RC-SMPL : Real-time Cumulative SMPL-based Avatar Body Generation

Hail Song*
KAIST UVR Lab

Boram Yoon†
KAIST UVR Lab

Woojin Cho‡
KAIST UVR Lab

Woontack Woo§
KAIST UVR Lab
KAIST KI-ITC ARRC

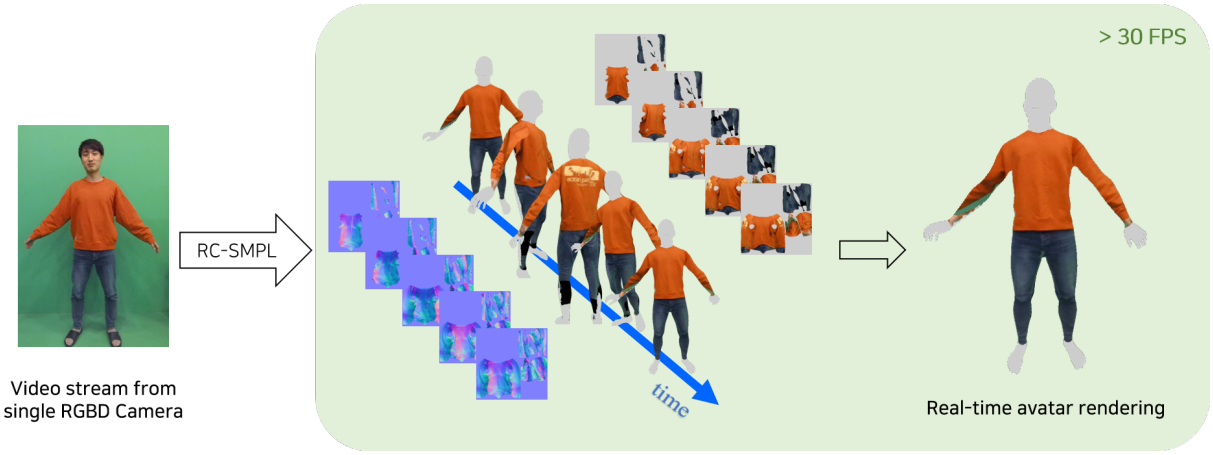

Figure 1: A teaser image showing our real-time 3D avatar body generation. Our system RC-SMPL transfers point clouds from an RGBD camera onto the animating body model to generate texture and normal maps. As the RGBD video stream comes in, our system progressively completes the full texture and normal maps using acquired partial information.

Abstract

We present a novel method for avatar body generation that cumulatively updates the texture and normal map in real-time. Multiple images or videos have been broadly adopted to create detailed 3D human models that capture more realistic user identities in both Augmented Reality (AR) and Virtual Reality (VR) environments. However, this approach has a higher spatiotemporal cost because it requires a complex camera setup and extensive computational resources. For lightweight reconstruction of personalized avatar bodies, we design a system that progressively captures the texture and normal values using a single RGBD camera to generate the widely-accepted 3D parametric body model, SMPL-X. Quantitatively, our system maintains real-time performance while delivering reconstruction quality comparable to the state-of-the-art method. Moreover, user studies reveal the benefits of real-time avatar creation and its applicability in various collaborative scenarios. By enabling the production of high-fidelity avatars at a lower cost, our method provides more general way to create personalized avatar in AR/VR applications, thereby fostering more expressive self-representation in the metaverse.

Index Terms: Computing methodologies—Computer graphics—Image manipulation—Texturing; Computing methodologies—Artificial intelligence—Computer vision—Reconstruction

1 Introduction

In the AR/VR area, a realistic avatar is one of the essential factors for representing the user's ego. On the Metaverse, the user utilizes an avatar as an interface to express himself and communicate with remote users. To generate the avatar efficiently, many studies have attempted to automatically reconstruct the 3D model of the human body with images or video input through deep learning techniques. Considering the statistical validity of the human body model and the need for animatable features, numerous studies have employed parametric body models. The SMPL model [32], a controllable 3D body model that expresses individual body types, is the most commonly used model for 3D human shape reconstruction. The SMPL model has also developed follow-up studies such as SMPL-X and STAR [39, 40], which are parametric models that make more use of the details of human expression based on the SMPL model.

Several techniques have been studied to reconstruct the parametric body models through visual information such as images [4,10,28,40] or a video [5, 41, 50] using deep learning techniques to generate avatars. SMPLify [10] and SMPLify-X [40] successfully reconstruct each model with a single image input, but texture information is not considered. Also, Lazova et al. [28] proposed a method that adopts a generative adversarial network (GAN) to generate a full body texture map, which still fails to deliver high-quality results for the details of invisible parts. Video-based approaches [5, 41, 50] can create full-body texture maps precisely. However, they require a long processing time and show insufficient quality for dynamic sequences such as moving arms or legs. Recent studies [48, 53] obtain models with clothes and the human body in an animatable form, but they have limitations that require high-quality scans. Recently, the technique using radiance fields presented in [36] is widely used to gather the realistic human body model and has shown good performance [8, 13, 54]. However, learning the radiance field was time-consuming, thus incorporating it into an interactive application had a significant challenge. In addition, there was a limit to securing real-time due to the high amount of computational resources to render on AR/VR HMD. Additionally, existing studies have constraints as they require separate pre-processing stages for recording images/videos and creating avatars.

Therefore, we propose a real-time avatar generation system with

*e-mail: hail96@kaist.ac.kr
†e-mail: boram.yoon1206@kaist.ac.kr
‡e-mail: dnwls2416@kaist.ac.kr
§e-mail: wwoo@kaist.ac.kr

a single RGBD camera by progressively updating the texture and normal map without complex pre-processing or initialization. Our key idea is that the texture and normal maps can be gradually refined when the user's body is within the camera's field of view and movements are captured. Notably, the UV coordinates used by the texture map and normal map are defined by the SMPL-X body model, even as the avatar animates. This ensures consistent alignment between the 3D coordinates on the mesh surfaces and the 2D UV space, referred to as UV correspondence. By maintaining this UV correspondence defined by the SMPL-X body model, our approach guarantees accurate alignment of the texture and normal maps with the avatar's geometry. The progressive update of the texture map and normal map with UV correspondence is a key feature that distinguishes our method from previous studies and enables real-time performance in interactive 3D human avatar interactions.

In summary, our contribution is the development of a system that rapidly reconstructs human models, including texture and normal information. In particular, our system enables real-time reconstruction, allowing users to create their own avatars by observing the rendering results in real-time and adjusting their poses to improve the quality of specific parts. Our 'Real-time Cumulative SMPL-based Avatar Body Generation(RC-SMPL)' system, which uses a single RGBD video stream as input, exhibits the following characteristics:

- Real-time reconstruction of a 3D human avatar via asynchronous RGBD frame projection.
- Including full texture map and normal map of the avatar body model.
- Enabling users to observe the texture of the progressively generated avatar.

We validate the system through experiments including real-time verification, image similarity between input RGB images and rendering results, and user study with existing video-based reconstruction method.

2 Related works

2.1 Parametric body model

Parametric body models are mathematical representations of the human body that allow for the efficient and accurate generation of 3D human shapes and poses. These models capture the underlying structure of the human body, including variations in shape and pose, by using a set of parameters that can be adjusted to create a wide range of realistic human forms. By leveraging prior knowledge and statistical analysis of human body shapes and poses, parametric body models can reduce computational complexity and simplify the process of human body reconstruction. In recent years, parametric body models have become increasingly important in various applications, such as computer graphics, VR, and AR.

In recent years, the Skinned Multi-Person Linear Model (SMPL) [32] has gained widespread use in human reconstruction. SMPL leverages large datasets to learn a mixture of shapes, representing human posture as a linear combination of rotation matrices. Compared to Linear Blending Skinning (LBS), SMPL offers a more standardized, simplified, and realistic approach with superior generalization capabilities. However, SMPL lacks a detailed representation of facial expressions and hand gestures, which convey significant interactive information. To address this limitation, Pavlakos et al. [40] proposed the SMPL-X model, which emphasizes local details to improve the representation of facial expressions and hand gestures. Both SMPL and SMPL-X employ low-dimensional parameters as inputs to generate high-dimensional human models. In this study, we use the SMPL-X model as our base template human model.

2.2 3d human model reconstruction

Reconstructing 3D human models has been a long-standing challenge in the field of AR/VR, computer graphics, and computer vision. High-quality reconstruction has been achieved in studies such as [22, 30, 47], using an image as an input of a deep learning network or fusing observations from dense arrays of RGBD camera. However, these studies primarily focused on the 3D reconstruction of humans in fixed poses, making their approaches unsuitable for scenarios requiring new pose data to animate the model. Recent advancements in 3D human model reconstruction have focused on leveraging personalized or parametric mesh models, like SMPL or SMPL-X [32, 40], to reconstruct 3D human models from monocular video input [5, 21, 24, 56]. These methods primarily deform the template mesh to fit 2D joints and silhouettes, allowing for the reconstruction of 3D human models without the need for complex hardware setups. Additionally, studies such as [48, 53] have made it possible to create animatable 3D human models that include clothing information using dense scans. Nonetheless, a limitation of these studies has been the requirement of several hours for inference, which constrains their practical applicability in real-time scenarios.

In recent years, neural representations [36] have emerged as a powerful approach for modeling 3D humans [8, 13, 16, 31, 43, 54] and head [11, 17, 19, 58] utilizing neural representations have demonstrated the ability to directly reconstruct high-fidelity neural human avatars from sparse sets of views or monocular videos, eliminating the need for pre-scanning personalized templates. These techniques employ neural radiance fields [36] or texture fields in a pose-independent canonical space to model 3D human shape and appearance. Subsequently, the models are deformed and rendered into various body poses to learn from posed observations. In particular, the study by [25] substantially reduced the training time required for achieving high-quality results, building upon the existing methods that necessitated several hours of training. Although these approaches have achieved impressive quality and can learn avatars from monocular videos, they showed slow training and rendering speeds. They usally display low frame rates and resolutions, which restricts their applicability in AR/VR environments. These studies also focus on learning radiance fields that produce 2D rendering outputs, challenging their integration into commercial game engines or rendering pipelines. Furthermore, these limitations have difficulties directly expanding their applications to AR and VR, where real-time interaction is essential.

In our work, we developed a system that cumulatively acquires color and depth information from a sequence of frames to make the details of 3D models from real-time RGBD video streams. Furthermore, unlike existing video-based restoration techniques, we propose a method that places more weight on the color and normal information of the most recent frame and incorporates an algorithm that satisfies real-time requirements.

2.3 Human body model texture generation

Texture generation for human model is a challenging task, especially when working with multi-view images or video. The complexity arises from the need to combine partial textures generated from different views seamlessly, ensuring minimal ghosting and stitching artifacts. Various techniques have been proposed to address this issue, including blending [9, 14, 49] and mosaicing [6, 29]. Learning-based models that leverage multi-view images for training [20, 38] still face similar challenges. Recent studies have explored the usage of a single image with GAN [12, 28] or a single image with re-identification supervision [52] for generating full-body texture maps. However, these approaches have been challenged by issues related to processing time and texture quality.

Our proposed approach, RC-SMPL, cumulatively completes texture maps obtained through 3D registrations to mitigate these problems. For more detail, we apply a local weight regarding the pro-

portion of the newly incoming image's texture and normal to use, refining our approach. This method allows for the high-quality, real-time generation of texture and normal maps for the SMPL-X body model, surpassing the limitations of existing methods.

2.4 Real-time 3D Human Model Generation

Recent advancements in real-time 3D human model generation have aimed to overcome the limitations of previous studies that required extensive processing and computation. In this regard, Lu et al. [33] introduced an approach that directly utilizes depth information for 3D reconstruction, enabling the real-time generation of 3D human models. However, this study primarily focused on 3D shape reconstruction and utilized only front and rear images of the scanned subject for texture generation, employing offline processing techniques. Moreover, their method did not utilize UV coordinate-based texture maps but relied on vertex colors, resulting in the inability to achieve high-fidelity representations. In contrast, our proposed RC-SMPL approach specifically focuses on texture map generation, addressing these limitations by offering solution for creating high-quality, real-time cumulative texture and normal maps for the SMPL-X based avatar.

3 METHODOLOGY

The core idea is to asynchronously project color and normal values onto an animating SMPL-X body model [40], updating the texture and normal maps of the visible regions of the model. Transferring raw values from the RGBD frame to the base body model and implementing this asynchronously can enable real-time execution of the system. Additionally, a seamless and continuous texture acquisition can be achieved by referencing the texture map generated from previous frames and blending it with the currently obtained information. Fig. 2 illustrates the system's overall structure.

The entire reconstruction proceeds as follows. Our system acquires image streaming and body pose tracking information from an RGBD camera for each frame. The SMPL-X body model is then positioned using the body pose tracking and pre-acquired SMPL-X shape parameters β. The RGBD image is converted to a point cloud using camera parameters. The SMPL-X mesh and point cloud array information are loosely overlapped at this stage. In order to achieve real-time performance, the process of acquiring textures and normals is conducted asynchronously. For each point in the point cloud, we initiate a raycasting process from the camera origin, tracing the path of the ray until it intersects with the mesh. We calculate an orthogonality weight based on the inner product of the mesh's normal value and the direction of the ray, ensuring the reliability of the acquired values. This weight is used to populate the corresponding texture and normal maps at the point of intersection. The normal value is calculated using the information from the neighboring points in the point cloud array. The cumulatively filled texture and normal maps in the UV coordinates are applied to the SMPL-X model's render during each asynchronous update. The system maintains real-time performance through asynchronous UV map updates. For the application of our system in AR/VR environments, the program has been implemented using Unity, a widely used AR/VR/gaming engine. The pre-required information before system operation, processes executed for every frame, and asynchronous processes can be summarized as follows.

- Pre-required information
  - SMPL-X shape parameters (β)
- Processes executed for every frame
  - RGBD video streaming
  - Pose tracking (θ parameters of SMPL-X model)
  - Avatar(SMPL-X model) animation
  - Rendering of the resulting body model
- Asynchronous processes
  - Raycasting and collision testing
  - Calculation of normal values
  - Conversion of points to UV space of SMPL-X body model
  - Texture and normal map updates

3.1 Preliminaries

We utilized the Kinect DK [2], a commercial RGBD sensor, and the SMPL-X model to animate a realistic human body model in real-time. The SMPL-X [40] model, denoted as $M(\beta,\theta)$, requires parameters for body shape β and pose θ.

Basic Body Model. The shape parameters β are obtained from a single image in advance using the SMPLify-X [40] model. It extract shape parameter using CNN-based method and interpenetration error term.

Pose Estimation for 3D Model Animation. The pose parameters θ are derived from the pose tracking data provided by Kinect DK. The pose-tracking data is generated through the inference results of a pre-trained neural network, which uses depth and color frames as inputs. This ensures a robust foundation for real-time applications.

Animating of base body model. In our system, the pose parameters and the SMPL-X model are updated per every frame, so we set the notation as follows.

$$M_t(\beta,\theta_t) \tag{1}$$

The pose parameters and the animated avatar mesh are expressed as θ_t and M_t for the current frame t.

$$P(I_t) = \{p_t^{0,0},\cdots,p_t^{w-1,h-1}\} \tag{2}$$

To apply texture to the animated SMPL-X body model, we utilized the point clouds data of Kinect DK, which has converted from RGBD stream. We represent RGBD image sequence as $\{I_1,I_2,\cdots,I_t\}$ and each image is converted to a point cloud array $P(I_t)$. The point cloud array $P(I_t)$ comprises p_t points, organized in a grid with dimensions $w \times h$. w and h denote the number of pixels in the width and height of the image, respectively.

$$p_t^{i,j} = (x,y,z;r,g,b) \tag{3}$$

The notation $p_t^{i,j}$ represents a point cloud at the image coordinates in the frame t for pixel i and j. Each point cloud includes position information relative to the x, y, and z axes, as well as color information, denoted by the r, g, and b components. Through these processes in every frame, the point clouds and visible parts of the SMPL-X body model are loosely aligned in the virtual space.

3.2 Acquisition of Texture and Normal Maps

This subsection describes an asynchronous loop process for real-time cumulative texture map generation. Each point cloud $p_t^{i,j}$ belonging to the point clouds array $P(I_t)$ is utilized for ray casting to calculate the intersection with the animating avatar mesh $M_t(\beta,\theta_t)$.

$$\overrightarrow{d} = \frac{p_t^{i,j} - \tilde{O}}{|p_t^{i,j} - \tilde{O}|} \tag{4}$$

$$r(\lambda) = \tilde{O} + \overrightarrow{d} \cdot \lambda \quad (0 < \lambda < \lambda_{max}) \tag{5}$$

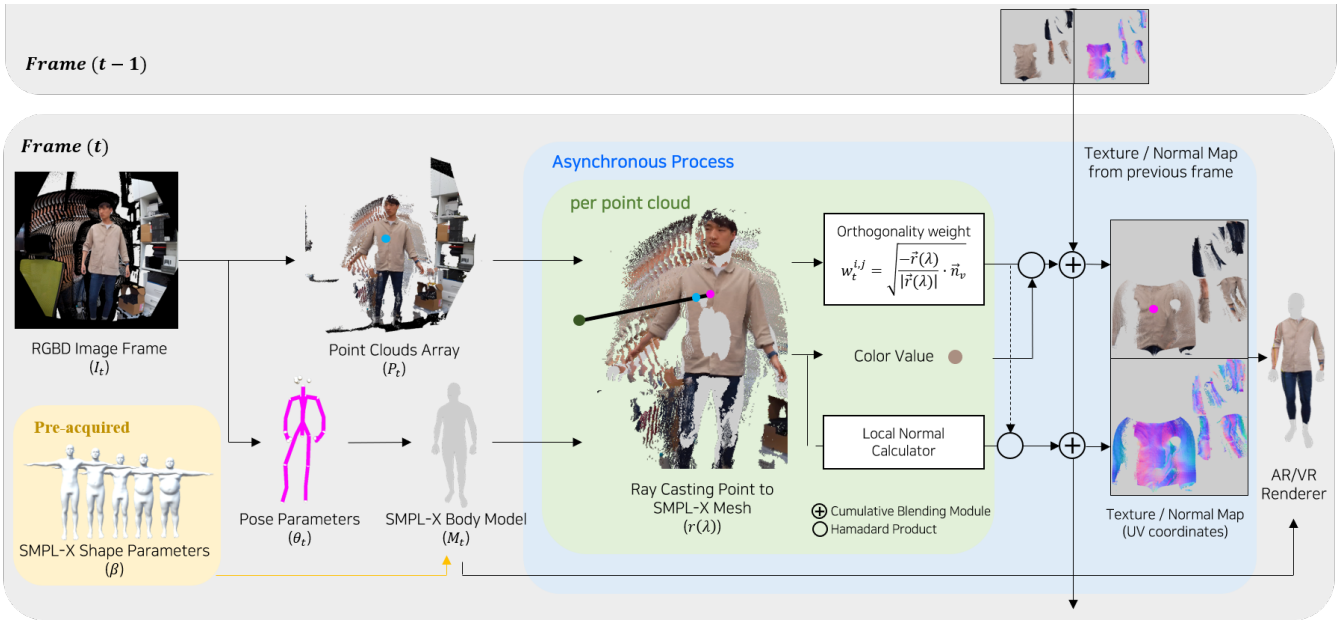

Figure 2: System diagram of proposed method for real-time 3D human body model reconstruction and enhancement of texture and normal maps. The diagram presents the various components of the system, which are detailed in Sections 3.1 (Preliminaries) through 3.3 (Normal Map Generation).

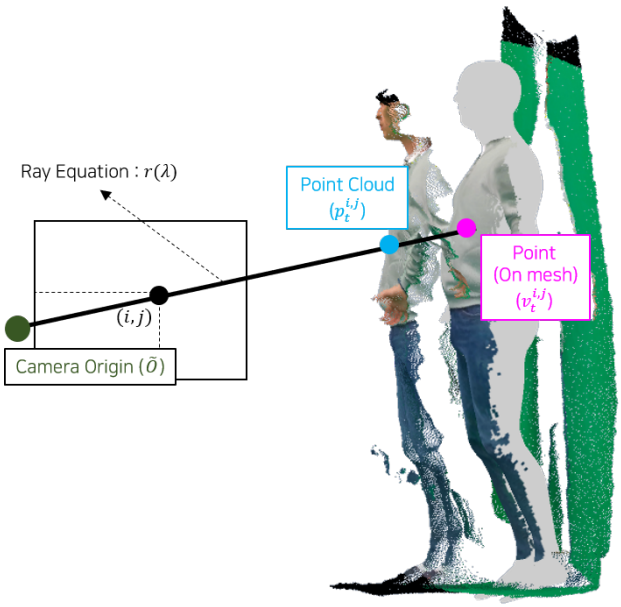

Figure 3: An illustration of ray casting from the point clouds obtained from the RGBD video stream onto the animated virtual SMPL-X model. To better illustrate the concept of ray casting, we have exaggerated the distance between the 3D body model and point clouds in the figure, which is closely overlapped in reality.

Each point cloud creates a ray that passes through the camera origin $\tilde{O}$ and the point cloud $p_t^{i,j}$. The direction of each ray is determined by normalizing the distance between $\tilde{O}$ and $p_t^{i,j}$, as shown in Equation 4. In Equation 4, $\overrightarrow{d}$ represents a unit vector indicating the ray direction. The camera origin $\tilde{O}$ serves as the ray origin. The system detects a collision between the calculated ray $r(\lambda)$ and the SMPL-X mesh $M_t(\beta, \theta_t)$. The collision point, $v_t^{i,j}$, is the point on the mesh closest to the camera origin $\tilde{O}$ where the ray and the avatar mesh intersect. The coordinates of $v_t^{i,j}$ are used to transfer the color value of $p_t^{i,j}$ on the UV coordinate-based texture map.

The mathematical representation of the collision detection between ray $r(\lambda)$ and avatar mesh $M_t(\beta, \theta_t)$ can be written as:

$$v_t^{i,j} = Collision(r(\lambda), M_t(\beta, \theta_t)) \tag{6}$$

where $Collision(r(\lambda), M_t(\beta, \theta_t))$ is a function that finds the intersection point $v_t^{i,j}$ between the ray and the mesh that is closest to the camera origin $\tilde{O}$. The collision detection is calculated using the method described by [1].

$$(u, v) = F(v_t^{i,j}) \tag{7}$$

The transformed colored point $v_t^{i,j}$ passed through $Collision(\cdot)$ is converted into the UV coordinates (u, v) of the corresponding texture map in the model through the UV mapping function F, as shown in the Equation 7.

Instead of directly assigning RGB values in the texture mapping process, our system updates the color values of the corresponding areas in the UV coordinates using a weighted combination of the previous color values and the new color values from $v_t^{i,j}$. The weights are calculated using the angle information between the ray direction $\overrightarrow{d}$ and the normal vector $\overrightarrow{n_v}$ at $v_t^{i,j}$, as shown in the following Equation 8).

$$w_t^{i,j} = \sqrt{-\overrightarrow{d} \cdot \overrightarrow{n_v}} \tag{8}$$

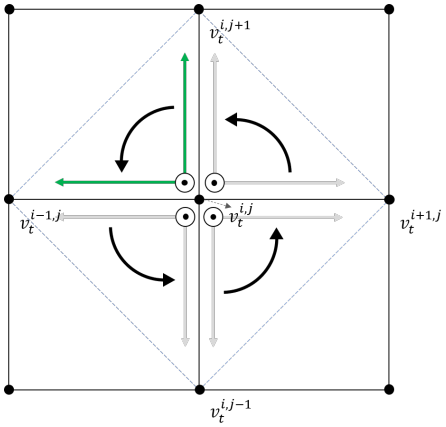

Figure 4: Illustration of the lightweight technique for computing normal values from acquired RGBD images. The normal values for a point cloud $v^{i,j}$ are calculated using the cross product of the vectors formed by its neighboring point clouds. This approach enables real-time normal value computation while minimizing computing resources.

$$c_t^{u,v} = (1 - w_t^{i,j}) * c_{t-1}^{u,v} + w_t^{i,j} * c_{new} \tag{9}$$

The updated color values $c_t^{u,v}$ are acquired according to Equation 9. Here, c_{new} stands for the color information of $v_t^{i,j}$, identical to the color value of $p_t^{i,j}$.

The points transformed into the UV coordinate system are relatively sparse compared to the model's texture map, which can potentially lead to seams during the texture map update process. For online texture updating, our approach does not employ texture stitching techniques, such as blending [9, 14, 49] and mosaicing [6, 29] considering the efficiency of the process. Instead, we utilize surrounding pixel's value during the coloring process. The color around each point is blended according to a Gaussian distribution, which helps to ensure a seamless integration between the colored regions and their adjacent points on the avatar mesh.

3.3 Normal Map Generation

In this subsection, we introduce the real-time normal map generation method incorporated into our system to represent the detailed features of the human body model alongside the texture map. Typically, surface normals are calculated from point clouds by finding neighboring point clouds and computing eigenvectors. The computational complexity for normal calculation, as explored in [37], involves adaptively calculating the neighborhood size, resulting in a complexity of $O(Nlog(N))$. This presents challenges when calculating normals in real-time from a large number of point clouds.

To compute normal values, studies on [3,23] have leveraged batch information from point clouds obtained using Kinect. Our system follows a similar approach, as the point cloud information is gathered from RGBD images, which are already ordered in a grid structure. This ordered structure allows us to reduce the computational complexity by specifying neighboring point clouds. Specifically, for each point cloud $v^{i,j}$, we utilize the positions of neighboring point clouds $v^{i-1,j}$, $v^{i+1,j}$, $v^{i,j-1}$ and $v^{i,j+1}$. Fig. 4 illustrates our system's local normal calculation method.

$$n_1^{i,j} = (v^{i+1,j} - v^{i,j}) \times (v^{i,j+1} - v^{i,j}) \tag{10}$$
$$n_2^{i,j} = (v^{i,j+1} - v^{i,j}) \times (v^{i-1,j} - v^{i,j}) \tag{11}$$
$$n_3^{i,j} = (v^{i-1,j} - v^{i,j}) \times (v^{i,j-1} - v^{i,j}) \tag{12}$$
$$n_4^{i,j} = (v^{i,j-1} - v^{i,j}) \times (v^{i+1,j} - v^{i,j}) \tag{13}$$

$$\overrightarrow{n^{i,j}} = \frac{\sum_{k=1}^{4} \widehat{n_k^{i,j}}}{4} \tag{14}$$

Cross products of vectors formed by neighboring point clouds $v^{i,j}$ are calculated, as shown in Equation 10 to Equation 13. Each of these vectors is normalized to create unit vectors, $\widehat{n_k^{i,j}}$, representing the direction of the original vectors ($n_k^{i,j}$). The final normal vector $\overrightarrow{n^{i,j}}$ for the point cloud $v^{i,j}$ is derived from the average of these unit vectors, as presented in Equation 14.

This method allows us to compute normal values for each point cloud in real-time, taking advantage of the spatial relationships between adjacent point clouds in the image domain.

4 Experiments

In this section, we present the evaluation of our proposed system through two user studies and system evaluation including image similarity and computational performance. We compare our method with existing approaches to demonstrate its effectiveness. Our method requires RGBD frame stream data and a corresponding avatar model. As no existing dataset satisfies these conditions, we collected our dataset for qualitative and quantitative evaluations during the first user study. Our experimental environments for processing are: Intel Core i7-8700K CPU, RTX3080 GPU, 32G RAM.

4.1 User Studies

We conducted two human-subjects studies to verify that our method successfully represents the real user's body and delivers a sufficient feeling of the avatar's body embodiment by comparing it with the existing method. For this, we designed Study 1 to generate avatar models of real users for evaluation and evaluate the rendering results in terms of avatar embodiment. In Study 2, the perceived similarity of the created avatar models with the ground truth was evaluated in the 3D devices' environment from the third person's point of view in a more subjective way. In both user studies, the two experimental conditions were the reconstructed avatars based on the methods of (1) *Video* (Alldieck et al. [5]) and (2) *Ours* (RC-SMPL). A total of 12 different avatar models (shown in Appendix Figure 1), which consisted of two types of avatars derived from each condition, were utilized for both subjective evaluations: We performed avatar scans to prepare these model sets in Study 1. The study content and procedures were approved by Institutional Review Board in advance.

4.1.1 Study 1: Sense of Embodiment

The first study aimed to not only generate samples of real user's avatars based on each condition but also evaluate whether our method sufficiently conveys the Sense of Embodiment (SoE) to users, which is defined as the representation of users with appropriate body images [7, 26], as their virtual replica compared to the existing method. Since a virtual avatar's appearance can greatly influence its user's behavior, attitude, and overall perception during the virtual experience [55, 57], the SoE has been broadly utilized to evaluate a user's body illusion toward the given virtual avatar [18, 44]. Based on the previous studies investigating the avatar appearance [15, 27, 34, 51, 55], which evaluated the user's SoE with various measurements, we also utilized the three representative tools: (1) the Avatar Embodiment (AE) by Peck and

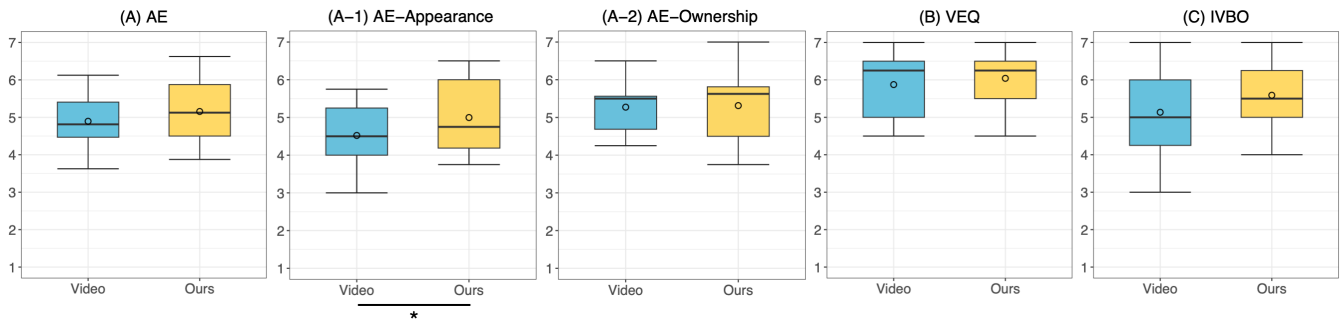

Figure 5: (Study 1) Results of Likert scale rating (1: strongly disagree – 7: strongly agree) for (A) Avatar Embodiment (AE), (A-1) AE–Appearance, (A-2) AE–Ownership; (B) Virtual Embodiment Questionnaire (VEQ); and (C) Illusion of Virtual Body Ownership (IVBO). (statistical significance between the experimental conditions: $^{*}p < .05$)

Gonzalez-Franco [42]; (2) the Virtual Embodiment Questionnaire (VEQ) by Roth and Latoschik [45]; and (3) the Illusion of Virtual Body Ownership (IVBO) by Roth et al. [46].

The AE [42] and VEQ [45] evaluate the user's body embodiment of their virtual avatar, generally composed of self-identification, agency, and self-location. From the AE measurement, the two subscales—Appearance and Ownership—were adopted based on our study purpose; the first subscale specifically focuses on appearance elements such as posture, shape, and clothes. The items in VEQ measure the perception of perceiving their own body and feeling of control over a virtual body. From the IVBO [46], which indicates the effect of users perceiving a virtual avatar's body as their own, we utilized items to measure the extent to the appropriate visual feedback of motion control and changes in self-perception. As a result, a total of 11 items on a seven-point Likert scale were measured, and items with low relevance or unmeasurable due to the study setup were deleted.

We recruited 12 participants—half of them were male, and half were female—through the campus website, and their ages ranged from 23 to 30 years ($M = 25.83$, $SD = 2.41$). The average height was 166 ($SD = 4.43$) and 175.5 centimeters ($SD = 5.75$) for female and male participants, respectively. Their previous experience with avatar-mediated applications (e.g., Social Network Services (SNS) or games) was asked: nine of them had never (six, 50%) or once (five, 41.67%), and one participant only had up to five times (8.33%).

Due to the *Video* condition requiring about 6 hours to reconstruct the avatar in our hardware setup [5], the study consisted of two separate phases, and therefore the participants had to visit twice on different dates. In the first phase, we used SMPLify-X [40] to obtain an image-based body model and create a base SMPL-X avatar for both methods. During the image collection process, participants were instructed to assume specific postures: They were asked to stand with their arms spread out in front of the camera and rotate twice, capturing the necessary input for the *Video* condition. Once we generated the avatars for both conditions, the participants were invited to the second phase. In this phase, they performed the simple task of experiencing the created avatars and evaluating them. Each participant evaluated both avatar conditions in a balanced order. The *Video* condition used a pre-generated avatar wearing the same clothes the participant had on during phase 1; *Ours* condition also started with a pre-generated avatar, but it was updated in real-time based on the clothes that the participant wore during phase 2.

The main task involved controlling a virtual avatar displayed in mirror mode on a 2D screen. Participants stood in front of a camera to capture their movements and synchronize them with the avatar in real-time. The virtual scene featured six floating spheres that could be interacted with (Fig. 6(A)). When the user-controlled avatar touched these spheres, the background of the scene changed based on the sphere's material. After freely exploring the avatar and its appearance, participants reported their subjective evaluation on the given questionnaire. Once they finished the task with both conditions, a post-experiment interview was conducted to gather general feedback.

Results

To analyze the quantitative results of the questionnaires, we used a Paired Sample T-Test ($\alpha = .05$) because our participants experienced both types of avatar conditions. The normality of data distribution and homogeneity of variances were first examined through the Shapiro-Wilk and Levene's tests. We excluded one data point in IVBO as an outlier because of the invalid responses. The results are illustrated in Fig. 5.

Avatar Embodiment (AE): The data was normally distributed (*Video*: $W = .982$, $p = .991$; *Ours*: $W = .932$, $p = .400$), and the variances were also homogeneous ($F(1,22) = 1.432$, $p = .244$). There was no significant difference in AE for the avatar conditions ($t(11) = 1.192$, $p = .258$). However, for the subscale of Appearance, we found a significant difference between avatar conditions ($t(11) = 2.300$, $p = .041$): Participants working with the avatar generated based on our method perceived higher embodiment related to their appearance than the one based on the existing method (*Video*: $M = 4.52$, $SD = .82$; *Ours*: $M = 5.00$, $SD = 1.01$). The other subscale of Ownership showed no significant difference between avatars ($t(11) = .155$, $p = .879$).

Virtual Embodiment Questionnaire (VEQ): The assumption of the data's normality (*Video*: $W = .870$, $p = .066$; *Ours*: $W = .920$, $p = .288$) and the variance's homogeneity were also satisfied ($F(1,22) = .444$, $p = .512$). It was revealed that there was no significant difference in VEQ for the avatar conditions ($t(11) = 1.483$, $p = .166$).

Illusion of Virtual Body Ownership (IVBO): The data of the IVBO measurement also satisfied the normality (*Video*: $W = .953$, $p = .685$; *Ours*: $W = .973$, $p = .914$) and the homogeneity assumption ($F(1,22) = 1.707$, $p = .206$). We found no significant difference in IVBO between the two avatar conditions ($t(10) = 1.910$, $p = .085$).

To the question of how the two avatars felt different or similar, participants responded similarly: First, they mostly answered that they felt not much difference because both avatars reflected their body shape and feature such as height, and the way of control their movement was also the same (P1–7, 10, 12). However, some participants also emphasized that they felt more embodiment when their clothes and appearance were updated in real-time under our method (P1, 2, 6, 9), including P11's comment—*"Because the avatar changed to reflect my appearance and the wrinkles on the clothes were expressed, it felt more like my own avatar and seemed realistic."*).

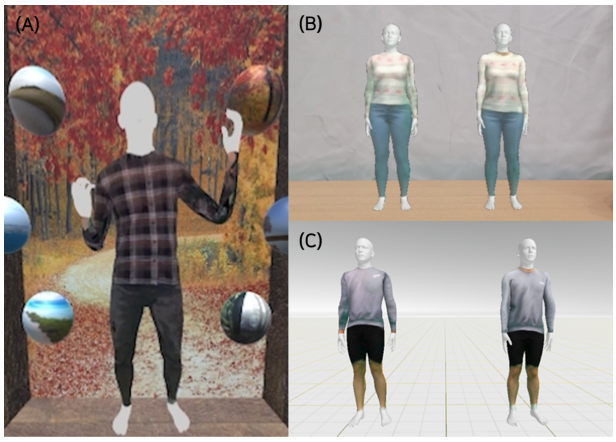

Figure 6: (A) In Study 1, participants engaged in the Sense of Embodiment task, controlling a motion-synchronized virtual avatar and interacting with spheres in the virtual space; and (B) and (C) illustrated rendering results of avatar models in AR and VR, respectively.

4.1.2 Study 2: Subjective Perception on Avatar Similarity

The second study focused on directly comparing two avatars, each generated by our approach and the existing method, following a similar procedure to other texture reconstruction and image generation studies. Using the 12 avatar models generated in Study 1, one avatar type most similar to the ground truth image was requested to be selected. Since it has been assumed that our approach would have the potential to be utilized in an XR environment, we evaluated the rendered results in two different devices (1) VR and (2) AR: For the VR device, a Meta Quest Pro[1] was utilized, and a Microsoft HoloLens 2[2] was used for the AR device, which is an Optical See-Through Head-Mounted Display (OST HMD).

A total of 30 participants were recruited for Study 2, also through the campus online community board. 17 of them identified as female, 12 as male, and one answered as other. Their ages ranged from 21 to 33 years ($M = 25.73$, $SD = 3.37$). We asked about their previous experience related to avatar-mediated applications and AR/VR technologies such as wearable devices: Half of them had less than four times of avatar-mediated experience (53.34%), and the rest of them had more than five times (46.66%). Regarding the experience with AR/VR, only one participant had never experienced it, ten had less than four times (33.33%), and the majority of them had more than five times of AR/VR experience (63.33%).

All participants experienced both devices and selected the avatar based on their subjective perception of avatar similarity: Each device was assigned in a balanced order, and 12 avatar models were randomly distributed to each participant but ensured that there were no repetitions for both devices. In Study 2, the main task was to observe both avatars' movements shown in the virtual scene for a certain period of time (Fig. 6(B) and (C)) and select one avatar thought to be more similar. After they had experienced both devices, we also asked them about the reason for their choices during the retrospective interview.

Results

For the analysis, we first calculated the frequency and compared the proportions of the choice of each avatar condition according to the type of device and avatar models. To verify whether observed proportions have statistically significant differences, we also used the Chi-square test for non-parametric analysis.

[1] https://www.meta.com/quest/quest-pro/

[2] https://www.microsoft.com/en-us/hololens/

VR and AR Device: Even though the *Video* condition was answered 20% and 18.33% more than the *Ours* condition for VR and AR devices, respectively, the Chi-square test for statistical verification showed that a significant difference between proportions was found in VR device ($\chi^2(1) = 4.800$, $p = .029$). Conversely, the difference of the proportions between avatar conditions in AR device was not statistically significant ($\chi^2(1) = 4.030$, $p = .055$): More votes for the *Video* method represent a statistically significant over our method only in a VR device, not in an AR.

Avatar Models: We further analyzed whether 12 avatar models' clothes and distinguished appearances differently impact the user's perceived similarity. We found that three avatar types (Avatar 1, 5, and 11) had significant differences between proportions of the collected response for each generating method (Avatar 1: $\chi^2(1) = 5.000$, $p = .025$; Avatar 5: $\chi^2(1) = 5.000$, $p = .025$; Avatar 11: $\chi^2(1) = 9.800$, $p = .002$). These three avatars had more responses for the *Video* condition than the *Ours* condition (50% more in Avatar 1 and 5, and 70% more in Avatar 11). On the other hand, there were no significant differences between observed proportions in the other nine avatar models (all $p > .05$).

In the interview, we asked about the reason for the choice of avatar, and participants' responses were aligned with the above results. It was commented that different characteristics in two devices influenced the selection of similar avatars, as stated by P18 (*"details shown in VR and AR made me feel different about the avatar"*): In the case of VR, in which the existing *Video* method had more votes than *Ours*, many participants could focus on more details and distinguish colors better due to the relatively high resolution, opaque, and vivid display (P4, 8–10, 19, 25, 28, 29). Oppositely for the OST AR device, due to its default transparency and additive color blending, the details of the avatar were less visible, so the difference was not greatly captured; even in the cases of wearing dark-colored clothes, they looked more similar (P4, 6, 7, 10, 19, 25).

Regarding different avatar models, participants also mentioned that the difference between two avatars was weakened when their clothes had the features such as dark colors or plain patterns (P6, 21). However, in the case of avatar models where the *Video* method had more choices, the majority of participants answered that they considered the following factors: the color of the arm looks strange (as greatly observed in Avatar 1), representation around joints parts such as the neck, and clearly distinct details in clothes (e.g., horizontal stripes, neck collar, patterns, or logo). Likewise, Avatar 11 of our method received negative evaluations because it had limited representation in its neck collar area, as we also observed in the qualitative results in Section 4.2.3 (Fig. 7).

4.2 System Evaluation

4.2.1 Image Similarity

In assessing image similarity, we computed the likeness between the rendered results of generated avatar model and the RGB frame images. We adopted the Structural Similarity (SSIM), a masked version of SSIM (mask-SSIM), and the Peak Signal-to-Noise Ratio (PSNR) as our evaluation metrics, in line with the methods of [35, 52, 59].

For a quantitative evaluation, we contrasted our method with the video-based reconstruction method proposed by Alldieck et al. [5]. We generated full-body textures for the 12 participants in Study 1 using both our method and the [5], based on the acquired RGBD video streams. To generate the textures, videos were recorded with the camera placed directly in front of the participants while they rotated twice in place with their arms extended. Subsequently, each texture was applied to the body mesh M_t created using the pose information θ_t and SMPL-X beta parameters β obtained from the stored video streams in the study. Then the image similarity is computed with the ground truth input RGB frames. The mask for mask-SSIM is generated with the frame-by-frame body mesh. The

Table 1: Quantitative comparison of image similarity metrics (SSIM, mask-SSIM, and PSNR) between our method(*Ours*) and the video-based reconstruction method(*Video*) proposed by Alldieck et al. [5].

Methods	SSIM (↑)	mask-SSIM (↑)	PSNR [dB] (↑)
Video [5]	0.2325	0.9431	**9.490**
Ours	**0.2327**	**0.9432**	9.485

Table 2: FPS and texture completion ratio according to system configuration

Method	Texture completion	FPS
Mesh animation	-	69.34
Ours (No async)	-	4.04
Ours (async, 2 seconds)	58.54%	61.10
Ours (async, 1 second)	72.30%	50.65
Ours (async, 0.5 second)	86.64%	42.89
Ours (async, 0.3 second)	96.21%	34.76
Ours (async, 0.1 second)	99.30%	3.71

experimental results for image similarity are summarized in Table 1. The portion marked as *Video* in the Table 1 refers to the video-based approach [5]. Our rendering results showed slightly higher values in both SSIM and mask-SSIM, while a marginally lower score was observed for PSNR.

4.2.2 FPS Performance Evaluation

To evaluate our system's FPS performance, we measured by altering the transfer cycle during the asynchronous texture transfer process. This assessment demonstrates the degree to which real-time performance can be achieved. Furthermore, to evaluate the texture generation performance, we defined the texture completion ratio as the proportion of the texture completed during a 10-second turn in front of the Kinect DK. Furthermore, to verify the real-time capabilities of our system, we quantitatively compared the FPS levels throughout the texture completion process. Lastly, as there are no existing studies on real-time body texture completion during an interaction, we made comparisons with the following systems.

- Mesh animation: Performs only mesh animation and rendering
- No async: Synchronous texture mapping
- Async: Asynchronous texture mapping (with a variation of texture mapping frequency)

The experimental results are shown in Table 2. It shows the system can generate high-quality texture maps while satisfying the real-time FPS level with asynchronous texture updates. Also, while the update cycle is shortened, the FPS decreases, but the texture completion ratio increases.

4.2.3 Qualitative Results

We present the qualitative results in Fig. 7, which depicts the rendering results of our approach in comparison with the ground truth image and the avatar generation from the video-based restoration method by Alldieck et al. [5]. In the cases of Avatar 6 and 8, our method captures more detailed clothing patterns compared to the results from Alldieck et al.'s method (denoted as *Video*). However, for Avatar 11, our method struggles to accurately capture the complex structure of the clothing around the neck region. The rendering results of all 12 avatars used in the experiments are provided in the Appendix.

5 Conclusion

In this paper, we propose a method to generate and render the texture and normal maps of an avatar's body model in real-time. Our system progressively updates the texture and normal maps using a single commercial RGBD camera, eliminating the need for complex pre-procedures or extensive computational resources. Our novel algorithms facilitate the rapid generation of texture and normal maps, thereby ensuring real-time performance during 3D human avatar interactions. We introduce weight values designed to determine the reliability of newly acquired color values and preserve existing textures. In addition, we propose the use of local normal values for efficient calculations, which represents a novel approach not previously utilized in real-time systems. Through quantitative evaluation, we verified that our real-time generated avatars achieve comparable quality to those produced with the existing video-based restoration method, which typically requires longer time. We validate the convenience of the method for avatar creation and its viability for use in AR and VR environments via two user studies. However, our method has certain limitations; the current system does not generate textures for body parts that are not visible within the system, and reconstruction performance tends to decline when dealing with complex structures such as neck and arms. To overcome these limitations, we plan to leverage learning-based methodologies in our future work, aiming to enhance the efficacy and performance of our proposed system. We will also incorporate advanced tracking methods and animation of the parametric body model's face and hands, creating a more robust system for the rapid generation of full-body avatars.

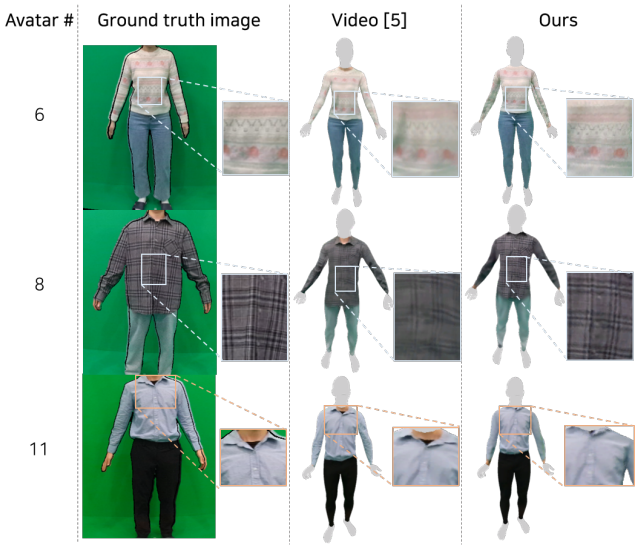

Figure 7: Comparative rendering results. From left to right: Ground truth image, avatar generated by the video-based restoration method by Alldieck et al. [5] (Video), and avatar generated by our proposed method (Ours). Our method captures more distinctive clothing patterns for Avatar 6 and Avatar 8 compared to the video-based method. However, it failed to capture the bumpy patterns of the clothing around the neck for Avatar 11.

Acknowledgments

This research was supported by National Research Council of Science and Technology(NST) funded by the Ministry of Science and ICT(MSIT), Republic of Korea(Grant No. CRC 21011) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-01270, WISE AR UI/UX Platform Development for Smartglasses)

REFERENCES

[1] https://docs.nvidia.com/gameworks/content/gameworkslibrary/physx/.

[2] https://learn.microsoft.com/en-us/azure/kinect-dk/.

[3] D. S. Alexiadis, G. Kordelas, K. C. Apostolakis, J. D. Agapito, J. Vegas, E. Izquierdo, and P. Daras. Reconstruction for 3d immersive virtual environments. In *2012 13th International Workshop on Image Analysis for Multimedia Interactive Services*, pp. 1–4. IEEE.

[4] T. Alldieck, M. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1175–1186, 2019.

[5] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll. Video based reconstruction of 3d people models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8387–8397, 2018.

[6] A. Baumberg. Blending images for texturing 3d models. In *Bmvc*, vol. 3, p. 5, 2002.

[7] S. Benford, J. Bowers, L. E. Fahlén, C. Greenhalgh, and D. Snowdon. User embodiment in collaborative virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 242–249, 1995.

[8] A. Bergman, P. Kellnhofer, W. Yifan, E. Chan, D. Lindell, and G. Wetzstein. Generative neural articulated radiance fields. *Advances in Neural Information Processing Systems*, 35:19900–19916, 2022.

[9] F. Bernardini, I. M. Martin, and H. Rushmeier. High-quality texture reconstruction from multiple scans. *IEEE Transactions on Visualization and Computer Graphics*, 7(4):318–332, 2001.

[10] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European conference on computer vision*, pp. 561–578. Springer, 2016.

[11] C. Cao, T. Simon, J. K. Kim, G. Schwartz, M. Zollhoefer, S.-S. Saito, S. Lombardi, S.-E. Wei, D. Belko, S.-I. Yu, et al. Authentic volumetric avatars from a phone scan. *ACM Transactions on Graphics (TOG)*, 41(4):1–19, 2022.

[12] S. Cha, K. Seo, A. Ashtari, and J. Noh. Generating 3d human texture from a single image with sampling and refinement. In *ACM SIGGRAPH 2022 Posters*, pp. 1–2. 2022.

[13] J. Chen, Y. Zhang, D. Kang, X. Zhe, L. Bao, X. Jia, and H. Lu. Animatable neural radiance fields from monocular rgb videos. *arXiv preprint arXiv:2106.13629*, 2021.

[14] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pp. 11–20, 1996.

[15] J. C. Eubanks, A. G. Moore, P. A. Fishwick, and R. P. McMahan. The effects of body tracking fidelity on embodiment of an inverse-kinematic avatar for male participants. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 54–63. IEEE, 2020.

[16] Y. Feng, J. Yang, M. Pollefeys, M. J. Black, and T. Bolkart. Capturing and animation of body and clothing from monocular video. *arXiv preprint arXiv:2210.01868*, 2022.

[17] X. Gao, C. Zhong, J. Xiang, Y. Hong, Y. Guo, and J. Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Transactions on Graphics (TOG)*, 41(6):1–12, 2022.

[18] A. Genay, A. Lécuyer, and M. Hachet. Being an avatar "for real": a survey on virtual embodiment in augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):5071–5090, 2021.

[19] P.-W. Grassal, M. Prinzler, T. Leistner, C. Rother, M. Nießner, and J. Thies. Neural head avatars from monocular rgb videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18653–18664, 2022.

[20] A. Grigorev, A. Sevastopolsky, A. Vakhitov, and V. Lempitsky. Coordinate-based texture inpainting for pose-guided image generation. *arXiv preprint arXiv:1811.11459*, 2018.

[21] C. Guo, X. Chen, J. Song, and O. Hilliges. Human performance capture from monocular video in the wild. In *2021 International Conference on 3D Vision (3DV)*, pp. 889–898. IEEE, 2021.

[22] K. Guo, P. Lincoln, P. Davidson, J. Busch, X. Yu, M. Whalen, G. Harvey, S. Orts-Escolano, R. Pandey, J. Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Transactions on Graphics (ToG)*, 38(6):1–19, 2019.

[23] S. Holzer, R. B. Rusu, M. Dixon, S. Gedikli, and N. Navab. Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2684–2689. IEEE, 2012.

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

[25] T. Jiang, X. Chen, J. Song, and O. Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. *arXiv preprint arXiv:2212.10550*, 2022.

[26] K. Kilteni, R. Groten, and M. Slater. The sense of embodiment in virtual reality. *Presence: Teleoperators and Virtual Environments*, 21(4):373–387, 2012.

[27] M. E. Latoschik, J.-L. Lugrin, and D. Roth. Fakemi: A fake mirror system for avatar embodiment studies. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*, pp. 73–76, 2016.

[28] V. Lazova, E. Insafutdinov, and G. Pons-Moll. 360-degree textures of people in clothing from a single image. In *2019 International Conference on 3D Vision (3DV)*, pp. 643–653. IEEE, 2019.

[29] H. P. Lensch, W. Heidrich, and H.-P. Seidel. A silhouette-based algorithm for texture registration and stitching. *Graphical Models*, 63(4):245–262, 2001.

[30] K. Lin, L. Wang, and Z. Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1954–1963, 2021.

[31] L. Liu, M. Habermann, V. Rudnev, K. Sarkar, J. Gu, and C. Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM Transactions on Graphics (TOG)*, 40(6):1–16, 2021.

[32] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.

[33] Y. Lu, H. Yu, W. Ni, and L. Song. 3d real-time human reconstruction with a single rgbd camera. *Applied Intelligence*, pp. 1–11, 2022.

[34] J.-L. Lugrin, M. Landeck, and M. E. Latoschik. Avatar embodiment realism and virtual fitness training. In *2015 IEEE Virtual Reality (VR)*, pp. 225–226. IEEE, 2015.

[35] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. *Advances in neural information processing systems*, 30, 2017.

[36] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[37] N. J. Mitra and A. Nguyen. Estimating surface normals in noisy point cloud data. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pp. 322–328, 2003.

[38] N. Neverova, R. A. Guler, and I. Kokkinos. Dense pose transfer. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 123–138, 2018.

[39] A. A. Osman, T. Bolkart, and M. J. Black. Star: Sparse trained articulated human body regressor. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pp. 598–613. Springer, 2020.

[40] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black. 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.

[41] G. Pavlakos, N. Kolotouros, and K. Daniilidis. Texturepose: Supervising human mesh estimation with texture consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 803–812, 2019.

[42] T. C. Peck and M. Gonzalez-Franco. Avatar embodiment. a standardized questionnaire. *Frontiers in Virtual Reality*, 1:575943, 2021.

[43] A. Raj, J. Tanke, J. Hays, M. Vo, C. Stoll, and C. Lassner. Anr: Articulated neural rendering for virtual avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3722–3731, 2021.

[44] R. Ratan, D. Beyea, B. J. Li, and L. Graciano. Avatar characteristics induce users' behavioral conformity with small-to-medium effect sizes: a meta-analysis of the proteus effect. *Media Psychology*, 23(5):651–675, 2020.

[45] D. Roth and M. E. Latoschik. Construction of the virtual embodiment questionnaire (veq). *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3546–3556, 2020.

[46] D. Roth, J.-L. Lugrin, M. E. Latoschik, and S. Huber. Alpha ivbo-construction of a scale to measure the illusion of virtual body ownership. In *Proceedings of the 2017 CHI conference extended abstracts on human factors in computing systems*, pp. 2875–2883, 2017.

[47] S. Saito, T. Simon, J. Saragih, and H. Joo. 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.

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

[49] J. Starck and A. Hilton. Model-based multiple view reconstruction of people. In *IEEE international conference on computer vision*, pp. 915–922, 2003.

[50] F. Tan, H. Zhu, Z. Cui, S. Zhu, M. Pollefeys, and P. Tan. Self-supervised human depth estimation from monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 650–659, 2020.

[51] T. Waltemate, D. Gall, D. Roth, M. Botsch, and M. E. Latoschik. The impact of avatar personalization and immersion on virtual body ownership, presence, and emotional response. *IEEE transactions on visualization and computer graphics*, 24(4):1643–1652, 2018.

[52] J. Wang, Y. Zhong, Y. Li, C. Zhang, and Y. Wei. Re-identification supervised texture generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11846–11856, 2019.

[53] S. Wang, M. Mihajlovic, Q. Ma, A. Geiger, and S. Tang. Metaavatar: Learning animatable clothed human models from few depth images. *Advances in Neural Information Processing Systems*, 34:2810–2822, 2021.

[54] T. Wang, N. Sarafianos, M.-H. Yang, and T. Tung. Animatable neural radiance fields from monocular rgb-d. *arXiv preprint arXiv:2204.01218*, 2022.

[55] E. Wolf, M. L. Fiedler, N. Döllinger, C. Wienrich, and M. E. Latoschik. Exploring presence, avatar embodiment, and body perception with a holographic augmented reality mirror. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 350–359. IEEE, 2022.

[56] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, D. Mehta, H.-P. Seidel, and C. Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018.

[57] N. Yee and J. Bailenson. The proteus effect: The effect of transformed self-representation on behavior. *Human communication research*, 33(3):271–290, 2007.

[58] Y. Zheng, V. F. Abrevaya, M. C. Bühler, X. Chen, M. J. Black, and O. Hilliges. Im avatar: Implicit morphable head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13545–13555, 2022.

[59] T. Zhi, C. Lassner, T. Tung, C. Stoll, S. G. Narasimhan, and M. Vo. Texmesh: Reconstructing detailed human texture and geometry from rgb-d video. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pp. 492–509. Springer, 2020.