

# CanonicalFusion: Generating Drivable 3D Human Avatars from Multiple Images

Jisu Shin<sup>1</sup>, Junmyeong Lee<sup>1</sup>\*, Seongmin Lee<sup>1</sup>, Min-Gyu Park<sup>2,3</sup>,  
Ju-Mi Kang<sup>2</sup>, Ju Hong Yoon<sup>2,3</sup>, and Hae-Gon Jeon<sup>1</sup>\*\*

<sup>1</sup> GIST AI Graduate School

<sup>2</sup> Korea Electronics Technology Institute (KETI)

<sup>3</sup> polygom

**Abstract.** We present a novel framework for reconstructing animatable human avatars from multiple images, termed CanonicalFusion. Our central concept involves integrating individual reconstruction results into the canonical space. To be specific, we first predict Linear Blend Skinning (LBS) weight maps and depth maps using a shared-encoder-dual-decoder network, enabling direct canonicalization of the 3D mesh from the predicted depth maps. Here, instead of predicting high-dimensional skinning weights, we infer compressed skinning weights, i.e., 3-dimensional vector, with the aid of pre-trained MLP networks. We also introduce a forward skinning-based differentiable rendering scheme to merge the reconstructed results from multiple images. This scheme refines the initial mesh by reposing the canonical mesh via the forward skinning and by minimizing photometric and geometric errors between the rendered and the predicted results. Our optimization scheme considers the position and color of vertices as well as the joint angles for each image, thereby mitigating the negative effects of pose errors. We conduct extensive experiments to demonstrate the effectiveness of our method and compare our CanonicalFusion with state-of-the-art methods. Our source codes are available at <https://github.com/jsshin98/CanonicalFusion>.

**Keywords:** Drivable 3D Avatar · Canonical Fusion · Forward Skinning-based Differentiable Rendering

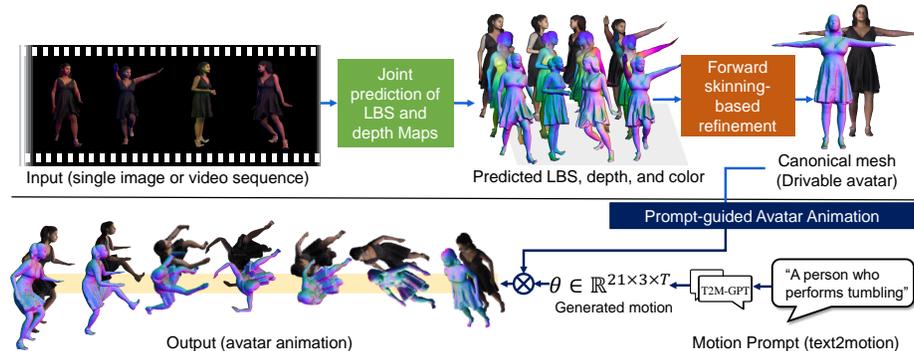
## 1 Introduction

Generating human avatars from images has gained significant interest, which is one of key technologies for various applications to metaverse and AR/VR. Traditionally, this process required huge manual efforts of skilled artists with expensive equipments in controlled environments. The advances in neural networks have simplified this process, anticipating automatic avatar creation shortly.

---

\* Currently at POSTECH

\*\* Corresponding author



**Fig. 1:** Our framework, CanonicalFusion, generates a drivable avatar from multiple images.

Existing approaches address the human reconstruction problem through various methods, including training deep implicit functions [1, 2, 37, 42, 47], explicitly predicting depth maps [7, 41, 46], or leveraging hybrid representations [13] such as deformable tetrahedron [39]. Deep implicit functions compute signed distances within a pre-defined volume where the iso-surface can be extracted from the signed distance volume or the novel view can be rendered via volume rendering. The explicit approaches infer explicit geometric information, such as depth maps or surface normal, allowing for higher resolution handling than the implicit approach. In addition, the reconstructed mesh or point cloud can be refined using predicted results, notably through, differentiable rendering. DMTet [39] and flexicube [40] leverage both explicit and implicit representations as hybrid approaches. They apply both volumetric and differentiable rendering to enhance the quality of 3D models. Thanks to the huge progress, researchers have moved their eyes onto animating reconstruction as in the real world.

To bring life into a virtual human, it is essential to estimate the position of joints and skinning weights for each vertex as well as the shape and color of a human model. The majority of studies employ the skinned template models [25, 30] to take their skinning weights and joints as weak supervision signals. For example, one can fit a template human model to a reconstructed mesh and assign skinning weights to each vertex using nearest neighbor search. We refer to this process as inverse skinning that assigns skinning weights to the posed mesh, allowing for the acquisition of a canonical mesh by warping each vertex to canonical space. This can serve as a post-processing step for animating human mesh, and the weights can be further refined by using multiple scan models as in SCANimate [36]. Moreover, several studies [43, 48] attempt to directly predict skinning weights in the image space together with depth values. The predicted depth map back-projects pixels into 3D points, and these points are animated using the predicted skinning weights. Here, the dimension of skinning weights aligns with the number of joints and typically exhibits sparsity (*i.e.*, zero values) as only a few joints influence the position of one vertex.

On the other hand, numerous studies have evolved the canonical template mesh while changing the posture, using a single image [4, 9, 11, 12, 14, 24], multiple frames [6, 15, 17, 31, 50, 52, 54], or generated images [19] through diffusion networks. This approach has double-sided edges, which retains the robustness of reconstruction, but often fails to recover loose clothing and hair.

In this work, we focus on generating drivable 3D human avatars from multiple images by jointly predicting depth and Linear Blend Skinning (LBS) maps, followed by a forward skinning-based differentiable rendering. Similar to sandwich-like approaches [5, 41, 46], we initially infer double-sided depth and LBS maps for each image. Here, we utilize a stacked autoencoder MLP model to represent the LBS weights as a low-dimensional vector and decode them back to full-dimensional skinning weights, *i.e.*, 55 dimensions. We can reconstruct the canonical mesh directly from an input image, predicted depth map, and predicted LBS map. After generating the initial canonical mesh, we refine the position and color of vertices through forward skinning-based differentiable rendering. Our method warps the canonical mesh w.r.t. multiple images via forward skinning and renders the normal maps and color images using a differentiable rasterizer. Then, we minimize the color and normal map errors to enhance the geometrical accuracy of the mesh; moreover, we consider the error of the joints, *i.e.*, 3D human pose parameters, as one of the loss functions, which comes from the inconsistency between the initial pose and the reconstructed mesh.

Our pipeline, dubbed *CanonicalFusion*, can animate by using motions generated from texts [51], as shown in Fig. 1. Furthermore, we conducted an extensive ablation study to validate each component of our method with publicly available benchmarks, demonstrating superior performance compared to the previous baseline. We also show the in-the-wild results to validate the practical utility.

## 2 Related Works

We review relevant studies by categorizing them into two groups: clothed human reconstruction and drivable avatar generation from images.

### 2.1 Clothed Human Reconstruction

Existing 3D clothed human reconstruction methods can be roughly divided into two major categories depending on their 3D representation: implicit and explicit. Implicit approaches train a deep implicit function defined for a continuous 3D space, that is capable of representing arbitrary humans. PIFu and PIFuHD [34, 35] are pioneering studies that exploit pixel-aligned image features to predict the occupancy value and color at a queried position. Following studies [4, 8, 9, 14, 22, 24, 47, 48, 53] incorporate geometric priors such as depth, LiDAR, 3D volumetric features, or parametric body models to alleviate the limitations of PIFu. For example, Geo-PIFu [8] employs latent voxel features to generate occupancy volume coarsely. ARCH [14] and ARCH++ [9] utilize the geometry encoder to estimate the 3D body shape of a given subject.

Explicit methods predict explicit geometric representation such as depth maps [5] or point clouds [27] to reconstruct 3D humans. Predicting front and back depth maps of given RGB images and then merging them in a "sandwich-like" manner [5] to form explicit meshes became a popular approach because of its efficiency in predicting 3D human models. To further generate high fidelity 3D clothed humans, several studies [41,46] derive normal maps from depth maps facilitating the reconstruction of loose clothes and challenging postures. Recent approaches incorporate diffusion with text-level guidance [13,19] or latent code inversion from given reference image [45]. With the generated normal maps with text prompts, differentiable rendering [44] is used to optimize the clothed human mesh from skinned body models [25,30].

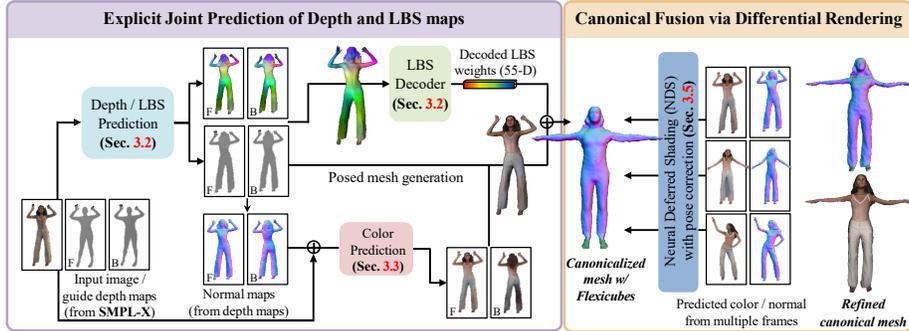
Recently, on the other hand, the boundary between implicit and explicit approaches has disappeared thanks to the hybrid representations [39,40]. The hybrid representations provide differentiable mesh generation with optimized implicit volumes, therefore, both volume rendering and differentiable rendering can be applicable. TeCH [13], for example, exploits deep marching tetradedron [39] to reconstruct human models.

## 2.2 Drivable Human Avatar Generation

Drivable 3D human avatars can be generated from pre-scanned models [36], images [9,12,14,24], and texts [10], where we focus on generating them from images including a single image input. Several studies [9,12,14,24] train animatable implicit neural representations from a single image with the guidance of the SMPL body model. ARCH [14] and ARCH++ [9] assign skinning weights to the canonical surface from the implicit skinning field, initialized with skinning weights from the underlying canonical SMPL model. However, distant points, *e.g.*, loose cloth, from the surface of a canonical template model are not correctly initialized, resulting in poor results. CAR [24] addresses this by using front and back normal maps as geometric cues. They learn the canonical implicit representation by adopting a canonical Signed-Distance Field (SDF) and further refines normal maps to generate high-fidelity clothed avatars.

Another group of studies takes an explicit approach that directly estimates the skinning weight map from a given image. Weng *et al.* [43] propose to learn an explicit warping function that warps normal and skinning maps of a predicted SMPL model into the silhouette of an image and S3 [48] predicts the skinning field along with occupancy and pose fields, enabling reconstruction and animation of clothed meshes simultaneously.

Recent methods take multi-view images [31], videos [4,6,17,50,52,54], or multiple scans [3,23,36,38] as inputs to enhance the quality of drivable avatar. For example, SCANimate [36] ensures cycle consistency between posed scans and canonical scans by utilizing an implicit skinning field to transform posed scans into canonical scans and back under the guidance of the body model. SNARF [3] proposed a differentiable forward skinning field representation, which is trained without a canonical avatar using an iterative root-finding algorithm. However,



**Fig. 2:** An overview of our framework, CanonicalFusion. It takes RGB image and depth maps generated from SMPL-X and estimates dual-sided depth and 3-dimensional LBS weight map. Original skinning weights are decoded from compressed LBS weight maps and used to generate a canonicalized mesh. To further increase the quality, canonical mesh is refined by integrating multiple frames with forward skinning based differentiable rendering.

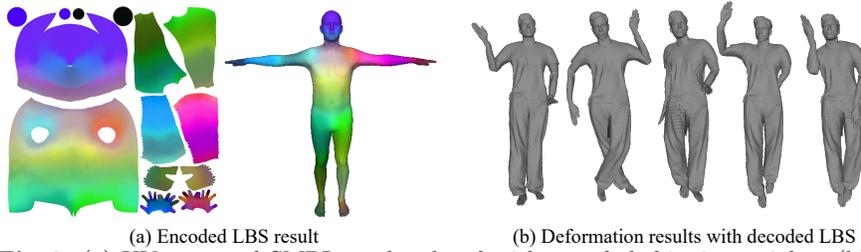
they require detailed 4D scanned data such as CAPE [26] to train a forward skinning network.

### 3 The Proposed Method

Our pipeline consists of two main steps in Fig. 2. First, we jointly predict geometry and skinning weights from a single input image followed by an additional texture prediction network. Afterward, we generate an initial mesh from the predicted results and then canonicalize the mesh to feed the canonical mesh to the next step. The second step involves the forward skinning-based differentiable rendering, refining the canonical mesh while minimizing geometric and photometric errors between the predicted and the rendered images. Note that our framework, CanonicalFusion, has no restrictions on the number of images, viewpoints, and pose variations.

#### 3.1 Preliminaries on Linear Blend skinning

Since the template model, *i.e.*, SMPL-X [30], can encompass various characteristics of a human avatar such as shape, pose, joints, semantic body parts, and skinning weights, we leverage its concept to define skinning weights for clothed humans. The position of vertices changes w.r.t.  $\theta$  and  $\beta$  in which  $\beta$  is an N-dimensional vector representing the shape and  $\theta \in \mathbb{R}^{J \times 3}$  refers to pose parameters, *i.e.* angles for each joint, with  $J$  denoting the number of joints, *i.e.* 55.  $\theta$  can be converted to a set of rotation matrices,  $\mathbf{T} = \{T_i \in \mathbb{R}^{3 \times 3}, i = 1, \dots, J\}$ . Given a mesh  $\mathcal{M} = \{\mathcal{V}, \mathcal{F}\}$ , where  $\mathcal{V}, \mathcal{F}$  denotes vertices and faces, respectively, the position of joints  $\mathbf{J} = \{J_i \in \mathbb{R}^3, i = 1, \dots, J\}$  can also be determined by



**Fig. 3:** (a) UV map and SMPL mesh colored with encoded skinning weights. (b) Re-posed mesh using decoded LBS weight from the pretrained decoder.

regressing designated vertices in  $\mathcal{V}$ . Here,  $i$  of  $T_i$  and  $J_i$  indicates an index of a joint. Once pose parameters are determined,  $\mathcal{V}$  can be moved accordingly,

$$\begin{aligned} \mathbf{x}^p &= \text{LBS}(\mathbf{x}^c, \mathbf{T}) = \left( \sum_i w_i(\mathbf{x}) T_i \right) \mathbf{x}^c, \\ \mathbf{x}^c &= \text{LBS}^{-1}(\mathbf{x}^p, \mathbf{T}) = \left( \sum_i w_i(\mathbf{x}) T_i \right)^{-1} \mathbf{x}^p \end{aligned} \quad (1)$$

where  $\text{LBS}(\cdot)$  warps a point as a linear combination of transformed vertices by  $\mathbf{T}$  and  $\mathbf{x}^c$ ,  $\mathbf{x}^p$  are vertices in canonical and posed space respectively. The inverse of  $\text{LBS}(\cdot)$  transforms a point from the posed to the canonical space.  $w_i(\mathbf{x})$  is a linear blend skinning (LBS) weight of vertex  $\mathbf{x}$  for the  $i^{\text{th}}$  joint, and  $\mathbf{x}^c$ ,  $\mathbf{x}^p$  are vertex in canonical and posed space respectively.

### 3.2 Joint Depth and LBS Prediction

Given an RGB image, we first predict the depth and skinning weight maps for both the front and back side views using a shared-encoder-double-decoder network, denoted as  $\mathcal{S}(\cdot)$ , employing the ATUNet architecture [28],

$$(\hat{\mathbf{D}}, \hat{\mathbf{L}}) = \mathcal{S}(\mathbf{I}, \bar{\mathbf{D}}) \quad (2)$$

where  $\mathbf{I}$  is an input image and  $\bar{\mathbf{D}}$  is dual-sided depth maps generated from the predicted SMPL-X parameters. Here, the network  $\mathcal{S}(\cdot)$  has a shared encoder and two decoder networks in which the first decoder predicts depth maps  $\hat{\mathbf{D}}$  and the second one infers LBS maps  $\hat{\mathbf{L}}$ . Due to the sparsity nature of skinning weights, we embed these skinning weights into a low-dimensional space to effectively compute linear blend skinning (LBS) weights. Then, with the aid of LBS weights and depth maps, it becomes feasible to canonicalize the predicted mesh directly. Lastly, we estimate color images using an additional network, which takes input images and dual-sided normal maps as input which are converted from the predicted depth map.

**Compact representation of skinning weights.** We employ a stacked autoencoder MLP network to compress skinning weights into a 3-dimensional latent space considering the sparsity nature of skinning weights. Here, we denote the low-dimensional LBS weight map as  $\mathbf{L}$  and the full-dimensional LBS weight map as  $\mathbf{L}^F$ . We utilize the SMPL-X model’s skinning weights, which are 55-dimensional, and interpolate skinning weights in the SMPL-X’s UV coordinate via barycentric interpolation, where the resolution of the UV map is  $1024 \times 1024$ . Consequently, the number of training samples to train the autoencoder network is about 800K. With these training samples, we minimize the following objective function:

$$\mathcal{L}_{\mathcal{P}} = \lambda_{L1} L_1(\mathbf{L}^F, \hat{\mathbf{L}}^F) + \lambda_{NZ} L_{\text{nonzero}}(\mathbf{L}^F, \hat{\mathbf{L}}^F) + \lambda_{KL} L_{KL}(\mathbf{L}^F, \hat{\mathbf{L}}^F), \quad (3)$$

where  $L_{\text{nonzero}}$  ensures that the estimated LBS weight map,  $\hat{\mathbf{L}}^F$ , has the same number of nonzero elements as the ground truth weights,  $\mathbf{L}^F$ , after decoding back to the original dimension. We use the approximation for the nonzero count using differentiable relaxation by employing a sum of narrow Gaussian basis functions.  $L_1$  represents the  $L_1$  loss and  $L_{KL}$  is the KL divergence loss. Hyperparameters  $\lambda_{L1}$ ,  $\lambda_{NZ}$  and  $\lambda_{KL}$  are empirically set to 1.0, 0.1 and 0.1, respectively. Since the sum of skinning weights is 1, we apply the softmax operation after the last layer of MLP. Fig. 3(b) shows that our skinning weight can deform the clothed mesh naturally, which is decoded back from the latent space. Fig. 3(a) provides a visualization of the latent code. Note that this LBS compression network is pre-trained before training  $\mathcal{S}(\cdot)$ . During inference, we only utilize the decoder to decode the predicted LBS weights.

**Objective Function.** Our loss function for the initial mesh prediction network  $\mathcal{S}(\cdot)$  is defined as:

$$\begin{aligned} \mathcal{L}_{\mathcal{S}} = & \lambda_1 L_{L2}(\mathbf{D}, \hat{\mathbf{D}}) + \lambda_2 L_{\text{ssim}}(\mathbf{D}, \hat{\mathbf{D}}) + \lambda_3 L_{\text{cos}}(\mathbf{N}, \hat{\mathbf{N}}) \\ & + \lambda_4 L_{L2}(\mathbf{N}, \hat{\mathbf{N}}) + \lambda_5 L_{L2}(\mathbf{L}, \hat{\mathbf{L}}), \end{aligned} \quad (4)$$

where  $\mathbf{D}$  is the ground truth depth map and  $\mathbf{N}$  is the normal map calculated from the depth map. Similarly,  $\mathbf{L}$  denotes the compressed LBS weight map. We utilize  $L_2$ , structural dissimilarity, and cosine loss functions to compare depth, normal, and LBS weight maps. Note depth maps and LBS weight maps are predicted for the front and hidden views. In addition,  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$ ,  $\lambda_4$  and  $\lambda_5$  are set to 0.9, 0.1, 1, 0.15, and 1, respectively.

### 3.3 Texture Prediction

Given the pretrained network  $\mathcal{S}(\cdot)$ , we train the color prediction network  $\mathcal{C}(\cdot)$  to infer shade-removed images for the front and hidden surfaces. We employ a UNet architecture that takes an input front image and the normal maps from the predicted depth map,

$$\hat{\mathbf{C}} = \mathcal{C}(\mathbf{I} \oplus \hat{\mathbf{N}}), \quad (5)$$

where  $\oplus$  indicates a channel-wise concatenation and  $\hat{\mathbf{N}}$  is the predicted normal map. To train  $\mathcal{C}(\cdot)$ , we minimize

$$\mathcal{L}_{\mathcal{C}} = \lambda_{L_2} L_{L_2}(\mathbf{C}, \hat{\mathbf{C}}) + \lambda_{\text{VGG}} L_{\text{VGG}}(\mathbf{C}, \hat{\mathbf{C}}), \quad (6)$$

where  $\hat{\mathbf{C}}$  and  $\mathbf{C}$  are the predicted and ground truth color images for front and back views, respectively.  $\lambda_{L_2}$  and  $\lambda_{\text{VGG}}$  are hyperparameters for the  $L_2$  loss and the perceptual loss which are set to 0.85 and 0.15, respectively.

### 3.4 Canonical Mesh Reconstruction

The reconstructed avatar from  $\hat{\mathbf{D}}$  is defined in the posed space. With the predicted LBS maps  $\hat{\mathbf{L}}$ , it can be warped to its canonical space without an additional optimization procedure to fit the template human model. Here, we define the canonical mesh as  $\mathcal{M}^c = \{\mathcal{V}^c, \mathcal{C}, \mathcal{F}\}$ , consisting of a set of vertices, vertex colors, and faces, which is obtained via an inverse skinning operation from the posed mesh,  $\mathcal{M}^p = \{\mathcal{V}^p, \mathcal{C}, \mathcal{F}\}$ ,

$$\mathcal{V}^c = \{\text{LBS}^{-1}(\mathbf{x}^p, \mathbf{T}) \mid \forall \mathbf{x}^p \in \mathcal{V}^p\}, \quad (7)$$

where  $\text{LBS}^{-1}(\cdot)$  utilizes the skinning weights decoded back from the encoded LBS map,  $\hat{\mathbf{L}}$ . However, this inverse skinning process fails to recover unseen geometry, such as armpit and thigh regions where the reconstructed mesh is not separated. Therefore, if we remove the faces stretched above the pre-defined value, these regions remain empty in the canonical space. To fill these empty regions, we combine the canonicalized mesh and canonical template mesh through signed distance integration as below:

$$\begin{aligned} \mathbf{S}_c(\mathbf{x}^c) &= SDF(\mathbf{x}^c, \partial\mathcal{M}_{\text{SMPL-X}}^c), \\ \mathbf{S}_p(\mathbf{x}^c) &= SDF(\text{LBS}(\mathbf{x}^c, \mathbf{T}), \partial\mathcal{M}^p) = SDF(\mathbf{x}^p, \partial\mathcal{M}^p), \end{aligned} \quad (8)$$

where  $SDF(\cdot)$  computes the signed distance from 3D point  $\mathbf{x}^c$  to the nearest surface of  $\partial\mathcal{M}$ ,  $\partial\mathcal{M}$  represents the surface of mesh  $\mathcal{M}$ , and  $\mathcal{M}_{\text{SMPL-X}}^c$  is the canonical template mesh. To integrate two signed distance functions in the canonical space, we integrate signed distance values within a pre-defined volume, *i.e.*,  $2m^3$ , where the LBS weights are obtained by taking the LBS weights of the nearest vertex of  $\mathcal{M}_{\text{SMPL-X}}^c$ . Then, we warp  $\mathbf{x}^c$  to the posed space with the LBS weights and compute the signed distance in the posed space. Here, direct canonicalization of  $\mathcal{M}^p$  generates holes and artifacts, therefore, we warp  $\mathcal{M}^c$  back to the posed space to sample the signed distance values. Finally, we take the signed distance of  $\mathbf{S}_p(\mathbf{x}^c)$  if there exists canonical vertices  $\mathcal{V}^c$  in the vicinity of  $\mathbf{x}^c$ ,

$$\mathbf{S}(\mathbf{x}^c) = \begin{cases} \mathbf{S}_p(\mathbf{x}^c) & \min|\mathbf{x}^c - \mathcal{V}^c| < \tau, \\ \mathbf{S}_c(\mathbf{x}^c) & \text{otherwise,} \end{cases} \quad (9)$$

where  $\tau$  is empirically set to 5 by observing average discrepancy between ground truth SMPL-X and posed meshes.

Afterward, we obtain a canonical mesh via the marching cubes and re-parametrize it using Flexicubes [40], transforming it into a differentiable, watertight and compact mesh via a gradient-based optimization. Here, a reference frame with the smallest Chamfer distance over the canonical SMPL-X mesh for multiple input images is chosen as the initial canonical mesh. In the rest of the paper, we refer  $\mathcal{M}^c$  as the canonical mesh after applying Flexicubes.

### 3.5 Incorporating Multiple Images into Forward Skinning-based Differentiable Rendering

As the final step, we rasterize the posed mesh deformed from the initial canonical mesh using Neural Deferred Shading (NDS) [44] to minimize the discrepancy between the rendered image and the predicted image from the previous step. In this process, we refine the 3D human pose as well as the shape and color of the canonical mesh. This is crucial because errors in human pose can substantially impair the quality of deformation results generating undesirable artifacts.

Formally, we have depth maps,  $\mathbf{D} = \{\mathbf{D}_1, \dots, \mathbf{D}_n\}$ , color images,  $\mathbf{C} = \{\mathbf{C}_1, \dots, \mathbf{C}_n\}$ , posed meshes,  $\mathcal{M}^P = \{\mathcal{M}_1^P, \dots, \mathcal{M}_n^P\}$  and canonical mesh  $\mathcal{M}^c$  obtained from the previous steps, where  $\mathbf{D}$  and  $\mathbf{C}$  are double-sided depth maps and images and  $n$  is the number of images. Here, the vertex color and faces of the canonical mesh  $\mathcal{M}^c$  (selected in Sec. 3.2) are invariant w.r.t. the change of vertex positions. With the linear skinning weights of the canonical mesh, we can apply forward skinning-based warping w.r.t given transformation as follows:

$$\hat{\mathcal{V}}_i^p = \{\text{LBS}(\mathbf{x}^c, T_i) \mid \forall \mathbf{x}^c \in \mathcal{V}^c\}, \quad (10)$$

where  $\hat{\mathcal{M}}_i^p = \{\mathcal{V}_i^p, \mathcal{C}, \mathcal{F}\}$  is the predicted posed mesh for  $i^{\text{th}}$  image via forward warping and  $T_i$  is the transformation matrix calculated from the 3D human pose. Before the optimization, we downsample the canonical mesh and progressively upsample the mesh while updating the canonical mesh.

To this end, we define our objective function,

$$\begin{aligned} \mathcal{L} = & \lambda_1 L_{\text{laplacian}}(\mathcal{M}^c) + \lambda_2 L_{\text{normal}}^{\text{reg}}(\mathcal{M}^c) \\ & + \sum_{i=1}^n (\lambda_3 L_{\text{normal}}(\mathbf{N}_i^p, \hat{\mathbf{N}}_i^p) + \lambda_4 L_{\text{mask}}(\mathbf{M}_i^p, \hat{\mathbf{M}}_i^p), \\ & + \lambda_5 L_{\text{chamfer}}(\mathcal{M}_i^p, \hat{\mathcal{M}}_i^p) + \lambda_6 L_{\text{color}}(\mathbf{C}_i^p, \hat{\mathbf{C}}_i^p)), \end{aligned} \quad (11)$$

which updates the canonical mesh  $\mathcal{M}^c$  and 3D human pose  $\mathbf{T}$  using differential renderer [21].  $L_{\text{normal}}$  minimizes the  $L_1$  loss between normal maps, and  $L_{\text{mask}}$  and  $L_{\text{color}}$  minimize the discrepancy between binary masks and color images using MSE-based losses, respectively. Note that our refinement procedure can take an arbitrary number of input images as input, which means  $n$  can range from at least 1 to tens of images. To prevent the abrupt change of vertices, we adopt the Laplacian loss function in NDS [44],

$$L_{\text{laplacian}} = \frac{1}{m} \sum_{i=1}^m \|\delta_i\|_2^2, \quad (12)$$

where  $\delta_i = (L\mathcal{V}^c)_i \in \mathbb{R}^3$  are the differential coordinates of vertex  $i$  with the graph Laplacian  $L$ . We also utilize normal consistency loss as:

$$L_{\text{normal}}^{\text{reg}} = \frac{1}{|\mathcal{F}|} \sum_{(i,j) \in \bar{\mathcal{F}}} (1 - n_i \cdot n_j)^2, \quad (13)$$

where  $|\mathcal{F}|$  is the number of the faces,  $\mathcal{F}$ , which is paired with a shared edge and  $n_i \in \mathbb{R}^3$  is the normal of triangle  $i$  in  $\mathcal{M}^c$ . We further minimize the Chamfer distance [18] between  $\hat{\mathcal{M}}_i^p$  and  $\mathcal{M}_i^p$  for the  $i^{\text{th}}$  frame. Due to the potential misalignment between each 3D human pose and the reconstructed mesh, we update the initial human pose parameters to consider pose errors while updating the shape and color of the canonical mesh.

We initially set the hyperparameters  $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6$  to 40, 0.1, 1, 2, 0.0001, and 0 respectively. We employ a progressive update scheme in NDS [44] for  $\mathbf{T}$  and  $\mathcal{V}^c$ , where we upsample  $\mathcal{V}^c$  by a factor of about 4 and multiply  $\lambda_3$  and  $\lambda_4$  by 4 every 500 iterations, up to 2000 iterations. After optimizing  $\mathbf{T}$ , we fix it and proceed to recover  $\mathcal{C}$  and  $\mathcal{V}^c$  by minimizing the  $L_2$  loss between the input color maps,  $\mathbf{C}_i^p$ , and the predicted color maps,  $\hat{\mathbf{C}}_i^p$ , for the  $i^{\text{th}}$  frame, denoted as  $L_{\text{color}}$ . During this step, we set  $\lambda_6$  to 10 to prioritize the color optimization.

## 4 Experimental Results

**Datasets.** We employed popularly used commercial [33] and public [49] datasets. First, we obtained 412 human models from RenderPeople <sup>4</sup>, denoted as RP. The ground truth SMPL parameters for RP dataset were sourced from Agora [29]. Within 412 models, 332 models are mostly standing, whereas 80 models are in T-pose. To augment our training data, we generated 800 human models from the 80 T-posed models while changing body poses using V-poser [30]. In addition, we used the THuman 2.0 [49] dataset for training our model, containing 526 models. For testing, we selected 20 scan models from RP and TH3.0 datasets that were not used for training. The list of tested models is given in the supplementary material.

**Training data generation.** To generate training data, we followed the protocols of prior studies [1, 35] and rendered synthetic images and depth maps as follows. Initially, human models were centered at the origin aligned based on the centroid of each mesh to ensuring them to be positioned at the center of the image. Afterward, we put a perspective camera with 52 degrees of field-of-view that is 3 meters away from the origin. Next, the human models were rotated 0, 1, 2, 3, 4, 5, 10, 15, 20, 30, 40, 60, 90, 120, 150, 180, 210, 240, 270, 300, 330, 340, 345, 350, 355, 356, 357, 358 and 359 degrees, along the vertical axis. To synthesize images, we sampled spherical harmonics mimicking natural illuminations [34] and rendered 20 images with different lighting conditions. Ground truth data includes color albedo maps, depth maps, and LBS maps for both the front and back sides, and silhouette masks. The resolution of the image is  $512 \times 512$ .

<sup>4</sup> <https://renderpeople.com/>

**Table 1:** Quantitative comparison of monocular human reconstruction methods. \* denotes the ground truth SMPL(-X) parameters were used as input. PIFu\* and PaMIR\* are newly devised in ICON [47] to analyze the dependency on the template model. † and ‡ denote our result trained on TH2.0 and combination of TH2.0 and RP, respectively. Average point-to-surface distance, chamfer distance, and average surface normal error are denoted as P2S, CF and NR, respectively. The first(dark color) and second-best(light color) results are marked in color.

	DATASET	RP			2K2K			TH3.0		
		P2S↓	CF↓	NR↓	P2S↓	CF↓	NR↓	P2S↓	CF↓	NR↓
PIFuHD [35]	RP	1.420	1.434	1.401	1.334	1.399	1.266	1.534	1.527	1.628
PIFu* [34]	RP	1.787	1.816	1.822	1.390	1.433	1.764	1.747	1.742	1.820
PaMIR* [53]	RP	1.545	1.616	1.679	1.464	1.536	1.654	1.706	1.722	1.715
ICON* [47]	RP	1.296	1.364	1.447	1.582	1.675	2.024	1.371	1.437	1.666
PHORHUM [1]	CUSTOM	1.512	1.617	1.633	1.638	1.663	2.206	2.073	2.120	2.716
ECON* [46]	TH2.0	2.066	2.290	1.756	2.382	2.769	1.806	1.944	2.139	1.821
2K2K* [7]	TH2.0+RP	1.097	1.195	1.507	1.310	1.265	1.339	1.416	1.542	1.952
TeCH* [13]	N/A	1.489	1.523	2.068	1.428	1.467	1.808	1.721	1.795	2.551
Ours*	RP	0.901	0.978	0.976	0.914	1.007	0.761	1.086	1.189	1.413
Ours*†	TH2.0	1.043	1.126	1.049	1.011	1.102	0.862	1.074	1.169	1.281
Ours*‡	TH2.0+RP	0.886	0.943	0.987	0.897	0.981	0.793	1.072	1.165	1.278

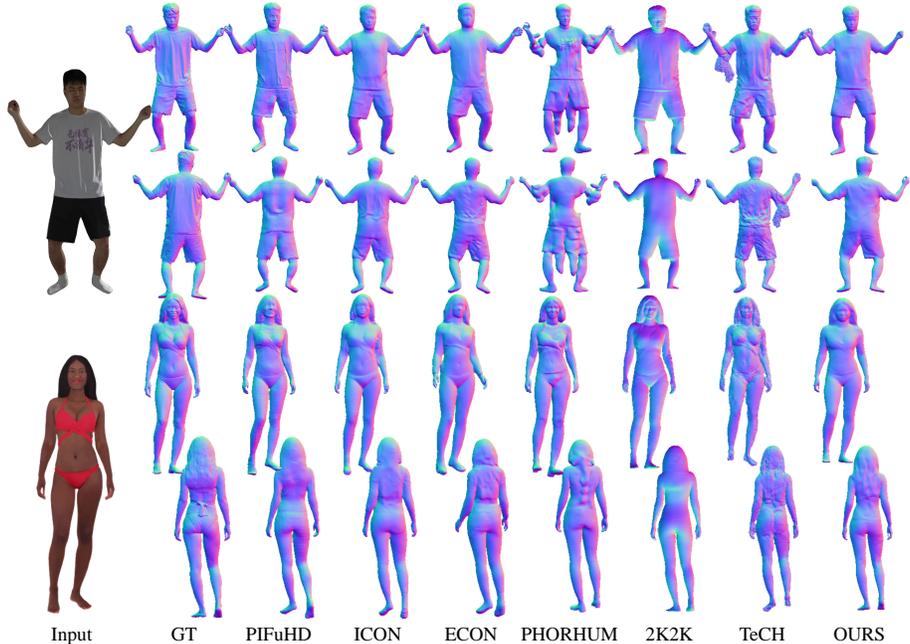
**Training settings.** We used the Adam [20] optimizer with  $\beta_1=0.9$  and  $\beta_2=0.999$ , and set a learning rate to 0.001. We trained our main network (Sec. 3.2) using four NVIDIA RTX 3090 GPUs which took 2 days for training. Our pipeline takes about 11 minutes in total: 3 minutes for the initial canonicalization and 8 minutes for the forward skinning-based differentiable rendering.

#### 4.1 Quantitative and Qualitative Evaluations

We compare our method with existing static avatar reconstruction techniques, in Tab. 1. PHORHUM [1], PaMIR [53], PIFu [34], PIFuHD [35], ICON [47], 2K2K [7], ECON [46], and TeCH [13]. Here, PIFu\*, PaMIR\*, PIFuHD, and ICON were trained by using the RP dataset and ECON [46] used TH dataset, 2K2K [7] used 2K2K and RP datasets, respectively. Since comparison methods are trained with different dataset settings, we train our model in several ways for a fair comparison: train only with RP, train only with TH, and train with both RP and TH. For methods that take SMPL guidance, we evaluate them with the ground truth SMPL-X that we used.

To assess human reconstruction methods, we employed average point-to-surface distance (P2S), Chamfer distance, and average surface normal error. The units for P2S and Chamfer distances are centimeters. For normal errors, we projected normal maps to the front and back views, calculating the average L2 distance between ground truth and predicted normal maps. To enhance readability, the normal error values were multiplied by 100.

Our quantitative analysis presented in Tab. 1 shows that our method performs better than previous methods. Our analysis indicates that leveraging diverse datasets consistently enhances performance, as evidenced by all three sets from our approach achieving the highest scores across all evaluation metrics.



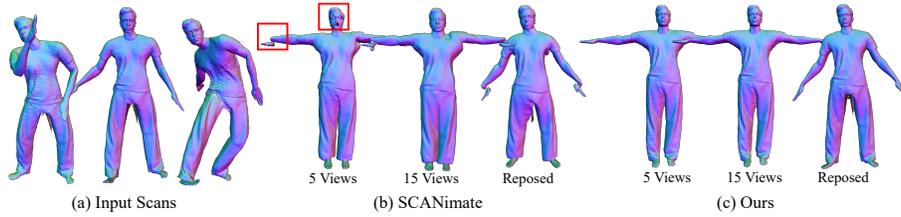
**Fig. 4:** Comparison of normal maps. The first two rows are from the TH3.0 test data, and the latter two are from the RP dataset.

**Table 2:** Quantitative comparison with SCANimate [36] in canonical space. We select 2 avatars from RP rigged T pose dataset and generate 15 posed scans for each avatar, where poses are obtained from CAPE dataset [26]. We differ the training sets to 5 and 15 views, and use 3D metrics for evaluation.

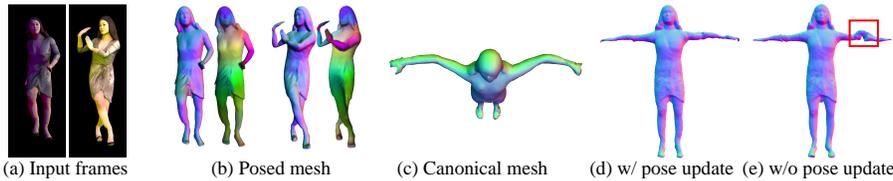
Methods	# of views	SET1			SET2		
		P2S↓	CF↓	NR↓	P2S↓	CF↓	NR↓
SCANimate [36]	5 view	1.362	1.316	2.921	1.076	1.039	2.071
	15 view	1.103	1.124	2.453	0.997	0.971	1.939
Ours	5 view	0.244	0.260	0.482	0.180	0.209	0.395
	15 view	0.199	0.218	0.412	0.149	0.175	0.307

From this, we can argue that explicitly predicting depth maps from a single image with the guidance of SMPL-X remains an effective approach, without necessitating complex techniques.

Fig. 4 shows the qualitative results of ours and comparison results, where ECON, PHORHUM, 2K2K, and our methods recover the details in wrinkles and facial regions. The diversity and quantity of training data are crucial for model-free methods because they directly regress the human model from the image, in that PHORHUM frequently lost arm regions for the TH3.0 dataset, which contains diverse postures. In the case of 2K2K, they could not recover the details of the backsides because part-based normal prediction possibly loses the global context for the backside. TeCH utilizes normal map-based SDS loss [32], which often generates different facial shapes and undesirable artifacts on limbs.



**Fig. 5:** Comparison of results between SCANimate and our method. We used the same SMPL pose parameters for SCANimate and ours. Five and fifteen scans were used to canonicalize the meshes.

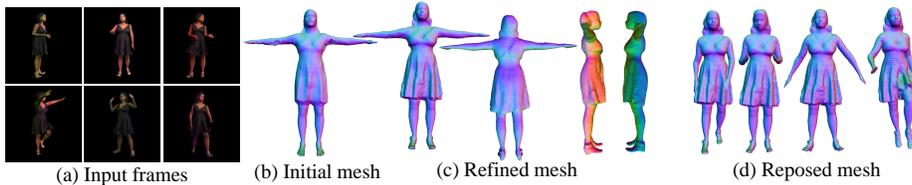


**Fig. 6:** Reconstructed results with and without pose error refinement when the canonical mesh is given with noisy pose.

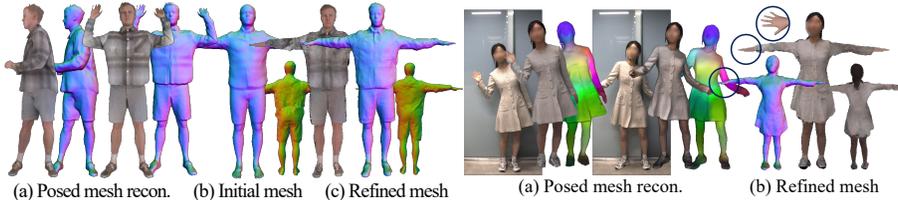
We compare our framework with SCANimate [36], which takes a set of raw scans to generate a canonical mesh. SCANimate relies on the SMPL model to canonicalize the raw scans and to fill missing regions, therefore, it fails in the presence of pose errors. To compare SCANimate with our method, we took the same input as SCANimate, *i.e.*, multiple raw scans, and used the same SMPL body pose parameters. The results are given in Tab. 2 and Fig. 5 that show our method accurately merges posed meshes into a canonical mesh.

## 4.2 Ablation Study

We conduct several experiments to further analyze the characteristics of our proposed method. We demonstrate its robustness under pose variations and its ability to capture loose cloth, leveraging the advantage of the usage of the initial canonicalized model. In general, multiple frames improve the completeness of the human model because invisible regions shrink. Both Fig. 6 and Fig. 7 indicate that our framework effectively integrates multiple results in the canonical space, even with errors in depth, LBS weights and joint positions. We present results for challenging cases in Fig. 6, where a woman shows difficult poses. In this case, the initial canonical mesh exhibits bent arms in Fig. 6(e). Our differentiable rendering scheme, however, alleviates this error because it is able to refine pose errors as well as the shape and color. Furthermore, several studies [13, 19] employ differential rendering to obtain clothed human meshes, starting from a canonical template mesh. This means that it often fails to recover an accurate shape when the topology of the target mesh significantly differs from the template mesh. Contrarily, our initial canonical mesh is in a similar shape to the target mesh because we warp initial posed meshes into canonical space with predicted LBS



**Fig. 7:** Example of reconstructed human wearing a loose cloth. (a) Given an input frame, (b) we initially reconstruct the canonical mesh. (c) With a forward skinning-based differentiable rendering, we integrate multiple frames and generate the refined avatar. (d) We can repose our mesh to unseen poses.



**Fig. 8:** Actors-HQ results.

**Fig. 9:** In-the-wild loose cloth results.

weights, as shown in Fig. 7. Importantly, our method does not require ground-truth canonical meshes, as opposed to ARCH and ARCH++ [9, 14]. We only compare errors w.r.t. the images and normal maps in the posed space.

We also carry out experiments on real-world scenarios, including non-rigid cloth deformations in Fig. 8 and Fig. 9, whose avatars are generated from Actors-HQ dataset [16] (Actor 8) and in-the-wild captured images, respectively. Here, we adopt hand replacement module from ECON [46] using the template model to demonstrate the capability of the hand refinement. Even with a large number of input frames, the blurry results occur when the cloth deformation is large, our approach can generate realistic 3D avatar with only a few frames. The experimental details and more results are provided in the supplementary material.

## 5 Conclusion

We have proposed CanonicalFusion which creates animatable human avatars from images via posed mesh prediction and forward skinning-based differentiable rendering. As a first step, we predict depth maps and LBS weights using a shared-encoder-dual-decoder network, where the LBS weights are compressed into a 3-dimensional latent vector without a loss of accuracy. Furthermore, we proposed a differentiable rendering technique to refine a canonical mesh by minimizing normal, color, and 3D human pose errors for an arbitrary number of images. We experimentally validated the effectiveness of our method from various aspects compared to existing methods. We believe our next direction is to handle the non-rigid deformation of cloth and hairs to simulate more realistic results and to combine our method with generative techniques to diversify our avatars.

**Acknowledgements** This research was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST) / No.2022-0-00566. The development of object media technology based on multiple video sources / RS-2024-00398830. The development of real-time digital human synthesis technology for real-world videos), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(RS-2024-00338439), the Ministry of Trade, Industry and Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) through the International Cooperative R&D program in part (P0019797), and the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land Infrastructure and Transport (Grant RS-2023-00256888).

## References

1. Alldieck, T., Zanfir, M., Sminchisescu, C.: Photorealistic monocular 3d reconstruction of humans wearing clothing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1506–1515 (2022)
2. Cao, Y., Han, K., Wong, K.Y.K.: Sesdf: Self-evolved signed distance field for implicit 3d clothed human reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4647–4657 (2023)
3. Chen, X., Zheng, Y., Black, M.J., Hilliges, O., Geiger, A.: Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11594–11604 (2021)
4. Dong, Z., Guo, C., Song, J., Chen, X., Geiger, A., Hilliges, O.: Pina: Learning a personalized implicit neural avatar from a single rgb-d video sequence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20470–20480 (2022)
5. Gabeur, V., Franco, J.S., Martin, X., Schmid, C., Rogez, G.: Moulding humans: Non-parametric 3d human shape estimation from single images. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2232–2241 (2019)
6. Guo, C., Jiang, T., Chen, X., Song, J., Hilliges, O.: Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12858–12868 (2023)
7. Han, S.H., Park, M.G., Yoon, J.H., Kang, J.M., Park, Y.J., Jeon, H.G.: High-fidelity 3d human digitization from single 2k resolution images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12869–12879 (2023)
8. He, T., Collomosse, J., Jin, H., Soatto, S.: Geo-pifu: Geometry and pixel aligned implicit functions for single-view human reconstruction. *Advances in Neural Information Processing Systems* **33**, 9276–9287 (2020)
9. He, T., Xu, Y., Saito, S., Soatto, S., Tung, T.: Arch++: Animation-ready clothed human reconstruction revisited. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11046–11056 (2021)

10. Hong, F., Zhang, M., Pan, L., Cai, Z., Yang, L., Liu, Z.: Avatarclip: Zero-shot text-driven generation and animation of 3d avatars. arXiv preprint arXiv:2205.08535 (2022)
11. Hu, S., Hong, F., Pan, L., Mei, H., Yang, L., Liu, Z.: Sherf: Generalizable human nerf from a single image. arXiv preprint arXiv:2303.12791 (2023)
12. Huang, Y., Yi, H., Liu, W., Wang, H., Wu, B., Wang, W., Lin, B., Zhang, D., Cai, D.: One-shot implicit animatable avatars with model-based priors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8974–8985 (2023)
13. Huang, Y., Yi, H., Xiu, Y., Liao, T., Tang, J., Cai, D., Thies, J.: Tech: Text-guided reconstruction of lifelike clothed humans. arXiv preprint arXiv:2308.08545 (2023)
14. Huang, Z., Xu, Y., Lassner, C., Li, H., Tung, T.: Arch: Animatable reconstruction of clothed humans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3093–3102 (2020)
15. Iqbal, U., Caliskan, A., Nagano, K., Khamis, S., Molchanov, P., Kautz, J.: Rana: Relightable articulated neural avatars. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23142–23153 (2023)
16. Işık, M., Rünz, M., Georgopoulos, M., Khakhulin, T., Starck, J., Agapito, L., Nießner, M.: Humanrf: High-fidelity neural radiance fields for humans in motion. *ACM Transactions on Graphics (TOG)* **42**(4), 1–12 (2023)
17. Jiang, B., Hong, Y., Bao, H., Zhang, J.: Selfrecon: Self reconstruction your digital avatar from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5605–5615 (2022)
18. Johnson, J., Ravi, N., Reizenstein, J., Novotny, D., Tulsiani, S., Lassner, C., Branson, S.: Accelerating 3d deep learning with pytorch3d. In: SIGGRAPH Asia 2020 Courses, pp. 1–1 (2020)
19. Kim, B., Kwon, P., Lee, K., Lee, M., Han, S., Kim, D., Joo, H.: Chupa: Carving 3d clothed humans from skinned shape priors using 2d diffusion probabilistic models. arXiv preprint arXiv:2305.11870 (2023)
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
21. Laine, S., Hellsten, J., Karras, T., Seol, Y., Lehtinen, J., Aila, T.: Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics (TOG)* **39**(6), 1–14 (2020)
22. Li, Z., Yu, T., Pan, C., Zheng, Z., Liu, Y.: Robust 3d self-portraits in seconds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1344–1353 (2020)
23. Li, Z., Zheng, Z., Zhang, H., Ji, C., Liu, Y.: Avatarcap: Animatable avatar conditioned monocular human volumetric capture. In: European Conference on Computer Vision. pp. 322–341. Springer (2022)
24. Liao, T., Zhang, X., Xiu, Y., Yi, H., Liu, X., Qi, G.J., Zhang, Y., Wang, X., Zhu, X., Lei, Z.: High-fidelity clothed avatar reconstruction from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8662–8672 (2023)
25. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: a skinned multi-person linear model. *Transactions of Graphics* **34**, 1–16 (2015)
26. Ma, Q., Yang, J., Ranjan, A., Pujades, S., Pons-Moll, G., Tang, S., Black, M.J.: Learning to Dress 3D People in Generative Clothing. In: Computer Vision and Pattern Recognition (CVPR) (2020)

27. Ma, Q., Yang, J., Tang, S., Black, M.J.: The power of points for modeling humans in clothing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10974–10984 (2021)
28. Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., et al.: Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999 (2018)
29. Patel, P., Huang, C.H.P., Tesch, J., Hoffmann, D.T., Tripathi, S., Black, M.J.: Agora: Avatars in geography optimized for regression analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13468–13478 (2021)
30. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10975–10985 (2019)
31. Peng, S., Xu, Z., Dong, J., Wang, Q., Zhang, S., Shuai, Q., Bao, H., Zhou, X.: Animatable implicit neural representations for creating realistic avatars from videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
32. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
33. RenderPeople: <https://renderpeople.com/3d-people> (2018)
34. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2304–2314 (2019)
35. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 84–93 (2020)
36. Saito, S., Yang, J., Ma, Q., Black, M.J.: Scanimate: Weakly supervised learning of skinned clothed avatar networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2886–2897 (2021)
37. Shao, R., Zhang, H., Zhang, H., Chen, M., Cao, Y.P., Yu, T., Liu, Y.: Doublefield: Bridging the neural surface and radiance fields for high-fidelity human reconstruction and rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15872–15882 (2022)
38. Shen, K., Guo, C., Kaufmann, M., Zarate, J.J., Valentin, J., Song, J., Hilliges, O.: X-avatar: Expressive human avatars. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16911–16921 (2023)
39. Shen, T., Gao, J., Yin, K., Liu, M.Y., Fidler, S.: Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. Advances in Neural Information Processing Systems **34**, 6087–6101 (2021)
40. Shen, T., Munkberg, J., Hasselgren, J., Yin, K., Wang, Z., Chen, W., Gojcic, Z., Fidler, S., Sharp, N., Gao, J.: Flexible isosurface extraction for gradient-based mesh optimization. ACM Transactions on Graphics (TOG) **42**(4), 1–16 (2023)
41. Smith, D., Loper, M., Hu, X., Mavroidis, P., Romero, J.: Facsimile: Fast and accurate scans from an image in less than a second. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5330–5339 (2019)
42. Wang, J., Yoon, J.S., Wang, T.Y., Singh, K.K., Neumann, U.: Complete 3d human reconstruction from a single incomplete image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8748–8758 (2023)

43. Weng, C.Y., Curless, B., Kemelmacher-Shlizerman, I.: Photo wake-up: 3d character animation from a single photo. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5908–5917 (2019)
44. Worchel, M., Diaz, R., Hu, W., Schreer, O., Feldmann, I., Eisert, P.: Multi-view mesh reconstruction with neural deferred shading. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6187–6197 (2022)
45. Xiong, Z., Kang, D., Jin, D., Chen, W., Bao, L., Cui, S., Han, X.: Get3dhuman: Lifting stylegan-human into a 3d generative model using pixel-aligned reconstruction priors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9287–9297 (2023)
46. Xiu, Y., Yang, J., Cao, X., Tzionas, D., Black, M.J.: Econ: Explicit clothed humans optimized via normal integration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 512–523 (2023)
47. Xiu, Y., Yang, J., Tzionas, D., Black, M.J.: Icon: Implicit clothed humans obtained from normals. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13286–13296. IEEE (2022)
48. Yang, Z., Wang, S., Manivasagam, S., Huang, Z., Ma, W.C., Yan, X., Yumer, E., Urtasun, R.: S3: Neural shape, skeleton, and skinning fields for 3d human modeling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13284–13293 (2021)
49. Yu, T., Zheng, Z., Guo, K., Liu, P., Dai, Q., Liu, Y.: Function4d: Real-time human volumetric capture from very sparse consumer rgb-d sensors. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5746–5756 (2021)
50. Yu, Z., Cheng, W., Liu, X., Wu, W., Lin, K.Y.: Monohuman: Animatable human neural field from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16943–16953 (2023)
51. Zhang, J., Zhang, Y., Cun, X., Huang, S., Zhang, Y., Zhao, H., Lu, H., Shen, X.: T2m-gpt: Generating human motion from textual descriptions with discrete representations. arXiv preprint arXiv:2301.06052 (2023)
52. Zheng, Z., Huang, H., Yu, T., Zhang, H., Guo, Y., Liu, Y.: Structured local radiance fields for human avatar modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15893–15903 (2022)
53. Zheng, Z., Yu, T., Liu, Y., Dai, Q.: Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *IEEE transactions on pattern analysis and machine intelligence* **44**(6), 3170–3184 (2021)
54. Zielonka, W., Bagautdinov, T., Saito, S., Zollhöfer, M., Thies, J., Romero, J.: Drivable 3d gaussian avatars. arXiv preprint arXiv:2311.08581 (2023)