NAIST-IS-MT1651214

Master Thesis

A Comparison of Body Capturing and Model Rigging Techniques for Creating Human Avatars

Jayzon Flores Ty

September 14, 2018

Graduate School of Information Science
Nara Institute of Science and Technology

A Master Thesis
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Master of ENGINEERING

Jayzon Flores Ty

Thesis Committee:

Professor Hirokazu Kato	(Supervisor)
Professor Kiyoshi Kiyokawa	(Co-supervisor)
Associate Professor Christian Sandor	(Co-supervisor)
Assistant Professor Alexander Plopski	(Co-supervisor)

A Comparison of Body Capturing and Model Rigging Techniques for Creating Human Avatars*

Jayzon Flores Ty

Abstract

Over the past few years, human 3D reconstruction technology has seen rapid developments, enabling us to capture a person's 3D model with very high visual fidelity. By applying a skeleton rigging technique to this model, we can animate them, e.g. make the model run, jump, or dance. While it is a very time consuming and complicated process, it can be simplified through automatic rigging techniques. This pipeline enables us to generate realistic avatars in terms of appearance and movement. However, with the abundance of methods developed for each component of the pipeline, it can be difficult to determine which reconstruction system works well with which rigging system. In this study, we aim to help answer this question by comparing avatars generated by different combinations of human 3D reconstruction and rigging systems. We conducted a pilot study with 5 participants, where we asked them to rate the avatars based on reconstruction and rigging quality. Among the reconstruction methods (itSeez3d, Autodesk Recap Pro, COLMAP, COLMAP with OpenMVS), COLMAP yielded the best looking avatar, while among the rigging systems (USC autorigger, Kinect, OpenPose, Autodesk Maya autorigger), the USC autorigger performed best in terms of motion appearance. Finally, the combination of COLMAP and the USC autorigger yielded the best avatar in terms of appearance and rigging quality.

Keywords:

3D reconstruction, model rigging, realistic 3D avatars, comparative study

*Master Thesis, Graduate School of Information Science,
Nara Institute of Science and Technology, NAIST-IS-MT1651214, September 14, 2018.

Contents

List of Figures

1 Introduction

Virtual 3D models that resemble human beings are being used in a lot of computer applications, e.g., training [64], education [61], and video games [41], an expert 3D model artist has to sculpt the 3D model by hand. Depending on the quality of the appearance as well as the complexity of the 3D model, this process can take anywhere from a couple of hours to even days. If we want to animate the model, it also takes an artist a few hours to create a skeleton hierarchy that best fits the model, as well as to assign skin weights to determine how the model deforms when the bones in the skeleton move. Thus, a better work-flow is necessary if we want to quickly produce human-like 3D models.

Recently, 3D reconstruction technology has seen rapid developments to the point that it has become possible to reconstruct virtually any existing object, e.g. a table, a room, or even an entire building, into a virtual 3D model with very high visual fidelity. Technology for performing 3D reconstructions specifically of the human body is also seeing rapid development, allowing us to create 3D models that highly resemble our own appearances. With the cost of depth-sensing cameras decreasing dramatically for the past few years, various depth cameras such as the Microsoft Kinect and the Structure Depth Sensor became widely available to consumers, almost anyone can quickly set up a system for generating 3D models that resemble their appearance. This has the potential to make the process of generating human-like 3D models much easier. Such models can then be used in different scenarios, from being 3D printed to serve as physical mementos, to being rigged and animated in order to serve as personal avatars in virtual environments, e.g., simulations, virtual training environments, telepresence, and video games.

Depending on the use case, there are times when we want the reconstructed version of ourselves to be as realistic as possible, i.e., closely resemble the actual person both in terms of appearance and behavior. For example, we might want to

(a) Training [64] (b) Telepresence [53] (c) Gaming [12]

Figure 1.1: Reconstructed human 3D models used in different applications such as training, telepresence, and video games.

create a 3D model that looks exactly like ourselves, observe it perform different actions, and see the consequences of doing so, in hopes of changing how we feel towards performing them, and adjusting our behavior accordingly. We might also want to store digital versions of ourselves, slightly modify this digital version of ourselves into an idealized form of ourselves, and use them as avatars in virtual worlds, such as Second Life [15].

However, as the realism of the models increases, there comes a point where humans start to feel a sort of eeriness towards them. This results in decreased appeal, before it rises again until humans feel that the 3D model feels totally human. This phenomenon is called the "Uncanny Valley" [50]. Corpses and zombies usually fall in this "valley". Corpses are human in nature, but the fact that they not moving, i.e., lifeless, makes them creepy. Zombies, on the other hand, are human-like and are moving, but the fact that their behavior is very different from an ordinary human makes them creepy as well. Previous studies [37, 57] have shown that the more a 3D model resembles a real person, the more sensitive we are to slight flaws in the appearance or behavior of the 3D model, which automatically breaks our belief that the 3D model is human-like.

In order to overcome the Uncanny Valley, theoretically we need to make the reconstructed 3D models as realistic and accurate as possible, and thus there has been a lot of focus on developing systems that generate very accurate reconstructions of humans [46, 65]. High-grade systems, such as Staramba's Instagraph [17], Artec 3D's Shapify [16], and Vitronic's Vitus [1], have been developed that provide very high quality reconstructions, but are usually expensive to build and set up that typically involve purchasing many high-quality cameras. Low-cost

consumer-grade systems have also been developed which are cheaper to build, but at the cost of quality of the resulting 3D model.

While the appearance of the 3D models that resemble ourselves is important, how these models move can also impact our perception of how realistic they are. Previous studies have shown that the avatar's movement, e.g., its gestures and gait, also play a role in identifying the model to be similar to the actual person [33, 51]. If the model performs the same movements and gestures, as well as possess the same gait, then the model is perceived to be similar to the original person. In order to imbue the model with the ability to act in the same way as the owner of the model, proper rigging of the model is necessary. Thus, several rigging systems (e.g., [24,32]) have also been developed to provide the 3D models with an accurate skeleton that enables it to move as realistically as possible.

With the abundance of methods for performing human body 3D reconstruction, as well as methods for rigging the reconstructed human models, it can be difficult to decide what combination of methods for reconstruction and rigging produces the best-looking avatar in terms of appearance and rig quality. In this study, we aim to help address this problem. Thus, the main contribution of this work is the comparison of different approaches for body scanning and model rigging, and evaluation of which combination of these techniques performs the best in terms of the accuracy of the reconstruction, and also the visual quality of the motion of the 3D model when performing animations. We perform the comparison by taking a small subset of body reconstruction systems (itSeez3d [5], Autodesk Recap Photo [13], COLMAP [3], COLMAP with OpenMVS [10]), as well as a small subset of body rigging systems (USC Autorigger and Reshaper [32], Autodesk Maya [7] Auto-rig Tool, rigging based on Kinect v2 [6] and OpenPose [26] skeleton data), and performing a comparison of the avatars generated by different combinations of these systems in terms of the 3D model appearance, as well as how the avatar moves when performing animations. We performed a pilot study with 5 students from our laboratory, wherein we ask them to subjectively rate the avatars produced by the different system combinations based on reconstruction and rigging quality. We found that the avatar generated by the combination of COLMAP for the body reconstruction, and USC Autorigger for the model rigging, was preferred the most.

2 Related Work

Generating a realistic 3D model of a person consists of two main steps: 3D human reconstruction and model rigging; and that these 3D models can be used in different applications. In this chapter, we will first take a look at previous work that has been done on the field of 3D human reconstruction and model rigging. We then discuss numerous examples of how these reconstructed 3D models of humans have been used in different applications.

2.1 Realistic Human Avatar Acquisition

To reconstruct an accurate avatar of a person, we require the following two components: the 3D mesh of the body that contains information about the shape of the human being captured, and the skeleton rig that is necessary to support movement of the 3D model. Thus, the process of capturing a human into a 3D model can be split into two major steps: 3D human reconstruction and body rigging. This section explores previous work that describe systems for 3D human reconstruction, model rigging, as well as systems that perform both at the same time.

2.1.1 3D Human Reconstruction

Reconstructing humans into 3D models has already been explored for a long time. Early human reconstruction systems generally used multiple images of a person taken from multiple angles to estimate the shape of the person and generate the person's 3D model, a process also known as photogrammetry [40]. Some systems use the silhouette of the person computed from the images to morph an existing human 3D model so that it fits the person being scanned, such as the work done by Hilton et al. [40] where they use 3 cameras to capture the front, back, and

side of the person, calculate the silhouette of the person, and use it to morph an existing 3D model to fit the person being scanned. Gu et al. [38] increased the number of cameras to 4, and used a huge database of previously generated 3D models to pick a 3D model that best fits the person, and tweaked the 3D model as necessary. Other systems rely solely on the silhouette of the person computed from the images to generate the 3D model of the person, such as the work done by Kakadiaris and Metaxas [43] where they utilized 3 cameras to capture the front, back and side of the person. The silhouette of the person is computed from each image, and is used to construct the 3D model of the person. Fua et al. [36] uses a similar concept, but uses video sequences instead of still images of the person.

Recently, the cost of depth cameras have been decreasing, which allowed us to develop low-cost systems for 3D human reconstruction with much more accuracy compared to just using images. As such, there have been a lot of previous work that explored the use of multiple depth cameras for body reconstruction. Tong et al. [63] developed a body scanning system that utilizes three Kinect v1's in order to capture three non-overlapping parts of a person, and later combines these 3 parts into a single 3D mesh. Dou et al. [29] uses as many as 8 Kinects to capture the subject from more viewpoints, and combines all the data into a single 3D mesh. Kowalski et al. [44] described an easy-to-setup system for real-time 3D reconstruction using multiple Kinect v2 sensors to capture 3D point clouds of the environment from different viewpoints, and using iterative closest point (ICP) to align all these point clouds together to create a combined point cloud. Dou et al. [30] developed Fusion4D, a system for performing real-time 3D reconstruction of challenging scenes, e.g., moving humans, with a very high visual fidelity using multiple RGB-D sensors.

While the use of multiple depth sensors can yield very nice 3d reconstructions, it can be very hard to acquire and to set up multiple depth sensors, especially in settings where space is very limited. Thus, there has also been previous work that explored the use of a single depth sensor for 3D reconstruction, including humans. One of the first systems developed for 3D reconstruction using a single depth sensor is KinectFusion [52], which requires the user to move the Kinect sensor around in order to reconstruct a static scene, e.g. a room, an object, or a non-moving human. Succeeding works built on the idea of having the human

rotate, either self-rotation or through a turntable, while keeping the same pose in front of a static depth sensor, such as the works done by Cui et al. [27], Wang et al. [65], Li et al. [46], and Dou et al. [31].

2.1.2 Model Rigging

After capturing the 3D mesh of a human subject, the next step is to imbue the scanned 3D model with the ability to move. The first step is to generate a skeleton for the 3D mesh in order to facilitate the movement of the 3D model. A skeleton consists of joints and bones, which is similar to the skeleton of the human body. A 3D model can assume different poses by moving and orienting the different joints and bones in the skeleton. After generating the skeleton, the next step is to assign vertices of the 3D mesh to the bones in the skeleton in order to dictate which set of vertices moves when a certain bone moves. This process is called skinning. Both of these processes are usually done by experts by hand, which takes a lot of effort and time. Thus, there has been a lot of work done in developing systems for automatically rigging and skinning an arbitrary 3D model.

Given a template skeleton and the 3D model to be rigged and skinned, the work of Baran and Popovic [23] attempts to rig the 3D model by modifying the template skeleton in order to best fit the 3D model. Pan et al. [54] developed an automatic rigging system using what they call 3D silhouettes to estimate the shape of the skeleton of the 3D model. Given a set of partial rigs from a set of characters, as well as the target 3D mesh to rig, Miller et al.'s Frankenrigs [49] generates a skeleton out of the different partial rigs that best fits the target 3D mesh. Bharaj et al. [24] developed an automatic rigging system for 3D models of a wide range of characters, including characters with multiple separate components. Feng et al. [32] developed an automatic rigging system for human 3D models wherein the 3D model is matched to a database of template models, and the rig copied from the matched model to the target model.

2.1.3 Combined Systems

While the two previous sections described works solely focused on either body reconstruction or body rigging, there have also been work that performs both tasks at the same time. Feng et al. [34] developed an avatar acquisition system that reconstructs the 3D mesh of the human subject, and also generates a skeleton for the mesh afterwards. In their system, they ask the subject to turn in front of a Kinect, and are asked to stay still at 4 key poses. For each key pose, they utilize the controllable motor of the Kinect v1 sensor in order to scan the subject up and down. All scans are then combined into one single mesh using the KinectFusion [52] algorithm. Once the mesh is obtained, they generate a skeleton using a method similar to [24]. Avatars generated by their system were recognizable from afar, but are not suitable for close viewing, due to the lack of details on the avatars face and hands. The system of Malleson et al. [48] also generates full-body animateable avatars, but with the addition of the capability for facial animations. Their system consists of a single Kinect sensor and two DSLR cameras. Their system performs body measurements of the subject through the Kinect, and uses this data to morph a pre-rigged template body. The two DSLR cameras are used to capture the face of the subject, which is then fed to OpenFace [22], a face tracking library, in order to detect face action units and use them to generate face blendshapes for facial animations.

2.1.4 Comparison of Combinations of Body Scanning and Model Rigging Systems

At the time of writing, to our knowledge, work has not yet been done in comparing different combinations of body scanning and model rigging systems for realistic avatar creation. This is because papers describing body reconstruction methods compare their systems with other body reconstruction systems, and the same goes for model rigging systems. The closest work to ours was the study done by Daanen and Ter Haar [28], where they enumerated different high-cost 3D body scanning systems that currently exist in the market, but did not perform a comparison on the 3D models produced by each system. Thus, the main contribution of this study is the comparison between the combination of the different body scanning

and model rigging systems, and how they work together in order to create realistic human avatars, both in terms of reconstruction quality and rigging quality.

2.2 Applications of Realistic Human Avatars

Realistic human avatars generated from avatar acquisition systems are recently being used in different applications, such as gaming, teleconferencing, and experiments for studying human social behavior. The following sections describe studies in the aforementioned fields that utilized realistic human avatars.

2.2.1 Gaming

Although few, there have been studies that investigated whether having a player of a game controlling an avatar that looks similar to them can affect their behavior in a video game, as well as their sense of enjoyment. For example, in a study performed by Hollingdale and Greitemeyer [41], they found that people who piloted an avatar that looks similar to them exhibited greater aggressive behavior after playing a violent video game. In a study by Lucas et al. [47], they asked players to play a maze game wherein they had to traverse a maze filled with mines while piloting an avatar that either looks like them or not, and measured their enjoyment and performance. They found that while there was no effect in terms of performance, male players tend to enjoy the game more when piloting the avatar that looks similar to them, while female players tend to enjoy the game more when piloting an avatar that does not resemble them. Wauck et al. [66] extended this study to investigate whether having players pilot an avatar that looks like their friend has an effect on performance and sense of enjoyment in a game. However, they did not find any significant effect of the avatar appearance to the players' performance and enjoyment, regardless of gender.

2.2.2 Teleconferencing

Teleconferencing applications that involve 3D reconstruction of the human users have been developed for the past few years, enabling us to see our conversation partners, as well as the environment they are in, in 3D space in real-time, thus

reinforcing our sense of "being there" with our conversation partner. Tanikawa et al. [62] did not necessarily do 3D reconstruction, but still attempted to display the human users in 3D by utilizing multiple cameras to capture the user from multiple angles, and a revolving display to display the captured user from different angles. Kurillo et al. [45] developed a system consisting of 48 cameras to 3D reconstruct the users in real-time for remote collaboration. They demonstrated their system in different scenarios, such as Tai Chi learning, 3D remote interaction, and tele-immersive dancing. Orts-Escolano et al. [53] developed Holoportation, a teleconferencing system for virtual and augmented reality that utilizes real-time 3D reconstruction to capture and to reconstruct a high quality mesh of the speakers' environments. They stream the mesh data to each other's devices for real-time communication. Jo et al. [42] investigated the effect of the environment background (VR versus AR) and the nature of the avatar (reconstructed human model versus pre-built model) in terms of users' sense of co-presence and level of trust with the avatar in a teleconference experience. Their results show that users experienced greater sense of co-presence in the AR environment, and that users perceived the reconstructed human model as more trustworthy.

2.2.3 Studies on Human Behavior

Self-similar avatars have also been used for psychological studies to observe how people behave or respond in different situations. Several studies investigated whether seeing one's own 3D model in a virtual environment performing specific actions and witnessing the consequences of such actions can affect their behavior in the real environment. Fox et al. [35] investigated whether having people witness their own avatars eating candy and seeing the body of their avatar change can affect their real-life eating habits. They found that men who experienced more immersion with the virtual environment tended to mimic the avatar and ate more candy, while women who experienced more immersion tended to suppress eating candy. Aymerich-Franch and Bailenson [20] studied whether having people observe their own avatar giving a speech in front of a large audience in a virtual environment can help them overcome their public speaking anxiety, and the results showed that this approach was more effective for men. In an effort to encourage the youth to more seriously think about their future, Hershfield et

al. [39] showed people 3D models depicting an older version of themselves with hopes of encouraging them to allocate more resources for their future, and found that people indeed expressed more interest towards investing for the future when seeing the older version of themselves. Segovia and Bailenson [56] focused their attention on the effects of these avatars on children, and looked at whether showing children their own avatars performing different actions in an immersive VR environment can produce false memories, and their results showed that the children had a high tendency to develop false memories through the experience. Slater et al. [59] conducted a study with fans of Arsenal football club where they looked at their tendency to intervene in a violent situation in immersive VR when the victim is either another fan or not. They found that participants intervened more when the victim is also a fan of the football club, just like themselves. Bouchard et al. [25] investigated whether people express more sense of empathy towards an avatar expressing pain when the avatar depicts someone that they personally know (e.g. a close friend or a relative), as opposed to a stranger, and found that people generally felt more sense of empathy towards the avatar depicting their friend.

3 Current System

In order to make use of most of the body scanning and model rigging systems described in the next chapters (Chapters 4 and 5), we built a system that captures images from multiple viewpoints and estimates the pose of the user. In this chapter, we describe the system that we built, along with the necessary procedures to make the whole system work.

3.1 Setup

One of the main requirements of 3D reconstruction is a set of photos of the subject taken from multiple angles. To achieve this, we built a setup consisting of multiple cameras arranged in a fashion that covers a 360 degree view of the subject. While high-grade, professional 3D body scanners, e.g., Staramba's Instagraph [17], Artec 3D's Shapify [16], and Vitronic's Vitus [1], are able to produce very high quality 3D models of the user, they usually utilize a number of high-end cameras, and thus are very expensive to set up. Thus, we decided to use an open-source design in order to also evaluate how open-source designs fare. For this study, we used the design provided by pi3dscan [11], wherein the original design consists of 100 raspberry pis and 100 raspberry pi cameras mounted to multiple poles arranged in an almost circular fashion. The person to be scanned stands in the middle of the enclosure, and the cameras are arranged around the person so that they cover a 360 degree view of the person. For our system, we used the Raspberry PI Camera Module V2 camera, and the Raspberry PI 3 Model B, all running the Raspbian OS, to operate the camera. Finally, the Raspberry PIs receive commands from a main computer over the local network to ensure that the cameras take the images at the same time.

Over the course of this study, we implemented two versions of the pi3dscan

Figure 3.1: First version of the avatar acquisition system based on design from pi3dscan.

design. The first version consisted of 35 Raspberry PIs arranged in a square enclosure (see figure 3.1). Each vertex of the square enclosure has a pole with 5 cameras attached. The sides of the square to the left, right and front of the person to be scanned also have poles with 5 cameras each. Upon testing reconstructions achieved with this setup, we found that the spacing between the poles was too large. Thus, there were not enough overlapping images to perform a proper reconstruction of the person. To overcome this, we decided to add more Raspberry PIs and cameras to bring the total up to 56, and restructured the setup to an octagon-shaped enclosure, with each vertex of the octagon containing a pole that holds 6 Raspberry PIs (see figure 3.2). Some of the cameras are mounted higher up in order to cover views of the person from above.

The current system also has three Kinect v2's, which are connected to three separate computers. The Kinects can be used to evaluate reconstruction methods based on depth scans. For this study, however, we used the Kinects to evaluate rigging methods using skeleton data provided by the Kinect SDK, as well as to take full-body images of the target person which will be fed into the OpenPose [26] library to get the skeleton data.

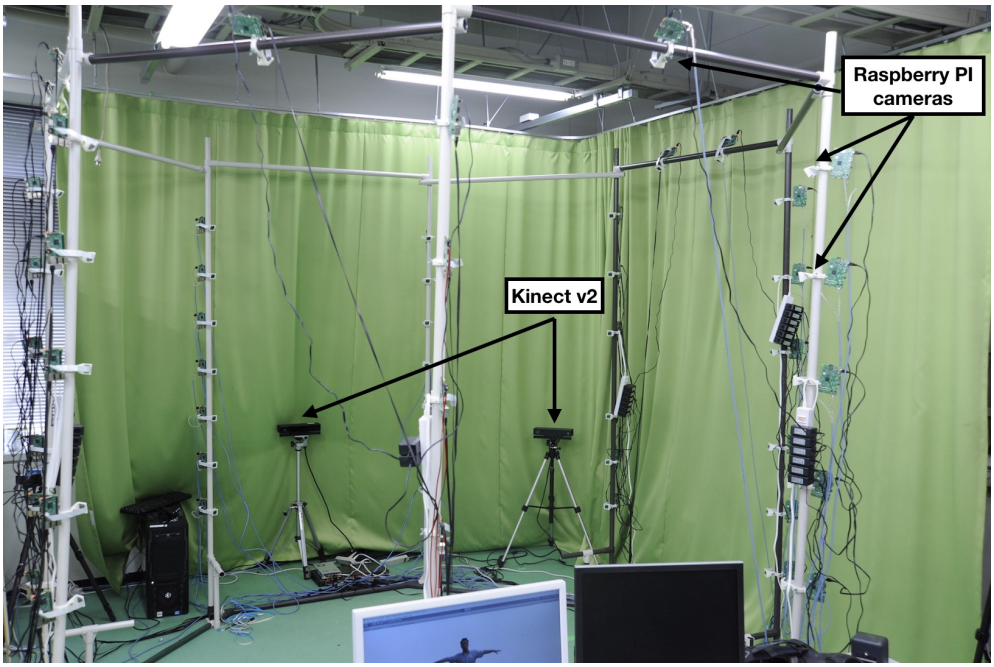


Figure 3.2: Current avatar acquisition system based on design from pi3dscan.

3.2 Calibration

In order to increase the accuracy of the reconstructed 3d models, we performed intrinsic calibration of all cameras in the system. This is to ensure that the images captured by the cameras have no distortions. For the intrinsic calibration, we used a Charuco board alongside the Charuco module of the OpenCV library (version 3.4.1). We used Charuco boards due to their high accuracy, as well as the fact that they support occlusion and partial views, which makes the calibration procedure easier and faster. Each camera took 30 images of the Charuco board, and we made sure that the board covered different parts of the image, and that the board was captured from different distances. Since there are a lot of cameras to calibrate, instead of calibrating the intrinsic parameters of each camera one-by-one, we grouped 4-5 cameras as closely as possible and took multiple images of the Charuco board simultaneously. This is possible since calibration of the intrinsic parameters of the camera does not rely on the positioning of the camera. Unfortunately, due to time constraints, external calibration for all Raspberry PIs was not performed.

Aside from the Raspberry PI cameras, the 3 Kinect v2's were also calibrated. The three Kinects were calibrated using the calibration module of the OpenPose library, which is based on the OpenCV calibration module [2].

3.3 Body Scan Procedure

In order to perform body scanning of a person, the person is asked to stand in the center of the enclosure. The person is then asked to stand in a T-pose, a standard resting pose for 3D models. The T-pose was chosen in order to make sure that the underside of the person's arms can be captured. The person only needs to maintain the pose for around 1 to 2 seconds. During that time, all 56 cameras, as well as the 3 Kinects, take a photo of the subject at the same time. At the same, the Kinect directly in front of the person also gets a snapshot of the estimated pose of the person. The captured images, along with the estimated pose from the Kinect, are used as input of the body scanning and model rigging systems described in chapters 4 and 5, and will ultimately result in the final rigged 3D model of the person.

4 Reconstruction Methods

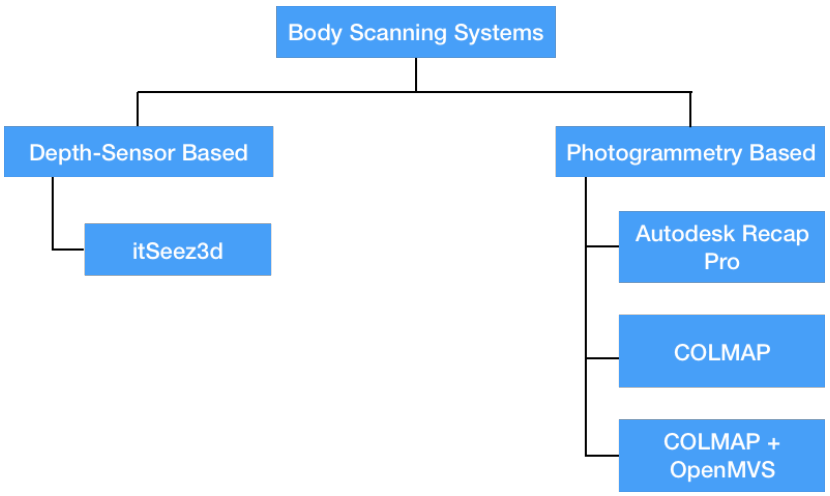


Figure 4.1: Reconstruction methods explored in this study

This chapter discusses the different 3D reconstruction methods that were considered for the evaluation of body scanning methods. Figure 4.1 shows an overview of the systems that were explored in this study, while figure 4.2 shows an example of body scans produced by the different body scanning systems.

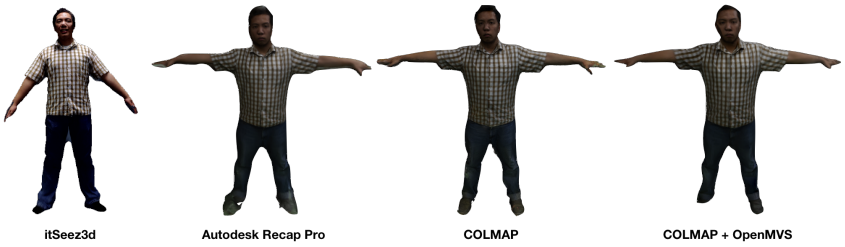


Figure 4.2: Sample results of body scans obtained from the different reconstruction systems

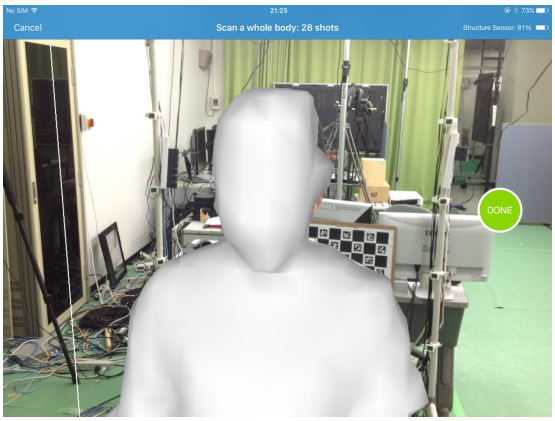


Figure 4.3: Screen-shot of the itSeez3d iPad application.

4.1 Depth-camera-based Systems

Depth-camera-based systems are systems that primarily use the depth data acquired from one or more depth-sensing cameras in order to perform a 3d reconstruction of the environment. For this study, we primarily considered the itSeez3d body scanning system.

4.1.1 itSeez3d

itSeez3d [5] (figure 4.3) is a body scanning app that uses an iPad with a Structure depth sensor attached. itSeez3d uses the depth data from the Sturcture sensor to reconstruct the mesh, and the RGB image from the iPad's camera to calculate the texture of the reconstructed mesh. In order to scan a person, the person needs to stand still, while another person goes around the target person with the iPad, taking multiple photos from different angles. For this study, since it is very difficult for the person to hold a T-pose for 1 to 2 minutes, we opted to have the person assume an A-pose instead. Processing of the model is done on itSeez3d's server, which takes around 5 - 10 minutes on average. In this study, we used itSeez3d 4.7.4. As for the hardware, we used a 4th generation iPad (model A1460), with the "Structure" depth sensor [18] developed by Occipital (running firmware 2.0) that is attached to the iPad. Finally, during the scanning process, we usually take somewhere between 50 to 60 photos, which approximately

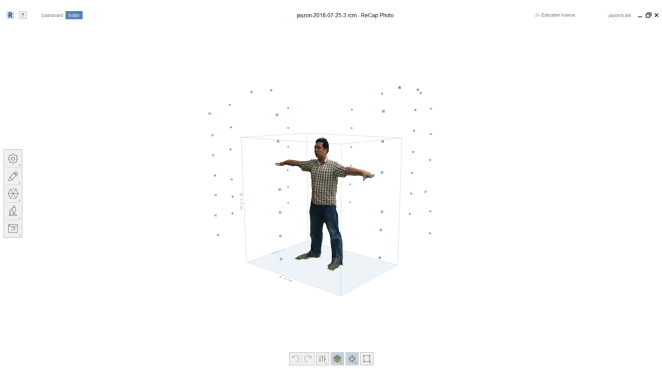


Figure 4.4: Screen-shot of the Autodesk Recap Photo application.

matches the 56 cameras from the photogrammetry setup.

4.2 Photogrammetry-based Systems

In contrast to depth-camera-based systems, photogrammetry-based systems do not rely on using depth data obtained from depth-cameras in order to generate a 3D reconstruction of the environment. Instead, photogrammetry-based systems rely on multiple images of the target taken from different angles, detecting features within those images, and computing the disparity between these features in order to create a 3D reconstruction of the target. For this study, we considered a mix of paid and open-source photogrammetry software, namely Autodesk Recap Photo, COLMAP, and a combination of COLMAP and OpenMVS, in order to compare open-source and commercial photogrammetry software.

4.2.1 Autodesk Recap Photo

Autodesk Recap Photo [13] (figure 4.4) is a photogrammetry software developed by Autodesk. For this study, we used Autodesk Recap Photo 2019. The software provides a graphical user interface for uploading photos of the object to be reconstructed taken from different angles, and the 3D reconstruction process is done on the server. Autodesk Recap Photo requires a minimum of 20 photos. It also has an option to automatically crop the target being scanned, which is always selected for this study. The processing time depends on the server load. For the

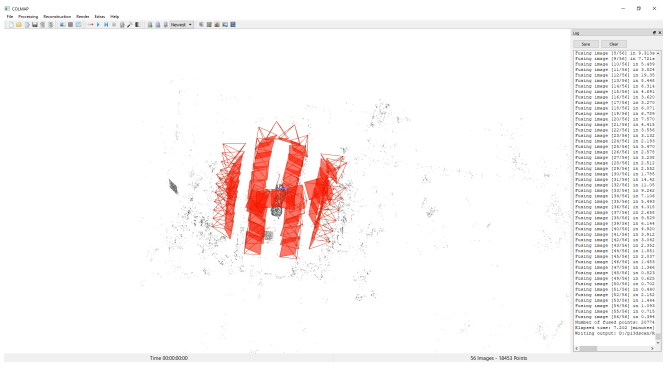


Figure 4.5: Screen-shot of the COLMAP application.

purposes of this study, we used the educational version of the software, which has the limitation that the reconstruction requests are of low priority. This results in variable processing time, ranging from 2 hours to a full day before the output can be downloaded.

4.2.2 COLMAP

COLMAP [3] (figure 4.5) is an open-source software for multi-view stereo and structure-from-motion. Given a set of images of a scene taken from multiple angles, the software can perform a sparse and dense reconstruction of the environment. For this study, we used version 3.4. While COLMAP is capable of automatically estimating the intrinsic and extrinsic parameters of the cameras used to take the images before performing the reconstruction, COLMAP also provides the user the capability to supply the intrinsic parameters obtained from a previous calibration, which we chose to do in this study.

Since the output of the software is a reconstruction of the whole environment, it is necessary to manually crop out the person in the scene. For this task, we used MeshLab [8], an application for modifying 3D mesh. The dense reconstruction generated by COLMAP is in .ply format that uses vertex colors to store color information. We needed to convert this into an .obj file that stores the color data of the mesh in texture form. We also used MeshLab to generate texture mapping using data about the camera parameters, as well as the images taken by the cameras. This is necessary since the 3D model will eventually be converted

into the .fbx format, a 3D model file format that also stores animation data, but does not support vertex colors.

COLMAP offers different parameters that control how the reconstruction is performed. For the feature extraction step, we set COLMAP to use the OpenCV camera model since we calibrated the cameras using OpenCV (see chapter 3). This allows us to feed COLMAP the intrinsic parameters that we obtained from the calibration. After empirical testing we used exhaustive matching with a block size of 10 for the feature matching step. After performing feature extraction and matching with the aforementioned parameters, we used "automatic reconstruction" to reconstruct the model.

4.2.3 COLMAP + OpenMVS

While COLMAP by its own can create a dense reconstruction of the scene, it is possible to use the camera parameters computed by COLMAP and feed them into another software that specializes on performing dense reconstruction via multi-view stereo. As such, we also explored the combination of COLMAP and an open-source multi-view stereo software OpenMVS [10]. For this study, we used version 0.8, and at the time of writing, OpenMVS is available only as a command-line application.

As discussed in the previous section about COLMAP, we provided COLMAP with the intrinsic parameters of the cameras, and used it to estimate the extrinsic parameters using the images taken from the cameras. After estimating the extrinsic parameters, COLMAP can export both intrinsic and extrinsic parameters into different formats, such as NVM, Bundler, and text files. OpenMVS supports the NVM file format. However, the NVM format only supports the radial camera model. As such, we needed to convert from OpenCV camera model to a pinhole camera model using the image undistorter module of COLMAP to undistort the images into a pinhole camera model. Since the pinhole camera model models the focal distance into two different parameters fx and fy, and radial only uses one value for the focal distance f, we computed f to be just the average between fx and fy.

5 Rigging

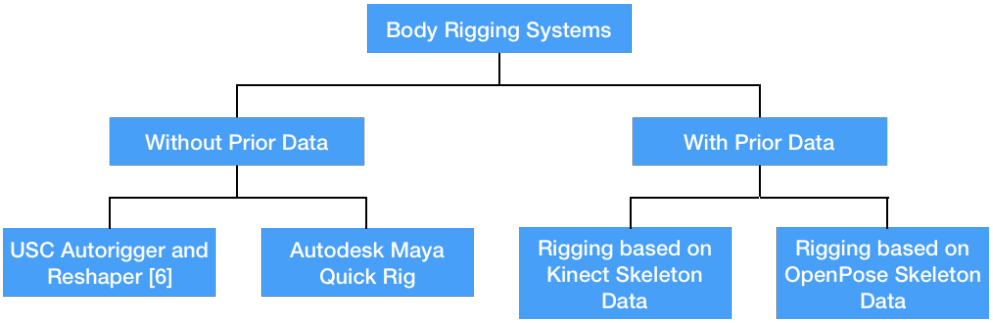


Figure 5.1: Overview of the rigging systems explored in this study

This chapter describes the different rigging systems that were explored in this study (see Figure 5.1 for an overview) in order to rig the body scans generated by the body scanning methods from the previous chapter. The rigging systems described in this chapter can be split into two categories, mainly systems that are able to generate a skeleton hierarchy and the corresponding skin weights without using any form of prior data, and systems that utilize some form of prior data, e.g., known joint positions, assists the generation of the skeleton hierarchy, the skin weights, or both.

5.1 Rigging Without Prior Data

Rigging systems that do not require prior data are usually automatic in nature, i.e., user input is not necessary. Given a 3D model, these systems automatically compute the skeletal hierarchy that best fits the structure of the model, assigns the skin weights automatically, and produces an animate-able model. In this study, we considered the following two systems: the autorigger and reshaper

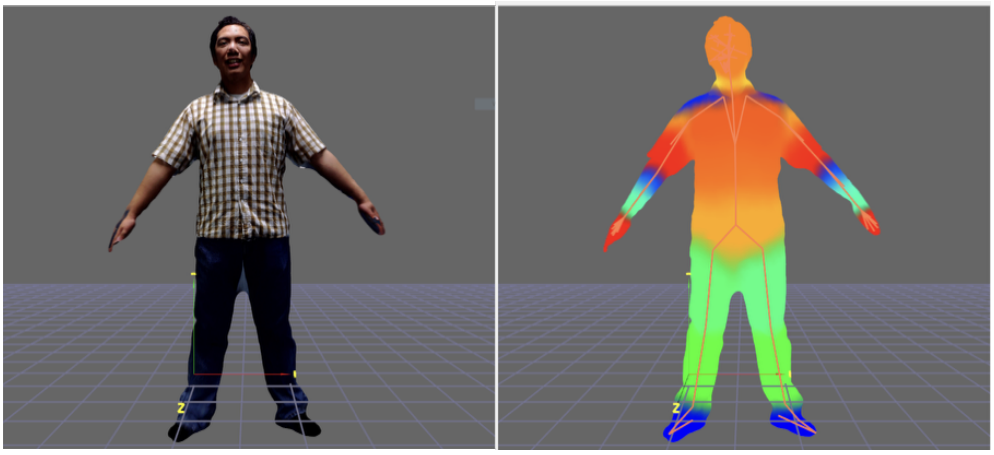


Figure 5.2: Skeleton generated by the USC Autorigger and Reshaper (right) based on an input 3D model (left).

system developed by Feng et al. [32], and the quick-rig tool of Autodesk Maya, a software built for creating 3D models.

5.1.1 Autorigger and Reshaper

The University of South California Institute for Creative Technologies developed an automatic model rigging tool, that estimates the skeleton hierarchy of a human-like 3D model by morphing a pre-rigged 3D model to best match the structure of the input model, and transferring the skeleton hierarchy of the pre-rigged model to the input 3D model. The skin weights are then re-computed for the input 3D model after the transfer. Figure 5.2 shows the skeleton and the set of skin weights generated for the supplied 3D model.

The software provides a graphical user interface that facilitates the model rigging process. The user can specify the 3D model to rig and parameters that can influence the rigging process. In this study, the default values were used, except for the "model height" parameter, wherein the actual height of the scanned person was supplied. After the rigging process is finished, the user can export the rigged model in COLLADA (.dae) format. However, for the purposes of this study, the rigged model needed to be converted to the FBX (.fbx) format. We used the 3D modeling software Blender [4] to perform the conversion.

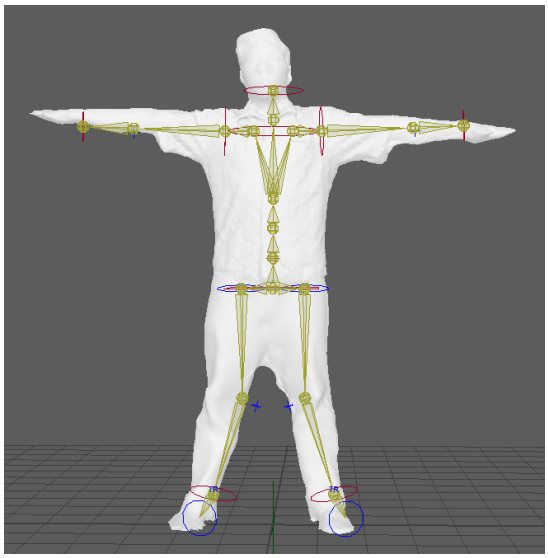


Figure 5.3: Skeleton generated by the quick rig tool of Autdesk Maya for an input 3D model.

5.1.2 Autodesk Maya Quick Rig Tool

Autodesk Maya is a software that provides various tools for the creation of 3D models. Among these tools is a built-in automatic character rigging tool that attempts to create a skeleton that best fits the 3D model supplied by the user. There are two sub-categories for the rigging tool: a one-click rigging tool and a step-by-step rigging tool. The one-click rigging tool performs the whole process automatically, from estimating the skeleton to assigning the skin weights, while the step-by-step rigging tool computes the initial position of the joints of the skeleton and allows the user to manually adjust the joint positions. In this study, we used the one-click rigging tool as a representative for the category of rigging systems without prior data. After the rigging process, the rigged 3D model can be directly exported to the FBX format.

5.2 Rigging from Prior Data

In contrast to automatic rigging systems such as the ones mentioned in the previous section, there are rigging systems that use some form of prior data to rig

the 3D model. For example, this data can be manual input from the user or joint position data from body tracking systems. The rigging system can then use this prior data to construct the skeleton for the 3D model and to assign the corresponding skin weights.

For this study, two forms of data priors were used: body tracking data provided by the Microsoft Kinect, and body tracking data from the body tracking software OpenPose. Both data priors were used to construct the skeleton for the 3D model, which must then be manually aligned to the body scanned 3D model. Since the skeleton is based on the estimated skeleton from the person being scanned, the user only needs to align the skeleton to the body, without the need to manually adjust the position and orientation of the individual joints in the skeleton. Once the skeleton has been aligned, a 3D modeling software can then be used to rig the 3D model with the skeleton. For this study, the 3D modeling software Blender [4] was used alongside the "Rigify" plugin [14] to rig the 3D model using the previously constructed skeleton. This whole process can be considered as a semi-automatic system for rigging the body scanned 3D model.

5.2.1 Skeleton Data from Kinect

Aside from providing depth information of the environment captured by the Kinect sensor, the Kinect SDK also provides an interface for body tracking, which can track the 3D joint positions of up to 6 people that are captured by the sensor. It is then possible to use these 3D joint positions to construct the 3D skeleton of the person being scanned, which can be used to rig the body scanned 3D model (see Figure 5.4). In this study, the Kinect sensor stationed directly in front of the person being scanned captured these joint positions. When using this approach alongside the itSeez3d body scanning system, the joint positions are captured first by the Kinect before proceeding with the itSeez3d body scanning process. When using this alongside the body scanning setup with multiple Raspberry PIs, the Kinect captures the skeleton data at the same time as the raspberry PI cameras.

For this study, 17 out of the 25 joint locations provided by the Kinect were used to construct the skeleton for the body scanned 3D model. Figure 5.5 describes the structure of the skeleton hierarchy constructed using joint data from the Kinect.

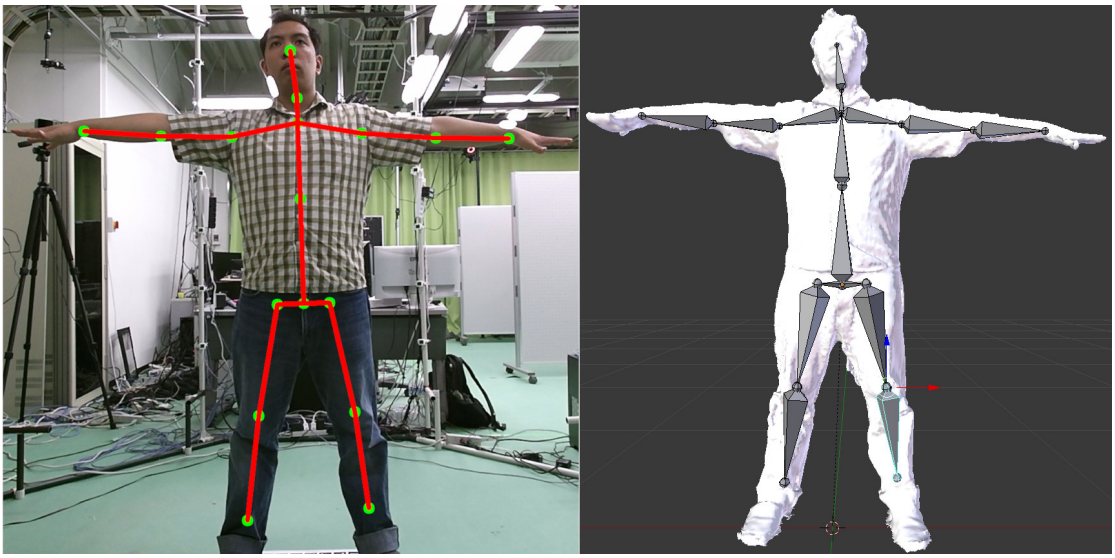


Figure 5.4: Skeleton of the 3D model generated in Blender using joint location data from Kinect.

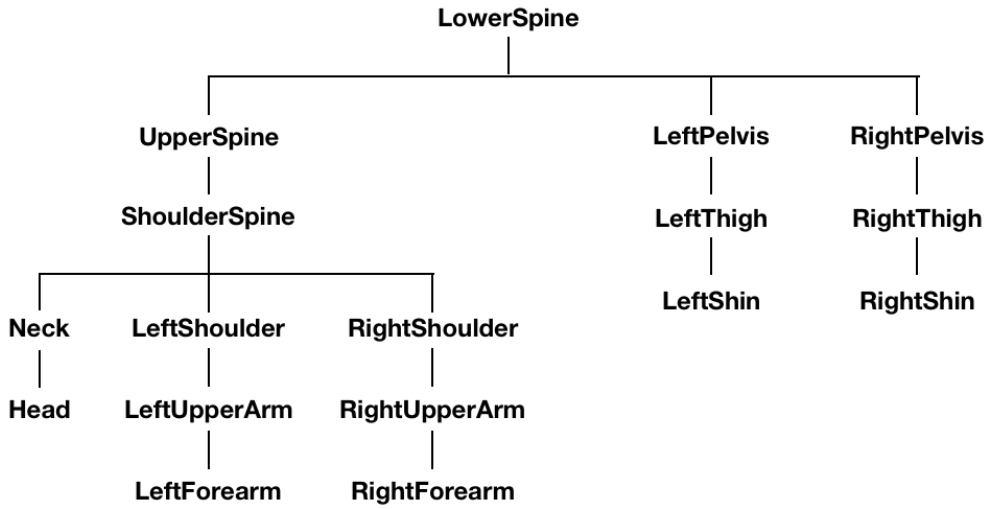


Figure 5.5: Kinect Rigging Skeleton Joint Hierarchy. Joints are represented as nodes, while the bones are represented as edges.

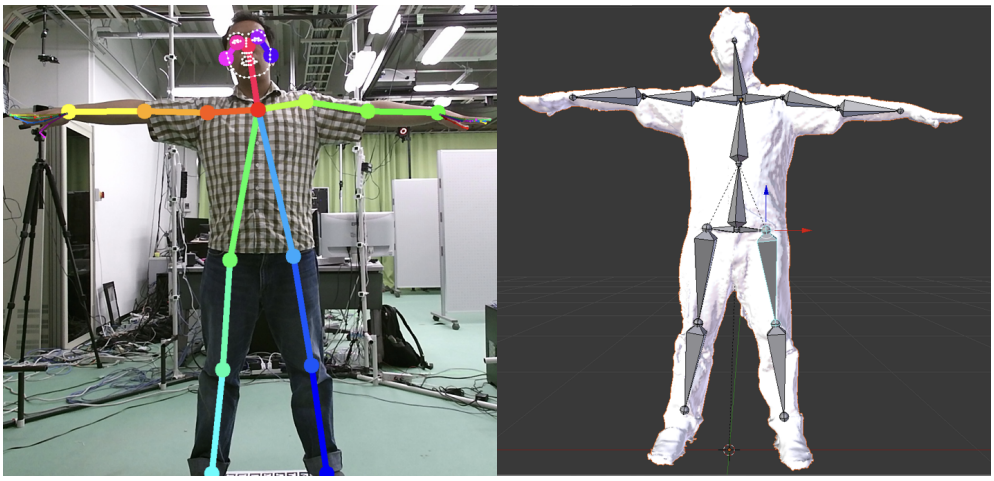


Figure 5.6: Skeleton hierarchy generated in Blender using data from OpenPose

5.2.2 Skeleton Data from OpenPose

Recently, machine-learning based human pose estimation systems have received a lot of attention. One such system is OpenPose [26], which can detect the 2D locations of human body joints, as well as facial key points, from images and videos. As such, OpenPose can potentially be used not only to generate the skeleton for the 3D model, but also to generate facial action units for the 3D model to support facial animations for the body scanned model.

By using multiple images of a human taken from different cameras, we can obtain the joint locations in 3D through triangulation. OpenPose provides a built-in 3D reconstruction module that combines different sets of 2D joint locations detected in different images into an integrated set of joint locations in 3D. In this study, the three Kinects in front of the body scanning setup (see Chapter 3) were used to obtain the images for the 3D reconstruction module of OpenPose. To use the 3D reconstruction module of OpenPose, it is necessary to calibrate the cameras. For this, we used the calibration module included in OpenPose. We used 200 images for the intrinsic calibration of each Kinect, and 1,500 images (500 images/Kinect x 3 Kinects) for the extrinsic calibration.

We constructed the skeleton from 14 of the 18 joints, excluding the points for the left and right ears, as well as the left and right eyes. To approximately match this skeleton hierarchy with the one generated using the Kinect, we generated two

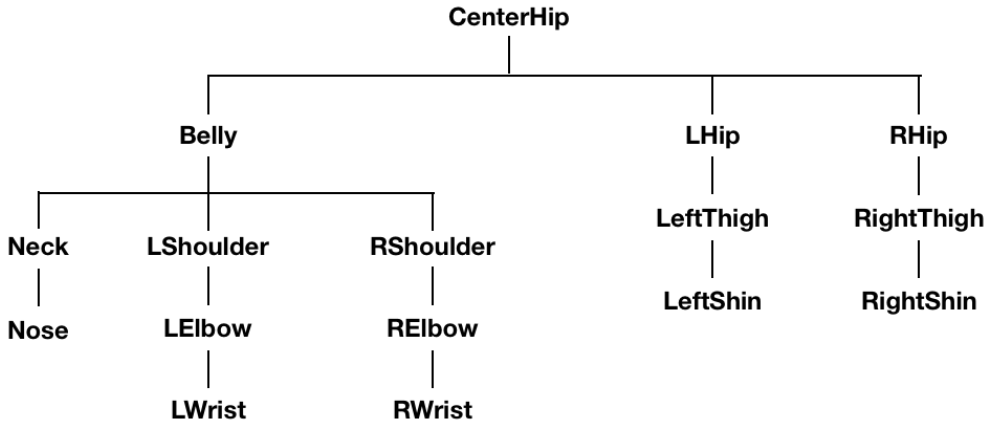


Figure 5.7: OpenPose Rigging Skeleton Joint Hierarchy. Joints are represented as nodes, while the bones are represented as edges.

custom joints: CenterHip and MiddleSpine, which correspond to the LowerSpine and UpperSpine joints of the Kinect, respectively. We generated the CenterHip joint by taking the midpoint between the left and right thigh joints, and the MiddleSpine joint by taking the midpoint between the CenterHip and Neck joints. Figure 5.7 describes the structure of the skeleton generated using joint location data from OpenPose.

6 Evaluation

For the evaluation of the avatars generated by the combinations of different body scanning and model rigging systems, we performed measurements on the reconstructed 3D models to determine whether they matched with the original person, and we also conducted a pilot study to get a subjective evaluation of the generated avatars.

6.1 Mesh Measurements

To evaluate the accuracy of the generated mesh by the body scanning systems, we conducted a quantitative evaluation by comparing measurements of different body parts of the body scanned 3D models against the actual person that was scanned to determine whether the generated body scans matched the actual person being scanned. We used the built-in measurement tool of the 3D mesh processing software MeshLab to perform the measurements on the 3d models. However, MeshLab measures 3D models using its own unit of measurement. Thus, we needed to convert the measurements from MeshLab into physical units, in this case, meters. To do this, we used the height of the 3D models that were measured using MeshLab, and the actual height of the scanned person to establish the relationship between MeshLab units and physical units (in meters). Given the height h_p (in meters) of the person that was scanned, and the height h_m (in MeshLab units) of a body scan, we establish the ratio $r_{a,m}$ between physical units (in meters) and MeshLab units through the equation:

$$r_{a,m} = h_a/h_m$$

Given any measurement x_m of the 3d model, we can then convert it to a

Measurement	Error (cm)			
	itSeez3d	**Autodesk**	**COLMAP**	**COLMAP + OpenMVS**
Shoulder Width	1.25	0.56	2.78	0.20
Arm Length	0.32	0.59	8.02	5.46
Arm Width	1.99	1.39	0.12	0.26
Leg Height	4.49	7.39	10.83	5.55
Leg Width	0.51	1.52	0.99	1.93

Table 6.1: Measurements for the body scanned 3d model and the reference person

measurement x_a in physical units through the equation:

$$x_a = x_m * r_{a,m}$$

We then evaluated each generated avatar by getting the difference between measurements for different body parts of the 3d model and the measurements for the corresponding body parts of the actual person. In this study, we used the body scan of the author for the evaluations. In order to ensure consistency, we used the same set of images for all the photogrammetry-based methods. Table 6.1 shows the results of the measurements. Based on the results, itSeez3d has the smallest range of error, meaning that the reconstruction generated by itSeez3d was the most up-to-scale. This can be attributed to the fact that itSeez3d requires the user to take photos of the target in close-range, and thus, is able to more clearly capture the features of the target, resulting in a geometrically-accurate reconstruction.

6.2 Pilot Study

We conducted a pilot study in order to gather subjective ratings about the appearance and accuracy of the reconstructed 3D models, as well as the quality of the rigged model generated by the different model rigging systems. Five students from our laboratory volunteered to be participants for the pilot study. We developed the application for the pilot study using Unity 2018.2.0f2 [19], and deployed it as a desktop application.

Reconstruction Rating	Rigging Quality Rating										
All Body Scans	Body Scan 1					...	Body Scan 4				
	A1	A2	A3	A4	A5		A1	A2	A3	A4	A5

Figure 6.1: Time-line for the pilot study. A1 to A5 represents the different animations played during the rigging quality rating phase.

Figure 6.2: Participants' view during the reconstruction rating part.

The pilot study involved two parts: reconstruction rating and rigging quality rating, wherein participants were asked to subjectively rate the quality of the reconstructed 3D models generated by the body scanning systems, as well as the skeletal rig produced by the model rigging systems respectively. Figure 6.1 shows the time-line of the pilot study. For the reconstruction rating part, participants were shown 4 different 3D models that were generated by the 4 different body scanning systems (see figure 6.2), and were asked to subjectively rate the appearance of each 3D model on a scale of 0 - 5, with floating values allowed, in terms of the 3D model's resemblance to the actual person, reconstruction accuracy (shape, body build), and quality (presence of artifacts, untextured areas). Participants were also given the capability to move around the scene using the WSAD control scheme to closely inspect each 3D model.

After the reconstruction rating part, participants proceeded to the rigging quality rating part, where they evaluated the quality of the skeleton and the corresponding skin weights that were generated by the different rigging methods. The

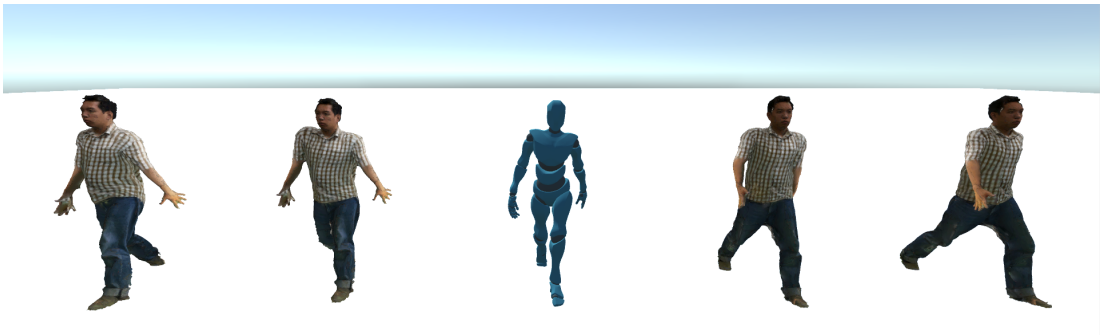


Figure 6.3: Participants' view during the rigging quality rating part.

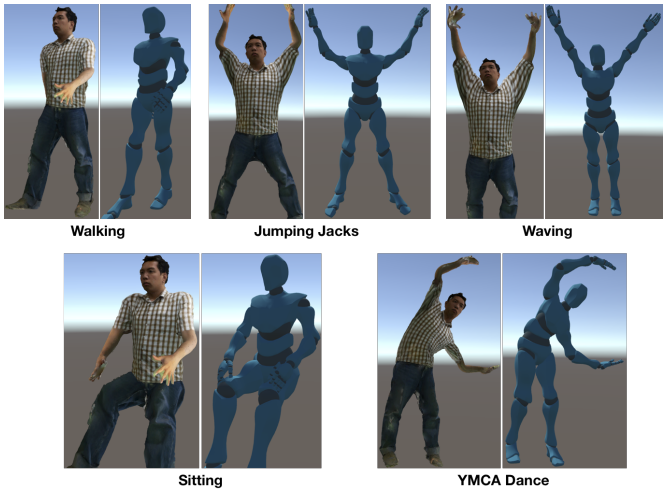


Figure 6.4: Animations used in the pilot user study

rigging quality rating part of the pilot study consisted of 4 phases, with each phase representing one body scanning method. For each phase, participants were shown 4 copies of a 3D model generated by one body scanning system, with each copy being rigged by one of the different model rigging method discussed in chapter 5. Figure 6.3 shows a screen-shot of the view that the participant sees. For each phase, 5 animations were played, and participants were asked to rate each avatar performing an animation on a scale of 1 - 5 based on the appearance and accuracy of the motion.

The five animations used for the study (see figure 6.4) were obtained from Mixamo [9], a website containing a database of different animations that can be downloaded for free. The animations chosen for the pilot test involved large mo-

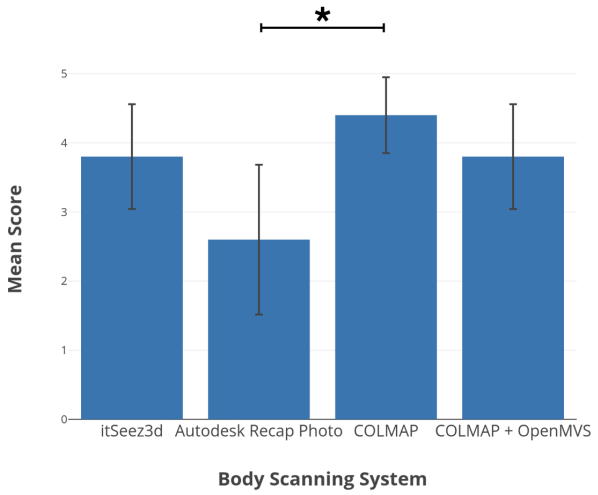


Figure 6.5: Subjective scores from the pilot study ($n = 5$) for each body scan generated by the reconstruction systems

tions of the limbs to accentuate potential errors in the skeleton and skin weights, as well as a constricted pose to see whether the skeleton and skin weights allows the avatar to perform meticulous poses. A pre-rigged character was placed in the center of the 4 avatars to serve as a reference for how the animation is supposed to look like, as well as to serve as a basis for comparison. Participants were also given the capability to move around the scene, as well as to pause the animation to let the participants look for possible errors on the avatars' movements. After rating all avatars for the five animations, the participants repeat the whole process 3 more times for the 3D models generated by the remaining body scanning systems. Finally, for each participant, the order of the sets was shuffled to eliminate potential order effects.

Figure 6.5 shows the mean scores given by the participants for each 3d model generated by the different 3d reconstruction systems. Based on the results, the body scan generated by COLMAP is the most preferred. Kruskal-Wallis test revealed a significant difference between the different pairings ($\chi^2 = 9.105, p = 0.0279$), and Dunn post-hoc test revealed a significant difference between the scores for Autodesk Recap Pro and COLMAP ($Z = 2.817, p = 0.029$). Furthermore, there were no significant differences between the scores for photogrammetry-

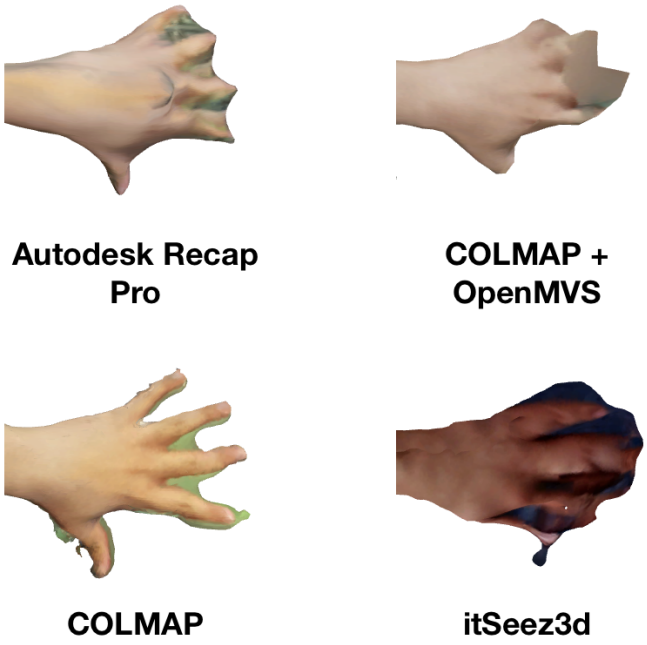


Figure 6.6: Reconstruction of the fingers for each body scanning system

based methods (Autodesk Recap Pro, COLMAP, COLMAP + OpenMVS) and itSeez3d.

Based on the feedback from the participants, COLMAP yielded the highest score due to the fact that the 3D model generated by COLMAP had the shape of the fingers properly reconstructed, while the 3D models generated by the other systems had the fingers cut off (see figure 6.6). Looking at the characteristics of the generated 3D models from previous scan attempts, we observed that COLMAP tends to produce 3D models that are raw in nature, i.e., the surface appears rougher, and has a lot of untextured areas. This can mean that COLMAP employs less filtering of the images taken, and puts more focus on reconstructing the shape of the person rather than the appearance. On the other hand, the 3D models generated by Autodesk Recap Pro, OpenMVS, and itSeez3d have smoother surfaces and less untextured areas, which can mean that these 3 systems focus more on producing a smoother mesh with higher texture quality, thus cutting out parts of the person that does not have sufficient data. However, this also reveals that the current body scanning setup does not capture enough

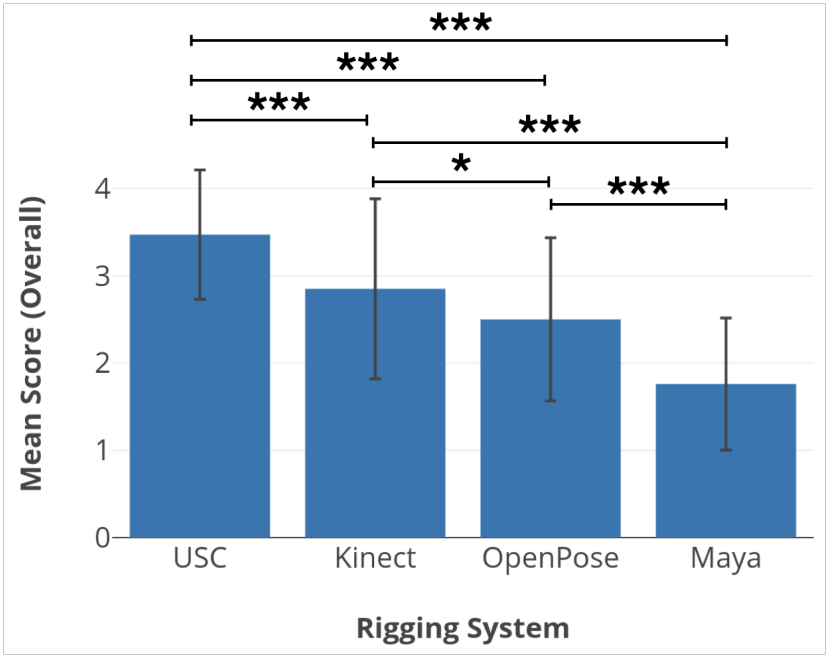


Figure 6.7: Overall mean scores for each rigging systems based on the appearance of the avatars' motions.

photos of the person's hands, and thus, future iterations should dedicate more cameras for taking photos of the person's hands.

Figure 6.7 shows the overall mean scores for the rigging systems, and based on the results, the skeleton generated by the USC autorigger yielded the best overall performance based on the appearance of the motion performed by the models when playing animations. Kruskal-Wallis test revealed significant differences within the data ($\chi^2 = 138.43, p < 0.001$), and Dunn post-hoc test revealed significant differences in all pairs ([Kinect-Maya: Z=7.099, p<0.001], [Kinect-OpenPose: Z=2.220, p=0.002], [Maya-OpenPose: Z=-4.878, p<0.001], [Kinect-USC: Z=-4.450, p<0.001], [Maya-USC: Z=-11.550, p<0.001], [OpenPose-USC: Z=-6.671, p<0.001]). Based on the comments gathered from the participants, this is mostly attributed to the observation that the 3D models rigged by the other rigging systems had weird deformations when moving, e.g., shoulders in an awkward position, chest area getting bloated when raising the arms, unnatural orientation of the limbs, and the neck stretching out when performing some of

Animation	USC	Kinect	OpenPose	Maya
Walking	2.875 (± 0.6858)	3.775 (± 0.8346)	2.7 (± 1.0686)	2 (± 0.5687)
Jumping Jacks	3.868 (± 0.6633)	2.3 (± 0.9651)	1.875 (± 0.7926)	1.875 (± 0.9983)
Waving	3.65 (± 0.6509)	2.325 (± 0.6934)	2.21 (± 0.5606)	1.45 (± 0.7052)
Sitting	3.05 (± 0.5596)	3.425 (± 0.9357)	3.2 (± 0.9514)	1.8 (± 0.5231)
YMCA Dance	3.975 (± 0.4127)	2.425 (± 0.7122)	2.525 (± 0.7158)	1.575 (± 0.7482)

Table 6.2: Mean scores for each rigging system based on appearance of avatars' motions.

the animations.

Table 6.2 shows the mean score for each rigging system based on the animation played. While the rigged model produced by the USC autorigger yielded the best overall performance, there were some cases wherein another rigging system was preferred. For instance, rigging the body scan based on the Kinect data yielded better looking walking and sitting animations, and is mostly attributed to the participants' observation that the shoulder appears bloated for the USC-rigged model. This implies that the USC autorigger system has some issues determining the actual position of the shoulder joints, most probably because the clothes affected the shape of the reconstructed 3D model. The reason why the Kinect-based rigging system is more desirable in these animations is in part due to the fact that the Kinect-based rigging system is semi-automatic, allowing the user to align the skeleton so that the shoulder joints are approximately in the correct location. The walking and sitting animations also did not involve raising the arms, and thus there was no unnatural chest deformation, which is the reason why the Kinect-based rigging performed better for the walking and sitting animations, but performed worse for the other three animations.

Table 6.3 shows the mean scores for each combination of body reconstruction system and body rigging system, and the results show that the avatar generated by the combination of COLMAP and the USC autorigger was the most preferable. In a way, this result is to be expected, since rigging the 3D model generated by the most-preferred body reconstruction system with the rigging system with the most preferable performance should theoretically generate the best-looking

Body Scanning System	**USC**	**Kinect**	**OpenPose**	**Maya**
COLMAP + OpenMVS	3.52 (± 0.7702)	2.92 (± 0.9205)	2.8 (± 0.8164)	1.86 (± 0.3958)
Autodesk Recap Photo	3.34 (± 0.7599)	2.8 (± 1.0307)	2.08 (± 0.7023)	1.96 (± 0.6910)
COLMAP	3.64 (± 0.7291)	3.04 (± 0.8650)	2.28 (± 0.8301)	2 (± 0.9464)
itSeez3d	3.38 (± 0.7112)	2.64 (± 1.2789)	2.84 (± 1.1430)	1.22 (± 0.6467)

Table 6.3: Average scores of rigging systems against reconstruction methods.

avatar in terms of both visual appearance and motion appearance. Based on the feedback from the participants, seeing the avatars move but not having the fingers intact felt very weird for them, which is why they gave higher scores to the 3D model generated by COLMAP. For the animation, their main focus was the movement of the shoulders, which is why they gave higher scores to the skeleton hierarchy generated by USC autorigger. Based on these results, when building a body scanning system on a limited budget, it may be better to focus on being able to reconstruct all the body parts of the person, rather than focusing on the appearance. As for future model rigging systems, more focus should be done on improving the rigging of the shoulders of the 3D model, due to people being more sensitive to flaws in the shoulder movement, rather than the other body parts.

6.3 Follow-up Evaluation

One of the main limitations of the pilot study is the fact that only one avatar was used for the evaluation. In order to supplement the results of the pilot study, we also attempted to perform body scanning on 4 other people from our laboratory in order to observe how well the body scanning systems perform under different circumstances (e.g., different people wearing different kinds of clothes, different body builds) based on the current body capture setup. Figure 6.8 shows the resulting body scans obtained from the 4 people using the different body scanning systems.

Figure 6.8: Body scans of 4 other people from the laboratory using the different body scanning systems.

Looking at the resulting body scans, it appears that while COLMAP works well when scanning areas that have distinct features (e.g., printed design in the shirt, creases), it has difficulty scanning texture-less areas (plain-colored shirt of the person on the top-most row), and dark-colored areas (dark pants of the

person on the bottom-most row), resulting in a lot of untextured areas and holes in the reconstruction. COLMAP with OpenMVS, on the other hand, still works even for texture-less or dark-colored areas, but does not work well with body parts that are thin, (e.g., the foot of the 3rd and 4th person). Autodesk Recap Photo works well in both cases, but has problems with the scanning the top of the head (notice the bulge on the head), which can be due to not having enough cameras that cover the top part of the person's head. Finally, itSeez3d works well in all cases, but still suffers from not being able to reconstruct the shape of the individual fingers of the hands. Based on these observations, it appears that Autodesk Recap Photo and itSeez3d are more robust to the type of clothing the person is wearing, as well as the person's body build.

7 Application Scenarios

Avatars generated through the reconstruction and rigging process can be used in a variety of applications. In this section, we describe two example applications that we developed.

7.1 Telecommunication

As mentioned in chapter 2, various teleconferencing applications utilize 3D reconstruction technology in order to reconstruct the human users in real-time, and stream the reconstruction over the network to the other user, e.g., [53]. This allows users to see their conversation partner in 3D space, which is expected to further reinforce their sense of being in the same space. However, this requires sending a large volume of data over the network. In order to reduce the amount of data that must be transferred, we suggest to send the 3D reconstruction of the user to the remote user beforehand, and animate it through motion tracking, instead of sending the reconstruction of the user at every frame. This also reduces the requirements of the telecommunication system, as it no longer has to continuously reconstruct the user.

We created a simple prototype of this idea for our Creative and International Competitiveness Project (CICP) 2017, and showed a demo during the open campus held last February 24, 2018. Figure 7.1 shows a diagram of the demo. During the demo, visitors of the open campus wore a Hololens that showed the 3D model of a remote user located in a different room. The movement of the remote user was tracked by a Microsoft Kinect v2. This information was then streamed to the Hololens worn by the visitor to animate the 3D model of the remote user.

Unfortunately, at the time of the demo, the photogrammetry setup was not completed. Therefore, we could not capture the 3D models of the visitors. Thus,

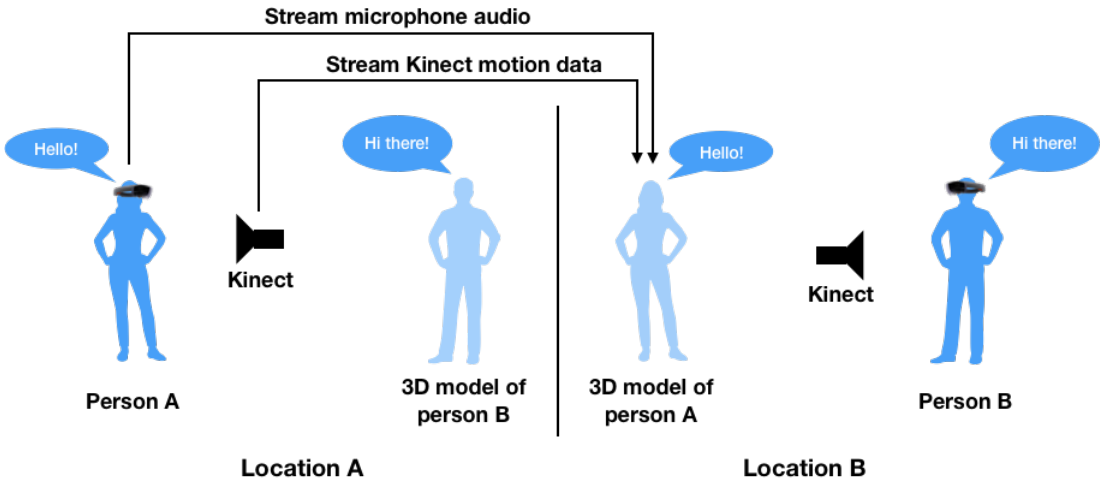
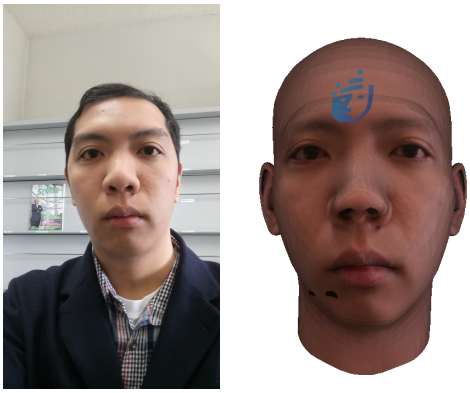


Figure 7.1: Diagram of the telepresence prototype application demonstrated during open campus.



(a) Face Photo (b) Reconstructed Head

Figure 7.2: Sample head reconstruction using the AvatarSDK library.

instead of having a full-body reconstruction of the visitors, we opted to have a reconstruction of only the visitor's head, and attached the reconstructed head into a pre-rigged body. This concept is similar to the work of Malleson et al. [48], who acquired a user's avatar by reconstructing a 3D model of the user's head from a facial photo, and attaching the head model into a pre-rigged body that is morphed based on body measurements obtained from multiple Kinect v2's. This allowed us to give the visitors a glimpse of human reconstruction technology, and to show the demo to more visitors, since a full-body reconstruction requires time. In order to place the head of the visitors into a 3D model, we used the AvatarSDK [12]. This library reconstructs the user's head from a single photo (see figure 7.2).

Finally, feedback from the visitors of the open campus was generally positive, with most of them saying that seeing the 3D model of your conversation partner made them feel like the other person is actually there in front of them and talking to them. However, most of the visitors disliked that the hair was not reconstructed. This resulted in low similarity of the user and the reconstructed 3D model did not resemble them at all.

7.2 Empathy Towards Avatars

Imagine a scenario wherein you are playing a game either in virtual reality (VR), wherein your perception of the environment is replaced by a totally virtual environment; or in augmented reality (AR), wherein virtual objects are superimposed onto the real environment. As you are playing the game, you suddenly see an avatar that looks like your best friend or family member trying to shoot you in the game. As you try to pull out your gun and shoot that avatar, you realize that you are shooting an avatar that looks like your best friend or family member. Will you be hesitant to shoot, or would you think that it is just a game, and thus it does not matter if the avatar looks like someone you know very well. We can explore this question through interaction with realistic avatars created with our system. In this section, we describe a preliminary study in which we explored the idea of hurting avatars that look like your close friend. In this study, we asked participants to drop a heavy box onto avatars of strangers and a close friend, and

investigated whether there was a difference when doing the action in VR versus in AR.

7.2.1 Research Questions and Hypotheses

For this study, we formulated the following research questions and their corresponding hypotheses:

1. *How does the identity of the avatar affect the psychological response of the participants when hurting the avatar* (**RQ1**)? We expected that hurting a friend, whether in real-life or virtually, would elicit a negative emotional response. Thus, we hypothesized that participants would exhibit more negative response when hurting the avatar of their friend (**H1**).

2. *How does the platform (VR or AR) affect the participants' sense of presence with the avatar, and as a result, their response* (**RQ2**)? We expected participants to be more immersed in the VR condition because of the lack of contrast between the nature of the avatar and the environment, i.e., virtual avatar in a virtual environment versus a virtual avatar in a real environment, and thus, be more likely to perceive the avatar to belong in the environment. Thus, participants would experience a greater sense of presence with the avatar in the VR environment, resulting in a greater negative response when hurting the avatars (**H2**).

3. *How does gaming experience affect the participants' response* (**RQ3**)? We expected participants who are frequent gamers to be more desensitized towards hurting people in video games. Thus, participants with less gaming experience would exhibit greater negative response when hurting the avatars (**H3**).

7.2.2 Experiment Scenario

We asked participants to sit in the middle of a room, and wear a head-mounted display (in this case, the Microsoft Hololens) which displayed the avatars of strangers as well as their friend. These avatars would appear in front of the participant one at a time. As soon as an avatar appears, the participants had to press a button as quick as possible, which triggered a heavy box that fell onto the avatar, causing the avatar to fall over. We used 5 different avatars, with each avatar appearing 12 times, where 4 avatars were obtained from actual people that

the participants don't know, and the remaining avatar being the friend's avatar.

7.2.3 Experiment Design and Procedure

We incorporated a within-subjects experiment design, with platform (VR, AR) as the within-subject factor. Participants were randomly assigned to first start with either the VR or the AR condition. In the AR condition, the avatars and the virtual box are superimposed into the real environment, while in the VR condition, everything is placed in a virtual room that has the same layout as the experiment room (Fig. 7.3). For both conditions, we used the Microsoft Hololens to render the virtual content. Since the Hololens is an optical see-through head-mounted display, we attached a custom mask to occlude the participant's view of the real world for the VR condition (Fig. 7.4). To ensure a similar field of view for both conditions, a mask was made for the AR condition to only show the area augmented by the display.

(a) VR (b) AR

Figure 7.3: User's view in the VR condition (a) and AR condition (b).

(a) VR (b) AR

Figure 7.4: Custom masks for VR condition (a) and AR condition (b).

The experiment started with a briefing session, wherein participants were told a cover-up story to hide the real nature of the experiment. After gaining the pair's consent, one had their body scanned, while the other answered a pre-experiment questionnaire. They switched places once the capture was completed. We used an iPad with an Occipital Structure depth camera, along with the itSeez3d app, to acquire the participants' 3D models. We used the system developed by [32] to add a skeleton rig to the 3D model. Participants were then asked to come back on the next day for the actual experiment, this time, individually.

Participants were first given an explanation on how to use the Microsoft Hololens. They were then equipped with the heart-rate measuring device, and were asked to sit in the designated spot. A one-minute rest period was given to stabilize their heart rate. After completing the first condition, participants were asked to come back 2 days after for the second condition. After finishing both conditions, the participants were then asked to fill a post-experiment questionnaire. A debriefing session followed, wherein the real goal of the experiment was revealed. No participant expressed discontent towards the deception. They were then asked for free-form feedback about the experiment. All participants received 1000JPY compensation for their time.

7.2.4 Participants

We recruited 5 pairs of participants for the study (3 male pairs, 1 female pair, and 1 mixed pair) who were 22 to 27 (25 ± 1.56) years old. However, due to problems encountered during the experiment, we had to exclude the results of 1 participant. The pairs were selected such that they are close friends with each other (non-intimate), have known each other for at least a year, and meet each other weekly.

7.2.5 Measures

At the start of the experiment, we recorded information about the participants such as their age, gender, gaming habits, and prior experience with VR/AR. We then considered whether empathy and immersive tendencies have an effect on the results, which we measured using the Toronto Empathy Questionnaire

(TEQ) [60], and the Immersive Tendencies Questionnaire (ITQ) [67], respectively. At the end of each condition, participants rated their sense of presence with the avatar via the social presence questionnaire used in [21]. Finally, a post-experiment questionnaire (7-point Likert-scale type) was used to assess participants' enjoyment and feeling of guilt when dropping the box onto the avatars.

For physiological measures, we considered response time (time taken to press the button), to measure hesitation. We also utilized the heart-rate measurement tool by Yamakawa et al. [68] to observe participants' anxiety levels throughout the experiment.

7.2.6 Results and Discussion

Table 7.1 shows the mean scores of the social presence questionnaire in both VR and AR conditions. A one-way ANOVA test revealed no significant difference in sense of presence with avatar ($F_{1,8} = 2.571, p = 0.147$) between both conditions.

Table 7.1: Social presence questionnaire scores.

Question	Mean (VR)	Mean (AR)
Sense of presence with avatar	4.22±1.31	3.22±2.10
Perception of avatar being not a real person	5.44±0.83	4.56±1.95
Perception of avatar appearing sentient, conscious, and alive	2.78±1.47	2.78±1.31
Perception of avatar being only a computerized image	5.33±1.15	5.78±1.03

We then look at the response times of participants for both conditions and for both avatar types. In the VR condition, the mean response time for the avatar of strangers and the friend is 0.5935 ± 0.0711 and 0.5901 ± 0.0632 seconds, respectively, while in the AR condition, it is 0.5862 ± 0.0804 and 0.6348 ± 0.1111 seconds, respectively. Two-way ANOVA revealed no interaction between condition and response time ($F_{1,8} = 0.899, p = 0.371$), but revealed that avatar type had an effect on the participants' response time ($F_{1,8} = 6.347, p = 0.0358$), particularly in the

AR condition wherein participants had higher response times when dropping the box onto their friend's avatar. Results from t-tests revealed no significant difference between response time of frequent and casual/non-gamers for both stranger ($t = 0.106, p = 0.9206$) and friend ($t = -0.394, p = 0.7139$) avatars in AR.

Observations from the heart-rate measurements revealed a general trend wherein participants experienced an increase in heart-rate upon seeing their friend's avatar for the first time, but remains stable for the remainder of the experiment. Participants were most likely surprised the first time they see their friend's avatar, but then realized that the consequences of their actions were not so bad, and thus they got used to it as the experiment goes on.

Results from the post-experiment questionnaire revealed no significant difference in terms of feeling of guilt for both avatars. In fact, all participants gave the same score for both avatars (mean score: 2.3 ± 1). This also implies no difference between frequent and casual/non-gamers in subjective feelings of guilt.

Finally, post-experiment interviews were also conducted to gather feedback from the participants. In general, participants found the act of dropping a box onto the avatars to be comical. The avatars also did not look realistic enough for them to believe the scenario. Finally, they were more focused on completing the task than being engaged in the experience.

As a follow-up to this preliminary study, we also considered conducting another user study involving a moral dilemma situation, similar to the studies done by Skulmowski et al. [58] and Pan and Slater [55], but involving realistic reconstructions of actual people. We plan to conduct this kind of study in the future using the body scanning system described in chapter 3 for generating the avatars.

8 Conclusion and Future Work

In this study, we explored combinations of 4 different body scanning systems (Autodesk Recap Photo, COLMAP, COLMAP + OpenMVS, itSeez3d) and 4 different model rigging methods (USC Autorigger and Reshaper, Autodesk Maya Autorigger, Kinect-based rigging, and OpenPose-based rigging) for generating realistic human avatars, and evaluated the resulting avatar generated by each combination in terms of the appearance of the reconstructed 3D model, as well as the appearance and accuracy of the animations performed by the avatars when rigged using the different rigging methods.

A pilot study was conducted to determine which body scanning method goes well with which model rigging method to produce appealing and realistic avatars in terms of the appearance of the mesh and the motion of the avatar when animations are played. Results of the pilot study showed that the body scanned 3D model generated by COLMAP was the most preferable in terms of model appearance. This was attributed to the fact that COLMAP was able to reconstruct the shape of the fingers of the scanned person. In terms of the rigging system, the USC autorigger scored the highest overall performance in terms of the appearance of the motion performed by the 3D models when playing an animation. However, participants still noticed some problems with the shoulders of the avatars, especially when playing animations involving large arm movements. This can be attributed to the fact that most auto-rigging systems assume that the subject is either naked, or is wearing tight-fit clothes, and thus the clothing of the subject greatly affects the resulting skeleton rig. Finally, the combination of both COLMAP and the USC autorigger yielded the most preferable avatar, and in this case, the best-performing body scanning system and model rigging system yielded the most preferable avatar.

Aside from the pilot study, we also looked at how well each body scanning

system performed based on the person's attire and body build. It turns out that while COLMAP did well in the pilot study, it exhibited difficulties in scanning texture-less areas, e.g., plain-colored shirts. COLMAP with OpenMVS had no problems with texture-less areas, but had difficulties scanning body parts of a person that are thin. Autodesk Recap Photo and itSeez3d turned out to be the most robust to the attire and body build of the person.

Based on the results of this study, we can say that when building a body scanning setup, especially on a limited budget, it may be better to build the system such that it prioritizes accurate reconstruction of the shape of all the body parts of the person being scanned, especially the hands and the fingers. This means allocating more cameras to body parts that contain fine details, e.g., separation between fingers of the hands, and the hair of the person. When choosing which reconstruction software to use, it is important to consider whether the resulting reconstruction is up-to-scale with the person being scanned, and whether it is robust to texture-less areas and very thin areas, to be able to get accurate scans of people regardless of apparel and body build. As for rigging systems, future automatic rigging systems should focus on the rigging accuracy of the shoulders, for the motion of the shoulders is one of the main things that people notice when looking at a moving human 3D model.

Finally, we also discussed two sample applications we developed that made use of avatars generated using body scanning and model rigging systems. Specifically, we discussed how realistic avatars of people can be used in telepresence application to potentially enhance the feeling of being in the same space with a person in a remote location, and also in psychological experiments where we can potentially observe how people interact with other people in different circumstances that may be impossible to perform in real life. Based on the feedback gathered from users of the two applications, people were generally sensitive about completeness of the body parts of the avatar (users not liking the fact that the hair is missing in the telepresence application), as well as the overall shape of the avatar (users being sensitive to the body build of the avatar in the empathy experiment), which is similar to the results that we got for the pilot study.

There are several limitations to this study that should be addressed in future studies. One such limitation is the fact the number of participants in the user

study was small. Future iterations of this study should definitely use a larger number of participants. Another limitation is that the body scans of only one person (in this case, the author) was used for the pilot study. Future studies should consider evaluating body scans of different people with different body sizes. Another limitation of this study is the fact that the evaluation of the effect of the different model rigging methods to the appearance and accuracy of the avatars' motions only involved subjective ratings from the participants. Future studies should devise a way to also quantitatively measure how the motion of the avatars change as a result of being rigged with different skeletons from different model rigging methods. Another limitation is the fact that the rigging system based on Kinect and OpenPose involved manually aligning the generated skeleton with the 3D model, and the aligning was done by an amateur, which could have affected the results. Future studies should consider either automating the process of aligning the generated skeleton with the 3D model, or having an expert 3D model artist perform the alignment of the skeleton to the 3D model. Finally, another limitation of this study is the fact that avatars generated by integrated avatar acquisition systems, i.e., systems that perform both body scanning and model rigging as a set, were not considered for comparisons, which should be addressed in future studies.

As future work, we plan to explore more combinations of body scanning methods and model rigging methods for creating realistic human avatars. In this study, body reconstruction systems based on the use of depth sensors such as the Kinect were not explored. As such, we plan to also include these types of systems in the evaluation. Future work should also investigate ways of determining the optimal positioning of a set of cameras in a photogrammetry system, and in the process, conduct a study on how the number of cameras affect the resulting avatar. Finally, we also plan to use the generated avatars in different psychological experiment to study how people interact with these avatars in different situations, and to see how the use of realistic avatars can affect user experience in different applications.

Acknowledgements

First of all, I would like to thank the Ministry of Education, Culture, Sports, Science and Technology (MEXT) and the screening committee at NAIST for providing me with the opportunity to study at NAIST as a master's student under a scholarship from the International Priority Graduate Programs.

I would like to thank Professor Hirokazu Kato for providing me with the opportunity to work under the Interactive Media Laboratory, and for imparting his wisdom to us students on how to become better researchers; my adviser, Professor Alexander Plopski, for always giving me advice on how to go about with my research, and for always being patient with me whenever I make mistakes, which I do a lot of; and Professor Takafumi Taketomi and Professor Christian Sandor for giving assistance and advice for my research. I would also like to thank Professor Takatomi Kubo from Mathematical Informatics for lending me the heart-rate measurement device that I used for my preliminary study. I would also like to thank everyone in the Interactive Media Design laboratory for providing a great atmosphere for conducting research.

I would like to thank Dr. Ma. Mercedes Rodrigo from the Ateneo Laboratory for the Learning Sciences, and Dr. Marc Ericson Santos for providing me with the opportunity to visit NAIST as an intern back in April 2013, which ultimately led to my decision of applying to NAIST as a master's degree student.

I would like to thank my friends for their company and support, making my stay in NAIST really enjoyable. I would also like to thank the Filipino community in NAIST for always providing me with a sense of home here in Japan.

Finally, I would like to thank my mom for being open to the idea of me going abroad for further studies, and for always supporting me throughout my journey in life. She is one of the main reasons for me being able to arrive to where I am right now.

References

[1] 3D Body Scanner - Vitronic. https://www.vitronic.com/industrial-and-logistics-automation/sectors/3d-body-scanner.html. Accessed 2018-07-29.

[2] Camera Calibration and 3D Reconstruction - OpenCV 2.4.13.7 Documentation. https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html. Accessed 2018-09-11.

[3] COLMAP - COLMAP 3.6 Documentation. https://colmap.github.io/. Accessed 2018-09-11.

[4] Home of the Blender Project - Free and Open 3D Creation Software. https://www.blender.org/. Accessed 2018-09-11.

[5] itSeez3D. https://itseez3d.com/. Accessed 2018-09-11.

[6] Kinect - Windows App Development. https://developer.microsoft.com/en-us/windows/kinect. Accessed 2018-09-11.

[7] Maya - Computer Animation and Modeling Software - Autodesk. https://www.autodesk.com/products/maya/overview. Accessed 2018-09-11.

[8] MeshLab. http://www.meshlab.net/. Accessed 2018-09-11.

[9] Mixamo. http://www.mixamo.com. Accessed 2018-07-27.

[10] OpenMVS - Open Multi-View Stereo Reconstruction Library. https://github.com/cdcseacave/openMVS. Accessed 2018-09-11.

[11] pi3dscan. http://www.pi3dscan.com. Accessed 2018-07-17.

[12] Realistic 3D Avatars for Games, VR, and AR - Avatar SDK. https://avatarsdk.com/. Accessed 2018-09-03.

[13] ReCap - Reality Capture and 3D Scanning Software - Autodesk. https://www.autodesk.com/products/recap/overview. Accessed 2018-09-11.

[14] Rigify - BlenderWiki. https://en.blender.org/index.php/Extensions:2.6/Py/Scripts/Rigging/Rigify. Accessed 2018-09-11.

[15] Second Life - Virtual Worlds, Virtual Reality, VR, Avatars, Free 3D Chat. https://secondlife.com/. Accessed 2018-09-13.

[16] Shapify Booth - 3D Full Body Scanner. https://www.shapify.me/partner/booth. Accessed 2018-07-29.

[17] Staramba - 3D Instagraph. https://company.staramba.com/3d-instagraph. Accessed 2018-07-29.

[18] Structure Sensor - 3D Scanning, Augmented Reality, and More for Mobile Devices. https://structure.io/. Accessed 2018-09-11.

[19] Unity. https://unity3d.com/. Accessed 2018-09-11.

[20] Laura Aymerich-Franch and Jeremy Bailenson. The Use of Doppelgangers in Virtual Reality to Treat Public Speaking Anxiety: A Gender Comparison. In *Proceedings of the International Society for Presence Research (ISPR'14)*, pages 173–186, Vienna, Austria, 2014.

[21] Jeremy Bailenson, Jim Blascovich, Andrew Beall, and Jack Loomis. Interpersonal Distance in Immersive Virtual Environments. *Personality and Social Psychology Bulletin*, 29(7):819–833, July 2003.

[22] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. OpenFace: An Open Source Facial Behavior Analysis Toolkit. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1–10, New York, USA, 2016.

[23] Ilya Baran and Jovan Popovic. Automatic Rigging and Animation of 3D Characters. *ACM Transactions on Graphics*, 26(3):72:1–72:8, August 2007.

[24] Gaurav Bharaj, Thorsten Thormahlen, Hans-Peter Seidel, and Christian Theobalt. Automatically Rigging Multi-Component Characters. *Computer Graphics Forum*, 31(2):755–764, May 2012.

[25] Stephane Bouchard, Francois Bernier, Eric Boivin, Stephanie Dumoulin, Mylene Laforest, Tanya Guitard, Genevieve Robillard, Johana Monthuy-Blanc, and Patrice Renaud. Empathy Toward Virtual Humans Depicting a Known or Unknown Person Expressing Pain. *Cyberpsychology, Behavior, and Social Networking*, 16(1):61–71, January 2013.

[26] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7291–7299, Honolulu, USA, 2017.

[27] Yan Cui, Will Chang, Tobias Nöll, and Didier Stricker. KinectAvatar: Fully Automatic Body Capture Using a Single Kinect. In *Asian Conference on Computer Vision*, pages 133–147, Daejeon, Korea, 2012.

[28] Hein A. M. Daanen and Frank B. Ter Haar. 3D Whole Body Scanners Revisited. *Displays*, 34(4):270–275, October 2013.

[29] Mingsong Dou, Henry Fuchs, and Jan-Michael Frahm. Scanning and Tracking Dynamic Objects with Commodity Depth Cameras. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 99–106, Adelaide, Australia, 2013.

[30] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdie, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, Pushmeet Kohli, Vladimir Tankovich, and Sharam Izadi. Fusion4D: Real-Time Performance Capture of Challenging Scenes. *ACM Transactions on Graphics*, 35(4):114:1–114:13, July 2016.

[31] Mingsong Dou, Jonathan Taylor, Henry Fuchs, Andrew Fitzgibbon, and Shahram Izadi. 3D Scanning Deformable Objects With a Single RGBD Sensor. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 493–501, Boston, USA, 2015.

[32] Andrew Feng, Dan Casas, and Ari Shapiro. Avatar Reshaping and Automatic Rigging using a Deformable Model. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, pages 57–64, Paris, France, 2015.

[33] Andrew Feng, Gale Lucas, Stacy Marsella, Evan Suma, Chung-Cheng Chiu, Dan Casas, and Ari Shapiro. Acting the Part: The Role of Gesture on Avatar Identity. In *Proceedings of the International Conference on Motion in Games*, pages 49–54, California, USA, 2014.

[34] Andrew Feng, Ari Shapiro, Wang Ruizhe, Mark Bolas, Gerard Medioni, and Evan Suma. Rapid Avatar Capture and Simulation Using Commodity Depth Sensors. *Computer Animation and Virtual Worlds*, 25(3–4):201–211, May 2014.

[35] Jesse Fox, Jeremy Bailenson, and Joseph Binney. Virtual Experiences, Physical Behaviors: The Effect of Presence on Imitation of an Eating Avatar. *Presence: Teleoperators and Virtual Environments*, 18(4):294–303, August 2009.

[36] Pascal Fua, Armin Gruen, Ralf Plankers, Nicola D'Apuzzo, and Daniel Thalmann. Human Body Modeling And Motion Analysis from Video Sequences. pages 866–873, Hakodate, Japan, 1998.

[37] Maia Garau, Mel Slater, Vinoba Vinayagamoorthy, Andrea Brogni, Anthony Steed, and M. Angela Sasse. The Impact of Avatar Realism and Eye Gaze Control on Perceived Quality of Communication in a Shared Immersive Virtual Environment. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 529–536, New York, USA, 2003.

[38] Jin Gu, Terry Chang, Ivan Mak, S. Gopalsamy, H. C. Shen, and M. M. F. Yuen. A 3D Reconstruction System for Human Body Modeling. In *Proceedings of the International Workshop on Capture Techniques for Virtual Environments*, pages 229–241, Geneva, Switzerland, 1998.

[39] Hal Hershfield, Daniel Goldstein, William Sharpe, Jesse Fox, Leo Yeykelis, Laura Carstensen, and Jeremy Bailenson. Increasing Saving Behavior

Through Age-Progressed Renderings of the Future Self. *Journal of Marketing Research*, 48(SPL):S23–S37, November 2011.

[40] Adrian Hilton, Daniel Beresford, Thomas Gentils, Raymond Smith, and Wei Sun. Virtual People: Capturing Human Models to Populate Virtual Worlds. In *Proceedings of the Computer Animation Conference*, pages 174–185, Geneva, Switzerland, 1999.

[41] Jack Hollingdale and Tobias Greitemeyer. The Changing Face of Aggression: The Effect of Personalized Avatars in a Violent Video Game on Levels of Aggressive Behavior. *Journal of Applied Social Psychology*, 43(9):1862–1868, 2013.

[42] Dongsik Jo, Ki-Hong Kim, and Gerard Jounghyun Kim. Effects of Avatar and Background Types on User's Co-presence and Trust for Mixed Reality-Based Teleconference Systems. In *Proceedings of the International Conference on Computer Animation and Social Agents*, pages 27–36, Seoul, South Korea, 2017.

[43] Ioannis Kakadiaris and Dimitri Metaxas. Three-Dimensional Human Body Model Acquisition from Multiple Views. *International Journal of Computer Vision*, 30(3):191–218, December 1998.

[44] Marek Kowalski, Jacek Naruniec, and Michal Daniluk. Livescan3D: A Fast and Inexpensive 3D Data Acquisition System for Multiple Kinect v2 Sensors. In *International Conference on 3D Vision*, pages 318–325, Lyon, France, 2015.

[45] Gregorij Kurillo, Ruzena Bajcsy, Klara Nahrsted, and Oliver Kreylos. Immersive 3D Environment for Remote Collaboration and Training of Physical Activities. In *Proceedings of the IEEE Virtual Reality Conference*, pages 269–270, Nevada, USA, 2008.

[46] Hao Li, Etienne Vouga, Anton Gudym, Linjie Luo, Jonathan Barron, and Gleb Gusev. 3D Self-Portraits. *ACM Transactions on Graphics*, 32(6):187:1–187:9, November 2013.

[47] Gale Lucas, Evan Szablowski, Jonathan Gratch, Andrew Feng, and Tiffany Huang. The Effect of Operating a Virtual Doppelganger in a 3D Simulation. In *Proceedings of the 9th International Conference on Motion in Games*, pages 167–174, Burlingame, California, 2016.

[48] Charles Malleson, Maggie Kosek, Martin Klaudiny, Ivan Huerta, Jean-Charles Bazin, Alexander Sorkine-Hornung, Mark Mine, and Kenny Mitchell. Rapid One-Shot Acquisition of Dynamic VR Avatars. In *IEEE Virtual Reality*, pages 131–140, Los Angeles, USA, 2017.

[49] Christian Miller, Okan Arikan, and Don Fussell. Frankenrigs: Building Character Rigs from Multiple Sources. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 31–38, Washington, USA, 2010.

[50] Masahiro Mori. *Bukimi no Tani [The Uncanny Valley]*. 1970.

[51] Sahil Narang, Andrew Best, Ari Shapiro, and Dinesh Manocha. Generating Virtual Avatars with Personalized Walking Gaits using Commodity Hardware. In *Proceedings of the Thematic Workshops of ACM Multimedia*, pages 219–227, California, USA, 2017.

[52] Richard Newcombe, Sharam Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, Basel, Switzerland, 2011.

[53] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip Davidson, Sameh Khamis, Mingsong Dou, Vladimir Tankovich, Charles Loop, Qin Cai, Philip Chou, Sarah Mennicken, Julien Valentin, Vivek Pradeep, Shenlong Wang, Sing Bing Kang, Pushmeet Kohli, Yuliya Lutchyn, Cem Keskin, and Shahram Izadi. Holoportation: Virtual 3D Teleportation in Real-Time. In *Proceedings of the Annual Symposium on User Interface Software and Technology*, pages 741–754, Tokyo, Japan, 2016.

[54] Junjun Pan, Xiaosong Yang, Xin Xie, Philip Willis, and Jian Zhang. Automatic Rigging for Animation Characters with 3D Silhouette. *Computer Animation and Virtual Worlds*, 20(2–3):121–131, June 2009.

[55] Xueni Pan and Mel Slater. Confronting a Moral Dilemma in Virtual Reality: A Pilot Study. In *Proceedings of the BCS Conference on Human-Computer Interaction*, pages 46–51, Newcastle-upon-Tyne, United Kingdom, 2011.

[56] Kathryn Segovia and Jeremy Bailenson. Virtually True: Children's Acquisition of False Memories in Virtual Reality. *Media Psychology*, 12(4):371–393, December 2009.

[57] Junichiro Seyama and Ruth Nagayama. The Uncanny Valley: Effect of Realism on the Impression of Artificial Human Faces. *Presence Teleoperators & Virtual Environments*, 16(4):337–351, August 2007.

[58] Alexander Skulmowski, Andreas Bunge, Kai Kaspar, and Gordon Pipa. Forced-Choice Decision-Making in Modified Trolley Dilemma Situations: A Virtual Reality and Eye Tracking Study. *Frontiers in Behavioral Neuroscience*, 8(1):426:1–16, December 2014.

[59] Mel Slater, Aitor Rovira, Richard Southern, David Swapp, Jian Zhang, Claire Campbell, and Mark Levine. Bystander Responses to a Violent Incident in an Immersive Virtual Environment. *PLoS ONE*, 8(1):e52766:1–13, January 2013.

[60] R. Nathan Spreng, Margaret McKinnon, Raymond Mar, and Brian Levine. The Toronto Empathy Questionnaire Scale: Scale Development and Initial Validation of a Factor-Analytic Solution to Multiple Empathy Measures. *Journal of Personality Assessment*, 91(1):62–71, July 2010.

[61] William Swartout, Ron Artstein, Eric Forbell, Susan Foutz, H. Chad Lane, Belinda Lange, Belinda Lange, Jacquelyn Ford Morie, Albert Skip Rizzo, and David Traum. Virtual Humans for Learning. *AI Magazine*, 34(4):13–30, December 2013.

[62] Tomohiro Tanikawa, Yasuhiro Suzuki, Koichi Hirota, and Michitaka Hirose. Real World Video Avatar: Real-time and Real-size Transmission and Presentation of Human Figure. In *Proceedings of the International Conference on Augmented Tele-Existence*, pages 112–118, Christchurch, New Zealand, 2005.

[63] Jing Tong, Jin Zhou, Ligang Liu, Zhigeng Pan, and Hao Yan. Scanning 3D Full Human Bodies using Kinects. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):643–650, April 2012.

[64] Matias Volonte, Sabarish Babu, Himanshu Chaturvedi, Nathan Newsome, Elham Ebrahimi, Tania Roy, Shaundra Daily, and Tracy Fasolino. Effects of Virtual Human Appearance Fidelity on Emotion Contagion in Affective Inter-Personal Simulations. *IEEE Transactions on Visualization and Computer Graphics*, 22(4):1326–1335, April 2016.

[65] Ruizhe Wang, Jongmoo Choi, and Gerard Medioni. Accurate Full Body Scanning from a Single Fixed 3D Camera. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 432–439, Zurich, Switzerland, 2012.

[66] Helen Wauck, Gale Lucas, Ari Shapiro, Andrew Feng, Jill Boberg, and Jonathan Gratch. Analyzing the Effect of Avatar Self-Similarity on Men and Women in a Search and Rescue Game. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 485:1–485:12, Montréal, Canada, 2018.

[67] Bob Witmer and Michael Singer. Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3):225–240, June 1998.

[68] Toshitaka Yamakawa, Koichi Fujiwara, Miho Miyajima, Erika Abe, Manabu Kano, and Yuichi Eda. Real-time Heart Rate Variability Monitoring Employing a Wearable Telemeter and a Smartphone. In *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–4, Siem Reap, Cambodia, 2014.