beta}|} \times \mathbb{R}^{|\boldsymbol{\psi}|} \rightarrow \mathbb{R}^{3N}$, where $\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\psi}$ are the parameters for pose, shape, and facial expression, respectively. The function of SMPL-X is formulated as follows:

$$\mathbf{M}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}) = W(\mathbf{T}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}), J(\boldsymbol{\beta}), \boldsymbol{\theta}, \mathbf{W}), \tag{1}$$

$$\mathbf{T}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}) = \bar{\mathbf{T}} + B_S(\boldsymbol{\beta}) + B_E(\boldsymbol{\psi}) + B_P(\boldsymbol{\theta}), \tag{2}$$

where $B_P(\cdot)$, $B_S(\cdot)$ and $B_E(\cdot)$ denote pose, shape, and expression blend functions, respectively, and $\mathbf{W}$ is a set of blend weights. The pose, expression and shape corrective blend shapes, *i.e.,* $B_P(\boldsymbol{\theta})$,

$B_E(\boldsymbol{\psi})$ and $B_S(\boldsymbol{\beta})$, add corrective vertex displacements to the template human mesh $\bar{\mathbf{T}}$. After that, linear blend skinning $W(\cdot)$ is applied to rotate the vertices in the template mesh around the joints $J(\boldsymbol{\beta})$, smoothed by the blend weights $\mathbf{W}$. This generates the final human mesh.

3D Gaussian Splatting. Methods based on NeRF [27] model the scene with an implicit representation and render novel views using volume rendering. In contract, 3D Gaussian Splatting [19] (3DGS) models a 3D scene with a set of discrete 3D Gaussians and performs rendering through a tile-based rasterization operation, which can reach real-time rendering speeds. Specifically, each Gaussian is defined with its central position p, 3D covariance matrix Σ as follows:

$$G(x) = exp(-\frac{1}{2}(x-p)^T\Sigma^{-1}(x-p)), \tag{3}$$

where x is an arbitrary position in the 3D scene. The covariance matrix Σ is decomposed into two learnable components to make the optimization easier, $\Sigma = RSS^TR^T$, where R and S denote the rotation matrix and scaling vector, respectively. During rendering, each 3D Gaussian $G(x)$ is first transformed to a 2D Gaussian $G^{'}(x)$ on the image plane. Then, a tile-based rasterizer is designed to efficiently sort the 2D Gaussians and employ α-blending:

$$C(r) = \sum_{i\in N} c_i\sigma_i \prod_{j=1}^{i-1}(1-\sigma_j), \quad \sigma_i = \alpha_i G'(r), \tag{4}$$

where r is the queried pixel position and N denotes the number of sorted 2D Gaussians associated with the queried pixel. c_i and α_i denote the color and opacity of the i-th Gaussian, which are modeled by Spherical harmonics.

Articulated 3D human modeling. Our formulation takes inspiration from works employing 3D Gaussians for modeling articulated objects [12, 22]. We optimize 3D Gaussians in a canonical space, which corresponds to a human in a rest pose [15]. The Gaussians are transformed from the canonical space to the frame space via linear blend skinning (LBS):

$$\mathbf{p}^f = \mathbf{G}(\mathbf{J}^f, \boldsymbol{\theta}^f)\mathbf{p}^c + \mathbf{b}(\mathbf{J}^f, \boldsymbol{\theta}^f, \boldsymbol{\beta}^f), \tag{5}$$

$$\boldsymbol{\Sigma}^f = \mathbf{G}(\mathbf{J}^f, \boldsymbol{\theta}^f)\boldsymbol{\Sigma}^c\mathbf{G}^{\mathbf{T}}(\mathbf{J}^f, \boldsymbol{\theta}^f), \tag{6}$$

where $\mathbf{p}^f, \boldsymbol{\Sigma}^f, \mathbf{p}^c, \boldsymbol{\Sigma}^c$ are the Gaussian mean and covariance in frame space and canonical space, respectively. $\mathbf{G}(\mathbf{J}^f, \boldsymbol{\theta}^f) = \sum_{k=1}^K w_k\mathbf{G}_k(\mathbf{J}^f, \boldsymbol{\theta}^f)$, $\mathbf{b}(\mathbf{J}^f, \boldsymbol{\theta}^f, \boldsymbol{\beta}^f) = \sum_{k=1}^K w_k\mathbf{b}_k(\mathbf{J}^f, \boldsymbol{\theta}^f, \boldsymbol{\beta}^f)$ are the rotation and translation, respectively, with respect to the K joints, and $\mathbf{G}_k(\mathbf{J}^f, \boldsymbol{\theta}^f)$, $\mathbf{b}_k(\mathbf{J}^f, \boldsymbol{\theta}^f, \boldsymbol{\beta}^f)$ are the rotation and translation, respectively, with respect to the k-th joint. w_k is the LBS weight.

To perform Linear Blend Skinning, we need two important components, *i.e.,* the LBS weights and the input pose parameters. Learning LBS weights w_k from scratch would be inefficient and can lead to a local optimum in the early stage of the training. Therefore, for each Gaussian, we start with the LBS weight of the nearest SMPL-X vertex and use an MLP f_{Θ_w} to predict an LBS weight offset $w_k^{'}$ using the positionally encoded [27] Gaussian centers $\gamma(\boldsymbol{p}^c)$. w_k is therefore defined as:

$$w_k = \frac{e^{\log(w_k^{\text{SMPL-X}}+\epsilon)+w_k^{'}}}{\sum_{j=1}^K e^{\log(w_j^{\text{SMPL-X}}+\epsilon)+w_j^{'}}}, \tag{7}$$

$$w_k^{'} = f_{\Theta_w}(\gamma(\boldsymbol{p}^c)[k]), \tag{8}$$

where $w_k^{\text{SMPL-X}}$ is the LBS weight of the nearest SMPL-X vertex. We set $\epsilon = 10^{-8}$.

Starting from the input pose $\boldsymbol{\theta}^{\text{SMPL-X}}$, we further fine-tune it via an MLP-based network $f_{\Theta_{\boldsymbol{\theta}}}$, which is jointly optimized during the 3D Gaussian optimization process. The actual poses $\boldsymbol{\theta}$ used for optimization and rendering are therefore obtained as follows:

$$\boldsymbol{\theta} = \boldsymbol{\theta}^{\text{SMPL-X}} \otimes f_{\Theta_{\boldsymbol{\theta}}}(\boldsymbol{\theta}^{\text{SMPL-X}}), \tag{9}$$

where $\otimes$ represents the vector pointwise product.

3.2 SMPL-X Alignment for Real-World Videos

A key requirement for learning accurate human avatars is to initialize the optimization process with a reliable SMPL-X estimate. However, this can be challenging, particularly in real-world cases,

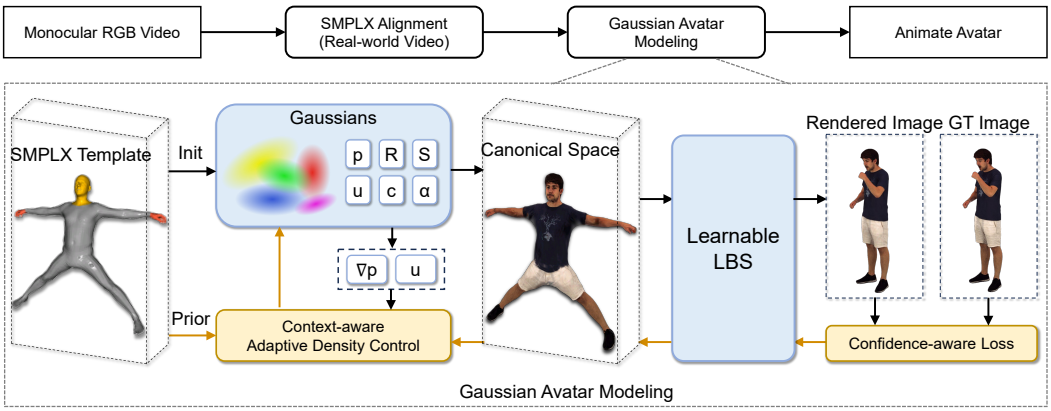

Figure 2: **Overview of our proposed EVA method**. Given a monocular RGB video, first we estimate a SMPL-X mesh that aligns well to the video frames using a reconstruction module. Then, EVA utilizes 3D Gaussian Splatting for avatar modeling, while inheriting the human shape prior from the SMPL-X model. To improve the optimization and the quality of the avatar, we propose context-aware adaptive density control and a confidence-aware loss.

given the limitations of current off-the-shelf methods [29] for SMPL-X fitting. This motivates us to propose a robust fitting procedure that leverages multiple sources, including the initial estimation of camera parameter, SMPL-X parameters, 2D keypoints and 3D hand parameters from off-the-shelf tools [30, 45, 1]. Given these initial estimates, the SMPL-X fitting procedure minimizes the following objective:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\psi}) = \mathcal{L}_{2D} + \lambda_{bp}\mathcal{L}_{bp} + \lambda_{hp}\mathcal{L}_{hp}. \tag{10}$$

The 2D keypoint term, $\mathcal{L}_{2D}$, encourages the projection of the 3D keypoints of the avatar to align with the detected 2D keypoints. It is formulated as follows:

$$\mathcal{L}_{2D} = \sum_{i \in J} \gamma_i \omega_i \psi(\Pi_K(R_{\boldsymbol{\theta}}(J(\boldsymbol{\beta}))_i) - J_i^{2D}), \tag{11}$$

where $\Pi_K(\cdot)$ represents the camera projection under the given camera intrinsic parameter K. $\psi(\cdot)$ is the robust Geman-McClure error function [36] which helps prevent the disturbance from noisy 2D keypoint detections. $R_{\boldsymbol{\theta}}(\cdot)$ denotes the function which rotates the joints $J(\boldsymbol{\beta})$ given the pose $\boldsymbol{\theta}$. J^{2D} is the detected 2D keypoints from [45]. γ and ω represent the predefined weighting parameters and the detection confidences respectively.

Our prior terms contain two components which focus on the coarse-grained body and fine-grained hand, respectively. For the body part, we utilize VPoser [28] to filter infeasible body poses. VPoser provides a compact mapping from a low-dimensional embedding η to the rotation matrices of the body pose θ_b. To better optimize the low-dimensional embedding η, the estimated 3D body joints J_b^{3D} [1] is treated as guidance, together with the added regularization term. The body prior loss term, $\mathcal{L}_{bp}$, is formulated as follows:

$$\mathcal{L}_{bp} = \psi(R_{\boldsymbol{\theta}}(J_b(\boldsymbol{\beta})) - J_b^{3D}) + ||\eta||^2. \tag{12}$$

Similary, for the hands, we utilize the 3D hand joints J_h^{3D} estimated from [30] to provide a better hand spatial relationship. This is formulated as follows:

$$\mathcal{L}_{hp} = \psi_z(R_{\boldsymbol{\theta}}(J_h(\boldsymbol{\beta})) - J_h^{3D}), \tag{13}$$

where the index h represents the joints corresponding to the hand. $\psi_z(\cdot)$ indicates that the robust error function only considers the coordinates along the z-axis coordinates for the error calculation.

3.3 Context-aware Adaptive Density Control

The core of the Gaussian Splatting optimization is adaptive density control, which generally contains two operations, *i.e.,* densification (split and clone) and pruning. The original strategy [19] selects a

fixed constant as the densification criteria. However, utilizing the fixed threshold does not leverage context information, leading to sub-optimal 3D Gaussian representations.

To this end, we propose a context-aware adaptive density control that leverages part attribute and history gradient information. Each Gaussian inherently possesses attributes associated with different body parts. These body parts vary in spatial size and characteristics. For instance, the hand is much smaller in size, compared to the body, exhibiting fine-grained textures and many degrees of freedom. These characteristics also lead to inherent differences in the Gaussian positional gradient changes across different body parts during the Gaussian optimization process. Furthermore, a continuous increase in the Gaussian positional gradient indicates that Gaussians need to be densified.

Specifically, we first initialize the 3D Gaussians with the vertices of the SMPL-X model. Since SMPL-X has a predefined topology, this initialization provides each Gaussian with the attribute information U on which part it belongs, *e.g.* body, hands or face. After that, considering attributes and gradient history information, the densification threshold for the i-th Gaussian in a certain attribute U is formulated as follows:

$$\epsilon_i = e + \frac{\lambda_t}{R}\left(\sum_{k=t-R}^{t} \nabla_{i,k} - \sum_{k=t-2R}^{t-R} \nabla_{i,k}\right), \tag{14}$$

where e is a constant, R represents the densification interval, and ∇_k denotes the position gradient of i-th Gaussian at the timestamp k. Note that e and λ_t have different values for different attributes. For the pruning strategy, we remove the points that are far away from the SMPL-X template vertices.

3.4 Objective Functions

The whole framework is optimized under the objective functions as follows:

$$\mathcal{L} = \mathcal{L}_c + \lambda_m \mathcal{L}_m + \lambda_s \mathcal{L}_{SSIM} + \lambda_l \mathcal{L}_{LPIPS}, \tag{15}$$

where λ_m, λ_s, and λ_l are loss weighting terms. The mask loss $\mathcal{L}_m$ calculates the consistency between accumulated volume density and the estimated mask. The SSIM [40] loss $\mathcal{L}_{SSIM}$ is adopted to improve the structural similarity between rendered image and input image. The LPIPS [47] loss $\mathcal{L}_{LPIPS}$ focuses on the perceptual quality of the rendered image.

Confidence-aware loss $\mathcal{L}_c$. It is inevitable that any training video will introduce some form of noise (*e.g.*, misalignment, motion blur), which will also affect the avatar optimization procedure. To address some of the effects of the noise, we introduce a feedback module to decide which pixels should be taken into consideration with higher or lower weight during training. This module takes as input the rendered image I_r and rendered depth D_r, and predicts a score for each pixel which represents the confidence value. More specifically:

$$C = \mu + exp(E(I_r, D_r)), \tag{16}$$

where μ is a constant. Then the confidence serves as an adaptive weighting factor on the per-pixel consistency. Eventually, we formulate our confidence-aware loss as follows:

$$\mathcal{L}_c = C \odot |I_r - I|_1. \tag{17}$$

4 Experiments

In this section, we first introduce our experimental setup, including datasets, implementation details and evaluation metrics. Then, we make comparisons with baseline methods both quantitatively and qualitatively. Finally, we perform ablation studies on the most important components of our approach.

4.1 Experimental Setup

Datasets. The experiments are conducted on two datasets, **XHumans** [35] and our collected **UPB** dataset. **XHumans** is a dataset captured in a controlled environment. It provides images with resolution of 1200 $\times$ 800, along with well-aligned SMPL-X meshes. There are 6 identities (3 male and 3 female) for evaluation. **UPB** consists of sign language videos from the web, which usually contain complicated hand gestures. The resolution of the videos is 1920 $\times$ 1080 and they do not contain any ground truth SMPL-X annotations. UPB includes 4 identities (2 male and 2 female).

Table 1: Comparison with three expressive avatar baselines, *i.e.*, GART + SMPLX, Splatting + SMPL-X and GauHuman + SMPLX, on the XHumans and UPB dataset. N-GS denotes the number of Gaussians. ↑ and ↓ represent the higher the better, and the lower the better, respectively.

Method	N-GS	Full			Hand			Face		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Controlled setting: XHumans dataset										
3DGS [19] + SMPLX	19,458	28.88	0.9609	44.93	25.28	0.9189	91.37	25.91	0.9087	101.04
GART [22] + SMPLX	89,571	27.73	0.9553	50.32	25.42	0.9151	99.53	25.86	0.9013	105.06
Splatting [34] + SMPLX	103,193	29.33	0.9606	44.39	26.19	0.9264	78.53	26.47	0.9103	92.51
GauHuman [12] + SMPLX	17,134	29.16	0.9623	41.16	25.69	0.9225	88.16	26.27	0.9124	93.35
EVA	19,993	**29.67**	**0.9632**	**33.05**	**26.27**	**0.9279**	**72.95**	**26.56**	**0.9157**	**72.30**
Real-world setting: UPB dataset										
3DGS [19] + SMPLX	21,008	25.31	0.9469	90.80	24.89	0.9425	66.19	24.57	0.9072	136.53
GART [22] + SMPLX	90,676	26.20	0.9511	78.90	25.25	0.9411	61.44	26.62	0.9253	93.28
Splatting [34] + SMPLX	257,811	25.13	0.9355	96.16	24.20	0.9298	70.91	24.48	0.8962	127.63
GauHuman [12] + SMPLX	12,372	25.17	0.9455	84.87	24.67	0.9418	67.61	24.33	0.9035	113.13
EVA	20,829	**26.78**	**0.9519**	**65.07**	**27.00**	**0.9524**	**45.90**	**26.85**	**0.9298**	**65.90**

Implementation details. Our framework is implemented on PyTorch and all experiments are performed on NVIDIA A5000. The hyperparameter λ_m, λ_s and λ_l are set to 0.1, 0.01 and 0.04, respectively. We use SGHM [4] to extract the human mask. The 3D Gaussian optimization lasts for 2,000 iterations with the densification performed between iteration 400 and 1,000. For other parameters, we follow the original settings of [19]. Our method can render a 1080p image with an inference speed of 361.02 fps on one RTX A5000. Since our method is reconstruction-based, our main requirement is that the areas of interest should be captured in the video. Note that we use the same input RGB frames for all methods to ensure a fair comparison. More details are presented in the Appendices.

Evaluation metrics. We evaluate the avatar quality through the rendered views following previous works [22, 12]. We adopt the widely-used Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [40], and Learned Perceptual Similarity (LPIPS) [47] to evaluate the rendered images. For a more fine-grained evaluation, we report the above metrics separately for the full body, for the hands region and for the face region.

4.2 Comparison with baselines

The baselines we use for comparison are adopted from related methods with a few modifications. 3DGS + SMPL-X is modified from 3DGS [19]. We add our articulated human modeling mechanism to make it fit the task requirement. GART and GauHuman are originally designed to model body-level avatars. They are animated via the SMPL parametric model, which lacks the capability of modeling hand articulation and facial expressions. Therefore, we replace the driven signal with the SMPL-X model and denote them as GART + SMPL-X and GauHuman + SMPL-X, respectively. Splatting selects the triangle mesh as the driving signal and we utilize SMPL-X mesh in Splatting + SMPL-X. Among these baselines, GauHuman is closer to our approach. Our differences are mainly reflected in the adaptive density control strategy, objective functions and the SMPL-X alignment for the real-world videos, which are also consistent with our main technical contributions.

As shown in Table 1, we conduct experiments on the XHumans and UPB datasets to evaluate the effectiveness of our method. XHumans is captured in a controlled setting with accurate SMPL-X ground truth, while UPB focuses on real-world video without pose information. Since it is hard to get accurate SMPL-X annotation for real videos, UPB is more challenging and more representative of learning a human avatar in the wild. We observe that our method achieves state-of-the-art performance on these two datasets. We note that we do not apply our designed alignment module in XHumans, since it already contains SMPL-X ground truth. Our method achieves 19.7%, 17.3% and 22.5% relative LPIPS gain on the full, hand and face regions, respectively. For the real-world UPB dataset, our performance gain is much larger, achieving over 25% relative LPIPS gain on the hand region. This indicates that EVA, unlike previous work, handles well the challenges of real-world videos. The qualitative comparison in Figure 3 also validates the effectiveness of our method.

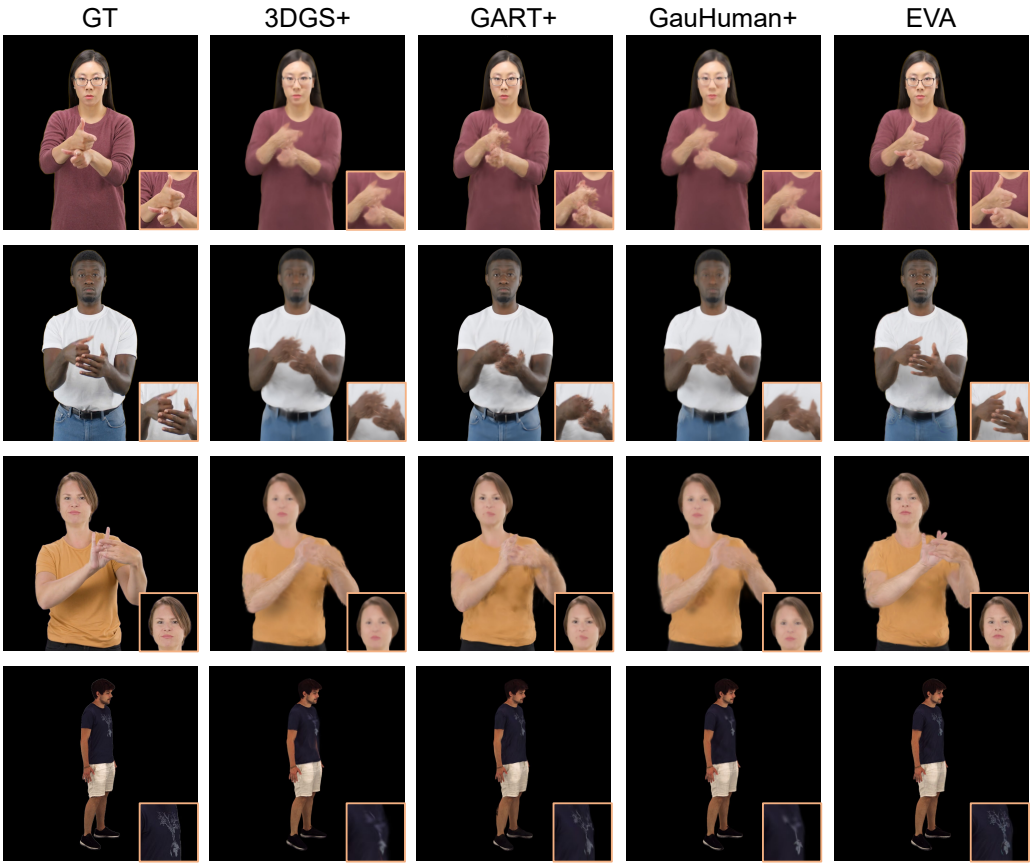

Figure 3: **Qualitative comparison with baselines.** We compare with 3DGS [19] + SMPL-X, GART [22] + SMPL-X, and GauHuman [12] + SMPL-X. First three rows are from UPB and last row is from the XHumans dataset. EVA exhibits the best visual quality. See the zoomed-in results in the box for comparison of the fine-grained details.

4.3 Ablation Studies

We perform ablation experiments to highlight the important components of EVA, *i.e.*, context-aware adaptive density control, confidence-aware loss and in-the-wild SMPL-X alignment module.

Effectiveness of context-aware adaptive density control (CADC) and confidence-aware loss (CL). These two components are designed for better optimization of *Gaussian Avatar Modeling*. As shown in Table 2, "w/o CADC" means that we replace it with the original density control [19], while "w/o CL" denotes removing the confidence weighting term in the RGB loss calculation. We observe that these two components both improve performance. More specifically, CADC improves performance for the metrics of all regions. Meanwhile, we observed that CL brings another benefit of a more compact representation (e.g., reducing the number of Gaussians from 21,038 to 19,993).

Effectiveness of in-the-wild SMPL-X alignment. We use our SMPL-X alignment module for real-world videos without accurate SMPL-X ground truth, so we conduct the corresponding ablation study on UPB dataset. As shown in Table 2, "w/o Align" means that we directly utilize the SMPL-X mesh extracted from current SOTA estimation method [1]. Notably, our proposed alignment module brings notable performance gains on all metrics of all regions. It also demonstrates the importance of the SMPL-X pose quality for final human avatar modeling. Without accurate SMPL-X alignment, the following Gaussian model will need to deal with inconsistent textures for a specific region across frames, leading to lower avatar quality. Furthermore, we visualize the SMPL-X alignments in Figure 4 and present qualitative comparison with state-of-the-art estimation methods in Figure 5, including fitting-based SMPLify-X and regression-based SMPLer-X. Visible improvements are clearly observed specifically for the hands.

Table 2: Effect of context-aware adaptive density control (CADC), confidence-aware loss (CL) and SMPL-X alignment. ↑ and ↓ represent "higher the better", and "lower the better", respectively.

Method	Full			Hand			Face		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
XHumans dataset									
w/o CADC	28.92	0.9611	35.14	25.23	0.9175	84.00	26.08	0.9080	80.76
w/o CL	29.63	**0.9634**	34.08	26.27	0.9279	74.02	**26.58**	**0.9166**	73.19
EVA	**29.66**	0.9632	**33.05**	**26.27**	**0.9279**	**72.95**	26.56	0.9156	**72.30**
UPB dataset									
w/o Align	25.02	0.9435	73.82	24.64	0.9396	63.64	24.27	0.9009	93.56
w/o CADC	26.53	0.9504	68.76	26.43	0.9494	50.59	26.45	0.9265	71.17
w/o CL	26.74	0.9507	67.90	26.74	0.9507	48.54	26.69	0.9293	69.60
EVA	**26.72**	**0.9519**	**65.37**	**26.90**	**0.9523**	**46.11**	**26.75**	**0.9298**	**66.41**

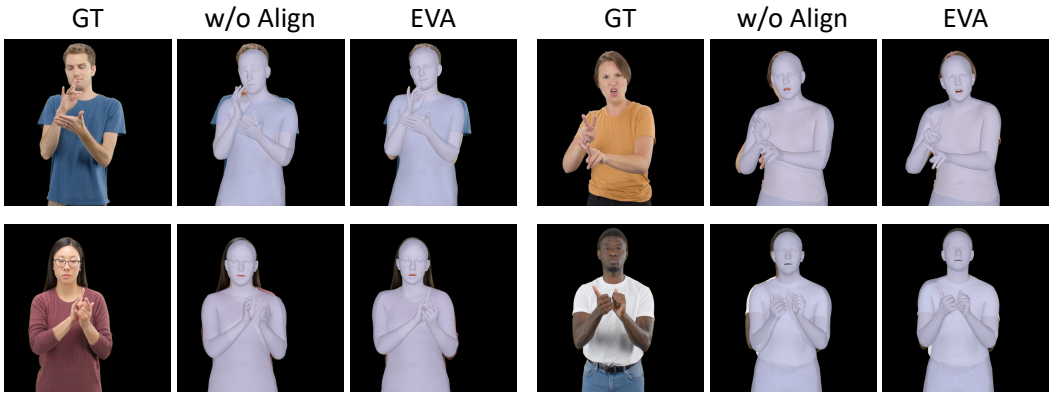

Figure 4: **Effect of our SMPL-X alignment module**. We can estimate a SMPL-X mesh that aligns well with the RGB frame, especially for the fine-grained hand regions.

4.4 More Discussion

In summary, we apply EVA on a variety of monocular RGB videos, including both controlled captures and videos collected from the Internet. The visual data contain large body motions (e.g., playing basketball, weight lifting, dancing), along with fine-grained motions (e.g. finger counting, using tools, sign language). EVA can handles well self-occlusions, which usually exist in every frame, especially for the hand areas. This can be attributed to our proposed SMPL-X fitting method which provides reliable correspondences from the pixels to the canonical space. Besides, we expect that EVA could also handle low lighting conditions (as long as the lighting is not changing significantly), since our SMPL-X fitting method should be robust to low lighting conditions. Given accurate SMPL-X estimates, we expect the avatar learning stage will perform well.

Limitations. The focus of this work is to capture expressive human avatars, by designing a pipeline that can take raw web videos as the input. Besides capturing the expressive details, other factors are challenging, yet worth investigating to further improve the quality of the avatar, *e.g.* non-rigid elements like clothes and hair. It is also worth extending our avatar reconstruction to more challenging input sources, *e.g.* occlusions or few-shot scenarios.

Failure Cases. We show some representative examples of our failure cases in Figure 6. For the fine-grained expressive areas, the failure cases happen when the video includes hand interactions. The failure is mainly caused by incorrect SMPL-X reconstruction (e.g., the estimated SMPL-X model that drives the reconstruction has implausible interpenetration between hands). Another failure case is the existence of floaters in the reconstruction which is a common issue of 3DGS modeling.

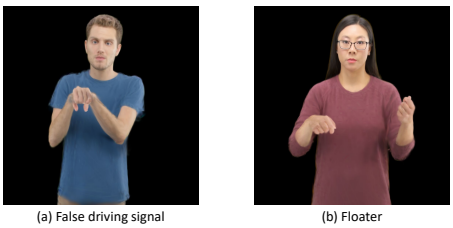

Figure 6: **Illustration of failure cases.**

Figure 5: **Comparison among methods for SMPL-X alignment.** We compare our results with a regression-based method (SMPLer-X), and a fitting-based method (SMPLify-X).

Broader Impact. In this work, our proposed framework helps push the boundaries of what is possible with human avatar modeling, allowing for more expressive results than previously achievable. However, with these advancements come certain risks, particularly concerning the potential misuse of the technology for deceptive practices or harassment. By advancing SOTA, our work makes these potential risks more pressing, as the barrier to creating such deceptive or harmful content is lowered. This highlights the need for careful consideration of the ethical implications and the development of strategies to mitigate these risks.

5 Conclusion

In this work, we present EVA, a drivable expressive human avatar learned from a real-world monocular RGB video. EVA is built on 3D Gaussians, in coordination with the human prior introduced by SMPL-X. To deal with the challenges brought by expressiveness, we first utilize a reconstruction module to solve the misalignment between SMPL-X model and RGB frames. During the optimization of the 3D Gaussians, we propose context-aware adaptive density control, which leverages attribute and historical gradient information to accommodate the varied granularity across body parts. A feedback mechanism is jointly designed to further guide learning. Extensive experiments on two benchmarks demonstrate that our method outperforms baselines both quantitatively and qualitatively, especially on the fine-grained hand and facial details.

Acknowledgments. This project was supported by LUCI program under the Basic Research Office and partially supported by ARL grants W911NF20-2-0158 and W911NF-21-2-0104 under the cooperative A2I2 program. It was also in part supported by NSF AI Institute for Foundations of Machine Learning (IFML). GP has received a research gift from Google.

References

[1] Zhongang Cai, Wanqi Yin, Ailing Zeng, Chen Wei, Qingping Sun, Wang Yanjun, Hui En Pang, Haiyi Mei, Mingyuan Zhang, Lei Zhang, et al. Smpler-x: Scaling up expressive human pose and shape estimation. *NeurIPS*, pages 1–15, 2024.

[2] Xu Chen, Tianjian Jiang, Jie Song, Max Rietmann, Andreas Geiger, Michael J. Black, and Otmar Hilliges. Fast-snarf: A fast deformer for articulated neural fields. *IEEE TPAMI*, 2023.

[3] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *ICCV*, 2021.

[4] Xiangguang Chen, Ye Zhu, Yu Li, Bingtao Fu, Lei Sun, Ying Shan, and Shan Liu. Robust human matting via semantic guidance. In *ACCV*, pages 2984–2999, 2022.

[5] Zhaoxi Chen, Fangzhou Hong, Haiyi Mei, Guangcong Wang, Lei Yang, and Ziwei Liu. Primdiffusion: Volumetric primitives diffusion for 3d human generation. In *NeurIPS*, pages 1–14, 2023.

[6] Junting Dong, Qi Fang, Yudong Guo, Sida Peng, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Totalselfscan: Learning full-body avatars from self-portrait videos of faces, hands, and bodies. In *NeurIPS*, pages 13654–13667, 2022.

[7] Maria-Paola Forte, Peter Kulits, Chun-Hao P Huang, Vasileios Choutas, Dimitrios Tzionas, Katherine J Kuchenbecker, and Michael J Black. Reconstructing signing avatars from video using linguistic priors. In *CVPR*, pages 12791–12801, 2023.

[8] Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Chen Qian, Chen Change Loy, Wayne Wu, and Ziwei Liu. Stylegan-human: A data-centric odyssey of human generation. In *ECCV*, pages 1–19, 2022.

[9] Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Wayne Wu, and Ziwei Liu. Unitedhuman: Harnessing multi-source data for high-resolution human generation. In *ICCV*, pages 7301–7311, 2023.

[10] Chen Geng, Sida Peng, Zhen Xu, Hujun Bao, and Xiaowei Zhou. Learning neural volumetric representations of dynamic humans in minutes. In *CVPR*, pages 8759–8770, 2023.

[11] 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*, pages 1–13, 2023.

[12] Shoukang Hu and Ziwei Liu. Gauhuman: Articulated gaussian splatting for real-time 3d human rendering. In *CVPR*, pages 1–16, 2024.

[13] Rohit Jena, Ganesh Subramanian Iyer, Siddharth Choudhary, Brandon Smith, Pratik Chaudhari, and James Gee. Splatarmor: Articulated gaussian splatting for animatable humans from monocular rgb videos. *arXiv*, pages 1–11, 2023.

[14] Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. In *CVPR*, pages 16922–16932, 2023.

[15] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video. In *ECCV*, pages 402–418, 2022.

[16] Yuheng Jiang, Zhehao Shen, Penghao Wang, Zhuo Su, Yu Hong, Yingliang Zhang, Jingyi Yu, and Lan Xu. Hifi4g: High-fidelity human performance rendering via compact gaussian splatting. In *CVPR*, pages 1–12, 2024.

[17] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *CVPR*, pages 8320–8329, 2018.

[18] HyunJun Jung, Nikolas Brasch, Jifei Song, Eduardo Perez-Pellitero, Yiren Zhou, Zhihao Li, Nassir Navab, and Benjamin Busam. Deformable 3d gaussian splatting for animatable human avatars. *arXiv*, pages 1–15, 2023.

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

[20] Muhammed Kocabas, Jen-Hao Rick Chang, James Gabriel, Oncel Tuzel, and Anurag Ranjan. Hugs: Human gaussian splats. In *CVPR*, pages 1–16, 2023.

[21] Youngjoong Kwon, Lingjie Liu, Henry Fuchs, Marc Habermann, and Christian Theobalt. Deliffas: Deformable light fields for fast avatar synthesis. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *NeurIPS*, pages 40944–40962, 2023.

[22] Jiahui Lei, Yufu Wang, Georgios Pavlakos, Lingjie Liu, and Kostas Daniilidis. Gart: Gaussian articulated template models. *arXiv*, pages 1–13, 2023.

[23] Mingwei Li, Jiachen Tao, Zongxin Yang, and Yi Yang. Human101: Training 100+ fps human gaussians in 100s from 1 view. *arXiv*, pages 1–20, 2023.

[24] Xinqi Liu, Chenming Wu, Jialun Liu, Xing Liu, Chen Zhao, Haocheng Feng, Errui Ding, and Jingdong Wang. Gva: Reconstructing vivid 3d gaussian avatars from monocular videos. *arxiv*, pages 1–13, 2024.

[25] Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason Saragih. Mixture of volumetric primitives for efficient neural rendering. *ACM TOG*, 40(4):1–13, 2021.

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

[27] 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*, pages 405–421, 2020.

[28] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, pages 10975–10985, 2019.

[29] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, pages 10975–10985, 2019.

[30] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *CVPR*, pages 1–11, 2023.

[31] 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.

[32] Zhiyin Qian, Shaofei Wang, Marko Mihajlovic, Andreas Geiger, and Siyu Tang. 3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting. In *CVPR*, pages 1–19, 2024.

[33] Edoardo Remelli, Timur Bagautdinov, Shunsuke Saito, Chenglei Wu, Tomas Simon, Shih-En Wei, Kaiwen Guo, Zhe Cao, Fabian Prada, Jason Saragih, et al. Drivable volumetric avatars using texel-aligned features. In *SIGGRAPH*, pages 1–9, 2022.

[34] 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 *CVPR*, pages 1606–1616, 2024.

[35] Kaiyue Shen, Chen Guo, Manuel Kaufmann, Juan Jose Zarate, Julien Valentin, Jie Song, and Otmar Hilliges. X-avatar: Expressive human avatars. In *CVPR*, pages 16911–16921, 2023.

[36] Geman Stuart and E. McClure Donald. Statistical methods for tomographic image reconstruction. *Bulletin of the International Statistical Institute*, 52:5–21, 1987.

[37] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. In *NeurIPS*, pages 12278–12291, 2021.

[38] David Svitov, Pietro Morerio, Lourdes Agapito, and Alessio Del Bue. Haha: Highly articulated gaussian human avatars with textured mesh prior. *arXiv*, pages 1–23, 2024.

[39] Shaofei Wang, Katja Schwarz, Andreas Geiger, and Siyu Tang. Arah: Animatable volume rendering of articulated human sdfs. In *ECCV*, pages 1–19, 2022.

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13:600–612, 2004.

[41] 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 *CVPR*, pages 16210–16220, 2022.

[42] Hongyi Xu, Thiemo Alldieck, and Cristian Sminchisescu. H-nerf: Neural radiance fields for rendering and temporal reconstruction of humans in motion. *NeurIPS*, 34:14955–14966, 2021.

[43] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *CVPR*, pages 6184–6193, 2020.

[44] Zhongcong Xu, Jianfeng Zhang, Junhao Liew, Jiashi Feng, and Mike Zheng Shou. Xagen: 3d expressive human avatars generation. In *NeurIPS*, pages 34852–34865, 2023.

[45] Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *ICCV*, pages 4210–4220, 2023.

[46] Zhengming Yu, Wei Cheng, Xian Liu, Wayne Wu, and Kwan-Yee Lin. Monohuman: Animatable human neural field from monocular video. In *CVPR*, pages 16943–16953, 2023.

[47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018.

[48] Zechuan Zhang, Li Sun, Zongxin Yang, Ling Chen, and Yi Yang. Global-correlated 3d-decoupling transformer for clothed avatar reconstruction. *NeurIPS*, pages 7818–7830, 2023.

[49] 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 *CVPR*, pages 7743–7753, 2022.

[50] Zerong Zheng, Xiaochen Zhao, Hongwen Zhang, Boning Liu, and Yebin Liu. Avatarrex: Real-time expressive full-body avatars. *ACM TOG*, 42:1–19, 2023.

[51] Wojciech Zielonka, Timur Bagautdinov, Shunsuke Saito, Michael Zollhöfer, Justus Thies, and Javier Romero. Drivable 3d gaussian avatars. *arXiv*, pages 1–11, 2023.

Appendices

This technical appendices provides more details which are not included in the main paper due to space limitations. For more visualization results, please refer to the project webpage.

Training/Testing split. For each identity in XHumans dataset, one video is selected as the training split, where the other videos are marked as testing. During training, we utilize all of the frames (150 frames). We sample 20 frames for each testing video with the sampling rate of 5. For UPB dataset, we uniformly sample the frames with the interval as 1 to split the training and testing frames. The number of training and testing frames are both 140.

Additional implementation details. μ is set to 1. λ_t is set as -9.0, -4.5 and -6.3 for body, hand, and face parts, respectively. We set e for body, hand, and face parts as 2e-4, 1e-4 and 1.4e-4, respectively. The feedback module $E(\cdot)$ consists of two 2D convolutional networks. For SMPL-X alignment, we utilize the L-BFGS optimizer with the Wolfe line search. The optimization contains three stages with different loss weighting factors. The first stage is designed to initialize the human body embedding to the initial estimation from the method [1]. Then the second stage mainly aims to get better spatial hand relationship leveraging the prediction from the method [30]. The third stage performs fine-tuning with more emphasize on the fine-grained hand and face regions.

Comparison baselines. The baselines we use for comparison are adopted from related methods with a few modifications. 3DGS + SMPL-X is modified from 3DGS [19]. Since the original 3DGS does not have the capability of animation, we add the same articulated human modeling in Section 3.1 as ours. For fair comparison, it also utilizes the same optimization schedule as ours, which lasts 2,000 iterations with the densification performed between 400 and 1,000 iterations. GART and GauHuman are originally designed to model body-level avatars. They are animated via the SMPL parametric model, which lacks the capability of modeling hand articulation and facial expressions. Therefore, we replace the driven signal with the SMPL-X model and denote them as GART + SMPL-X and GauHuman + SMPL-X, respectively. Since Splatting is embedded on a triangle mesh, we utilize the SMPL-X human mesh as its driving signal and denote it as Splatting + SMPL-X. We do not modify the optimization schedules of these three methods.

Future works. We outline the potential future works as follows,

- Modeling capability on non-rigid elements, such as loose cloth (dress). Modeling cloth on the avatar has been a challenging topic, even separately studied in several works. A potential solution could be parameterizing the cloth deformation or adding more prior on cloth type, so as to provide more driving signals.
- Generalizable human avatar from monocular RGB video. Current methods in this topic mostly need per-subject optimization, which needs to be re-trained to any new given subjects. It is worth exploring if we could get an avatar from a monocular RGB video of any given subject with a single feed-forward pass.
- Robustness. It is worth exploring if we could build the human avatar well with more limited source inputs, e.g. a few images with even mutual occlusion, a single image, etc.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have clearly stated the contributions and scope in Abstract and Introduction.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We describe the limitations in the Section 4.4.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper does not include theoretical results.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have presented the information in Section 4.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We will release our code after our paper gets accepted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training and test details are specified in Section 4.1.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We use the fixed seed for easy reproducing.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the type of resources used in the experiments in Section 4.1.

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The research conducted in the paper conform the NeurIPS Code of Ethics.

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discusses the broader impact in Section 4.4.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The released model does not have a high risk for misuse.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite corresponding papers for the asserts we use in Section 4.1.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.