A Framework for Realistic Clothed Avatar Reconstruction and Animation

by

Mengyuan Wang

Thesis submitted to the University of Ottawa

In partial fulfillment of the requirements

For the M.A.Sc. degree in

Electrical and Computer Engineering

School of Electrical Engineering and Computer Science

Faculty of Engineering

University of Ottawa

Ottawa, Canada

Abstract

The rise of metaverse technologies has sparked a growing need for reconstructing 3D realistic avatars, which could give users a real experience in various fields, such as fashion, movies, games, and so on. An avatar is essentially a digital replication of a physical human and plays a crucial role in this evolving field. Traditional approaches rely on expensive equipment and advanced algorithms for 3D avatar reconstruction, which is impractical for users. Subsequently, methods are based on images or videos reconstructing avatars without texture or avatars with texture but not animation. Therefore, this paper proposes a comprehensive framework for realistic clothed avatar reconstruction and animation utilizing a single video as input. The framework comprises four modules: video acquisition, video preprocessing, avatar reconstruction, and avatar animation. Videos can be captured in any wild environment with different devices, such as cameras or phones. 2D keypoints and masks are extracted through deep learning models and input into the reconstruction module to generate a precise 3D avatar mesh model and a full texture image. Avatar animation is achieved through an auto-rigger system and a bake action tool. Meanwhile, extensive experiments are conducted to evaluate the impact of four factors on avatar reconstruction performance: mask generation method, background, speed, and rotation. The results indicate that rotation has the most significant effect on avatar reconstruction, followed by the mask generation method, background, and speed. Finally, a four-layer metaverse is proposed and developed using blockchain and artificial intelligence (AI) technologies to verify the availability and generality of our avatar framework. Through this exploration, we not only showcase the potential of our approach but also underscore the convergence of digital realms and emerging technologies that are reshaping the future of human interaction and expression.

Acknowledgements

I wish to convey my profound gratitude to my mentor, Prof. Abdulmotaleb El Saddik, for granting me the privilege of being a part of MCRlab. It is a tremendous honor to engage in research under his guidance. His unwavering passion, visionary outlook, and motivational spirit have served as a wellspring of inspiration, propelling me to surmount challenges and diligently work toward my goals. Prof. El Saddik's insights have not only enriched my research endeavors but have also provided invaluable life guidance. I am genuinely appreciative of his kindness, sense of humor, and the friendship he has extended to me throughout our collaboration.

I would also like to express my gratitude to all the members of MCRlab. Their advice and mentorship during my thesis work have been invaluable, and their thoughtful questions during lab workshops have consistently illuminated the path forward for me. In particular, I extend my heartfelt thanks to Haopeng Wang for his patient mentorship and selfless assistance.

Lastly, I would like to extend my heartfelt appreciation to my parents for their unwavering support, both emotionally and financially. When I was anxious about the thesis, my friends supported and accompanied me. Their love, kindness, and understanding have been a constant source of motivation, spurring me to complete my thesis and prepare for my future.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Background

The Metaverse has recently entered mainstream conversation and captured widespread attention when Facebook rebranded itself as Meta. The concept of the metaverse, in its simplest form, describes a fully immersive virtual world where individuals come together to socialize, play, and work, as discussed in a study by Laeeq et al. [35]. This digital

realm represents a simulated environment that amalgamates various cutting-edge technologies, including augmented reality (AR), virtual reality (VR), blockchain, artificial intelligence (AI), and principles of social media. The ultimate goal is to create spaces for user interaction that closely mimic the experiences of the physical world. A critical component of the metaverse's success is the development of lifelike avatars. These digital representations of users have taken on increased significance and complexity. The generation and animation of avatars have become pivotal across a multitude of applications, spanning augmented and virtual reality (AR/VR), entertainment, and the fashion industry. For instance, in the world of entertainment, animation studios extensively employ motion capture technology. This advanced technique allows them to capture and digitize the movements of real actors, subsequently transferring these movements to virtual characters. The outcome is an array of digital characters capable of delivering remarkably realistic and intricate performances, replete with nuanced actions and emotions. The Metaverse with its ever-evolving avatars and immersive environments is poised to revolutionize how people interact, work, and entertain themselves in the digital age. This offers a glimpse into the limitless possibilities of this emerging virtual realm.

Traditional methods for reconstructing avatars typically rely on expensive 3D acquisition systems and intricate algorithms, these methods are represented in [32, 30, 67, 60, 68]. Nevertheless, the impractical size and cost of these scanning devices limit their viability for everyday consumers. Consequently, alternative approaches have been proposed to reconstruct avatars more conveniently. The majority of these methods, as mentioned in references [15, 20, 33, 34, 49, 52, 66], involve generating 3D human body avatars from images or videos by estimating parameters of statistical 3D mesh models such as SCAPE[5], Adam[31], SMPL/SMPL-X [41, 48], GHUM [62], or STAR [46], or through the use of implicit surface models such as imGHUM [3] and LEAP [43]. However, these models are typically trained by people who wear minimal clothes, which means they struggle to capture the shape and appearance variations of clothing. More recently, some methods [63, 65, 70, 61, 54] have been developed to restore the body with texture.

However, these methods typically only account for the folds and textures of the clothing, without considering the full range of color variations in the clothing.

In the pursuit of reconstructing clothed avatars, researchers have turned to the use of multi-view images or videos as these sources provide a wealth of information regarding both the human body and the clothing it adorns. However, a persistent challenge in this process is the accurate recovery of colors for occluded or concealed parts of the body. Even though there have been attempts, such as the work by Natsume et al. [44], to infer the colors of occluded regions using synthetic images, the results often exhibit disparities when compared to the original colors of visible body parts. This inconsistency can lead to a phenomenon known as chromatic aberration, which affects the overall visual quality and realism of the 3D avatar. To address these issues, various markerless monocular approaches for capturing human performance have emerged. These approaches, exemplified by Habermann [23] and MonoPerfCap [63], predominantly rely on explicit mesh representations. They utilize joint positions and silhouettes to estimate the pose and shape of a subject within a single frame. The development of deep learning networks has been instrumental in advancing the field of clothed avatar reconstruction, leading to significant achievements and a deeper understanding of the complexities involved in this process.

While strides have been made, it is worth noting that there is still room for improvement in the quality of clothed avatars generated by existing methods, including those mentioned in [25, 44, 26]. These approaches have contributed to the recovery of avatars with clothing but often fall short of achieving a level of realism that satisfies the demands of various applications. To address these limitations, we propose a novel framework aimed at generating high-quality and realistic avatars, as shown in Fig.1.1, particularly focusing on capturing the intricacies of clothing, thereby pushing the boundaries of avatar realism and enhancing the potential applications of such technology.

While the methods mentioned above are designed for the reconstruction of 3D avatars, a common limitation is that these avatars are static and lack animation. Animation in

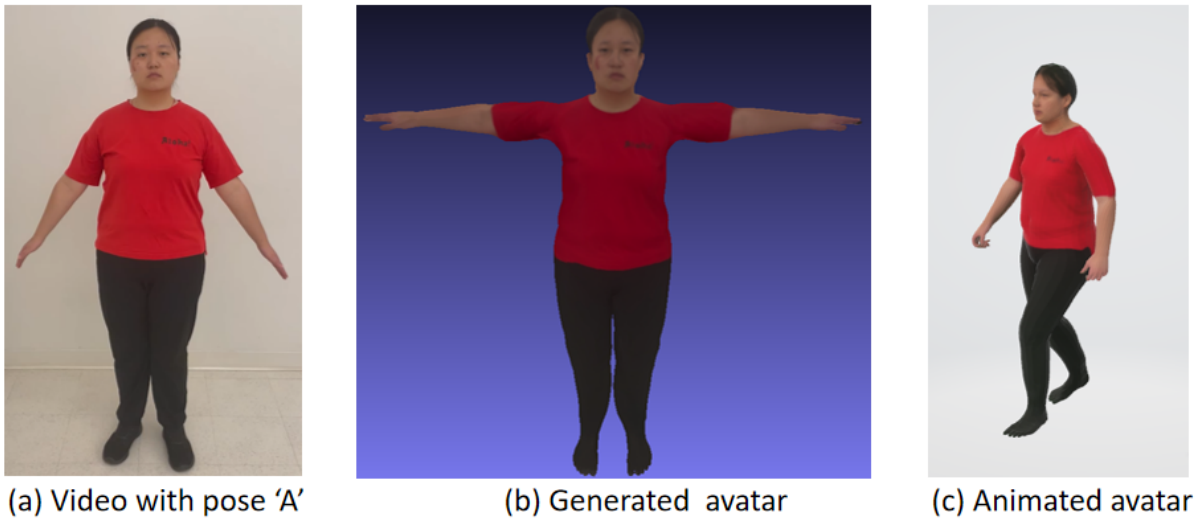


Figure 1.1: Overview of the proposed framework. (a) A subject rotates with pose 'A' in a video. (b) Generated a realistic clothed avatar from video (a). (c) Animated the generated avatar.

3D avatars typically involves a process called rigging, which entails creating a bone structure for the avatar, akin to a puppet, that can be manipulated to produce movements. Traditional skeletal animation often necessitates manual rigging, a skilled task that involves placing joints within the character manually [6]. While manual rigging offers a high level of precision and applicability, it is a time-consuming process and cannot be reused. To address the challenge of adding animations to avatars, various approaches are demonstrated in [21, 17, 9, 14, 55]. These methods aim to create animatable representations of human figures by utilizing input in the form of point clouds or meshes. However, a drawback in using human point clouds is their limited usability in many scenarios.

In our work, we propose a comprehensive framework designed to generate realistic clothed avatars with multiple movements. We take a video in which the person with an "A" pose rotates in a circle in front of the camera, as shown in Figure 1.1 (a). This video is subjected to a series of processing steps and leverages corresponding deep-learning models to obtain masks and keypoints. Following this preprocessing stage, the collected

data is then fed into a reconstruction module for generating a 3D avatar mesh model and a texture image.

The results of this reconstruction are subsequently rendered and visualized using a graphic tool to create a highly realistic avatar, as shown in Figure 1.1 (b). Our framework allows avatars to have multiple actions instead of a single action by utilizing an auto-rigger system and a bake action tool, as shown in Figure 1.1 (c), which automates the process of rigging, saving time and offers versatility in animating the avatar with a range of actions, thereby enhancing the overall appeal and usability of the generated avatars.

1.2 Challenges

The aim of our research is to reconstruct and animate a realistic clothed avatar, covering body shape, hair, facial features, clothing texture, and color. Current methods either reconstruct a 3D human body model without clothing texture or reconstruct avatars with clothing texture but lack animations. Some approaches involve a very complex process to independently complete avatar reconstruction and animation. In response to these challenges, we propose a framework capable of reconstructing and animating realistic clothed avatars seamlessly.

Reconstructing realistic clothed textures from a video is a challenging task, and mask generation plays a crucial role in this process. Masks are essential for accurately separating and delineating different elements in the scene, particularly the clothing and the underlying body. The importance of mask generation lies in its ability to distinguish between regions covered by clothing and those representing the exposed body. In the absence of accurate masks, the texture reconstruction process can result in misalignment or blending errors, which compromise the fidelity of the final result.

The animation process depends on rigging, which requires linking joints or relative positions on the character through virtual controllers. Previous methods rig skeletons by hand, which results in a lot of work and is error-prone. Subsequently, some works

utilize point clouds or mesh sequences as input to animate models, however, most of the time human point clouds cannot be used. Besides, an avatar can do only one action. We use automatic rigger and bake action tools not only to simplify the process of animating avatars but also to allow avatars to perform a series of natural and coherent movements according to user preferences, instead of being limited to only one action.

In summary, the challenges of this thesis are concluded as follows:

- Our research aims to reconstruct and animate realistic clothed avatars, encompassing body shape, hair, facial features, clothing texture, and color. Current methods face challenges, either lacking clothing texture or animations. We propose a streamlined framework to address these limitations.

- Reconstructing realistic clothed avatars from a single video involves not only ensuring human accurate segmentation, texture misalignment, and blending issues but also handling occlusions recovery and enhancing the overall visual fidelity and realism of the reconstructed clothed avatar. Mask generation is a crucial step in reconstructing a realistically dressed avatar from video data.

- To simplify the avatar animation process and reduce the likelihood of errors, we have replaced manual rigging with automatic tools. This not only streamlines the process but also enables avatars to perform a series of natural and coherent movements based on user preferences, rather than being limited to a single action.

1.3 Objectives

The objectives of our thesis are stated as follows:

1. Propose a framework to reconstruct a realistic clothed avatar from a single video and animate this avatar with a series of natural and coherent movements using an auto-rigger system and bake action tool. Meanwhile, we develop a metaverse and merge avatars into the metaverse to verify the availability of our framework.

2. Utilize a mask generation model during preprocessing videos to enhance the fidelity of reconstructed avatars. This helps mitigate issues related to texture misalignment and blending. Additionally, for more accurate foreground separation, we compare three different mask generation methods and evaluate their impact on avatar reconstruction performance.

3. Using an auto-rigger system and a bake action tool to streamline the avatar animation process and decrease the risk of errors, provides a more effective alternative to manual rigging, which not only simplifies the overall process but also empowers avatars to execute a variety of natural and coherent movements according to user preferences.

1.4 Thesis Statement

The growing demand for reconstructing the 3D human body in the metaverse has heightened the focus on reconstructing realistic clothed avatars. However, many existing methods rely on complex equipment and algorithms for scanning and reconstructing high-quality avatars. Additionally, most existing methods generate mesh models or they have textures but they do not have animations. This paper presents a framework for realistic clothed avatar reconstruction and animation from a single video as input using just one camera. The ubiquitous availability of cameras and low cost of 3D human models will make it possible for people to digitize themselves and use the 3D human models for VR applications, entertainment, biometrics, and online shopping. For example, for gaming, players often customize their avatars with different clothing options, realistic clothing simulation adds depth and realism to character interactions and movements; for health and rehabilitation, realistic avatars can be used to create interactive exercises and simulations that help patients recover from injuries or surgeries. As well, our reconstructed avatars can be set to perform appropriate actions based on user preferences. for example some basic movements, the avatar with standing, walking, jumping, clapping, and sitting. The framework includes four modules: video acquisition, video processing,

avatar reconstruction, and avatar animation. The approach involves keypoint detection, mask generation, texture generation, and auto-rigger animation. Meanwhile, extensive experiments are conducted to test and evaluate the impact of four factors on avatar reconstruction performance: mask generation method, background, speed, and rotation. Additionally, a four-layer metaverse is proposed and developed using blockchain and AI to verify the availability of our framework.

1.5 Contribution

The main contributions of our research are summarized as follows:

1. We propose a framework for realistic avatar reconstruction and animation. The realistic avatar can be reconstructed from a monocular RGB video using just one camera. Additionally, the avatar can do a series of coherent and natural movements. It integrates four stages: video acquisition, video preprocessing, avatar reconstruction, and avatar animation.

2. We test and evaluate four factors: mask generation method, background, speed and rotation. Experiment results indicate that rotation has the greatest impact on avatar reconstruction, followed by the mask generation method and background. The speed plays a minor role.

3. We propose and develop a four-layer metaverse using blockchain to write and deploy smart contracts for transactions and AI to verify the availability and generality of our framework. Avatars are capable of roaming randomly, asking questions, placing orders, and conducting transactions.

1.6 Thesis Outline

The thesis is organized as follows:

- Chapter 2 introduces the related works regarding the reconstruction and animation

of avatars.

- Chapter 3 provides an overview of the framework and introduces each module in detail, including video acquisition, video preprocessing, avatar reconstruction and avatar animation.

- Chapter 4 presents the avatar reconstruction results, conducts a user study and evaluates four factors: mask generation method, background, speed and rotation. Also, this chapter offers a detailed analysis and discussion.

- Chapter 5 shows a case study, a four-layer metaverse is proposed and developed and an avatar is merged into the metaverse using blockchain and AI technologies to verify the availability of realistic avatars reconstructed in real life.

- Chapter 6 concludes this thesis and points out the limitations of this work. Future work is discussed in the end.

Chapter 2

Related Work

The exploration of 3D avatars encompasses a wide-ranging domain of study, which includes aspects like full-body human reconstruction, clothed textures, and animation. The subsequent sections delve into the various methodologies employed in the realm of 3D avatars. In Section 2.1, avatar reconstruction-related literature is presented. In Section 2.2, the related works of animating an avatar are described.

2.1 Avatar Reconstruction

2.1.1 Traditional Reconstruction Methods

The traditional 3D human reconstruction methods directly reconstruct the high-dimensional human body surface mesh instead of the low-dimensional human body parameter representation in the parametric method, which is also called a non-parametric method. It generally requires the use of special data collection equipment, such as laser scanners, depth cameras, etc. For instance, Kanade et al. [32] employed an elaborate setup, including a substantial dome with a five-meter diameter, outfitted with an array of 51 cameras. This elaborate apparatus enabled the digitization of real objects, producing free-viewpoint videos, and advancing the field of 3D reconstruction.

Taking the Vitronic commercial human body scanner as another example, the person to be scanned wears tight-fitting clothing and stands on the platform in the center of the scanner. Then, four high-speed laser scanning probes move rapidly from the top of the head downward, scanning the entire body. Each laser scanning probe acquires a local point cloud from a single viewpoint. Within a few seconds after the scanning is complete, the accompanying software can directly reconstruct a three-dimensional human body mesh. Three-dimensional human body scanners are capable of quickly and accurately obtaining static three-dimensional human body models, and they can even reconstruct hard-to-reach body parts such as the armpits and groin. They are commonly used in applications such as movies, games, body measurements, and custom clothing. They can also be used to create a human body database, for example, SCAPE [5]. However, laser scanners are expensive and bulky. Many researchers are also attempting non-parametric human body reconstruction using consumer-grade depth cameras like Kinect.

KinectFusion [45] is a classic method for reconstructing three-dimensional scenes using Kinect. It reconstructs 3D scenes by incrementally merging the geometric information collected. Inspired by KinectFusion [45], subsequent researchers [57, 16]. proposes

non-parametric reconstruction methods for the human body. These methods typically require depth maps of the human body from multiple views, which can be obtained by multiple depth cameras placed around the subject [57, 59] or by using a single depth camera in motion around the subject [16]. However, during the scanning process, it is necessary for the subject to maintain a specific and unchanging posture and wear as little clothing as possible. Tong et al. [29] are setting up a human body scanning and reconstruction system, which includes three carefully positioned Kinect cameras, and this layout helps eliminate point cloud overlap areas. They first construct a rough human body template using depth maps captured by the three Kinect cameras. Subsequently, they use a non-rigid global registration method to align multiple depth frames for the complete reconstruction of the human body. Li et al. [38] are also using a single Kinect camera. They require the participants to stand in a fixed position and rotate by 45 degrees eight times to capture depth images from eight different viewpoints. Subsequently, they use the Iterative Closest Point (ICP) algorithm to merge the point clouds from the eight viewpoints. They utilize Poisson Surface Reconstruction to obtain a three-dimensional human body mesh model. Similar to Kinect-based parameterized human body reconstruction methods, this approach is susceptible to noise in Kinect's depth data. Nevertheless, slight variations in the participants' postures during data capture can lead to a decrease in reconstruction accuracy.

3D human body shape refers to the geometric shape model of the human body represented in the form of a three-dimensional mesh. Compared with the above non-parametric method of 3D human body reconstruction, the parametric human body shape reconstruction method relies on a human body parameterized model based on statistics and only needs a set of low-dimensional vectors (ie, human body parameters) to describe the human body shape. Currently, common parametric human body models include SCAPE[5], SMPL [41], SMPL-X [48], etc.

Taking SCAPE as an example, it defines two independent low-dimensional parameter spaces: the human body shape space and the human body pose space. Given a set of

human body shape parameters and human body pose parameters within these spaces, a human body shape can be directly synthesized. The human body shape space is represented by a subspace obtained through Principal Component Analysis (PCA) on a database of human bodies with the same pose but different shapes. The shape parameters correspond to the coefficients of the bases in this subspace. The parameter variations on the SCAPE shape basis affect changes in human body shape. SCAPE's pose parameters are represented by the rotations of 17 human body parts relative to the corresponding parts of a standard template human body. Following the success of the SCAPE model, several researchers have made continuous improvements and proposed various upgraded versions, some of the more notable ones being Blend Scape [24], Breath Scape [58], S-Scape [27], and so on. However, SCAPE model deformations rely on the rotation deformation of triangular faces, rather than the more commonly used vertex deformation methods in animation software (such as skeletal skinning). As a result, the human body geometric models generated by SCAPE are challenging to directly use in existing animation software, like Maya or Blender.

Subsequently, the Max Planck Institute for Intelligent Systems in Germany released an open-source human body parameterized model based on vertex deformation called SMPL. The SMPL model is also controlled by human body shape parameters and human body pose parameters. Its shape parameters are the same as SCAPE's shape parameters, represented using parameters extracted from shape deformation bases via PCA. Pose parameters are represented by global body rotations and rotations of 23 joints, and they are used for human body pose deformation through Linear Blend Skinning (LBS).

SMPLify [49] introduces a human body two-dimensional pose estimation model based on convolutional neural networks. They optimize SMPL parameters, including body shape and pose parameters, by minimizing the registration error between the synthesized three-dimensional human pose and the reprojected two-dimensional keypoints obtained from detection. They also incorporate human penetration constraints to reduce ambiguities in the transition from two-dimensional to three-dimensional. However, this method

does not impose constraints on the human body shape and is prone to getting stuck in local optima, leading to reconstruction failures. Building upon SMPLify, Lassner et al. [36] introduce additional human body landmark constraints (91 landmarks), resulting in more accurate pose reconstruction. They also propose using a Random Forest model to learn the mapping relationship between the human body contour and SMPL body shape parameters. However, the quality of the predicted human body contours is relatively poor, significantly affecting the accuracy of body shape predictions.

2.1.2 Mesh Models

In recent years, parameterized human body shape reconstruction methods based on deep learning have gained popularity. Dibra et al.[18] were among the first to utilize Convolutional Neural Networks (CNN) to estimate human body shape parameters. They used specific viewpoint masks of standing human bodies as inputs to the CNN and directly regressed to SCAPE's body shape parameters. Unlike manually designed features, CNNs can automatically extract body shape features, resulting in more accurate body shape predictions.

Subsequently, Dibra et al. [19] further improved the accuracy of body shape prediction. They first learned a feature latent space describing the same body shape under different viewpoints with a fixed posture, and then learned a regression model from this latent space to body shape parameters. This method can reliably predict body shape parameters from human body mask images in various viewpoints. Single-view human body mask images often lack some body shape information, such as the beer belly in males, which cannot be displayed on frontal mask images. To address this issue, Ji et al. [28] designed a novel dual-stream network structure that simultaneously takes front and side human body masks as inputs to predict SCAPE shape parameters.

HMR [33] incorporates the reprojection registration error of human body keypoints into the loss function to supervise SMPL's pose and body shape parameters. HMR draws

inspiration from Generative Adversarial Networks (GAN) [22] and includes a discriminator in the loss function to validate the legitimacy of the predicted human parameters. However, this method does not effectively supervise human body shape, resulting in predictions that are closer to average body shapes, and significant differences in body poses compared to the input images.

Xu et al. [64] innovatively introduce dense re-projection error of human body mesh vertices into the loss function. They utilize the IUV map predicted by Densepose [37], representing the correspondence between dense mesh vertices and image pixels, as input. They regress the human body mesh, render the predicted IUV map using a Differential Renderer, and calculate registration errors between the rendered and input IUV maps.

Subsequently, some researchers [34, 39] incorporate graph convolution from general object 3D reconstruction into 3D human body reconstruction. Kolotours et al. [34] explicitly model the topological relationships of the human body mesh using graph convolution. They design a progressively optimized network structure that reconstructs from low-resolution mesh to high-resolution mesh. This method achieves more accurate reconstruction of 3D human poses but exhibits significant differences from the input images in terms of body shape. [39] introduces a novel approach to deformable shape completion utilizing Graph Convolutional Autoencoders (GCAEs). Focused on addressing the challenge of reconstructing 3D shapes with missing or incomplete information, the method employs graph convolutional operations to capture intricate spatial relationships within 3D point clouds. By representing the point cloud as a graph and leveraging autoencoder architecture, the GCAE effectively predicts missing regions in a deformable and context-aware manner. However, one potential disadvantage of this method for 3D human body reconstruction is its sensitivity to noisy data, potentially leading to less accurate results in the presence of imperfections or incomplete information in the input scans.

2.1.3 Texture Models

The limitations of 3D mesh models have become apparent, making them less practical for widespread real-life applications where high fidelity is a crucial requirement. Fortunately, several methods, as introduced by [63, 65, 70, 61], have risen to the challenge by enabling 3D human reconstruction with the inclusion of texture.

DoubleFusion [65] presents an innovative method for real-time 3D human performance capture using a single depth sensor. The DoubleFusion system simultaneously reconstructs both the outer surface of a human subject and their inner body shape. This approach enables the capture of detailed body movements and deformations during dynamic performances. By fusing these two aspects, DoubleFusion achieves highly realistic and accurate representations of human performances, making it a valuable tool for applications in fields such as animation, gaming, and virtual reality. However, reliance on a single depth sensor may limit the system's adaptability to diverse environments and poses challenges in capturing occluded areas. The paper [70] presents an innovative approach to estimate detailed 3D human shapes from a single 2D image. The method utilizes a hierarchical mesh deformation framework, allowing for the precise reconstruction of intricate human body shapes, even in cases involving clothing and challenging poses. By hierarchically optimizing both the global and local deformations of the mesh to achieve a high level of accuracy and realistic representation. MonoPerfCap [63] presents a groundbreaking approach to human performance capture, leveraging monocular video input and deep learning techniques. The paper addresses the challenges of pose estimation, enabling 3D reconstruction of human body shapes from 2D pose data. The captured textures are meticulously mapped onto the 3D shapes to create high-quality avatars. However, the reconstruction accuracy may be affected when dealing with subjects wearing loose or non-standard clothing. [61] presents a novel approach for the temporal reconstruction of detailed clothing dynamics from monocular RGB video sequences. The paper proposes an end-to-end framework that leverages deep learning techniques to capture clothing de-

formations and dynamics over time. However, although these two methods are effective, they may involve computationally intensive processes, potentially impacting real-time applications or requiring substantial computational resources. DeepHuman [69] leverages a low-resolution 3D voxel representation of the human body encoded by parametric models, as well as a 2D semantic map, to provide a rough representation of the 3D human form. It takes these representations, alongside RGB images, as input and utilizes a voxel translation model to predict the geometric details of the human body surface. This approach involves the integration of parametric modeling concepts, which was subsequently adopted by [70]. However, 3D human representations using voxels often face challenges such as excessive computational demands and high memory usage. As a result, there is a need to lower the voxel resolution for prediction, but this comes at the cost of losing the fine-grained surface details of the human body. Additionally, it's important to note that these models typically account for clothing wrinkles but often do not consider clothing color information, leaving room for further developments in this aspect.

2.1.4 Clothed Models

Texture generation is a critical component in the creation of lifelike avatars, as textures enable the conveyance of material properties that cannot be captured by surface geometry models alone. Achieving a coherent texture involves integrating texture fragments collected from various angles, making it an essential step in the avatar creation process. Several methods [26, 37, 1] initially estimate the 3D pose and a portion of the texture and then employ a GAN network to generate the complete texture. However, this approach relies on accurate pose estimation as a prerequisite.

ARCH [26] approach addresses the challenges associated with capturing the intricate details of clothed human subjects, including clothing wrinkles, textures, and dynamic movements. By employing deep learning networks and a multi-view image dataset, the paper presents a comprehensive framework that not only reconstructs the 3D avatars

but also animates them with various actions. However, this method may require a substantial amount of training data and may not perform optimally with limited datasets. The accuracy of the results reconstructed is not high in scenarios with insufficient image views or challenging lighting conditions. Additionally, the application of ARCH for real-time performance capture and animation may necessitate powerful hardware, which could be a limitation for some users and applications. [37]is a data-driven approach that revolutionizes the creation of detailed clothing textures for 3D avatars. It employs deep learning to extract intricate features from a single input image, such as clothing patterns and structures. The core innovation lies in texture synthesis, where deep learning models are trained to predict clothing textures from various viewpoints. These synthesized textures are merged into a comprehensive 360-degree texture map, capturing clothing details and nuances from all angles. Addressing the challenging task of extracting comprehensive 3D body shapes from 2D visual input, [1] leverages a novel approach that combines texture synthesis and geometric reasoning. The model employs a convolutional neural network to predict both surface displacement maps and detailed texture maps, enabling the generation of a high-fidelity 3D representation. However, the training process is hindered by an over-reliance on 3D human body ground truth scans, requiring precise pose estimation as crucial prior and facing challenges in handling complex deformations like long hair and skirts.

The work in Huang et al. [25] has trained a network to extract features related to spatial point projection onto image positions, along with the prediction of occupancy values and RGB values combined with the point's position. Despite this approach, the resulting texture quality is not consistently high.

Saito et al.'s research [53] proposes a method for high-resolution digitization of clothed human bodies. It uses a pixel-aligned implicit function that aligns 3D surface points with 2D pixels for detailed human shape and clothing reconstruction. PIFu achieves high-quality clothed human digitization and allows for the generation of detailed texture maps. However, this method demands precise masks and camera parameters, making it

susceptible to reconstruction failures caused by even minor disturbances in the field of view (FOV) and camera distance.

Recent research [2, 4] proposes a multi-frame joint estimation of SMPL+D in a canonical T-pose, subsequently projecting it back to each frame to extract texture fusion. This has resulted in significant improvements in clothed avatar reconstruction. Nevertheless, for a truly realistic and immersive experience in the metaverse, the focus must not only be on creating more realistic avatars but also on animating them effectively.

2.2 Avatar Animation

Avatar animation plays a pivotal role in a wide range of applications, spanning from gaming and visual effects to digital human representation. The animation process hinges on rigging, which entails associating joints or relevant positions on the character through virtual controllers. This rigging procedure empowers us to orchestrate diverse actions for the model, such as walking, clapping, or jumping.

The works in Ilya Baran et al. [6] introduce a system that adapts to the character's skeleton, enabling the animation of 3D characters based on skeleton motion data. However, challenges may arise when dealing with characters characterized by substantial body mass and slender limbs, potentially leading to issues in connecting the limbs and affecting overall performance.

Various techniques for animating human characters have been developed by researchers [21, 17, 9, 14], which involve utilizing sequences of point clouds or meshes as input. Saito et al., as described in [55], presents an innovative approach. They initially train a skinning weight network to transform the body based on point clouds into canonical poses and then employ the transformed point clouds to learn implicit shapes. However, the human point clouds cannot be used the majority of the time.

In an endeavor to reduce the demands on the capture system, recent works, such as those highlighted in [40] and [56], strive to reconstruct 3D animated human models from

minimal multi-view images or videos. In a similar vein to works by Park et al. [47] and Pumarola et al. [51], Peng et al. [50] introduce Animatable NeRF, representing a video through a canonical NeRF and a series of deformation fields. These fields establish correspondences between observed spaces and canonical spaces, with the deformation field articulated through a combination of the human skeleton and a blend weight field based on a skeleton-driven deformation framework, enabling dynamic animations. Presently, the market offers several robust 3D animation applications equipped with built-in rigging functionalities or components. Notable examples include 3ds Max's Biped tool, Maya's Rig, Blender's Rigify, and Cryptic Studio's CrypticAR, all of which empower users with the capacity to animate avatars seamlessly.

Chapter 3

Framework

This paper introduces a comprehensive framework designed to reconstruct and animate a realistic clothed avatar using only a single RGB video as input. The framework consists of four main components: video acquisition, video preprocessing, avatar reconstruction, and avatar animation, each playing a crucial role in achieving the desired outcome. Figure 3.1 provides a visual representation of these integral components.

In the initial stage of video acquisition, footage of a subject performing pose "A" and

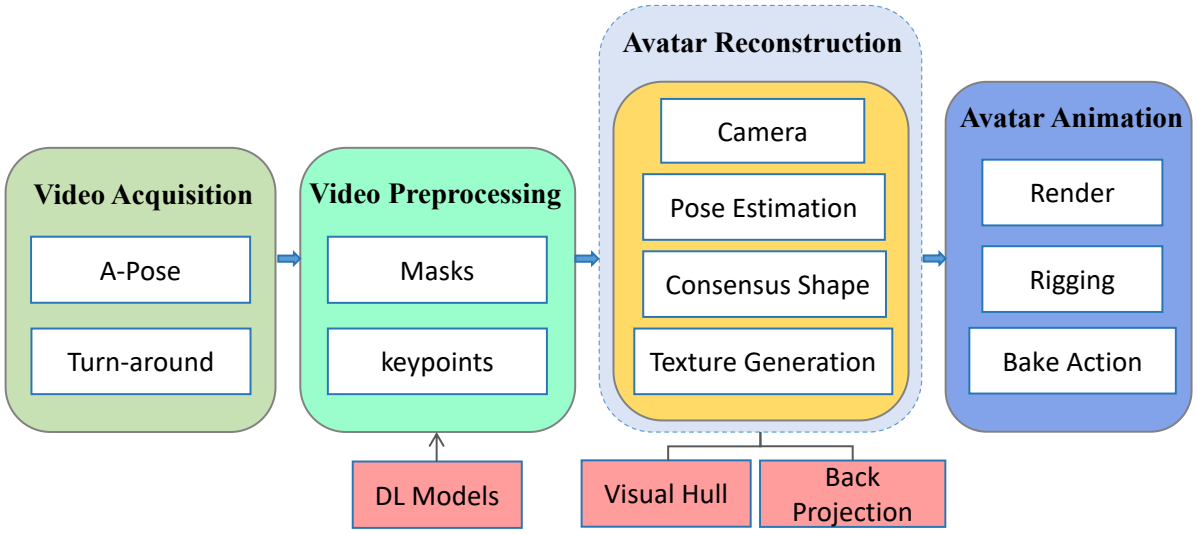


Figure 3.1: The proposed framework includes four modules: video acquisition, video preprocessing, avatar reconstruction and avatar animation. Video is processed by deep learning (DL) models. The visual hull [2] method is used to reconstruct the 3D avatar model.

feet apart is captured in front of a camera. By doing so, the texture in the obscured areas such as armpits and crotch could be restored to the greatest extent.

In the next step, two deep learning models are used to prepare data. The one is the OpenPose [11] model to obtain 2D keypoints of each frame. OpenPose utilizes deep neural networks, specifically convolutional neural networks (CNNs), to perform keypoint detection and pose estimation from images and videos. It is widely used in various fields such as computer vision, human-computer interaction, and motion analysis. OpenPose takes as input images or videos containing one or more persons whose poses need to be estimated. The input can be in various formats, including images, video streams, or pre-recorded video files. Before performing pose estimation, OpenPose preprocesses the input images or frames such as resizing, normalization, and color space conversion. The core functionality is the detection of keypoints, which represent key body joints and parts such as the head, neck, shoulders, elbows, wrists, hips, knees, and ankles.

Once keypoints are detected, OpenPose estimates the pose of each person in the image or video frame by connecting the detected keypoints to form a skeletal representation. The output of OpenPose includes the positions of keypoints and the corresponding pose skeletons for each detected person in the input images or frames. These results can be visualized, analyzed, or used as input for downstream applications such as action recognition, gesture recognition, or human-computer interaction. OpenPose is designed for real-time performance, allowing it to process images and video frames at high speed, typically achieving frame rates suitable for interactive applications. This real-time capability makes OpenPose suitable for use in applications such as interactive installations, sports analysis, virtual reality, and augmented reality.

Another one is the mask generation model. It includes three deep learning models, One-Shot Video Object Segmentation, Semantic Guided Human Matting and PP-Matting respectively. In this work, PP-Matting is used eventually because of high-accuracy segmentation. PP-Matting is a deep learning-based method for solving the problem of image matting. Image matting refers to the process of extracting a foreground object from an image while preserving fine details such as hair or fur. PP-Matting is designed to achieve high-quality results efficiently, especially in challenging scenarios where traditional matting methods may struggle. PP-Matting takes as input an image containing a foreground object that needs to be extracted. Additionally, it may also take an optional trimap, which provides rough estimates of the foreground, background, and unknown regions in the image. If a trimap is provided, PP-Matting first preprocesses it to create a probability map indicating the likelihood of each pixel belonging to the foreground or background. PP-Matting employs a deep neural network architecture to estimate alpha matte probabilities for each pixel in the image. The network is trained on a large dataset of paired images and ground truth alpha mattes. One key feature of this model is its propagation mechanism, where the estimated alpha matte probabilities are refined iteratively. This propagation process helps improve the accuracy of the matte estimation, especially in regions with complex textures or fine details. The output of

PP-Matting is a high-quality alpha matte, which accurately represents the transparency of each pixel in the foreground object. This matte can then be used to composite the foreground onto a new background or integrate it into other images or videos seamlessly. The acquired video then undergoes preprocessing, where a mask generation model and a keypoint detection model are employed to derive masks and keypoints, respectively. These processed data are subsequently utilized by the reconstruction module to generate a precise 3D avatar mesh model along with a complete texture image.

The processed data 2D keypoints and masks are then fed into the reconstruction module to generate both a precise 3D avatar mesh model and a full texture image. The avatar reconstruction module contains four steps: camera computation, pose estimation, consensus shape and texture generation. The core idea of reconstruction is to generalize a visual hull method to a video of people in motion and back-project the image color to all visible vertices. Finally, the texture image is applied to the 3D model to reconstruct a realistic clothed avatar with a graphic tool "Blender" [7]. The avatar animation allows the 3D avatar to engage in diverse actions based on user preferences, such as jumping, clapping, walking, dancing, sitting, etc. This animation is facilitated by an auto-rigger system that strategically places markers on several body joints, including the chin, wrists, elbows, knees, and groin. Each module within the framework operates with loose coupling, ensuring a degree of independence that allows for flexible updates and modifications to individual algorithms as needed. This design choice enhances the framework's adaptability and extensibility, enabling seamless integration of new modules and functionalities. Examples of potential expansions include 3D rendering for character attire alterations or scene adjustments, motion capture capabilities, and facial expression compositing, among others. The functionalities of each module are elaborated upon in subsequent sections, providing a detailed understanding of their roles within the framework. Overall, the proposed framework offers a robust solution for generating realistic clothed avatars from single RGB videos while also paving the way for future enhancements and advancements in the field.

3.1 Video Acquisition

The reconstruction of 3D realistic clothed avatars relies on comprehensive whole-body information obtained from a single video. To achieve this, the video must capture the subject's complete body during filming. Therefore, the subject should rotate in a circle in front of the camera is a key aspect of the video shoot. The poses should strike a balance—neither too casual nor too tight. This precaution aims to prevent insufficient whole-body information or occlusion of clothed texture in specific areas, such as the armpits and crotch. Failing to address this may lead to a notable absence of clothing texture in the occluded parts of the reconstructed results. Hence, it is crucial to ensure an accurate representation of the gaps between the subject's body and limbs. This is a precautionary measure to ensure clear separation of clothing from the human body in the 3D model during motion import. In alignment with these prerequisites, our research adopts posture "A" characterized by slightly downward-extended arms and spread-apart feet. This configuration allows the subject to execute a circular motion in front of the camera, as illustrated in Fig.3.2.

Figure 3.2: Video acquisition. Volunteers rotate with pose "A" in various environments.

3.2 Video Preprocessing

Several preprocessing steps are carried out once the qualified video is acquired, incorporating keypoint detection and mask generation. The initial steps involve resizing the video (set to 1080 * 1080 pixels) and converting it into individual video frames. The video could be converted into dependent frames using the Labelme tool or online. These frames are subsequently fed into both the keypoint detection and mask generation models for further processing.

The detection of human keypoints in each frame relies on OpenPose, a bottom-up detection algorithm employed in this context. OpenPose operates by initially identifying all keypoints of the human body and subsequently mapping these keypoints onto the human body, as shown in Fig.3.3 (b). The keypoints detected in this work are 18, it does not detect the heel, big toe, and small toe.

2D keypoints of the subject are required to adjust the SMPL model to better adapt to the pose and shape of the human body in each frame during the avatar model reconstruction. The Skinned Multi-Person Linear (SMPL) model is a versatile parametric 3D human body model. Comprising three main parameters—shape β, pose θ, and global translation—it adeptly captures the nuances of human body morphology and movement. The shape parameters govern overall body characteristics, while pose parameters dictate joint angles and positions. The global translation parameter accounts for the entire body's translation in 3D space. With approximately 6890 vertices, the SMPL model facilitates realistic simulations of human body shape and pose, making it invaluable for applications like virtual clothing fitting and motion capture. Also, we use the SMPL body model as a starting point for monocular 3D human shape reconstruction to overcome ambiguities

Meanwhile, achieving accurate segmentation of people's masks in images is of utmost importance in this process. Because mask segmentation has a direct impact on avatar

reconstruction. In this paper, PP-Matting [12] is utilized for mask generation in each frame. PP-Matting stands as a network architecture capable of directly achieving high-accuracy natural image matting without depending on auxiliary information as input, as illustrated in Fig.3.3 (c).

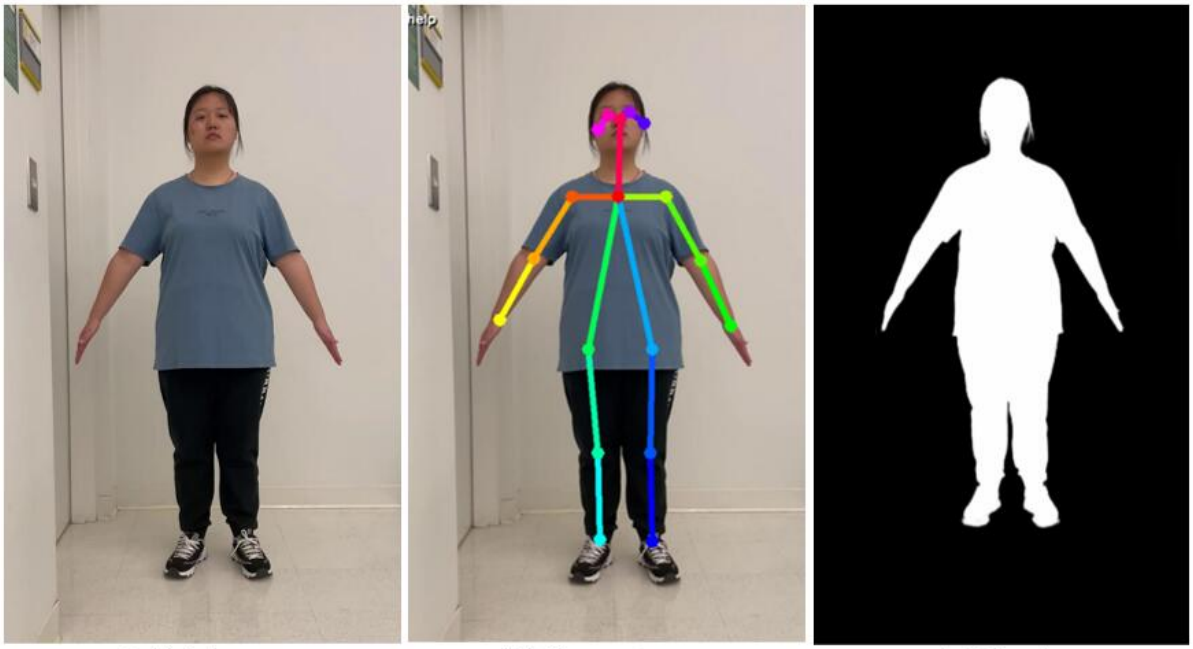


Figure 3.3: Video preprocessing. (a) The original video with the subject posing in the pose "A". (b) Generated 2D keypoints by a keypoint detection model. (c) Generated mask by a mask generation method.

3.3 Avatar Reconstruction

When the data is prepared, they are fed into the avatar reconstruction module for pose reconstruction, consistent shape optimization, and texture generation, which produces a 3D avatar mesh model and a corresponding clothed texture. The key idea in the avatar reconstruction module is to generalize the visual hull method [42] to videos of people in motion. The visual hull method builds on the assumption that foreground objects

in a scene can be discerned from the background, this method utilizes silhouettes as the 2D projection of the corresponding 3D foreground object. As a spatial object is observed from multiple viewpoints through perspective projection, the contour of the object is gathered from each viewpoint. The combination of these contours with the perspective projection center defines a roughly shaped cone in three dimensions. The visual hull of the object is determined by the intersection of all known outline contours with the corresponding perspective projection center. However, the person is moving in our video, which leads to minor inconsistencies induced by rotation despite asking the person to maintain the same pose. Consequently, it is necessary to eliminate these pose's inconsistencies. During the reconstruction process, we calculate the pixel focal length of the camera first. As stated earlier, the image dimensions are fixed at 1080 * 1080 pixels. Calculating the "pixel focal length" can be quite complex since standard imaging equipment usually provides focal length data in millimeters during capture. The value of the "pixel focal length" must be determined using the following equation:

$$\begin{aligned} FP_x &= (\frac{FM_x}{sensor\_size} * image\_x) \\ FP_y &= (\frac{FM_y}{sensor\_size} * image\_y) \end{aligned} \tag{3.1}$$

Within this context, FP_x and FP_y denote the focal lengths of pixels in the x and y dimensions, respectively. Similarly, FM_x and FM_y represent the focal lengths in millimeters along the x and y dimensions, respectively. The sensor_size serves as a critical hardware parameter for the camera, specifically indicating the CCD size of the sensor in millimeters. As for image_x and image_y, they denote the pixel count in the length and width of the image. In our processed videos, these values are standardized at 1080 pixels.

The original frames, keypoints, and masks are then inputted to estimate the initial body shape and 3D pose for each frame. This estimation is achieved by fitting the SMPL model to 2D detections, similar to [36, 8], providing a foundation for 3D-level pose information. The SMPL shape β and pose θ deformations are applied to a base

template T, which in the original SMPL model corresponds to the statistical mean shape in the training scans T_μ:

$$T(\theta, \beta, D) = T_\mu + B_s(\beta) + B_p(\theta) + D \tag{3.2}$$

where T_μ is statistical mean shape, $B_s(\beta)$ is shape-dependent deformations, $B_p(\theta)$ and D is offset.

Each silhouette point in every mask is linked to a 3D point in the body model through these fits. Depth information is then incorporated by integrating human segmentation masks with the 3D point positions. For pose reconstruction, the objective function contains a joint-based data term, three pose priors and a shape prior:

$$E_J(\beta, \theta, K, J_{\text{est}}) + \lambda_\theta E_\theta(\theta) + \lambda_\alpha E_\alpha(\theta) + \lambda_{\text{sp}} E_{\text{sp}}(\theta, \beta) + \lambda_\beta E_\beta(\beta) + S_{\text{silh}}(\theta) \tag{3.3}$$

where K is camera parameters and λ_θ, λ_α, λ_{sp}, λ_β are scalar weights. E_J is a joint-based data term that penalizes the weights 2D distance between estimated joints and corresponding projected SMPL joints. $\lambda_\theta E_\theta(\theta)$ is to fit SMPL poses to a public dataset to get average poses. $\lambda_\alpha E_\alpha(\theta)$ is a pose prior to penalizing elbows and knees that bend unnaturally. $\lambda_{\text{sp}} E_{\text{sp}}(\theta, \beta)$ term defines an interpenetration error term that exploits the capsule approximation. $\lambda_\beta E_\beta(\beta)$ term is a shape prior. The $S_{\text{silh}}(\theta)$ is a silhouette term.

Next, a set of rays from the camera to silhouette points defines a constraint cone. The intersection of these cones forms a visual hull. Because the person is moving, and poses are changing in our video, we need to remove the different poses caused by previous and behind frames to make each frame the same first and then reconstruct the shape. For pose inconsistency, we invert the SMPL function to adjust each ray from an A-pose to a canonical T-pose, this process is called "unpose". In SMPL, every vertex deforms according to the following equation:

$$\mathbf{v}'_i = \sum_{k=1}^{K} w_{k,i} G_k(\boldsymbol{\theta}, J(\boldsymbol{\beta})) \left(\mathbf{v}_i + b_{s,i}(\boldsymbol{\beta}) + b_{P,i}(\boldsymbol{\theta})\right) \tag{3.4}$$

where G_k is the global transformation of joint k. $b_{s,i}(\boldsymbol{\beta})$ and $b_{P,i}(\boldsymbol{\theta})$ are elements of $b_s(\beta)$ and $b_p(\theta)$ corresponding to vertex. For every ray r, we find its closest 3D model input. The inverse transformation applied to a ray r corresponding to model point $\mathbf{v}'_i$:

$$\mathbf{r} = \left(\sum_{k=1}^{K} w_{k,i} G_k(\boldsymbol{\theta}, J(\boldsymbol{\beta})) \right)^{-1} \mathbf{r}' - b_{P,i}(\boldsymbol{\theta}) \tag{3.5}$$

Doing this for every ray effectively unposes the silhouette cone and places constraints on a canonical T-pose.

For shape reconstruction, given the set of unposed rays, we formulate an optimization in the canonical frame:

$$E_{cons} = E_{data} + \omega_{lp} E_{lp} + \omega_{var} E_{var} + \omega_{sym} E_{sym} \tag{3.6}$$

where the objective E_{cons} consists of a data term E_{data} and three regularization term. E_{lp} is a Laplacian term, it enforces smooth deformation by adding the Laplacian mesh regularizer. E_{var} is the body model term, it penalizes deviations of the reconstructed free-form vertices from vertices explained by the SMPL model. E_{sym} is symmetry term. It is generally observed that humans are asymmetrical with respect to the Y-axis. The optimization method in this module is "Dog-leg". The "Dog-leg algorithm" is a numerical optimization method commonly used to solve unconstrained nonlinear optimization problems. It is a type of trust-region method that aims to find the minimum of a function by iteratively adjusting the parameters or variables of the function. After unposing all frames, a visual hull is generated that constrains the form of the body in a canonical T-pose. Subsequently, a 3D avatar model with a T-pose is generated by jointly optimizing the body shape parameters and 3D point positions to minimize the distance between unposed rays and the 3D model point position, as shown in Fig.3.4 (b).

For texture generation, an image is produced by warping the estimation of the canonical model back to each frame. This involves back-projecting the image color to all visible vertices and computing the median of the most orthogonal texels from each view. Finally, a full texture image is generated, as demonstrated in Fig.3.4 (c). The whole

process efficiently estimates consensus 3D shapes, textures, and embedded animation skeletons based on a large number of frames. One key benefit of back projection lies in the establishment of 3D spatial consistency, aligning video frames with the reconstructed clothed avatar and ensuring a seamless integration of textures onto the underlying 3D model. The accurate estimation of pose and shape is facilitated by back projection, enabling the textures to conform naturally to the dynamic contours and movements of the human body. This approach adeptly addresses challenges related to dynamic clothing deformations and ensures a faithful representation of how clothing responds to various body movements. Back projection also proves effective in mitigating occlusion issues and maintaining realism in regions where the body may be temporarily obscured. Furthermore, the adaptability to different poses, the effective use of depth information, and the potential for real-time applications underscore the versatility and efficiency of back projection in achieving lifelike reconstructions of clothed avatars from video data.

3.4 Avatar Animation

To visualize the avatar model described earlier, a lifelike clothed representation is meticulously reconstructed by applying the texture image onto the underlying avatar mesh model, as illustrated in Fig.3.4 (d). A pivotal stage in the 3D animation pipeline, rigging assumes a critical role in achieving fluid and realistic movement. For any 3D geometry intended for animation, an effective rigging setup is essential. Through this method, you can control the flexibility of each joint precisely throughout the animation process. This intricate control system is commonly known as rigs. Within the proposed framework, the utilization of pre-generated and custom rigs becomes instrumental in animating characters. This involves strategically placing markers on several body joints, such as the chin, wrists, elbows, knees, and groin, as shown in Fig.3.5. Through this methodology, avatars become easily animatable, enabling key points to seamlessly execute a diverse range of actions. These actions encompass activities such as sitting, clapping, walking,

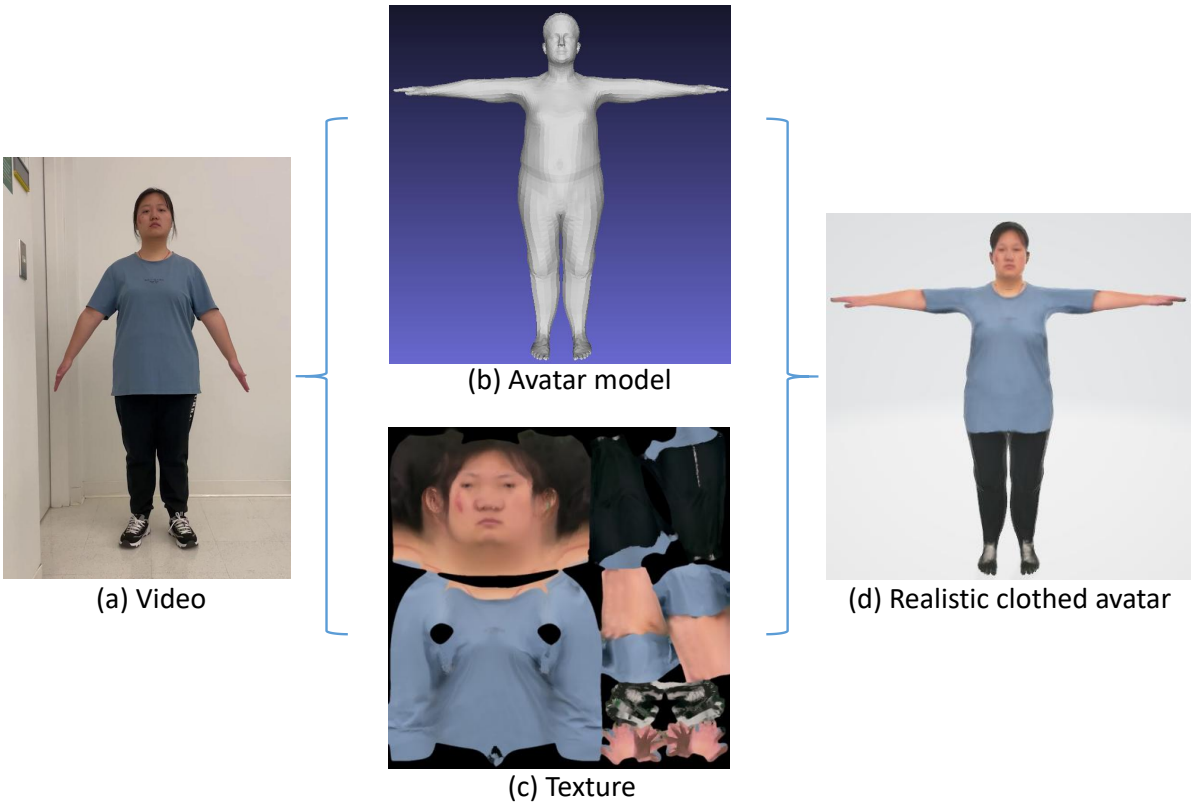


Figure 3.4: Avatar reconstruction. (a) The original video features the subject posing in "A-pose" (b) A generated 3D avatar mesh model. (c) A generated texture image. (d) An avatar is rendered by applying the texture image to the 3D avatar mesh model.

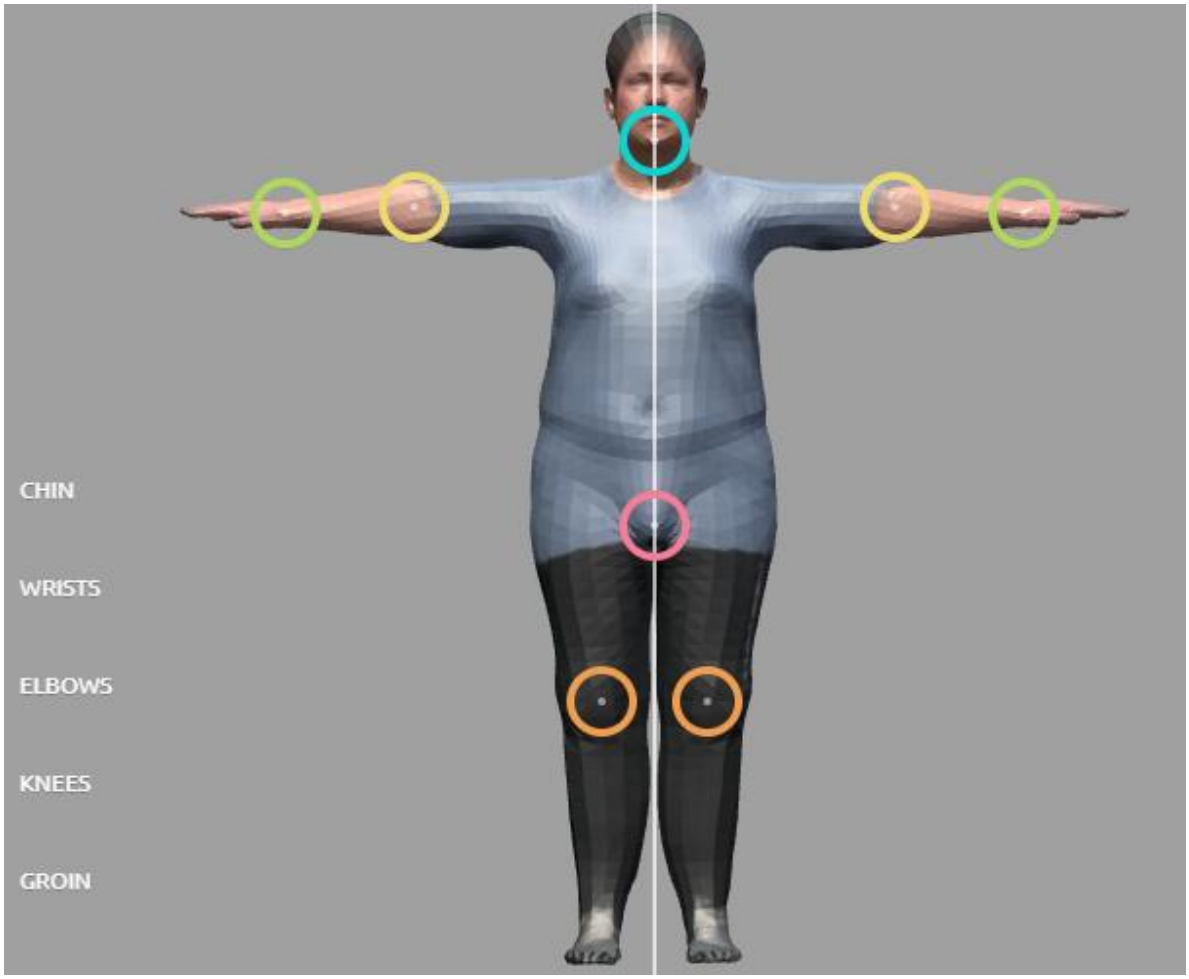


Figure 3.5: Auto-rigger system. Animated an avatar by placing markers on several body joints, including the chin, wrists, elbows, knees and groin.

running, stretching, jumping, and dancing, as shown in Fig.3.6. This approach not only ensures seamless control over joint movement but also contributes to the authenticity and dynamism of the animated characters within our animation ecosystem.

The animation process involves a limitation where only one movement can be animated on an avatar via the auto-rigger system. Consequently, Blender allows the same avatar to showcase multiple movements by integrating different actions into one unified avatar. Due to the constraint that avatars can only utilize one action at a time for editing, the non-linear action Editor becomes instrumental in blending multiple actions seamlessly. The primary approaches in this animation process employ the non-linear action editor and bake action. The non-linear action editor is employed for the operation and re-planning of actions. It facilitates the adjustment of action sequences and the incorporation of transitions between actions. This adjustment enhances the coherence and smoothness of movements by allowing control over the transition duration. Subsequently, the bake action function combines all actions simultaneously to create a new action comprising several actions. A cache is then created to store the resulting action, allowing automatic caching during animation playback. This step allows for faster playback in subsequent animations, drawing on the cached results from memory. It's important to note that once the bake action is implemented, the new action cannot be further edited. For instance, merging idle and walking actions into the same avatar yields an animated avatar with a 30-second animation after baking action—10 seconds of standing action with breathing and 20 seconds of walking, as shown in the top of Fig.3.6.

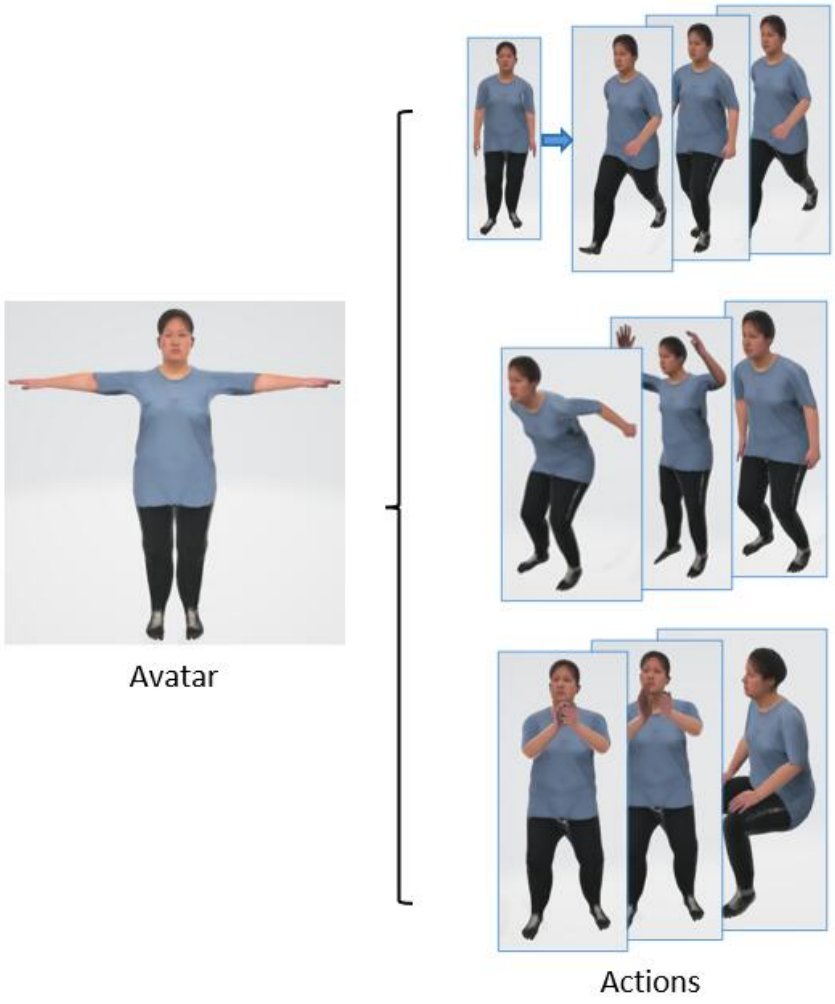


Figure 3.6: Avatar animation. An avatar with a sequence of natural and coherent movements.

Chapter 4

Experiments and Evaluation

4.1 Implementation Details

In our experiment, we test and evaluate the impact of four key factors on the accuracy of avatar reconstruction, including mask generation method, background, speed, and rotation. The overarching goal is to empower users to effortlessly reconstruct their avatars with high quality. Within the experimental setup, we define a subject's turning speed

as fast if she/he completes a full rotation in 4 or 5 seconds and slow if she/he takes 9 or 10 seconds. Videos capturing a subject's turn (1 rotation) typically consist of 120 to 210 frames, while videos of a subject rotating twice (2 rotations) generally comprise 225 to 380 frames. Additionally, we test the reconstruction under two different backgrounds: wild and green backgrounds. Consequently, a total of 8 videos are acquired, forming the basis for our comprehensive evaluation: subject rotates once slowly in wild word; subject rotates once quickly in wild word; subject rotates once slowly with a green background; subject rotates once quickly with a green background; subject rotates twice slowly in wild word; subject rotates twice quickly in wild word; subject rotates twice slowly with a green background; subject rotates twice quickly with a green background;

4.2 Visualization of Mask Generation Methods

Our objective is to maintain the fidelity of the subject's appearance in their avatar reconstruction. Achieving realistic clothed avatars, particularly in terms of facial features, is heavily contingent on the effectiveness of human masks. In our pursuit of high-quality avatar reconstruction, we conduct a comparative analysis of three prevalent segmentation methods. To identify which method produces the most accurate avatar outcomes, we need to discern the discrepancies between the results reconstructed using these methods. The three segmentation methods under evaluation are One-Shot Video Object Segmentation (OSVOS) [10], Semantic Guided Human Matting (SGHM) [13], and PP-Matting.

OSVOS represents an architecture founded on a fully convolutional neural network. Its operational principle involves taking a single manually annotated frame as input, enabling the network to learn the model of the specific object and autonomously segment it in the subsequent frames. The primary focus of this network is to address the challenge of semi-supervised video object segmentation. Notably, OSVOS demonstrates the ability to effectively transfer generic semantic information acquired from ImageNet to the task of foreground segmentation. In this way, it can learn the appearance of a single

annotated object in the test sequence, displaying remarkable one-shot learning capabilities. The results exhibit temporal coherence and stability despite processing each frame independently, which contributes to the overall efficacy of the segmentation process.

SGHM stands as a multi-stage semantic-guided human body matting framework. Its architecture comprises multiple stages and utilizes a semantic human segmentation network to sequentially predict semantic segmentation masks and matting alpha. This multi-stage approach allows the framework to effectively leverage coarse mask training data. By doing so, it diminishes the dependency on high-quality, large-scale annotations while still generating high-quality alpha details during the matting process. The result is a matting framework that balances accuracy with reduced annotation requirements, which contributes to its versatility and practicality in scenarios with limited annotated data.

In testing and evaluating the impact of mask generation methods, all videos undergo processing using the three different mask methods to generate corresponding clothed avatars. The results are exemplified in Fig. 4.1, showcasing the outcomes of one specific video where a subject executes two slow turns in a green background environment. The result comparison offers a visual representation highlighting the distinctions among each mask generation method and the extent of their impact on reconstructing realistic clothed avatars.

In the OSVOS-generated mask column shown in Fig. 4.1 OSVOS (a), the left knee is missing, indicating incomplete segmentation of the left knee of the foreground. Flaws are more noticeable in the arms and head of the clothed avatar, evident in the green areas in Fig. 4.1 OSVOS (c) and (d). This indicates weak performance in the OSVOS mask. SGHM-generated masks also exhibit issues in foot recognition, as shown in Fig. 4.1 SGHM (a). Nevertheless, the arms and head of the avatar display fewer flaws compared to OSVOS, as shown in Fig. 4.1 SGHM (c) and (d). Fig. 4.1 PP-Matting (a) showcases that PP-Matting accurately delineates the foreground and background, leading to a more precise recovery of the avatar's arms and head compared to OSOVS or SGHM. Moreover,

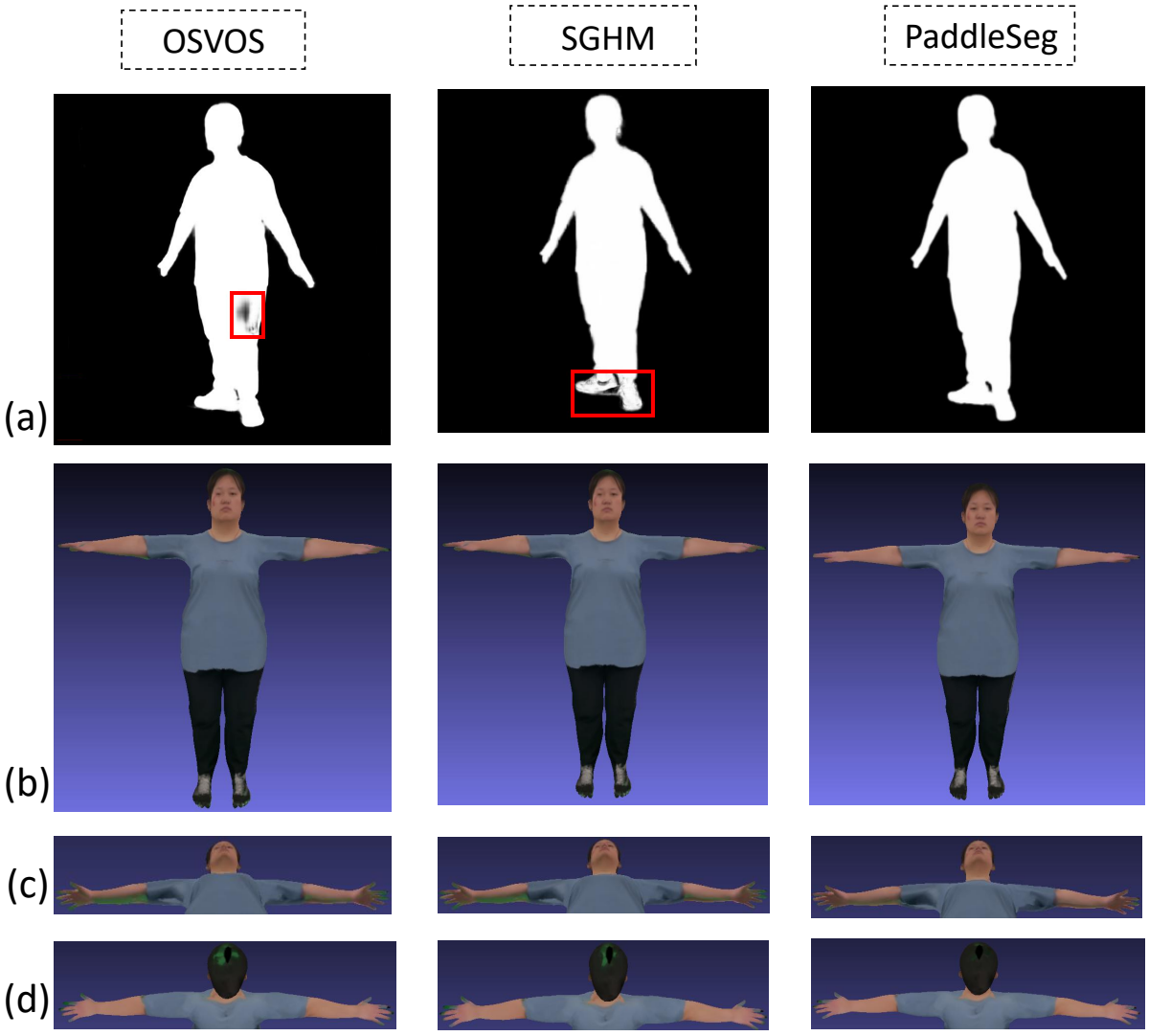


Figure 4.1: Avatars reconstruction results. (a) Generated masks based on three different mask methods. (b) Generated realistic avatars based on three mask methods. (c) Generated arms of avatars. (d) Generated neck and head of avatars.

the clothes near the neck in PP-Matting are more complete, as shown in Fig. 4.1 PP-Matting (c) and (d).

Comparing the reconstructed realistic avatars generated by the three mask methods in Fig. 4.1 (b), it's apparent that the OSVOS avatar exhibits a pronounced outward bulge on the left waist compared to the others. On the other hand, the avatars generated by SGHM and PP-Matting show no significant differences in clothes and body shape. However, a noticeable distinction arises in the clarity of the faces, particularly the left eye. The PP-Matting avatars are clearer than the SGHM avatars. Therefore, the PP-Matting model outperforms SGHM and OSVOS in terms of both image segmentation and realistic clothed avatars.

4.3 User Study

We conduct a user study. Each participant receives a questionnaire that contains 8 raw videos and 24 3D realistic clothed avatars derived from these raw videos. The questionnaire was designed using a control variable approach. The study focuses on evaluating four key factors for the impact on the realistic avatar reconstruction: mask generation method, background, speed, and rotation. To assess the impact of the mask generation method, the 24 avatars were categorized into 8 groups, ensuring consistency in background, speed, and rotation. Participants are asked to select the avatar that most closely resembles the subjects of the raw videos in their judgment. This process is repeated for each factor, allowing for a comprehensive evaluation of the various elements influencing the perception of realistic clothed avatars.

30 volunteers are invited to do our online questionnaire, ranging from people with no experience with 3D data to those who were proficient with it. Significantly, some participants had no prior familiarity with the subjects featured in the videos. They only compare the original videos with the avatars to facilitate more objective evaluations. Each participant receives specific instructions in the questionnaire, guided as follows:

- Which realistic avatar looks more like the subject in the video?
- Based on similarity, choose the avatar that best preserves the subject's identity and is visually appealing to you.

Within each set of 3D avatar models, each participant must make a choice. During the assessment process, the participants can zoom in on the model to see the avatar's details in greater detail. Furthermore, participants have the opportunity to provide feedback or articulate the reasons behind their choices or rejections directly within the questionnaire. This approach aims to gather more nuanced insights into the decision-making process and preferences of the participants.

4.4 Experiment Analysis

All selections made by 30 volunteers for each set of avatar models are summarized and shown in Table 4.1.

The avatars in the table are systematically named to reflect the order of background-speed-mask-rotation. Specifically, 'W' and 'G' designate the background (wild and green), 'S' and 'F' denote the speed (slow and fast), 'O', 'S', and 'P' represent the mask methods (OSVOS, SGHM, and PP-Matting), and the numbers '2' or '1' signify the rotation. For example, 'WSS2' signifies the video with a wild background, slow speed, SGHM mask method, and 2 rotations, while 'GFP1' designates the video with a green background, fast speed, PP-Matting mask method, and 1 rotation. With a total of 8 videos collected, it can generate 24 avatar results. So the mask method results are divided into 8 groups, each group comprising three avatar outcomes produced by three distinct mask methods for comparative analysis. The remaining factors encompass background, speed, and rotation, yielding 12 sets of avatars and each set presenting two avatar results for comparative evaluation.

As shown in Table 4.1, participants' votes indicate that among the mask generation

Table 4.1: Experiment results on four factors from participants.

Mask Generation Method			**Background**			**Speed**		**Rotation**	
Groups	**Avatars**	**PCT**	**Groups**	**Avatars**	**PCT**	**Avatars**	**PCT**	**Avatars**	**PCT**
Group1	WSO2	33%	Group1	GSO2	87%	WSO2	43%	WFO2	43%
	WSS2	20%		WSO2	13%	WFO2	57%	WFO1	57%
	WSP2	47%	Group2	GSS2	87%	WSS2	54%	WFS2	40%
Group2	WFO2	27%		WSS2	13%	WFS2	46%	WFS1	60%
	WFS2	50%	Group3	GSP2	77%	WSP2	57%	WFP2	27%
	WFP2	23%		WSP2	23%	WFP2	43%	WFP1	73%
Group3	GSO2	13%	Group4	GFO2	30%	GSO2	60%	WSO2	43%
	GSS2	20%		WFO2	70%	GFO2	40%	WSO1	57%
	GSP2	67%	Group5	GFS2	46%	GSS2	67%	WSS2	54%
Group4	GFO2	10%		WFS2	54%	GFS2	33%	WSS1	46%
	GFS2	40%	Group6	GFP2	70%	GSP2	63%	WSP2	63%
	GFP2	50%		WFP2	30%	GFP2	37%	WSP1	37%
Group5	WSO1	30%	Group7	GSO1	60%	WSO1	27%	GFO2	87%
	WSS1	50%		WSO1	40%	WFO1	73%	GFO1	13%
	WSP1	20%	Group8	GSS1	77%	WSS1	30%	GFS2	87%
Group6	WFO1	23%		WSS1	23%	WFS1	70%	GFS1	13%
	WFS1	50%	Group9	GSP1	73%	WSP1	43%	GFP2	90%
	WFP1	27%		WSP1	27%	WFP1	57%	GFP1	10%
Group7	GSO1	13%	Group10	GFO1	36%	GSO1	90%	GSO2	80%
	GSS1	30%		WFO1	64%	GFO1	10%	GSO1	20%
	GSP1	57%	Group11	GFS1	47%	GSS1	90%	GSS2	77%
Group8	GFO1	10%		WFS1	53%	GFS1	10%	GSS1	23%
	GFS1	37%	Group12	GFP1	41%	GSP1	80%	GSP2	60%
	GFP1	54%		WFP1	59%	GFP1	20%	GSP1	40%

methods, the PP-Matting method excels in performance across 5 out of 8 result categories. Specifically, group 1, 3, 4, 7, and 8 employing PP-Matting received the highest scores, with 47%, 67%, 50%, 57%, and 54% of participants respectively believing that the reconstructed virtual image and prototype had the highest degree of restoration. On the other hand, SGHM groups 2, 5, and 6 followed with notable performance, garnering recognition from 67%, 50%, and 57% of participants, respectively. However, no groups utilizing the OSVOS method received similar acclaim. The GSP2 avatar obtains the highest rate at 67%, highlighting the effectiveness of PP-Matting. Conversely, SGHM excels in 3 groups of results, with the highest rate reaching 50%. Consequently, PP-Matting enhances the accuracy of video masks more effectively compared to OSVOS and SGHM.

Concerning background evaluation, there are 7 groups (group 1: 87%, 2: 87%, 3: 77%, 6: 70%, 7: 60%, 8: 77%, 9: 73%) of green background and 5 groups (group 4: 70%, 5: 54%, 10: 64%, 11: 53%, 12: 59%) of wild background. Moreover, the difference between the results of the green background and the wild background is small in the 5 groups of wild background. This indicates volunteers think the performance of the reconstructed avatar from the green background is better than the wild background. Specifically, both GSO2 and GSS2 obtain the highest rate of 87%. Similarly, GSP2 and GSS1 share the second-highest score of 77%. This indicates that the avatars reconstructed from videos in a green environment exhibit better quality compared to those in the wild.

In terms of speed evaluation, there are 8 out of 12 groups of slow rotation results, they are group 2: 54%, group 3: 57%, group 4: 60%, group 5: 67%, group 6: 63%, group 10: 90%, group 11: 90%, group 12: 80%. And 4 out of 12 groups of fast rotation results, respectively group 1: 57%, group 7: 73%, group 8: 70%, and group 9: 57%. Avatars reconstructed from videos with subjects rotating slowly consistently exhibit significantly better quality than those at a faster speed. In particular, the last three groups, GSO1 and GSS1 achieve high scores of 90%, whereas GFO1 and GFS1 only score 10%. Therefore,

Table 4.2: Mean results on four factors from participants.

Mask Generation Methods	Mean	STD	Background	Mean	STD	Speed	Mean	STD	Rotations	Mean	STD
OSVOS	20%	9	Green	61%	20	Slow	59%	20	Twice	63%	21
SGHM	37%	12									
PP-Matting	43%	17	Wild	39%	20	Fast	41%	21	Once	37%	21

it is possible to reconstruct a more precise and beautiful 3D digital avatar of the person in the green background at a slow speed.

Regarding rotation, there are 8 out of 12 groups of two rotations results are good, with group 5: 54%, group 6: 63%, group 7: 87%, group 8: 87%, group 9: 90%, group 10: 80%, group 11: 77%, group 12: 60%. And 4 out of 12 groups of fast rotation results are good, with group 1: 57%, group 2: 60%, group 3: 73%, and group 4: 57%. Similarly, the difference between the results of two rotations and one rotation is obvious in the 8 groups of two rotations. The results illustrate that subjects who rotate twice achieve higher rates than those taking one rotation. Specifically, GFP2 obtains the highest rate of 90%, followed by GFO2 (87%), GFS2 (87%), and GSO2 (80%). They are all involving two rotations. Conversely, videos featuring subjects taking one rotation score between 10% and 20%, such as GFP1 (10%), GFO1 (13%), and GFS1 (13%).

Table 4.2 is the average percentage and STD of participants' votes for different factors computed using the data from Table 4.1. For instance, to compare the impact of green and wild backgrounds on avatars' performance, all data marked with "G" in Table I is extracted, and the sum is divided by the number of groups. Table 4.2 allows us to visually assess the influence of each factor on avatar reconstruction. Concerning the mask generation method, 43% of votes favor the PP-Matting method, followed by SGHM and OSVOS methods at 37% and 20%, respectively. Therefore, the PP-Matting model significantly enhances the quality of realistic clothed avatars, excelling in the precise

Table 4.3: The silhouette and symmetry errors on three mask generation methods.

Silhouette Error	OSVOS	3.12mm	SGHM	2.34mm	PP-Matting	2.30mm
Symmetry Error	OSVOS	3.81mm	SGHM	3.25mm	PP-Matting	3.21mm

segmentation of highly accurate human masks. Regarding the background, 52% of votes are cast for the green background and 48% of votes are cast for the wild background, indicating that avatars reconstructed from videos with a green background are more realistic than those reconstructed in a wild environment.

Meanwhile, avatars reconstructed from slow-speed videos receive 59% of votes, whereas only 41% of votes are for fast-speed videos, suggesting that slow speed could provide more details and information about subjects to improve avatar quality. In terms of rotation, avatars with two rotations receive 63% of votes, while 37% of votes are cast for avatars with only one rotation. Because videos with two rotations have double frames, supplements information from the first rotation and enhances mask training accuracy. Consequently, avatars reconstructed from videos featuring a single quick rotation may exhibit issues such as missing or blurred facial features or clothing with a more prominent background color.

Overall, rotation shows a 26% difference between two and one rotations, as indicated by the data in Table 4.2. Following this, the mask generation method exhibits a 23% difference between OSVOS and PP-Matting, while a 22% difference is observed between green and wild backgrounds. Speed has the least impact, with an 18% difference between slow and fast. Therefore, rotation has the most significant impact on avatar reconstruction performance, followed by the mask generation method and background. Conversely, speed has the least impact. It's noteworthy that the mask generation method, being a module of the framework for avatar reconstruction, directly influences the quality of the avatar.

We evaluate mask generation methods on the Videoavatar dataset [2] for silhouette

and symmetry errors, as shown in table 4.3. This table presents the silhouette and symmetry errors for three mask generation methods. Specifically, for silhouette errors, we achieve 3.12mm, 2.34mm, and 2.30mm with OSVOS, SGHM, and PP-Matting respectively. Similarly, for symmetry errors, we obtain 3.81mm, 3.25mm, and 3.21mm with OSVOS, SGHM, and PP-Matting respectively. These results demonstrate that our framework enhances reconstruction accuracy, and the PP-Matting model outperforms OSVOS and SGHM in avatar reconstruction performance.

Chapter 5

Case Study: Metaverse

5.1 Metaverse Framework

To verify the availability and generality of our proposed framework for realistic clothed avatar reconstruction and animation, we introduce a four-layer metaverse framework. This metaverse serves as a virtual world platform, seamlessly integrating various cutting-edge technologies such as artificial intelligence, blockchain, XR, digital twin, advanced

hardware, and more. The metaverse provides users with a collective virtual world where they can manipulate avatars to engage in activities ranging from playing and working to socializing and interacting with other avatars or virtual entities. As shown in Fig.5.1, our metaverse architecture is structured into four layers: infrastructure layer, interface layer, metaverse engine layer, and virtual world layer. This framework facilitates a holistic approach to avatar interaction and showcases the synergy of diverse technologies within the metaverse environment.

5.1.1 Infrastructure Layer

The infrastructure layer includes sensors, communication, computation, and storage. It serves as the backbone supporting data perception, transmission, processing, caching, and physical control within the metaverse. This layer enables efficient interaction with both digital and physical environments. Sensors play a pivotal role in facilitating high-accuracy device control and comprehensive data perception from the surrounding environment and human bodies. The communication infrastructure incorporates various wireless and wired networks to facilitate seamless connectivity. Given the demanding requirements of metaverse traffic, including necessary reliability, low latency, and high data throughput, the presence of 5G technology becomes crucial. The integration of local, edge, and cloud computing, along with storage facilities, offers robust computation and storage capacities. Through this process, vast amounts of metaverse data can be processed, resulting in faster response times, reduced bandwidth consumption, and enhanced metaverse performance.

5.1.2 Interface Layer

The interface layer comprises XR (AR/VR/MR) devices, computers, brain-computer interfaces (BCI), and smartphones. They can grant users access to the metaverse. Through these devices, users can immerse themselves in the metaverse. Because they can gain

the ability to control their digital avatars within virtual worlds. This interface facilitates a range of collective and social activities, allowing users to engage seamlessly with the metaverse environment.

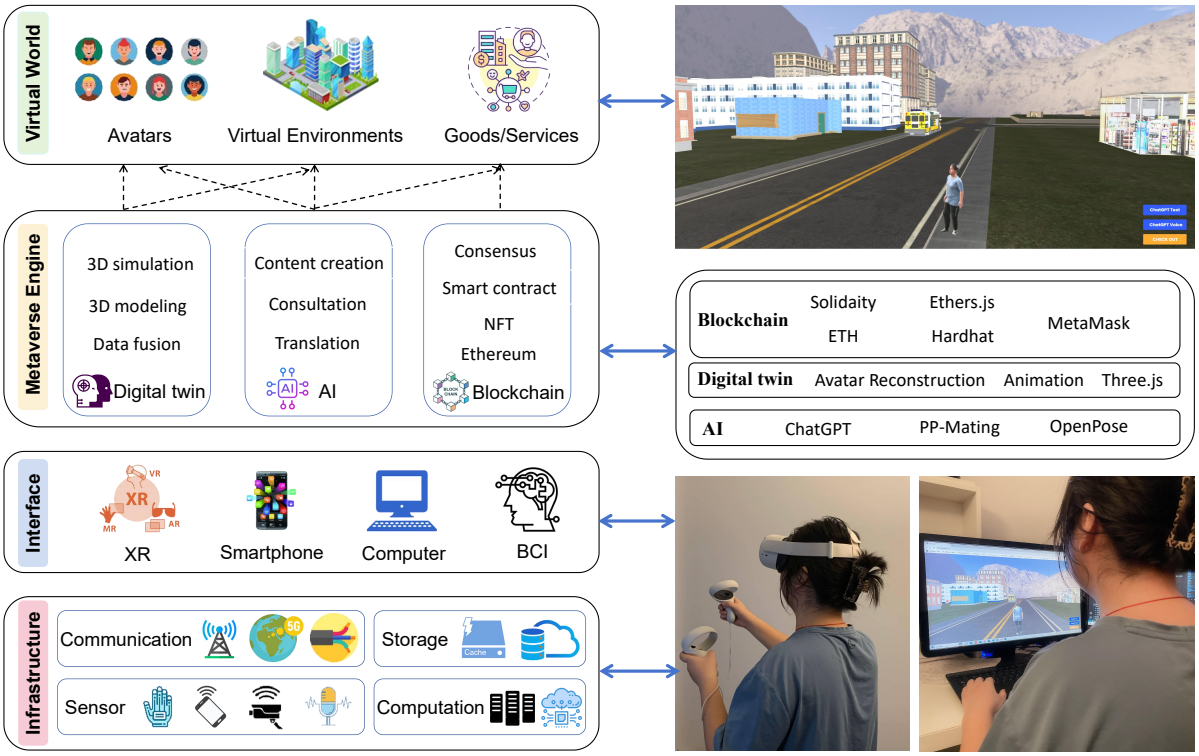
5.1.3 Metaverse Engine Layer

The metaverse engine layer harnesses real-world big data collected from the interface layer as input, employing digital twin, AI, and blockchain technologies to create, sustain, and update the virtual world. Digital twins enable the modeling of metaverse virtual worlds after their physical counterparts in real time, incorporating 3D modeling, simulation, and data fusion processes. AI plays a pivotal role in crafting personalized avatars and delivering intelligent services, such as enabling users to seek medical advice within the metaverse. To mitigate centralization risks, a decentralized architecture is deemed crucial for the metaverse. Blockchain technology provides a robust and secure foundation, ensuring ownership, security, and transparency within a decentralized virtual environment. It plays a vital role in establishing and maintaining the virtual economy and value system within the metaverse, contributing to a more decentralized and user-centric metaverse experience.

5.1.4 Virtual World Layer

The virtual world layer encompasses avatars, virtual environments, and goods/services, forming the interactive core of the metaverse. Users can access virtual goods and services within these environments through avatars, which can be created using our proposed framework for realistic clothed avatar reconstruction and animation. Virtual environments represent simulated spaces, whether mirroring reality or entirely imaginative, featuring 3D digital elements and their associated attributes. Within this layer, users can engage in a myriad of experiences, including education, shopping, medical consultation, social interactions, and concerts, all facilitated through digital currency in the metaverse.

This layer serves as the immersive playground where users interact with digital elements, fostering a dynamic and engaging metaverse experience.



(a) Metaverse framework (b) Metaverse development

Figure 5.1: The metaverse architecture. (a) The metaverse framework comprises four layers: infrastructure layer, interface layer, metaverse engine layer and virtual world layer. (b) The metaverse is developed using various technologies.

5.2 Metaverse Development

The development of our metaverse is based on the proposed four-layer metaverse framework. It incorporates squares, schools, hospitals, shops, buses, apartments, and more, as shown in Fig.5.1 (b). We use Blender for the creation of 3D models and scenes. While three.js is utilized to load, render the metaverse, and provide user interfaces for selecting various goods and services. Users can access the metaverse and control their avatars through XR headsets, such as the Oculus Quest, or via computers. Additionally,

users have the option to engage in text or voice-based consultations within the metaverse. What is more, the consultation platform seems to be a shopping guide, where users can seamlessly browse and select products through a combination of typing and voice commands. This innovative feature streamlines the shopping experience, allowing users to effortlessly add items to their virtual carts. For instance, users can simply type or speak the products they desire, ranging from everyday essentials like chicken, vegetables, and beef to more specific items such as artisanal cheeses or organic fruits. The platform's intuitive interface interprets these commands accurately, ensuring that users can quickly find and add their preferred items to their shopping lists. Moreover, the platform's voice recognition capabilities enable users to make orders verbally, further enhancing convenience and accessibility. Users can simply say phrases like, "I want a cup of iced American coffee," or "Add a loaf of whole-grain bread to my cart," and the platform will promptly execute their requests. This seamless integration of voice commands into the shopping experience reflects the platform's commitment to user-centric design and cutting-edge technology. By leveraging natural language processing and artificial intelligence, the platform creates a frictionless shopping journey for users, allowing them to effortlessly navigate through a diverse range of products and place orders with ease.

Voice consultations involve Whisper, an automatic speech recognition model that converts speech into text. To facilitate transactions within the diverse range of goods and services offered in the metaverse, users can enjoy these offerings through their avatars and conduct transactions using blockchain technology. Ethereum serves as the cryptocurrency in our metaverse. Solidity is employed to create the smart contract within the hardhat development environment, utilized for local development and testing. This smart contract is subsequently deployed on the Ethereum Mainnet network. Next.js is leveraged for designing user interfaces and interacting with blockchain networks and data, while ethers.js facilitates interactions with the Ethereum blockchain from our Next.js application and initiates transactions. Users can connect to the blockchain using MetaMask as an asset management tool and user signer. This comprehensive technological stack

enables a rich and interactive metaverse experience with diverse functionalities and services.

The three.js is used to load and render the metaverse. In the metaverse, users control the walking of avatars with the keyboard and control the orientation of avatars with the mouse. Collision detection and third-person perspective following technology are used for avatar first-person perspective scene tour, which provides users with a good experience. Collision detection needs to emit rays from each vertex of the avatar to the center point of another entity in the direction of movement and calculate the length of the ray. If the length is equal to or greater than the custom length, it indicates that the avatar is collided with another entity, such as walls or checkout counters. At this time, the avatar can stop immediately to prevent the body from crossing entities. For the third-person perspective following the character, we need to calculate the camera position: calculate the height of the animated model first and then calculate the character's world direction and normalize to obtain the direction. Next, we add the previously normalized vector to the character's world coordinates, so that the camera's position will move according to the character's position. After that, when the binding keyboard is pressed, the model's walking action will be intercepted for 30 seconds. The model's standing action will be intercepted for 10 seconds as soon as the keyboard is released. With the real-time acquisition of camera rotation and avatar movement vector, the 3D avatar loaded into the metaverse can roam around and visit every scene.

Users can also consult questions using text or voice in the metaverse, mainly based on the OpenAI API key and GPT-3.5 Turbo model. Users input a text question in the metaverse and the answer is displayed in the consultation interface, as shown in Fig.5.2 (a). OpenAI allows developers to interact directly with ChatGPT and use ChatGPT functions in their applications, websites, products, and services. API Key is the secret key used to authenticate access to the ChatGPT API. GPT-3.5 Turbo is a natural language processing model launched by OpenAI that can handle complex language processing and generation tasks. It provides users with personalized, accurate, and efficient services in

terms of conversation technology, speech recognition, and search engines. Voice consultation involves a "Whisper" model that can convert speech into text to improve accuracy and translation speed. Besides, we create an MP4 file to reserve the voice contents temporarily so that users can hear the answer and ensure whether the question is correct or not, as shown in Fig.5.2 (b).

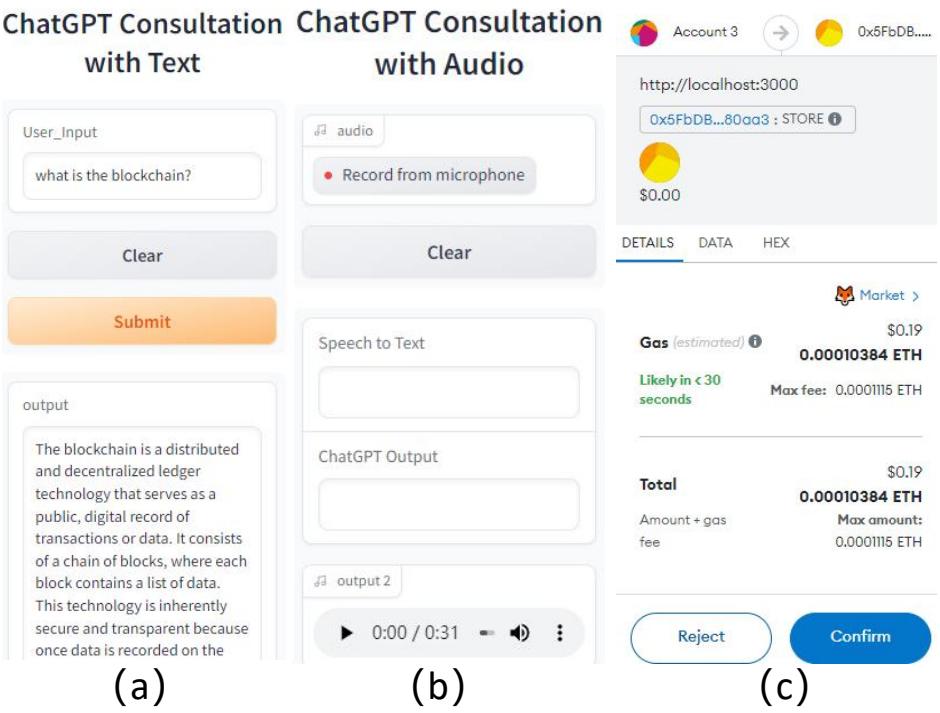


Figure 5.2: Consultation and transaction. (a) Question and order consultation by text. (b) Question and order consultation by voice. (c) Transaction information in detail.

In addition to enabling transactions through blockchain technology, avatars in the metaverse play a crucial role in utilizing the virtual economy and value system provided by blockchain. Blockchain technology ensures data integrity, decentralization, transparency, and auditability through its hash-chained blocks. Consensus protocols are pivotal for ledger consistency and blockchain scalability. Smart contracts deployed on blockchains facilitate automated tasks such as allowing untrusted parties to automate tasks such as agreements and asset transfers in a prescribed manner. Non-fungible tokens (NFTs) on the blockchain uniquely identify assets and maintain ownership provenance

can be used to identify assets and maintain ownership provenance. With De-Fi, financial services can be made more secure, transparent, and complex (e.g., stock/currency exchange) in the metaverse. In our case, Ethereum Coin is the cryptocurrency we trade.

Solidity language is used to write the smart contract on the blockchain. This smart contract is then built deployed and tested in the Ethereum blockchain using the Hardhat framework. The ether.js library enables developers to interact with the Ethereum blockchain and is used to connect our provider and send a transaction. HTML and Javascript frontend languages are used to allow users to operate the webpage menu. Next.js executes instructions from the user on the webpage. Finally, a transaction using Ethereum cryptocurrency was completed successfully, as shown in Fig.5.2 (c)

Transactions in the metaverse, we provide a method that people could use to connect their crypto wallet and use virtual currency to pay for items. Most "backends" to your web3/blockchain applications are going to be built with a framework like Hardhat, Brownie, DappTools, Anchor, or Foundry. Our frontends are going to use anything and everything in the traditional web2 space: HTML, JavaScript, CSS, and frameworks like NextJS and React. Whenever we want to read data from a blockchain, call a function, or make a transaction, we need to connect to the blockchain network. Connecting to blockchain networks and cryptocurrency wallets requires interfacing with blockchain nodes, typically provided by node-as-a-service providers. The same happens with cryptocurrency wallets, The Metamask we selected has a connection to a blockchain node built-in. Additionally, we added some functionality to our button by adding a script tag and creating a JavaScript function that looks for the window. Ethereum and if it finds it, it requests to connect. If you run the script in a browser, your Metamask will pop up and ask you to connect.

It's time to send a transaction after connecting our Metamask. This is where we can use packages like Ethers.js and Web3.js to connect our provider and then send a transaction. The only difference for sending transactions in a browser is that we change the provider to our window. Ethereum and our wallet will now come directly from our

provider. Since our metamask is both our provider and wallet (or signer). We work with NextJS for all of these because ReactJS is the most popular frontend framework on the planet right now, and NextJS is built on top of it, and in my opinion is much more user-friendly than raw ReactJS.

After setting up a basic NextJS setup, we will be testing executing functions. We set up a local hardhat blockchain and contract. It will give us an output that will start a local blockchain, give us some private keys, and deploy our SimpleStorage contract which has a store function. Ethereum is still used to interact with our smart contracts, we use hooks to enable our Metamask and any other wallet providers that we'd like. This method has a context provider and minimalistic built-in functionality for interacting with smart contracts. By following these steps, encryption wallets can be successfully linked, and transactions can be securely conducted within the metaverse.

Chapter 6

Conclusion and Future Work

The main goal of this work is to enable users to create their own digital avatars using a mobile phone or camera in any wild environment, which can be used in VR applications, entertainment, or virtual try-ons online. A novel framework for the reconstruction and animation of realistic clothed avatars from a single video is presented. The framework is structured into four modules: video acquisition, video preprocessing, avatar reconstruction, and avatar animation. Employing the OpenPose deep learning model to obtain

2D keypoints in each frame and OSVOS, SGHM and PP-Matting three deep learning foreground segmentation models to obtain masks of each frame, which serves as input for the reconstruction module. Moreover, we generalize a visual hull method to remove pose inconsistency caused by the previous and behind frames and then reconstruct the model shape for consensus shape through inverting the SMPL function. The back projection method is used for texture generation. Meanwhile, we conduct extensive experiments to test and evaluate the impact of four factors on avatar reconstruction performance: mask generation method, background, speed, and rotation. The experiment results show that rotation has the most significant impact, followed by the mask generation method, background, and speed. Avatar reconstruction from a video capturing the subject turning rotation twice yields higher accuracy. Among the mask generation methods, the quality of avatar result reconstructed by PP-Matting is the highest, which means that the PP-Matting model outperforms SGHM and OSVOS. The reconstructed avatar undergoes animation using an auto-rigger system and Blender's bake action tool, enabling the avatar to do a series of natural and coherent actions. Finally, a four-layer metaverse framework is proposed and developed in order to verify the framework's availability and generality. The metaverse framework contains an infrastructure layer, interface layer, metaverse engine layer, and virtual world layer. The metaverse is developed based on this framework using blockchain for transactions and AI to provide access to the metaverse for immersive interaction. This case study shows the potential of our avatar reconstruction and animation approach.

The reconstruction avatar has certain limitations: the lack of clarity in reconstructing clothing patterns and the inability to reconstruct the avatar's feet. Therefore, future efforts should be focused on improving the quality of reconstructed avatars. In addition, avatars do not have real-time animation capabilities. Future efforts should prioritize real-time motion capture. Furthermore, avatars will be able to change clothes in the future.

References

[1] Thiemo Alldieck et al. "Tex2shape: Detailed full human body geometry from a single image". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 2293–2303.

[2] Thiemo Alldieck et al. "Video Based Reconstruction of 3D People Models". In: *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8387–8397.

[3] Thiemo Alldieck, Hongyi Xu, and Cristian Sminchisescu. "imghum: Implicit generative models of 3d human shape and articulated pose". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 5461–5470.

[4] Alldieck, Thiemo et al. "Detailed human avatars from monocular video". In: *Proceedings of 2018 International Conference on 3D Vision*. 2018, pp. 98–109.

[5] Anguelov et al. "Scape: shape completion and animation of people". In: *ACM SIGGRAPH 2005 Papers*. 2005, pp. 408–416.

[6] Ilya Baran and Jovan Popović. "Automatic rigging and animation of 3d characters". In: *ACM Transactions on graphics* 26.3 (2007), 72–es.

[7] Blender Foundation. *Blender*. Accessed: April 1, 2024. Blender Foundation. Amsterdam, Netherlands, 2023. URL: `https://www.blender.org/`.

[8] Federica Bogo et al. "Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image". In: *Proceedings of Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*. 2016, pp. 561–578.

[9] Aljaz Bozic et al. “Neural deformation graphs for globally-consistent non-rigid reconstruction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 1450–1459.

[10] Sergi Caelles et al. “One-shot video object segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 221–230.

[11] Z. Cao et al. “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[12] Guowei Chen et al. “PP-Matting: high-accuracy natural image matting”. In: *arXiv preprint arXiv:2204.09433* (2022).

[13] Xiangguang Chen et al. “Robust Human Matting via Semantic Guidance”. In: *Proceedings of the Asian Conference on Computer Vision*. 2022.

[14] Xu Chen et al. “Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 11594–11604.

[15] Choutas et al. “Monocular expressive body regression through body-driven attention”. In: *Proceedings of Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. 2020, pp. 20–40.

[16] Yan Cui et al. “Kinectavatar: fully automatic body capture using a single kinect”. In: *Proceedings of Computer Vision-ACCV 2012 Workshops: ACCV 2012 International Workshops, Daejeon, Korea, November 5-6, 2012, Revised Selected Papers, Part II 11*. 2013, pp. 133–147.

[17] Boyang Deng et al. “Nasa neural articulated shape approximation”. In: *Proceedings of Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*. 2020, pp. 612–628.

[18] Endri Dibra et al. “Hs-nets: Estimating human body shape from silhouettes with convolutional neural networks”. In: *Proceedings of 2016 fourth international conference on 3D vision*. 2016, pp. 108–117.

[19] Endri Dibra et al. “Human shape from silhouettes using generative hks descriptors and cross-modal neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4826–4836.

[20] Feng et al. “Collaborative regression of expressive bodies using moderation”. In: *Proceedings of 2021 International Conference on 3D Vision*. 2021, pp. 792–804.

[21] Kyle Genova et al. “Deep structured implicit functions”. In: *arXiv preprint arXiv:1912.06126* 2 (2019).

[22] Ian Goodfellow et al. “Generative adversarial networks”. In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[23] Marc Habermann et al. “ReTiCaM: Real-time Human Performance Capture from Monocular Video”. In: *CoRR* abs/1810.02648 (2018).

[24] David A Hirshberg et al. “Coregistration: Simultaneous alignment and modeling of articulated 3D shape”. In: *Proceedings of Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*. 2012, pp. 242–255.

[25] Zeng Huang et al. “Deep volumetric video from very sparse multi-view performance capture”. In: *Proceedings of the European Conference on Computer Vision*. 2018, pp. 336–354.

[26] Huang, Zeng et al. “Arch: Animatable reconstruction of clothed humans”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 3093–3102.

[27] Arjun Jain et al. “Moviereshape: Tracking and reshaping of humans in videos”. In: *ACM Transactions on Graphics* 29.6 (2010), pp. 1–10.

[28] Zhongping Ji et al. “Shape-from-mask: A deep learning based human body shape reconstruction from binary mask images”. In: *arXiv preprint arXiv:1806.08485* (2018).

[29] Tong Jing. “3D object and human body scan reconstruction based on depth camera”. PhD thesis. Zhejiang University, 2012.

[30] Hanbyul Joo et al. “Panoptic studio: A massively multiview system for social motion capture”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 3334–3342.

[31] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. “Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies”. In: *CoRR* abs/1801.01615 (2018).

[32] Takeo Kanade and PJ Narayanan. “Virtualized reality: perspectives on 4D digitization of dynamic events”. In: *IEEE Computer Graphics and Applications* 27.3 (2007), pp. 32–40.

[33] Kanazawa et al. “End-to-end recovery of human shape and pose”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7122–7131.

[34] Nikos Kolotouros et al. “Learning to reconstruct 3D human pose and shape via model-fitting in the loop”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 2252–2261.

[35] Kashif Laeeq. “Metaverse: why, how and what”. In: *How and What* (2022).

[36] Christoph Lassner et al. “Unite the people: Closing the loop between 3d and 2d human representations”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6050–6059.

[37] Verica Lazova, Eldar Insafutdinov, and Gerard Pons-Moll. “360-degree textures of people in clothing from a single image”. In: *Proceedings of 2019 International Conference on 3D Vision*. 2019, pp. 643–653.

[38] Hao Li et al. “3D self-portraits”. In: *ACM Transactions on Graphics (TOG)* 32.6 (2013), pp. 1–9.

[39] Or Litany et al. “Deformable shape completion with graph convolutional autoencoders”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 1886–1895.

[40] Lingjie Liu et al. “Neural actor: Neural free-view synthesis of human actors with pose control”. In: *ACM transactions on graphics* 40.6 (2021), pp. 1–16.

[41] Matthew Loper et al. “SMPL: A Skinned Multi-Person Linear Model”. In: *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34.6 (2015), 248:1–248:16.

[42] Wojciech Matusik et al. “Image-based visual hulls”. In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. 2000, pp. 369–374.

[43] Marko Mihajlovic et al. “LEAP: Learning articulated occupancy of people”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10461–10471.

[44] Ryota Natsume et al. “Siclope: Silhouette-based clothed people”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 4480–4490.

[45] Richard A Newcombe et al. “Kinectfusion: Real-time dense surface mapping and tracking”. In: *Proceedings of 2011 10th IEEE international symposium on mixed and augmented reality*. 2011, pp. 127–136.

[46] Ahmed A. A. Osman, Timo Bolkart, and Michael J. Black. “STAR: Sparse Trained Articulated Human Body Regressor”. In: *CoRR* abs/2008.08535 (2020).

[47] Keunhong Park et al. “Nerfies: Deformable neural radiance fields”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 5865–5874.

[48] Pavlakos et al. “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 (CVPR)*. June 2019.

[49] Georgios Pavlakos et al. “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*. 2019, pp. 10975–10985.

[50] Sida Peng et al. “Animatable neural radiance fields for human body modeling”. In: *arXiv preprint arXiv:2105.02872* 2.3 (2021), p. 5.

[51] Albert Pumarola et al. “D-nerf: Neural radiance fields for dynamic scenes”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10318–10327.

[52] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. “Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 1749–1759.

[53] Shunsuke Saito et al. “Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 2304–2314.

[54] Shunsuke Saito et al. “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*. 2020, pp. 84–93.

[55] Shunsuke Saito et al. “SCANimate: Weakly supervised learning of skinned clothed avatar networks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2886–2897.

[56] Shih-Yang Su et al. “A-nerf: Surface-free human 3d pose refinement via neural rendering”. In: *arXiv preprint arXiv:2102.06199* 3 (2021).

[57] Jing Tong et al. “Scanning 3d full human bodies using kinects”. In: *IEEE transactions on visualization and computer graphics* 18.4 (2012), pp. 643–650.

[58] Aggeliki Tsoli, Naureen Mahmood, and Michael J Black. “Breathing life into shape: Capturing, modeling and animating 3D human breathing”. In: *ACM Transactions on graphics* 33.4 (2014), pp. 1–11.

[59] Ruizhe Wang et al. “Capturing dynamic textured surfaces of moving targets”. In: *Proceedings of Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*. 2016, pp. 271–288.

[60] Alexander Weiss, David Hirshberg, and Michael J Black. “Home 3D body scans from noisy image and range data”. In: *Proceedings of 2011 International Conference on Computer Vision*. 2011, pp. 1951–1958.

[61] Donglai Xiang et al. “Monoclothcap: Towards temporally coherent clothing capture from monocular rgb video”. In: *Proceedings of 2020 International Conference on 3D Vision (3DV)*. IEEE. 2020, pp. 322–332.

[62] Hongyi Xu et al. “Ghum & ghuml: Generative 3d human shape and articulated pose models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 6184–6193.

[63] Weipeng Xu et al. “Monoperfcap: Human performance capture from monocular video”. In: *ACM Transactions on Graphics* 37.2 (2018), pp. 1–15.

[64] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. “Denserac: Joint 3d pose and shape estimation by dense render-and-compare”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7760–7770.

[65] Tao Yu et al. “Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7287–7296.

[66] Andrei Zanfir et al. “Neural descent for visual 3d human pose and shape”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 14484–14493.

[67] Qing Zhang et al. “Quality dynamic human body modeling using a single low-cost depth camera”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 676–683.

[68] Tianhao Zhao et al. “3-D reconstruction of human body shape from a single commodity depth camera”. In: *IEEE Transactions on Multimedia* 21.1 (2018), pp. 114–123.

[69] Zerong Zheng et al. “Deephuman: 3d human reconstruction from a single image”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7739–7749.

[70] Hao Zhu et al. “Detailed human shape estimation from a single image by hierarchical mesh deformation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 4491–4500.