Creating a Virtual Mirror for Motor Learning in Virtual Reality

Dissertation

zur Erlangung des akademischen Grades des
Doktors der Naturwissenschaften (Dr. rer. nat.)
an der Technischen Fakultät der Universität Bielefeld

vorgelegt von
Thomas Waltemate

Bielefeld 2018

Versicherung

Hiermit versichere ich,

- dass mir die geltende Promotionsordnung der Fakultät bekannt ist,
- dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte von Dritten oder eigener Prüfungsarbeiten ohne Kennzeichnung übernommen und alle benutzten Hilfsmittel und Quellen in meiner Arbeit angegeben habe,
- dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Vermittlungstätigkeiten oder für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe und
- dass ich keine gleiche, oder in wesentlichen Teilen ähnliche oder eine andere Abhandlung bei einer anderen Hochschule als Dissertation eingereicht habe.

Bielefeld, 2018

Thomas Waltemate

The supplemental material can be found here:
`http://doi.org/10.4119/unibi/2932579`.

Printed on non-aging paper in compliance with DIN-ISO 9706.

Abstract

A mirror is an important and helpful tool, which we all use in everyday life. In general, it helps us to get a better idea of what we actually look like, i.e., it is used to build a richer mental representation of our own body. A mirror provides a perspective of the body or body parts, which otherwise would not be possible, since major parts of the body, e.g. the face, are not even visible to us without a mirror or a similar tool. Therefore, mirrors are essential and very important instruments of everyday life.

Apart from such everyday life usage, mirrors are regularly used in fitness and dance studios or in sports training in general. Here, they usually serve the purpose to observe the own movement to check, for instance, whether a given motor task is performed correctly or whether the body moves as we expect it to. For this *motor learning* scenario the mirror is the perfect tool, since it constitutes a simple yet powerful method to provide this kind of intuitive feedback.

However, the mirror metaphor is this much established in the real world, that no one would question their own mirror image — assuming that the mirror provides a perfect non-distorted reflection. This kind of recognition is not necessarily the case for a *virtual mirror* (an artificial mirror created in a virtual environment), since here the mirror does not provide a real reflection of the physical world. Therefore, it is a nontrivial task to create such a virtual mirror in a convincing way. However, a virtual mirror plays a central role in virtual reality as a tool to perform experiments concerning virtual embodiment as well as in virtual environments dedicated to motor learning.

Thus, this thesis explores how to thoroughly convey the mirror metaphor to Virtual Reality (VR) and further investigates major factors, that are important for such a mirror with a special focus on motor learning. Here, important requirements have to be fulfilled to provide proper motor learning with a virtual mirror in VR. These requirements lead to a specific set of hard- and software, which is introduced in this work. Additionally, based on this system different crucial factors to create a convincing virtual mirror are evaluated. While there are actually various factors influencing the effects of the virtual mirror, this work concentrates on three major factors: latency, personalization and immersion.

In order to investigate the factor personalization, personalized virtual characters of real persons are necessary. Therefore, this thesis additionally presents a technique to create personalized ready-to-animate virtual characters in a few minutes, in order to use them in such an experiment.

Contents

1

Introduction

A mirror is a useful tool of every day life. It usually serves the purpose to see ourselves from a different perspective. More important, some parts of our bodies, e.g., our face, are only visible with the help of mirrors (or similar tools). This metaphor can be used to build a richer mental representation of our outer selves or, in more simpler terms, it helps us to get to know how we actually look like. The mirror metaphor is long standing and well established; we usually recognize ourselves in a mirror. This is actually based on different cues. First of all, we do have a rich mental representation of ourselves and thus already *know* what we look like, which, for instance, also allows to recognize ourselves in videos or on pictures. Second, when we move in front of the mirror, our counterpart in the mirror does the same movement without any perceivable delay. This is a strong cue, which couples the mental representations of our sent motor commands with the visual result in the mirror. This is also known as *visuomotor synchrony*. Altogether, we know that it is our reflection in the mirror and thus can utilize our mirror image for different tasks, ranging from a simple check whether everything is alright with our outer appearance over everyday tasks like styling our hair or putting on make-up to rather complex tasks, e.g., practicing dance moves or performing fitness exercises. This makes mirrors interesting and highly useful tools to learn new or to improve already known motor tasks. Mirrors are actually frequently used to help in such motor learning scenarios, for instance, in sports/dance training as well as rehabilitation. In this context, we can observe ourselves and the movements we are performing from a different view, which otherwise would be impossible. This view allows us to check whether we perform a given motor task correctly or not.

A mirror, as we use it in every day life, is subject to the laws of physics. It presents our mirror image based on light reflected from a, usually flat, surface, e.g., a calm water surface or a manufactured mirror consisting of a layer of aluminum or silver beneath a flat glass plane. This provides a nearly perfect reflection of us and the surroundings. Additionally, while there is also delay for a real mirror, since light also needs time to travel, the delay is not perceivable for us. Now imagine moving in front of a mirror and the person looking back at you, doing the same movement, is not what you are used to see in the mirror, e.g., wearing different clothes or looks nothing like you are used to see in the mirror. Further, imagine the reflection you see in the mirror moves exactly as you do, but a noticeable moment later. Will you still accept the mirror image as your mirror image?

While these issues will not occur with a mirror in reality, this is totally possible or even likely with a *virtual mirror* (Figure 1.1) in Virtual Reality (VR). Such a virtual mirror,

Figure 1.1: The virtual mirror image created with the techniques and technologies introduced in this work.

while bound to a different set of rules and limitations, is neither subject nor bound to the laws of physics. This on the one hand opens the door to a lot of possibilities but on the other hand also introduces a lot of challenges and limitations. Therefore, when it comes to creating a virtual mirror many important issues have to be kept in mind. First of all, the right hardware, like display technology and motion tracking system, has to be chosen. Then, software architecture to record and stream motion data in real-time from the motion tracking system to the render engine is needed. The render engine in turn has to fulfill certain requirements to visualize the virtual mirror in a proper way, e.g., low latency and visual quality. Hence, in Chapter 2 this thesis tackles these important issues and explains how to assemble a decent system, which thoroughly creates the illusion of a virtual mirror. A virtual mirror system like this is the perfect tool to get used in VR systems dedicated to motor learning like the system we have developed in the Intelligent Coaching Space (ICSPACE) project. This project has the goal to create a virtual environment to help with learning of new or improvement of already known motor tasks. In this context we use the virtual mirror as our main feedback channel by on the one hand allowing self-monitoring and on the other hand augmenting the mirror with additional information like, e.g., color highlighting of erroneous movements. The acceptation of the virtual mirror image as the

own reflection is important in this scenario to create a training experience which is as natural and intuitive as possible. Therefore, this work additionally investigates the influence of crucial factors and their impact on the virtual mirror or, more precisely, on the effects of virtual embodiment.

Virtual embodiment means that we are represented by some kind of body in the virtual environment. This embodiment can elicit a set of psychophysical effects (see Section 3.2 for details). The virtual mirror is the perfect tool to experimentally investigate these effects, since the virtual mirror allows to show users their full-body mirror image in an intuitive and natural way, in contrast to, e.g., a third person perspective. This work concentrates on the impact of the end-to-end latency, personalization of the avatar and level of immersion on virtual embodiment effects as well as on motor performance.

In this context end-to-end latency means the delay of a user's movement until visualization on the display device. This is also called the motion-to-photon latency. As already mentioned above, a real mirror does not have any noticeable latency and therefore we are not used to any latency of the mirror image. Still, in VR there always is and will be a comparably high amount of latency, since processing and transport of data takes time. While this latency can be reduced to a minimum, it will never completely vanish. Therefore, this thesis provides proper advice and background information on how to minimize latency in virtual mirror systems. Moreover, another part of this thesis is to fathom what are the impacts of latency on the virtual mirror starting from low and usually not noticeable latency to rather high and clearly perceptible latency. Here, especially the effects on the different aspects of virtual embodiment, e.g., body ownership and agency, are of high interest as well as the impact on motor performance. Especially the latter is important when the virtual mirror is employed in the context of motor learning. The experiment concerning the impact of latency and its results are presented in Chapter 3.

Similar to the factor latency, which we do not expect and do not encounter in front of a real mirror, we do not expect to see anything different than ourselves in a real mirror. But, in the virtual mirror scenario it is actually even likely that we see someone or something different in the mirror, since the virtual mirror image can be represented in various ways. We, our virtual versions, in the virtual world are usually represented by *avatars*. Thus, also the virtual mirror image is usually created by using such *mirror avatars*. To this end, one can use various types of avatars. These range from rather abstract non-humanlike versions, like stick figures, mannequins or robots, over more humanlike but still generic and less realistic representations up to realistic personalized avatars created from real people. While all of these representations will allow to create a virtual mirror image of some kind, only the latter will provide the most intuitive one, since those personalized avatars will actually look like the user in front of the mirror. Therefore, they are closest to what we see in an actual mirror.

However, it is not a trivial task to create such personalized avatars and there are different techniques available to do so. Currently the most sophisticated one is to create them by 3D scanning. Here, at the time of writing of this thesis the state-of-the-art method is photogrammetry — also known as multi-view stereo reconstruction. The scans created with this technique are completely static and cannot be animated in any way. Though, in order to create an interactive virtual mirror with them, mirror avatars need to be ready-to-animate. They need at least an embedded skeleton rig, which is used as a proxy to drive the animation of the overlaying geometry representing the mirror avatar. Therefore, further processing is necessary to produce ready-to-animate avatars. This usually is a tedious and time consuming task, but in order to conduct studies with fully personalized avatars as presented in this work, avatars ideally need to be ready in a few minutes after the scan. Additionally, the resulting avatars must be of sufficient quality to create a convincing mirror image. Hence, in Chapter 4 this work additionally deals with the fast generation of such ready-to-animate virtual avatars at a decent quality level.
With the ability to create such *virtual humans* in hand, an experiment examining the impact of personalized virtual mirror avatars on various virtual embodiment effects was conducted. This work presents this experiment and its results in Chapter 5.
Furthermore, while motor learning in VR employing a virtual mirror is best performed in Cave Autonomous Virtual Environments (CAVE) for several reasons, which are motivated in Chapter 2, there are certainly other devices, which are capable to present a virtual mirror. Especially devices, which yield a high level of immersion, are of great interest. Currently Head Mounted Displays (HMD) have the edge when it comes to high immersion levels. Thus, in Chapter 5 this thesis additionally sheds light on which display device (CAVE or HMD) creates higher levels of presence and body ownership while visualizing the virtual mirror.

Contribution

Summarized, the contributions of this thesis are:

- A thorough investigation of how to create a virtual mirror hard- and software-wise, which is particularly suitable for motor learning in VR and in general applicable for virtual embodiment experiments.
- An evaluation of the impact of different levels of latency on virtual embodiment effects and motor performance.
- A holistic approach to create ready-to-animate virtual humans in a short amount of time at a decent quality level.
- An evaluation of the impact of personalized avatars and level of immersion on virtual embodiment effects as well as presence.

Publications

The contributions of this thesis have been published in the following publications:

Thomas Waltemate, Felix Hülsmann, Thies Pfeiffer, Stefan Kopp & Mario Botsch. Realizing a Low-latency Virtual Reality Environment for Motor Learning. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 139–147, 2015.

Thomas Waltemate, Irene Senna, Felix Hülsmann, Marieke Rohde, Stefan Kopp, Marc Ernst & Mario Botsch. The Impact of Latency on Perceptual Judgments and Motor Performance in Closed-loop Interaction in Virtual Reality. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 27–35, 2016.

Matthias Schröder, Thomas Waltemate, Jonathan Maycock, Tobias Röhlig, Helge Ritter & Mario Botsch. Design and Evaluation of Reduced Marker Layouts for Hand Motion Capture. *Computer Animation and Virtual Worlds*, e1751, 2017.

Jascha Achenbach, Thomas Waltemate, Marc Erich Latoschik & Mario Botsch. Fast Generation of Realistic Virtual Humans. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 2017.

Thomas Waltemate, Dominik Gall, Daniel Roth, Mario Botsch & Marc Erich Latoschik. The Impact of Avatar Personalization and Immersion on Virtual Body Ownership, Presence, and Emotional Response. *IEEE Transactions on Visualization and Computer Graphics*, 40:1643–1652, 2018.

Other publications that are not directly relevant to this work but to which I have contributed during the work on the ICSPACE project are:

Iwan de Kok, Felix Hülsmann, Thomas Waltemate, Cornelia Frank, Julian Hough, Thies Pfeiffer, David Schlangen, Thomas Schack, Mario Botsch & Stefan Kopp. The Intelligent Coaching Space: A Demonstration. In *Proceedings of the International Conference on Intelligent Virtual Agent*, pages 105–108, 2017.

Marc Erich Latoschik, Daniel Roth, Dominik Gall, Jascha Achenbach and Thomas Waltemate & Mario Botsch. The Effect of Avatar Realism in Immersive Social Virtual Realities. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 2017.

2

Low Latency VR Environment for Motor Learning

2.1 Introduction

Learning of new motor tasks or improvement of already known ones is essential in many domains such as fitness training or rehabilitation. Often it is useful and much more efficient to learn motor tasks with the help of a coach or another external source of feedback, since in this way the athlete gets information about whether or not the given motor task was executed correctly and what kind of errors were made.

The extensive capabilities of Virtual Reality (VR) seem to be an ideal candidate to facilitate and boost the learning process [RK05, SBK+14]: A VR environment can be equipped with high-precision sensors and can employ various feedback channels. Highly precise sensors gather data of the trainee, which are analyzed in real time, in order to provide directed purposeful feedback over various channels. This feedback can either be given after the movement execution or — more interestingly — already during the execution. Especially in the latter case it is important to ensure that the feedback is precisely timed, so that it is presented exactly when it is relevant. As a consequence, the environment has to be highly controlled, i.e., properties like the end-to-end latency or tracking robustness either must be controlled or at least have to be taken into account. It thus seems necessary to report such basic properties of a system in every research addressing issues of motor learning in VR. This would allow researchers to compare systems and to reproduce studies more reliably.

However, for many systems described in literature, no sufficient information on relevant aspects such as end-to-end latency, robustness of the motion capture system, et cetera is given. Thus when building up a new VR environment, one is faced with a vast number of potential techniques and technologies, but a well-informed choice is hardly possible.

Hence, we aim at improving this situation by

1. providing general requirements towards VR systems for motor learning,
2. evaluating and assessing state-of-the-art techniques and technologies,
3. presenting a system built according to the aforementioned requirements,
4. providing latency measurements of the virtual environment and giving hints on how to reduce latency.

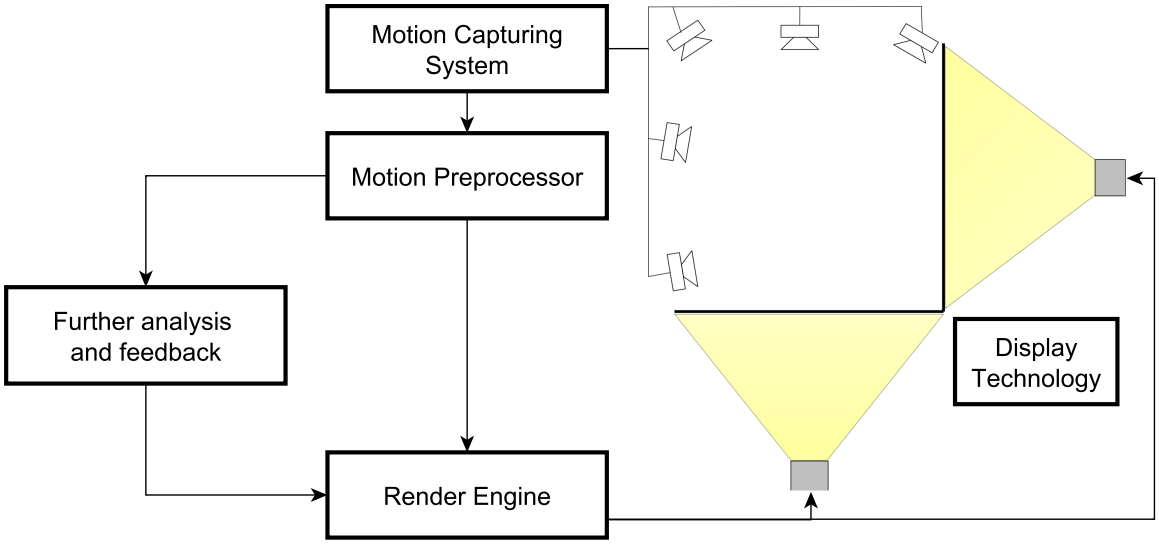


Figure 2.1: A minimal architecture for a VR environment for motor learning combines a motion capturing system, motion processing (e.g., for re-targeting or motion analysis), as well as a render engine for high-fidelity character rendering.

The system we present is actually used in the project ICSPACE[1]. A minimal version of such a VR system for motor learning would consist of components for motion capturing, pre-processing of motion data, motion analysis, feedback generation, rendering and display technology (see Figure 2.1). These components allow to create a virtual mirror, which is the core component of the presented VR environment for motor learning. As rendering and motion capturing are the backbones of such a VR environment in general and the virtual mirror in particular, this chapter focuses on these two components. We want to stress, however, that the choice of display technology is very important, as projectors can be found with latency in the range between 15 ms and 140 ms. But the decision for display hardware is relatively easy to make based on technical specifications.

In the following, we start by developing general requirements towards motor learning in VR applications (Section 2.2). This is followed by background information about latency and how to minimize latency in the context of the virtual mirror (Section 2.3). After discussing related motor learning approaches in Section 2.4, we present in Section 2.5 the essentials of our low-latency VR environment, while also assessing particular state-of-the-art techniques and technologies for motion capturing and real-time rendering. In Section 2.6, we present an evaluation of our system and report results of a pilot user study, before concluding in Section 2.7.

[1] http://graphics.uni-bielefeld.de/research/icspace/

My Contribution *The VR environment for motor learning presented in this chapter was mainly developed in close cooperation with Felix Hülsmann. My main contribution to this chapter is the render engine and the techniques to visualize the virtual mirror. Additionally, I contributed to setting up the whole system and the compilation of the requirements. Moreover, I executed the latency measurements together with Felix Hülsmann. Felix Hülsmann evaluated the motion tracking systems to eventually choose the one that is most suitable to our needs. Further, he worked on the motion preprocessing to convert the motion data so that it can be streamed into the render engine. Additionally, he conducted and evaluated the pilot study.*
Corresponding publication: Realizing a Low-latency Virtual Reality Environment for Motor Learning, VRST 2015.

2.2 Requirements

In this section we develop requirements necessary for an efficient motor learning system in VR. Many researchers already pointed out some of the most crucial requirements for VR applications in general: For instance, Bierbaum et al. [Bie00] provide an overview including general features like low latency, high frame rate, tracking robustness, but also engineering requirements such as extensibility and hardware abstraction. To our knowledge, this has not yet been done for VR systems specialized on motor learning. In the following we therefore carve out the most important requirements.

R1: Feedback on one's own motion

As a first requirement, users have to be able to verify the correct execution of a given motor task by getting feedback of whatever kind. This feedback should be as intuitive as possible, and one of the most intuitive ways is to let users observe their own motion by viewing their own body.
In real world scenarios, like fitness or dance studios, self-monitoring is usually achieved through a mirror. Thus, it seems desirable to provide mirror-like feedback in VR training environments as well [Häm04]. This is inter alia motivated by findings of Chau et al. [CCD+03], who found that none of their proposed layouts of students and teachers could improve upon a standard face-to-face configuration—similar to that of a mirror—when learning Tai Chi. A *virtual* mirror, as provided in our setup, may serve multiple purposes: it may show the optimal performance, just as a teacher would, to guide the performance of the trainee; it can simply reflect the real performance of the trainee to support self-monitoring; or it could add augmentations to the real performance, e.g., emphasizing errors. Finally, it serves as a perfect base for further feedback strategies. Besides face-to-

face layouts, a third person view could also improve training results, as has been recently shown by Covaci et al. [COM14]. In summary it can be said that self-monitoring is an essential ingredient on the way towards meaningful visual feedback.

R2: Low latency and high frame rate

The immediate real-time feedback provided by the virtual mirror (R1: Feedback on one's own motion) or other feedback during the interaction with our system is prone to the system's latency.

There are many studies which investigated the impact of latency on different tasks and in different environments. For controlling characters in computer games a latency as high as 150 ms might be acceptable, but higher latency is already directly noticeable for untrained users and affect players in several ways [JNS12]. Meehan et al. [MRWBJ03] showed that decreasing the latency from 90 ms to 50 ms already affects presence in virtual environments. Mackenzie and Ware [MW93] used Fitt's tapping task to investigate the influence of latency on performance: They found that the performance of participants is reduced when being exposed to a latency of 75 ms or higher. According to Ware and Balakrishnan [WB94] even a latency of 70 ms already affects performance in a VR reaching task. In a non-VR tapping task Jota et al. [JNDW13] found that performance improves only little using latency below 25 ms. Even for latency below 50 ms, only a very slight improvement was measured. Improvements in latency below 40 ms were not even noticed by most untrained participants. In a study on collaborative virtual environments, Park et al. [PK99] show that with increased latency, humans adopt a move-and-wait strategy, waiting several seconds to let their views synchronize, before continuing performing their tasks. They showed that in such setups jitter had a larger impact on collaborative performance than latency. The development of similar strategies has to be avoided in our target scenario, as it would hamper with the natural flow of movements.

In summary, depending on the task as well as the environment different acceptable levels of latency were found. Therefore, no clear picture emerged so far for tolerable delays in systems for motor learning featuring a virtual mirror in a CAVE like ours.

Nevertheless, it seems to be desirable to reach the lowest possible latency. According to the revised literature an optimal corridor of latency for visual feedback appears to be between 40 ms and 70 ms, depending on the specific application.

Please have a look at Chapter 3 for more information and related work concerning latency. In that chapter we present the results of an experiment investigating the impact of latency on closed-loop interaction with a virtual mirror and corresponding embodiment effects as well as on motor performance. Among other interesting findings, we found that latency of about 75 ms already significantly worsens motor performance in this scenario. Therefore,

these results substantiate the importance of a low latent and highly responsive system when it comes to motor learning in VR employing a virtual mirror.

R3: Minimal level of disturbance

To guarantee a natural and intuitive training, the user should be able to move freely, at least regarding the movements that are relevant for the motions to be trained. Thus the hardware attached to the user has to be as unobtrusive as possible, since otherwise the user would not be able to use his/her full range of motion. For instance, the use of long and stiff wires as well as heavy components should be prevented if possible: Users should perform the motor actions as they would in a real training scenario. Besides issues of naturalness of movements, obtrusive hardware could also make the optimal perception of the virtual environment more difficult [WS98]. Therefore motion capturing system and VR environment have to be chosen to offer a reasonable compromise between tracking precision, immersion, and obtrusiveness.

R4: Robust tracking

Many typical sports exercises include movements during which parts of the body are occluded for outside-in tracking systems. The motion capture system has to be as robust as possible against such kind of occlusions, where single or multiple markers/body parts might get lost. If the tracking is not robust enough, it might require a re-calibration of the human that is to be tracked. Thus, the training has to be interrupted and cannot be continued until the re-calibration is performed. The training is severely affected by such a re-calibration procedure to re-align tracking: If this happens, the naturalness of the application, as for instance demanded by Witmer and Singer [WS98], would be significantly reduced.

Notes

From all requirements, we identified low latency (R2) as one of the most crucial factors. We therefore argue that the latency of a VR environment for motor learning should be reported whenever presenting results of studies conducted in such a setup. This is important to exclude high latency as a potential side effect in the conducted experiments. More generally, when the exact specifications of an environment are known, the results of future experiments become better comparable as well as more reproducible. That is why we lay a special focus on latency measurement in this work.

2.3 Background

2.3.1 About Latency

Latency in general cannot be avoided. It can be only minimized. Even a real mirror has a minimum amount of latency t, which results from twice the distance to the mirror d divided by the speed of light $c = 299792458\,\frac{m}{s}$:

$$t = \frac{2d}{c}.$$

The factor two is there due to the fact that light has to travel from the person's body to the mirror and back to the eye. Thus, a person standing two meters away from a mirror, will perceive his/her mirror image with a latency of about $t \approx 13\,ns$. While this very low latency is not perceived by humans at all, it is still there.

However, the virtual mirror as discussed in this thesis has a much higher latency. The lowest possible latency measured using the proposed system while still rendering a minimal scene and a mannequin mirror avatar without shadows (see Figure 2.2), is about $45\,ms$ (at 240 fps). This is about $3.5\,M$ times higher than the real mirror. Still, this is actually a low latency for this kind of setup and the employed hardware. Moreover, this amount of latency is actually hardly noticeable at all, as is shown in Chapter 3. Nevertheless, it is a complex task to actually minimize latency. Therefore, the upcoming section reports what are the causes for latency and how they can be optimized in order to minimize latency.

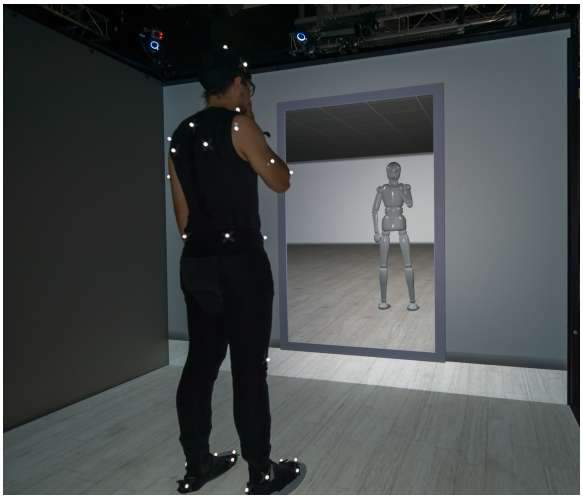


Figure 2.2: Minimal virtual mirror scene with a mannequin character and empty room.

2.3.2 Minimizing Latency

Every component of the virtual mirror pipeline generates latency and thus contributes to the final end-to-end latency of the system. These components are:

- Motion capturing
- Data transport
- Visualization.

Motion capturing In order to map the motion of the user moving in front of the virtual mirror to the mirror avatar, his/her full-body motions have to be tracked by a motion capturing system. These systems usually consist of multiple hard- and software components.
While there are also other motion capturing systems, the most popular ones use cameras and optical markers to generate animation data (see Section 2.5.2). The images of these cameras are streamed to a processing software, which then in turn generates animation data from these images. The computation of this animation data as well as recording and streaming of the images takes time. As a consequence, latency occurs already at this stage in the pipeline. However, this latency is majorly dependent on the used hardware (cameras, computer components and their connection) and software (algorithm to compute the joint angles). As motion capturing systems are in general commercial proprietary systems, the easiest and probably only convenient way to minimize latency of this component, is to choose the motion capturing system with the lowest latency (see Section 2.5.2).

Data transport Whenever a new set of animation data is generated by the motion capturing system, the resulting data has to be transported to the visualization system. In the simplest and actually most efficient case this boils down to store it somewhere in main memory where the visualization system can directly read the data. However, in practice visualization and motion capturing run on different independent computers. Therefore, the data is usually sent via network connection, which again adds latency.
Anyway, latency in this setup can be minimized by choosing the network protocol and hardware with the lowest possible latency. As data is normally transported in a local area network over short distances only, this usually results in a rather small latency offset (about 1 ms).

Visualization In order to visualize a new set of data, the data is received by the network interface of the render engine and stored in main memory. The receiving of data is best done asynchronous, to minimize blocking of the main render thread. Then, to update objects, data is processed so that it fits the local data format, e.g., incoming joint angles are converted to matrices, which are then in turn used to update the forward kinematic of a character's skeleton. This has to be done in the main render thread, to avoid thread concurrency issues. Otherwise a data block may be read during rendering, while it is still written from the network interface's thread. However, while it is useful to optimize this process, it may not be the main bottleneck causing latency in the visualization.
The rendering itself actually constitutes the more pressing issue. The more often and faster the render thread picks up new data and computes a new frame, the sooner it is visualized on the display. As a consequence, faster rendering results in lower latency. However,

the rendering performance itself is dependent on various factors, i.e., scene complexity, rendering techniques, shading quality, and hardware components.
Scene complexity is dependent on the task and on the desired richness of the visualized scene. For the virtual mirror, it may be kept quite simple. An empty room is actually often intentionally employed in this scenario to minimize distraction. For other applications, e.g., visualization of complex data sets for plant design, the scene might be much more complex and thus can cause massive performance drops. Therefore, while scene complexity influences the frame rate and thus latency, it is not the main bottleneck in the virtual mirror pipeline.
Performance issues due to scene complexity can be compensated to some extend by advanced rendering techniques, like view frustum culling, geometry instancing, deferred shading, or similar. Even when not as critical as for other applications, also the rendering of the virtual mirror will profit from advanced rendering techniques like these. For instance, view frustum culling prevents unnecessary draws of the whole scene for the mirror in a CAVE as it may not be visible on all screens, e.g., the floor projection (see Figure 3.1).
Employing higher quality shading creates more realistic visualization, but is more expensive to compute and thus causes a drop in performance. However, some effects, like (soft) shadows, smooth skinning as well as texturing, are necessary to generate a convincing rendering of the virtual mirror. Further, when implemented efficiently, none of them will cause extensive performance issues while at the same time offering significantly higher visual quality (see Section 2.6).
Moreover, rendering performance and thus the capability to increase shading quality is determined by the employed hardware setup. This is largely dependent on the installed graphics cards.
In general, the employed hardware components have a severe impact on latency:

- More powerful GPUs obviously will allow for faster rendering and thus lower latency.
- Using a single computer with multiple GPUs, instead of a render cluster, will significantly reduce latency (see Section 2.5.1).

Also other hardware components like CPU or main memory may influence rendering performance, but they usually do not constitute major bottlenecks in the overall pipeline.
Finally, when a new frame based on new data is computed and ready to get visualized on the display device, the back buffer of the GPU is swapped. This basically means the back buffer, where the rendered image was written into, is declared as the front buffer. The content of the front buffer is subsequently read out and visualized by the display device. This, of course, is only the case when Vertical Synchronization (VSync) is disabled. Otherwise

the buffer swap is delayed until the next refresh of the display, which delays the visualization of a new frame by a full refresh cycle. Still, even with VSync disabled, displays have discrete refresh rates and thus new frames are only visualized whenever the displays refreshes. Therefore, latency is additionally dependent on the refresh rate. The rule of thumb is: the higher the refresh rate, the lower the latency (provided the render frame rate is as high). Further, the display needs some time to actually receive the current frame, process it and build up the actual image. This additionally adds a small amount of latency. The only efficient way, to minimize latency of the display, is to choose display hardware accordingly.

2.4 Related Approaches

This section gives a short overview of state-of-the-art approaches to motor learning systems in VR with respect to requirements developed above. In the following, these are referenced as R1–R4.

Smeddinck et al. [SVHM14] present a training system that covers a large range of human movements. The system aims at improving motor performance for Parkinson's disease patients. Participants can monitor their own motion visualized through a coarsely rendered skeleton. Furthermore, the movement of the instructor can also be monitored, depending on the experimental condition. The authors evaluate the effect of different abstractions of instruction presentations on motor performance. A Microsoft Kinect camera was used for motion capturing. The authors fulfill R1 (feedback on one's own motion), but did not provide any information on system latency (R2). However, given the latency of the Kinect sensor, it can be expected to be well above 100 ms. Requirement R3 can be considered fulfilled as no hardware has to be attached to the user for Kinect-based motion tracking. The overall tracking robustness can be assumed to be sufficient for the task of rehabilitation for Parkinson's disease patients and the employed set of simple movements. However, using a Kinect camera might not be fast and robust enough for more complex motions (R4).

A yoga training game with a special focus on visually impaired people is presented by Rector et al. [RBK13]. They focus on spoken feedback to help trainees to reach a desired yoga posture. To get information about the performed movement, they also employ a Kinect camera. As the system targets visually impaired people, requirement R1, which demands for feedback on one's own motion can be seen as fulfilled via the provided spoken feedback. Indeed, the authors do not give any information on the system's latency (R2), which might be important to counter-steer over- and under-shooting movements caused by a high latency. For example, the system could state "Lean forward" based on a delayed measurement, although the user already exceeded the desired angle. Yet, yoga movements are typically rather slow, such that a high latency might only slightly influence the given

task. Requirement R3, which requires a minimal level of disturbance, is fulfilled due to the marker-less Kinect tracking. Concerning the robustness of the tracking for the desired type of motion (R4), no information is given. It can be assumed that the authors chose postures that are easy to track with the Kinect camera and do not require too many changes in user orientation or self-occlusions of body-parts.

A highly specialized VR training system for rowing is presented by [SRMC+14]. The user is placed in a modified boat, surrounded by projection walls. An extended version of the rowing blade is visualized and superimposed by the optimal blade position. Furthermore, the authors employ auditory feedback, which consists of a sonified oar blade and a sound which is played when the blade enters the virtual water. Haptic feedback is applied via resistance torques against the user's movement as soon as the user's blade moves away from the target position. Virtual self-monitoring (R1) is only possible via observing the virtual oar blade. Concerning latency and frame rate (R2), no information on the overall latency is given. Only the update rate of the projectors (> 30 Hz), movement sonification (30 Hz), and the frequency of the haptic device (1000 Hz) is described. The Unity engine is used to render the virtual ocean and the motion of the oar blade. Requirement R3 is satisfied as no additional hardware except from headphones has to be attached to the user and he/she is located inside a real boat. The tracking can be assumed to be sufficiently robust (R4), since the tracking task is not very complex.

Covaci et al. [COM14] present a training system that aims at high-precision tasks such as the basketball free throw. The system is located in a CAVE environment, hence the ball has to be attached to a special construction to prevent the walls from damage. The ball and the user are tracked by a Vicon MX motion capture system. Directly after throwing the ball, the system calculates the trajectory of the ball and visualizes the throw. The users can monitor their own motion (R1) either in first- or in third-person perspective. The third-person perspective can also be overlaid with the correct trajectory of the ball. The system's shutter glasses run at 30 Hz per eye, the motion capture system has a frequency of 120 Hz. Information on the system's latency is not stated (R2). In a user study, the authors showed that the overall latency did not disturb the users. Requirement R3 (minimal disturbance) was evaluated via questionnaires: The interaction was stated as natural by participants, such that R3 can be considered fulfilled. The tracking is described as being robust (R4) and the calculation of the ball trajectory leads to correct results in 87.5 % of 500 trials.

To summarize, many different approaches towards VR motor learning exist. However, information on end-to-end latency is only rarely given. Hence results are difficult to compare, e.g., concerning the achieved levels of performance, and it is difficult to exactly replicate experiments. Furthermore, some systems use sensors unable to provide a robust tracking for a broad set of possible motor actions. To the best of our knowledge, no approach

published until now aims at providing a general, highly controlled, efficient training environment that satisfies the above mentioned requirements and provides information on end-to-end latency. This work in this chapter tries to fill this gap via description, discussion, and evaluation of state-of-the-art techniques, leading to an exemplary realization of a system that satisfies the stated requirements. Furthermore, we provide information on the system's end-to-end latency, which enables replication, comparison, and assessment of future experiments, which have been or are to be performed in this particular VR system.

2.5 Realization of Low-Latency Environment

This section describes our hardware setup, provides an assessment of state-of-the-art techniques for building a low-latency VR environment for motor learning fulfilling the stated requirements, and finally presents our design choices and developments for this particular task. Figure 2.1 depicts the architecture of our system. It consists of three major parts: (i) display technology, (ii) render engine and (iii) motion capturing system / motion preprocessing.

To display our virtual world, we decided to use a CAVE environment. This ensures a *minimal level of disturbance* (R3), since the equipment attached to the user is limited to a pair of tracked 3D glasses. These glasses are usually much lighter and smaller than a full-sized HMD and there are no cables attached to the user. Moreover, the user is still able to see her own physical body and thus gets *feedback on her own motion* (R1) without any additional equipment. The still slightly narrow field of view of available HMDs impedes self monitoring by looking at one's own (virtually rendered) body: The user has to make larger head movements, especially when looking down along one's own body, which then may interfere with the training goals. In particular training situations, in which head and neck orientation and/or movements are essential, the additional weight imposed by the HMD also influences the trainee's posture compared to the optimal natural posture.

Our two-sided CAVE (L-Shape, 3 m $\times$ 2.3 m for each side) has a resolution of 2100 $\times$ 1600 pixels per side. Each side is driven by two projectors with INFITEC filters to enable passive stereoscopic vision by utilizing wavelength division. Both walls (floor and front) use back-projections. The four projectors are driven by a single computer (Intel Xeon CPU E5-2609 @2.4 GHz, 16 GB Ram, 2 Nvidia Quadro K5000 GPUs).

Our virtual world consists of the following components: a virtual fitness room with a virtual mirror mounted on the front wall. The user is placed in front of this mirror and his/her motions are mapped onto a mirror avatar visible in the mirror. This effectively generates a virtual reflection of the user's motions, which further enhances the fulfillment of the requirement for *feedback on one's own motion* (R1). The user's motions are captured by an

optical motion tracking system mounted at the top and the sides of the CAVE. Motion data is streamed into the Motion Preprocessor, which prepares the data for its use in the render engine and additional software packages for further analysis and feedback generation. The render engine then visualizes the scene while adapting the camera perspective(s) according to the user's head position/orientation and animates the virtual character in the mirror using the full-body tracking data.

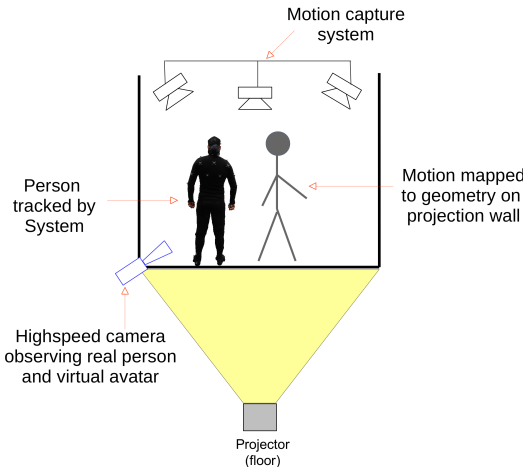


Figure 2.3: Overview on latency measurement: The person inside the CAVE is equipped with a motion capture suit. Motion is directly mapped on the virtual character. A high-speed camera records both: real person and virtual character. Later on, the number of milliseconds between the real person reaching a turning point and the virtual character reaching the turning point is determined.

In order to evaluate and compare the overall end-to-end latency of different rendering and tracking approaches, we adopted and extended a well-established latency measurement approach [LSG91, Ste08, FS14]: Typically, a pendulum is placed inside the tracking area, and the tracking data is visualized on a display behind the pendulum. A high speed camera records both the swinging real pendulum and the virtual pendulum on the screen. Afterwards, the recording is analyzed by hand, and the time-offset between the real and virtual pendulum is the end-to-end latency of the overall system. The following individual system latency values add up to the total latency: tracking latency, network latency, rendering latency, and display latency. The simple and periodic movement of a pendulum allows the application of automatic evaluation techniques [FS14]. However, in our case, we are not only interested in the end-to-end latency of a single marker tracked by the system, but in the latency induced by (more complex) full-body motion capture. Hence we replace the pendulum by a human standing in the center of the CAVE, who is fully tracked and instructed to move one arm up and down. The tracked motions are mapped onto the virtual character, which is rendered on the front screen (see Figure 2.3). The scene is again recorded by a high-speed camera (Ximea, 170 Hz), and the video is analyzed by hand (see the accompanying video). To reduce errors due to manual labelling, we average latency results over 30 trials.

In the following we first discuss our rendering solution, before presenting the full-body motion capturing approach.

2.5.1 Real-Time Rendering

In typical VR scenarios, the scene is rendered once per eye to provide stereoscopic vision. This means, even in the simplest VR-like scenario, e.g., single stereoscopic display/power-wall or HMD, the virtual world is rendered at least twice at high resolution. For the case of a CAVE things actually get worse, since it consists of multiple screens, which all have to be rendered independently and need stereoscopic visualization. In the end this means that for our two-sided CAVE, the virtual world has to be rendered at least four times, once for each eye and screen.
However, to keep up with requirement R1 (*feedback on one's own motion*), we visualize the movements of the user through a virtual character in a virtual mirror. Since the mirror reflects the virtual world as well, the world actually has to be rendered eight times in the worst case: once per eye and screen as well as per mirror in the world's scene. More generally, this adds one render pass per additional mirror in the scene for each eye. Fortunately, the default scenario used for this work includes one mirror only.
Obviously, when it comes to visualizing the virtual mirror, the mirror avatar is drawn multiple times as well. However, as the virtual mirror constitutes a first person perspective, the character is not drawn outside of the virtual mirror in the CAVE scenario, since it would be superimposed by the user of the virtual mirror. Thus, the virtual character would be occluded by the real body of the user anyway. Therefore, for our CAVE with two screens, the character is drawn four times in the worst case. Still, this imposes an additional render load. It actually often constitutes the most expensive part to render, since the characters, as used for the virtual mirror in this work, usually feature higher quality in terms of geometry compared to the rest of the scene.
Additionally, to achieve more realistic scene lighting, dynamic shadows are computed in the scene. To this end, shadow mapping is usually applied in the render pipeline. This, however, additionally increases the render load. Fortunately, only *shadow casters* have to be rendered into the shadow maps. Further, the shadow maps are independent from the current view. Therefore, they have to be rendered only once for each light source in each render pass. Unfortunately, the expensive-to-draw mirror characters are such shadow casters and thus they must be rendered once more for each shadow map. Moreover, as shadow mapping is computed on the GPU, each shadow map is computed individually on each GPU.
Still, requirement R2 (*low latency and high frame rate*) has to be satisfied. This is mainly dependent on the chosen hard- and software solution for real-time multi-view rendering. In the following we first asses and evaluate several rendering techniques and check how well they fit our requirements, before we present the rendering approach chosen for our low-latency VR system.

Evaluation of Existing Approaches

The rendering of the world including the mirror avatar, the mirror itself, and the rest of the scene on multiple views constitutes a high render load which is usually offloaded onto multiple GPUs, since even current GPUs would not be capable to manage this load all alone — apart from lacking sufficient hardware connections to plug in all displays/projectors. Hence, the usual approach for multi-view rendering in the CAVE scenario is to use a render cluster.
A typical render cluster consists of multiple render clients, each of which is equipped with at least one GPU which then drives one stereoscopic view. The alternative is to employ multiple GPUs in a single PC (multi-pipe rendering) where each GPU drives at least one of the CAVE walls.
In order to test which one of the two approaches is superior in terms of latency, we implemented both approaches and compared them. To this end, we set up two test cases:

1. Distributed rendering on a minimalist render cluster of two nodes with one GPU each, where each machine is responsible for driving one projection wall.

2. Multi-pipe rendering on a single computer, which is equipped with two GPUs for driving the two screens.

The cluster nodes and the single computer had identical system specifications (2 Intel Xeon 2.4 GHz, 16 GB RAM), with either one (cluster) or two (multi-pipe) Nvidia Quadro K5000 GPUs. The cluster nodes were connected by a fast 10 Gigabit Ethernet network.
We analyzed how much latency the network communication of even our minimalist render cluster introduces by employing the full-body latency measurement described in the previous section. To this end, we use an OptiTrack Prime13W system (240 Hz) for motion tracking (see Section 2.5.2) and map the motion onto a minimalist stick figure. Rendering is done using InstantReality[2], which is based on the distributed rendering framework OpenSG[3].
The scene was rendered at about 260 fps for the cluster setup and 280 fps for the multi-pipe setup. Over a range of 30 full-body latency measurements we determined a mean latency of 50 ms for the render cluster and 41 ms for the multi-pipe setup, with standard deviations of 12 ms and 10 ms, respectively. These results demonstrate that even a minimal cluster consisting of only two render nodes can already lead to an increased latency. The render cluster is inferior in terms of latency, since the final visualization is synchronized via network connection to provide a homogeneous image. Due to this network synchronization, the visualization of the final image takes some additional time in each frame. This

[2] http://www.instantreality.org
[3] http://www.opensg.org

suggests that in terms of latency a multi-pipe framework on a single machine should be preferred over a render cluster, at least when the number of projection screens/views allows for a multi-pipe approach (easily for up to four walls). Additional reasons for the single-machine solution are easy maintenance, easier implementation, and less expensive hardware setup.

In terms of software frameworks, stereoscopic multi-view rendering is a mature and well-established topic, and consequently numerous software solutions are available for this task. There exist a range of specialized VR render engines, e.g. Instant Reality, which are optimized for rendering in VR applications like the CAVE and which we employed for the cluster-vs-multi-pipe benchmark. Such a render engine seems to be the canonical first choice. However, being targeted at fast and easy prototyping of VR applications, the primary goal of InstantReality is not high-performance rendering. While it supports character animation, it does not exploit GPU-acceleration for the involved skinning computations. As a consequence, when tested on our high-quality mirror character consisting of 135k triangles, performance dropped down to about 5 fps even for a single window. Since fast character animation is crucial to ensure *low latency and high frame rate* (R2), InstantReality could not be employed. Of course, there exist more VR render engines like Instant Reality, which are maybe or maybe not more sophisticated, but these engines usually rely on cluster-based rendering, which have the disadvantage of having a higher end-to-end latency as already stated above.

Fast character animation and high-quality rendering are (besides many other features) provided by game engines. Therefore, another solution to drive CAVE-like environments is to employ a game engine, like the Unreal Engine[4] or Unity[5].

The issue with these engines is that they are usually optimized to render to only one single screen. Thus, to drive a CAVE with multiple views an augmentation of the software has to be made or an extra layer of software has to be implemented to distribute and synchronize rendering of multiple instances of the game engine.

The by now open source Unreal Engine may be a candidate for an augmentation. However, since such render engines usually consist of thousands of lines of code, it is a rather complex and tedious endeavor. Further, other game engines, e.g., Unity, are actually closed source, which makes it even impossible to augment these engines in a proper way.

In contrast, such as the above mentioned extra layers of software already exist. An example for such a piece of software is MiddleVR[6], which is a commercially available middleware for the Unity game engine. Apart from being quite expensive (about $ 21k for our

[4] `https://www.unrealengine.com`
[5] `http://www.unity3d.com`
[6] `http://www.middlevr.com`

setup), MiddleVR is actually designed as a cluster-based solution and therefore in fact synchronizes multiple instances of the Unity game engine. Still, even though being designed mainly as a cluster solution, MiddleVR also allows to set up a local cluster on a single computer, thereby effectively providing a multi-pipe rendering solution. This configuration will add less latency compared to a multi-machine rendering cluster. Nevertheless, the communication between single instances, even when running on the same machine, constitutes an additional overhead, which will result in higher latency compared to a single instance application, where data is kept in the same address space and is not distributed to multiple instances. CaveUDK [LCC+12] is another example of an extra software layer. It is a middle-ware for using the Unreal Engine 3 in a CAVE and is based on a distributed rendering approach. However, they report rather high end-to-end latency of 82 ms for pressing a button to trigger an event and 136 ms for user navigation, which rules out this rendering solution.

Further, another issue for any closed-source game engine in general might be that it is difficult to control the data flow inside the application to optimize for latency. This is due to the fact that they usually provide some plugin interface with predefined functions to feed data into the visualization, which are only called at discrete points in time.

While several other high-level and low-level software packages exist, such as OpenSceneGraph[7] or Equalizer [EMP09], respectively, we eventually decided to develop our own slim rendering framework, since this gives us full control to exploit low-level hardware acceleration and parallelization. The architecture and features of our rendering engine are presented in the next section.

Realization

In order to minimize latency, we developed a single-computer multi-pipe solution for rendering the scene in the CAVE. This render engine is implemented in C++, features modern OpenGL 4 and offers Qt-GUI[8] elements for configuration. Further, standard techniques are employed for visualization. Lighting is performed by the efficient Phong lighting model and soft shadow mapping is applied for real-time shadows.

In order to support multiple GPUs and views, the render engine queries for available GPUs and displays at start-up. The actual window configuration is then determined by a user-provided configuration file. The information provided in this configuration includes window positions and resolutions, coordinates of the actual physical screen and other additional settings, like shading effects and stereoscopic view modes. The desired windows are dynamically mapped to available display space and are created at the determined positions.

[7] `http://www.openscenegraph.org`
[8] `https://www.qt.io/`

Various window and screen configurations with or without stereoscopic visualization are possible. For instance, extending rendering to an arbitrary number of CAVE walls as well as to other multi-display setups is possible. The only restrictions are hardware limitations, e.g., the number of GPUs which fit into one PC and display connections provided by these GPUs. Nevertheless, CAVE-setups with up to four walls are easily feasible. Moreover, rendering in HMDs such as the HTC Vive is seamlessly possible as well.

For large display configurations, e.g., CAVEs or power walls, where the user moves closely to the generated image, it is crucial to adapt the perspective to the current head position of the user. To this end, the head or, to be more precise, the user's eye positions are tracked. This is commonly achieved by tracking the user's stereoscopic glasses that he/she has to wear anyway. From the position of the eyes and the coordinates of the display corners provided in the configuration of the render engine a projection frustum is computed in every frame for each eye so that the perspective is always correct from the current user's position. Similar, a HMD needs head tracking as well. But in contrast to the large display scenario, the perspective is fixed and instead the whole view is translated and rotated as the display inside the HMD actually moves with the user's head. This way the current view of the HMD is always conform with the user's position and viewing direction. Consequently, the render engine supports head tracking for both setups.

High performance rendering requires to offload all expensive computations to the available GPUs. This is even more important for a multi-pipe approach, because only this way it is ensured that the rendering performance scales properly with the number of GPUs. In addition, data transfer costs have to be reduced to a minimum, which is especially important for the implementation of character animation.

The characters used for the mirror are based on 3D meshes and feature additionally an underlying skeleton to support full-body animation. While many possibilities to animate virtual characters exist, many of them, like, e.g., sophisticated simulations of soft-tissue or clothes, are currently too computationally expensive to calculate them during real-time rendering for multiple views. Thus, even if a more sophisticated technique would be preferable quality-wise, performance due to the anyway high rendering load caused by feeding multiple views, while maintaining low latency and high frame rate (R2), is the more pressing issue here. Therefore, more simple geometry-based animation techniques are employed in this context. The de facto standard in computer animation is skinning. The typically applied skinning techniques are linear blend skinning (LBS) or dual quaternion skinning (DQS) [JDKL14].

We visually compared both approaches and noticed that the typical bulging artifacts of DQS are actually more disturbing for our use case than the candy wrapper and joint collapse artifacts of LBS. Therefore, we decided to animate the virtual characters using the

Figure 2.4: Real-time feedback using a virtual mirror requires an immersive Virtual Reality environment that provides full-body motion capturing, motion analysis, and realistic character rendering at a low end-to-end latency.

very efficient and simple LBS, where vertices $\mathbf{x}_i$ are transformed using a weighted linear blending of the joints' transformation matrices $\mathbf{T}_j$:

$$\mathbf{x}'_i = \left(\sum_{j=0}^{n} w_{i,j} \mathbf{T}_j \right) \mathbf{x}_i, \tag{2.1}$$

where the weights $w_{i,j}$ determine the influence of joint/bone j onto vertex i. We transform vertex normals $\mathbf{n}_i$ using the same equation except that only the linear part of the blended matrix is applied to $\mathbf{n}_i$. Note that this way of transforming the vertex normals is actually not accurate as explained in [TPSH14]. However, this solution is easier to implement and faster, while still providing convincing results.

Since this computation can be performed independently for each vertex, it can easily be mapped to the GPU. In this case, the rest-pose vertices $\mathbf{x}_i$ and normals $\mathbf{n}_i$, as well as the skinning weights $w_{i,j}$ remain constant in GPU memory, such that only the (rather small number of) joint transformations $\mathbf{T}_j$ have to be uploaded to GPU memory in each frame. In contrast, a CPU implementation (as done, e.g., in InstantReality) uses fewer computation cores and has to upload all new per-vertex data ($\mathbf{x}'_i$, $\mathbf{n}'_i$) to GPU memory. A performance comparison on our high-quality character (135k triangles) showed the GPU implementation to be about 50 times faster (2980 fps vs. 57 fps, single window, Intel Xeon E5-1620 3.6 GHz, Nvidia GeForce GTX 980).

The virtual mirror is implemented by first rendering the scene, including the animated character, from the correct mirrored perspective derived from the user's eye position. The content of the resulting framebuffer is then mapped as a texture onto the mirror geometry in the scene (see Figure 2.4, Figure 2.7 and Figure 2.9). To animate the character, we stream

the pre-processed motion data from the Motion Preprocessor to the render engine using a network interface (compare Figure 2.1 and Figure 2.5).

As the render engine has to handle rendering on an arbitrary number of GPUs, special data management is necessary to handle new incoming data from the network interface. First of all, each GPU needs its own OpenGL context, which is shared between all windows created on the displays connected to the same GPU. These contexts constitute interfaces to each GPU. Each context manages its own data such as buffers, arrays and textures. Further, it has its own current state as OpenGL acts as a state machine. As a consequence, data has to be distributed to all GPUs and their contexts separately. In order to handle this situation properly and to avoid redundant storage or transfer of data, the data is basically split into two types of data: data in main memory and data in GPU memory. This is implemented in terms of *scene objects*, which represent data in main memory, and *scene graph nodes*, which represent corresponding data in GPU memory. As, in fact, these nodes are organized in a scene graph, each GPU has its own scene graph. Consequently, for each new scene object a corresponding scene graph node is created per GPU and added to the GPU's scene graph. Each scene graph node in turn carries a pointer to its corresponding scene object in order to have access to the data in main memory (see Figure 2.5).

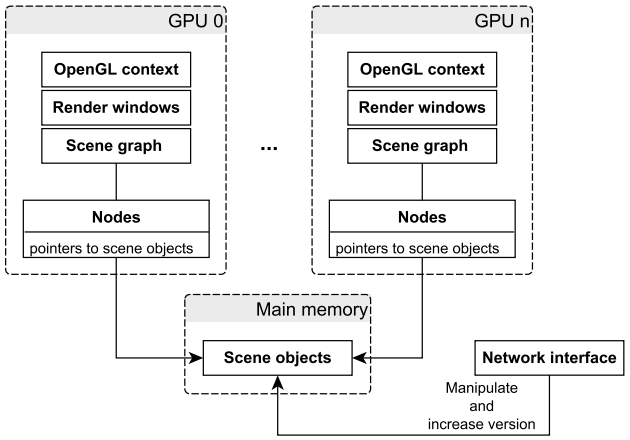


Figure 2.5: Data distribution in our render engine: Each GPU has its own OpenGL context and scene graph. The scene graph nodes represent an object in the GPU memory and carry a pointer to their corresponding object in main memory. The nodes are kept up-to-date using a versioning approach.

Whenever data of a scene object is updated, e.g., when new animation data was received via the network interface, all corresponding scene graph nodes have to be updated as well. To this end a simple versioning approach is implemented, which works as follows:
Whenever data of a scene object is modified, a corresponding version counter is incremented. Before the subsequent render pass, data in GPU memory is updated if the corresponding version counter in the scene graph node differs from the scene object's one. Therefore, the object is visualized with the modified data. This is how new incoming animation data is processed, to animate the character in the virtual mirror. This simple yet

powerful mechanism is more efficient than communicating with messages between scene objects and scene graph nodes, because data is directly read from main memory and thus no unnecessary copying and transfer of data is performed. However, this mechanism will not allow to seamlessly utilize the render engine as a cluster solution since data has to be in the same address space.
To parallelize rendering on multiple GPUs, each GPU has to render independently from the other GPUs. But, due to the fact that the consumer graphics cards manufactured by NVIDIA send OpenGL commands to *all* installed cards simultaneously, rendering is not independent. NVIDIA consumer cards are thus not applicable for multi-pipe rendering. Note that AMD GPUs do not suffer from this issue since OpenGL commands are only sent to the GPU which is connected to the display on which the window and corresponding OpenGL context were created [DSPB12]. Therefore, AMD consumer GPUs may be one possible solution, but they lack proper synchronization of the connected displays and visualized frames, which is especially important when it comes to stereoscopic visualization. In contrast to that, professional GPUs, e.g., from the NVIDIA Quadro series, offer OpenGL extensions to specifically send OpenGL commands to selected GPUs only. Consequently, the render engine applies the `WGL_NV_GPU_Affinity` extension on Microsoft Windows based computers in order to specifically send OpenGL commands to the right GPU during rendering to effectively provide parallel rendering on multiple GPUs. Furthermore, these professional GPUs also feature the more efficient quad buffer stereoscopic rendering, i.e., they are equipped with two additional hardware buffers, which allow rendering of both stereo images into these buffers followed by a simultaneous buffer swap of both images. This guarantees that both images are visualized at once and no incoherence between left and right eye images occur. Accordingly, quad buffer rendering is also seamlessly supported by the render engine.
In summary, the render engine is a lightweight application, which provides fast and optimized rendering of characters, the virtual mirror and medium-sized scenes in stereoscopic multi- and single-display environments such as CAVEs, powerwalls or HMDs. Thus, the resulting render engine provides all necessary features for our VR motor learning environment, while maintaining a slim software design and flexibility. Moreover, by offering a network interface to manipulate scene objects in various ways, it is actually suitable to be used in other contexts than the virtual mirror and motor learning scenario.

2.5.2 Motion Capture

Full-body motion capture is necessary to provide real-time augmented feedback on motor performance. In the following, we give an overview of state-of-the-art motion tracking approaches and assess them with respect to the aforementioned requirements.

Evaluation of Existing Approaches

To track full-body motion, the distinction between *outside-in* and *inside-out* approaches is important. For outside-in approaches, markers are attached to the human body and the actual capturing devices are placed at fixed positions outside the tracking area. The inside-out approach works the other way around, for instance, by attaching inertial trackers to the user. Although the outside-in approach has to deal with occluded markers, it has the important advantage that no sensitive and/or heavy devices have to be attached to the user. Furthermore, outside-in approaches do not suffer from drift due to time-integration of sensor data, and they provide the exact location of the user. Since we want to avoid attaching disturbing hardware on the user (R3) we only take outside-in approaches into account in the following.

For these systems, the next distinction is between *marker-based* and *marker-less* approaches. Many commercially available systems exist for both approaches, such as the marker-based systems Vicon and OptiTrack, or the marker-less systems Microsoft Kinect, The Captury and Organic Motion. The advantage of marker-less systems seems obvious: No hardware has to be attached to the users, thereby reducing setup time significantly and minimizing user disturbance. However, marker-less systems often need more controlled lighting conditions, a good view of the user's whole body, and sometimes even a controlled background. The tight space, bad lighting conditions, and frequently changing background in a CAVE thus impose a huge challenge for such marker-less systems, which interferes with the requirement R4 for robust tracking. Further, due to the computationally more complex tracking task, more latency is generated during the generation of new animation data compared to marker-based systems. Furthermore, due to the approximative nature of these approaches they often lack tracking precision, e.g., twist movements are not tracked faithfully. Nevertheless, there exists a rich body of literature introducing interesting techniques which perform marker-less real-time motion tracking [SHG+11, HMST13, MSS+17]. Still, these either do not meet the demanded tracking precision and latency requirements [SHG+11, MSS+17] and/or demand for a good view of the user and need additional inertial trackers (R3) [HMST13].

Therefore, we decided to focus on marker-based systems, since they provide lower latency, more robust and more accurate tracking results in a CAVE environment. Nevertheless, we also analyzed the marker-less Kinect sensor, since this device can also operate in rather dark environments and is used in many related approaches towards motor performance training (e.g., [RBK13, SVHM14]).

For the marker-based systems, one can use *active* or *passive* markers. Passive markers simply reflect the infrared light emitted by the tracking cameras. The tracking system captures a set of markers, which then have to be consistently labeled. As soon as markers

System	Price / Camera	Camera Res.	Max. FR	Used FR	Latency	Std. Dev.
Vicon T20	20,000 EUR	1600×1280	500 Hz	100 Hz	54.9 ms	13.18 ms
				240 Hz	44.7 ms	10.6 ms
				500 Hz	38 ms	8.4 ms
OptiTrack Prime 13W	2,500 EUR	1280×1024	240 Hz	100 Hz	59.7 ms	12.3 ms
				240 Hz	41 ms	9.9 ms
OptiTrack Flex 100	600 EUR	640×480	100 Hz	100 Hz	65.5 ms	21 ms
Microsoft Kinect 2	150 EUR	512×424	30 Hz	30 Hz	98.8 ms	19.17 ms

Table 2.1: Comparison of end-to-end latency values of the different motion capturing systems (averaged over 30 measurements), also listing price per camera, camera resolution, as well as the maximum and the employed frame rates (FR).

get lost and re-appear later on, the labeling step can produce errors. Active markers, as used in systems like PhaseSpace[9] avoid the labeling problem by emitting light at a unique frequency. The disadvantage of active markers is that they require more service and are more prone to get damaged during experiments. Furthermore, the marker suits are more difficult to clean. Additionally, active motion capture suits are often less comfortable to wear than suits for passive markers (R3).

Thus we decided to focus on outside-in tracking systems based on passive markers. We analyzed and compared the Vicon T20 system and the OptiTrack systems Flex 100 and Prime 13W. These systems require a motion capture suit with attached markers, or having the markers attached directly to the human skin. As motion capture suit any tightly fitting sports clothing can be used as long as it does not contain reflective materials. Thus these systems satisfy requirement R3. We evaluate the end-to-end latency and update rate (R2) of the different tracking systems using the latency measurement approach described in Section 2.5. In order to focus on the tracking latency, we only rendered a simple stick

[9] `http://www.phasespace.com`

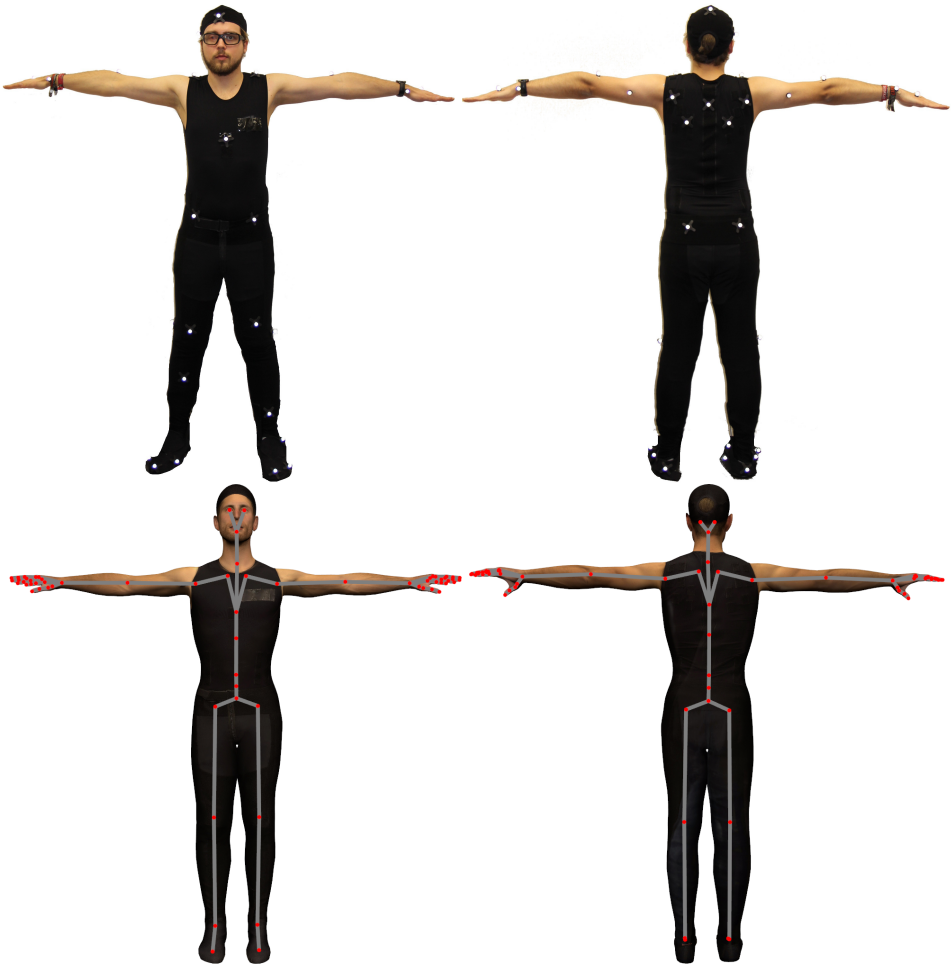


Figure 2.6: We use a marker-based motion capture suit with 41 markers (top) and extract 19 joint angles which we map onto our virtual characters' skeleton (bottom).

figure (at about 280 fps). Table 2.1 summarizes the resulting latency values for Vicon T20, OptiTrack Prime 13W, OptiTrack Flex 100, and Microsoft Kinect.

Concerning the robustness of the tracking (R4), the marker-based systems meet our demands: For most basic movements and exercises (e.g., squats, walking around, jumping), the user is tracked without the need for re-calibration or returning to the T-Pose during a session. In contrast, the tracking robustness for the Kinect camera was worse: Here, many kinds of exercises, e.g. squats, cannot be tracked reliably due to occluded body parts.

Realization

Based on the benchmark results shown in Table 2.1, we decided to use an OptiTrack Prime 13W system with 10 cameras. This marker-based solution is a good compromise as long as there is no marker-less option of similar performance and robustness. The Microsoft

Kinect was excluded due to its high latency (R2) and problems in dealing with occluded body parts (R4). The Vicon cameras' advantage in terms of temporal and spatial resolution did not justify the much higher price for our field of application. We decided to use the Prime 13W system instead of the Flex 100 cameras because of the wider field of view (82° vs. 58°) and the higher temporal and spatial resolution.

The cameras are arranged in a way that allows an almost failure-free tracking. Users are equipped with a marker suit with 41 markers for accurate skeleton tracking (Figure 2.6). This layout extends Optitrack's basic 37-marker layout by two additional markers on each foot. These increase the robustness of foot tracking and allow us to get information on metatarsal rotation. With the final setup, we are able to obtain animation data for 19 joints. The system provides joint positions and rotations, which are used to animate the virtual mirror character, for feature extraction (e.g., calculation of movement direction, speed, acceleration), and for motion analysis.

2.5.3 System Summary

Our VR environment for motor learning consists of a two-sided CAVE with front and floor projection (L-shape, 3 m × 2.3 m and 2100 × 1600 pixels for each side). Each wall is operated by two projectors (Projection Design F35 WQ), which run at 60 Hz. To separate the images for both eyes, we use passive filters by INFITEC. All four projectors are driven by our self-developed render engine running on a single computer to minimize latency. The render engine runs on a Microsoft Windows 7 desktop PC with two Nvidia Quadro K5000 GPUs and an Intel Xeon E5-2609 CPU with 4 × 2.4 GHz and 32 GB RAM. For each CAVE wall, we use one NVIDIA Quadro K5000 graphics card.

For motion capturing we use a marker-based Optitrack Prime13W motion capturing system consisting of 10 cameras. This system is used to obtain position and orientation of the 3D glasses for perspective adaption as well as the joint orientation of 19 joints to animate the virtual mirror avatar. The motion capturing software runs on a dedicated PC with Microsoft Windows 7 operating system and is equipped with a 4 × 2.4 GHz Intel Xeon E5-2609 CPU and 16 GB of RAM.

2.6 Benchmark

To evaluate the influence of rendering options on latency, we evaluated different quality levels to find the best trade-off between quality and performance. The same measurement procedure as described in Section 2.5 was used for 30 trials, and we report mean latency

Render Quality	Fps	Latency	Std. Dev.
Stick figure	690	36 ms	9 ms
Low resolution	114	54 ms	9 ms
Low resolution + Shadows	88	60 ms	10 ms
High resolution	86	62 ms	12 ms
High resolution + Shadows	62	81 ms	14 ms

Table 2.2: Latency and performance values for different rendering qualities (mean value of 30 trials and standard deviation). Rendering a minimalist stick figure without a virtual environment, or rendering the full gym scene and the virtual mirror, but using either a low-resolution (20k triangles) or high-resolution (135k triangles) virtual character, with optional shadow mapping.

values and standard deviations. The virtual scene used for the tests consists of a virtual fitness studio (about 100 k triangles) including the virtual mirror (see Figure 2.7).

The results are listed in Table 2.2. For our high-resolution character (135 k triangles), we observed a latency of 81 ms at 62 fps when using shadow mapping. Without shadows, a latency of 62 ms at 86 fps was measured. Using the low-resolution character (20 k triangles) reduces the latency to 60 ms at 88 fps (with shadows) and 54 ms at 114 fps (without shadows). As a baseline test, we also rendered a simple stick figure without the surrounding fitness studio, which resulted in a latency of 36 ms at 690 fps. We conducted an additional test which consists of a single marker attached to a pendulum instead of a tracked human. The pendulum was visualized as a box inside the CAVE. Here, we observed a latency of 32 ms (SD=9 ms).

Figure 2.7: The scene used for the quality versus latency tests.

Our end-to-end latency consists of the individual latency of the cameras (approx. 4 ms according to manufacturer), of the tracking software, the motion preprocessing (approx. 2 ms), the network communication (approx. 1 ms), as well as rendering, synchronization, and display hardware (approx. 19 ms according to manufacturer). Figure 2.8 shows exemplary frames of the recording filmed by the high-speed camera, showing the experiment using the high resolution character and real-time shadows.

Figure 2.8: Example frames from one of the latency test videos (highest quality character with shadows). The left image shows the user's arm approaching the lowest point. The image in the middle shows the turning point of the real arm, the picture on the right shows the turning point of the virtual arm, while the real arm already moves upwards.

In a pilot study we examined whether users have the feeling of being able to control the virtual character, i.e., have sense of agency, and whether the system induces simulator sickness (e.g., due to latency effects). For this study we rendered the high-resolution character including shadow mapping. 23 participants (15 female; age M=26.17, SD=8.94) interacted for 5–6 minutes with our system: They had to perform squats while getting simple textual feedback afterwards. The degree of perceived control was measured using a 7-point Likert scale ranging from 0 (no control) to 6 (highest level of control). The results were quite satisfying (M=5, SD=1). Furthermore, no increase of simulator sickness, using the Simulator Sickness questionnaire by Kennedy et al. [KLBL93], was detected due to the experiment. For this experiment, we also evaluated the robustness of our system in terms of full-body motion capture: Only one single time, too many markers were occluded which required a re-calibration of the participant. In all other trials, temporary loss of markers was compensated by the system.
Figure 2.9 shows a photograph of a user reflected in the virtual mirror inside our environment. The accompanying video also shows an example of an interaction with our system.

2.7 Conclusion

We developed and motivated requirements for VR motor learning. We examined state-of-the-art techniques and technologies for motion capturing and rendering with respect to these requirements, and propose a low-latency environment based on the most promising components and approaches. In terms of rendering, a single-PC multi-pipe approach was

Figure 2.9: The visual quality achieved in the proposed system, including artificial shadows for the trainee.

shown to achieve a lower latency than even a minimal render cluster using two nodes. Our slim custom-designed render engine maps all expensive computations to the GPUs and parallelizes well. For full-body motion capture, we decided to use the marker-based outside-in OptiTrack system. The resulting system provides a virtual environment featuring a virtual mirror employing high quality characters. This systems serves as a solid base for further developments and experiments in VR motor learning.

Using the 20k-triangle character, our system meets the stated requirements: The user is able to monitor his own motion in the virtual mirror (R1). The overall latency of the system is at around 60 ms, which is comparable or better then related systems (R2). The graphics engine runs at 88 fps, feeding four channels with 2100×1600 pixels each, which is sufficient to perceive smooth images. Requirement R3 is also satisfied as users only have to wear passive stereo glasses and tight clothing with attached markers. Of course marker-less motion tracking would be the ideal solution, but to our experience the available solutions are not fast or robust enough in a CAVE environment. Requirement R4 can also be considered as satisfied, as shown in the accompanying video.

Our proposed system lacks portability, since the display technology as well as the motion tracking system are fixed installations. A portable system could be achieved by using

components like a commodity depth sensor (e.g., Kinect) or inertial trackers for motion tracking and a HMD for visualization. However, any configuration of that sort will have the problems mentioned in this chapter. Still, developing a portable system for motor learning that can be used at home or in a small clinic is an interesting challenge. It is then to be evaluated in future work how the more obtrusive display hardware (HMDs) influences users' performance of motor actions and their ability of motor learning.

Up Next

The system presented in this chapter represents the basic system, on which we built to conduct the experiments introduced in Chapter 3 and Chapter 5.
In contrast to the work presented in this chapter, the upcoming chapters will concentrate less on the field of motor learning in VR but more on the virtual mirror. Nevertheless, the next chapters are highly relevant for the field of motor learning, since the virtual mirror is one of the core elements to provide proper motor learning in VR. Therefore, a thorough investigation of factors influencing the properties of the virtual mirror in general is of high interest.
The next chapter deals with the impact of latency on the virtual mirror, since latency is always present whenever users interact with their virtual mirror image. Therefore, it is important to know whether and to what extend latency influences or even disturbs users performing a motor task in front of a virtual mirror.

3

Impact of Latency

3.1 Introduction

Even though numerous studies investigating the effects of feedback delays in virtual environments have previously been conducted, no clear picture has emerged which amount of latency is acceptable for full-body closed-loop interaction scenarios like the virtual mirror. While Held and Durlach [HD91] report that delays as small as 60 ms significantly interfere with motor adaptation, others reported a threshold of 120 ms or 150 ms to perceive latency (e.g., [MVHE04, JNS12]). Furthermore, the impact of latency is often only considered with respect to one single factor (e.g., perception of simultaneity [MAEH04, REG14]). In other cases, only very simple tasks such as button pressing are examined without a focus on VR [KRN13, RSE14]. Also, Rohde et al. [RvDE14] found that feedback delays are processed differently in predictable motor tasks as opposed to unpredictable motor tasks. The effect of feedback delays is thus contingent on the display, the movement performed, and on the perceptual or motor task (see Section 3.3 for a review of the background literature). Most studies to date have focused on just one task and one movement, which makes it difficult to compare results from different studies and to derive acceptable levels of delay in VR across applications.

We conducted a study in which participants performed movements involving the whole upper body in front of a virtual mirror showing their full body. Our study contributes to a better understanding of the effects of feedback latency on full-body closed-loop behavior such as the virtual mirror. It combines three important factors:

Plausibility: Our VR system provides an immersive, full-body virtual mirror image and allows for full-body motor learning of complex tasks. Participants perform a motor task that requires fast and precise movements of the whole upper body. Thus, the present study contributes to the development of VR environments dedicated to motor learning, such as sports training, and will hopefully be useful for the design of VR environments.

Temporal precision: Our CAVE is characterized by a low baseline end-to-end latency of 45 ms, which is particularly low for an environment using full-body motion capturing. This allows us to evaluate our dependent variables using a latency of 45 ms and higher. In the present study, we investigate a range of delays, from 45 to 350

ms, which allows us to model the perceptual responses as a continuous psychometric function of the feedback delay. Testing latency as small as 45–125 ms allows us to study the impact of even small feedback delays in VR systems on perception and behavior. Exploring high latency allows us to identify the limits of the temporal windows of perceived simultaneity, agency, and ownership in VR while performing complex movements. This gives us the opportunity to compare our results on feedback delays in immersive VR with previous related work on visual feedback delays in simpler tasks and environments, such as button presses and manual control interfaces (e.g., [FVH13, RSE14]).

Range of measures: Apart from motor performance error, we record participants' perceptual judgments (simultaneity and virtual embodiment effects: agency and body ownership) after performing movements in our experimental setup. We test for interaction effects between these variables and pre-existing immersive tendencies.

This paradigm enables us to identify thresholds for tolerable delays in different domains of user performance and experience, to compare these thresholds across measures and to identify interactions between measures. We hope that our study will be a useful resource for the design of virtual environments and provides information about how transmission delays impair different aspects of human perception and performance.

My Contribution *This experiment was prepared, conducted and evaluated in close cooperation with Irene Senna. I contributed by setting up the apparatus and implementing all necessary hard- and software for the experiment. Together with Irene Senna and Felix Hülsmann I discussed different experimental designs, of which I later implemented and evaluated the most promising ones to eventually find the final, most suitable design. Moreover, in cooperation with Felix Hülsmann I conducted the latency measurements to determine and verify the latency of the system.*
Corresponding publication: The Impact of Latency on Perceptual Judgments and Motor Performance in Closed-loop Interaction in Virtual Reality, VRST 2016.

3.2 Background

3.2.1 Virtual Embodiment

Virtual embodiment means that we are embodied by some kind of body (or object) in the virtual environment. This body can be an anthropomorphic body or actually something

more abstract like a box or a tool. For the experiment presented in this chapter the participants are embodied by a rather abstract mannequin-like virtual character which mirrors the participants' movements by motion tracking.

Whenever users are embodied like this a set of psychophysical effects is elicited. The most prominent ones are sense of body ownership and sense of agency. This section provides a short overview on how they are defined and what are the typical factors to induce them.

Sense of Body Ownership

Sense of body ownership is the feeling of having or owning a body. We usually feel body ownership towards our own body and body parts. This also comes natural whenever we stand (or move) in front of a real mirror. The reason for this is twofold. First of all, we have a rich mental representation of our outer selves, which actually allows us to recognize ourselves, for instance, on photos or in prerecorded videos. Therefore, we recognize our mirror image and we *know* that it is us. Additionally, whenever we move in front of the mirror, our mirror image reflects the same movement without any perceivable delay. This couples our sent motor commands with their immediate perceived visual outcome, which creates a feeling of sense of agency and thus strengthens the sense of ownership. Altogether, we have the feeling that we own the body in front of us in the mirror — that the body is ours.

Body ownership seems self-evident for our own body, but we can also feel it towards artificial limbs or body parts. A famous example is the rubber hand illusion experiment [BC98]. In this experiment the participant's hand is hidden from his/her sight and superimposed by an artificial rubber hand. Then the participant's hands and the rubber hand are stroked simultaneously. Due to this multisensory input of seeing the stroke on the rubber hand (visual) and feeling the tactile feedback of the stroke on the own skin (haptic) ownership towards the artificial hand is elicited. This *visuotactile* coherence tricks our brain to think that the artificial hand is our own. To test whether the induction of body ownership towards the artificial hand was successful, a threat is usually opposed to it, e.g., hitting it with a hammer and measuring reaction. This is just one example of how our brain can be tricked by multisensory input to feel ownership to foreign and artificial body parts.

Similar, we can feel body ownership towards a non-real and fully virtual body or body part: the so called illusion of virtual body ownership (IVBO, [IdKH06, SPMESV08, LLL15a]), which is exploited in the virtual mirror metaphor. Here, mainly the simultaneous replication of the performed complex movements, similar to a real mirror, provided by the virtual mirror (also known as *visuomotor* coherence) causes sense of body ownership. This is essentially coupled with the sense of agency as described in the next paragraph.

Sense of Agency

Sense of agency is the feeling of being responsible for a certain effect or action. In other terms, sense of agency is the feeling of controlling something, e.g., the movement of an object or a body part. Applied to the mirror this means that we feel responsible or in control of what the virtual mirror avatar does. Consequently, agency has a strong impact on the illusion of body ownership in this scenario as it basically represents the coupling of our sent motor commands and the expected and visualized outcome.
Therefore, agency is a crucial factor for the virtual mirror as there are actually no other direct cues that may elicit body ownership — at least for the virtual mirror without personalized avatars. Moreover, latency (the elapsed time from an action to a reaction) plays an important role for the sense of agency, since we may not feel in control anymore whenever the expected outcome is delayed too long.

3.3 Related Work

In the physical world, causes and effects of an action are temporally contiguous. The human brain automatically compensates for small and natural delays introduced by inconsistencies in neural propagation across signals in different modalities, and between movement outputs and afferent signals. In VR setups and in computer games, the latency of the system always introduces further delay between motor commands and visual feedback [SNL16]. While the brain tolerates small but perceivable delays, and even temporally recalibrates to them [RvDE14, YSSRA13], larger delays might affect perception and behavior. There is no golden rule specifying the highest acceptable latency a system is allowed to have in order not to impair motor and perceptual performances. Studies involving different setups and sensorimotor tasks come to different conclusions when investigating the perceptual threshold for delay detection. Users might be unaware of delays below 120 ms [MVHE04], or below 150 ms when controlling characters in computer games [JNS12]. Even delays around 200 ms between mouse movements and cursor movements or between a button press and a visual stimulus might go unnoticed [REG14, RSE14]. Although the available body of literature suggests that participants might fail to detect delays below 120–200 ms, other studies report that some users are able to spot latency as small as 50 ms in the context of mouse-cursor movements [REG14], and even latency of around 15 ms while using a Head-Mounted Display [MAEH04]. Moreover, when asked to detect changes of system latency while wearing a Head-Mounted Display, users' perceptual stability across different virtual environments would require latency below 16 ms [EMAH04]
Delays between the execution of an action and its visual feedback can also impair two components of bodily self consciousness (sense of agency and sense of ownership over

one's body [Gal00]). The emergence of the sense of agency is thought to be based on an internal forward model, which compares the predicted sensory consequences of the motor commands with the actual sensory feedback [WGJ95, FBW00]. A mismatch between predicted and observed feedback, as in the case of delayed feedback, can result in a loss of the sense of agency [LH09]. When this happens, people might even attribute the observed delayed feedback of their movement to an external cause [FF02, FFG+01]. Although the sense of agency tends to decrease with increasing delay between an action and its sensory counterparts [FVH13, IA15], a certain amount of delay is acceptable for sense of agency to persist. Literature suggests that, when observing a delayed image of one's own hand, people attribute the observed movements to the self even at delays around 150 ms [FFG+01, LH09, TPH06, TLH10]. Above 100–150 ms, perceived agency toward the delayed visual feedback of hand movements decreases [FFG+01], and does not emerge at delays around 500 ms [LH09, TLH10]. However, when presented with visual flashes occurring after a voluntary button press, subjects report agency for flashes lagging even up to around 500 ms [RSE14]. It has been suggested that body ownership might allow a narrower temporal window of asynchrony than sense of agency, emerging with delays up to 100–150 ms [IA15, LH09].

A perceivable delay between movements and their observed consequences does not only impair sense of ownership and agency, but may also affect motor performance in 2D and 3D tasks (e.g., [TPSM09]). For instance, when controlling a character in computer games, delays greater than 150 ms affect performance [JNS12]. A latency of around 170 ms led to a longer time necessary to complete a grasping task [CS99]. In a simple coordination task, the introduction of a 200 ms delay significantly increases the error rate [Gut01]. Similarly, in a tracking task, latency of or above 110 ms dramatically increase the error rate [PS11]. A delay around 70–75 ms has been found already effective in decreasing performance in a VR reaching task [WB94], and in tasks which require moving a mouse cursor to a target [MW93]. In a non-VR and low latency scenario with a Fitts' law style pointing task a latency of about 16 ms already seemed to have an effect [FKS16]. Samaraweera et al. [SGQ13] showed in an experiment using a HMD that inducing a latency of 225 ms can change gait patterns. In another study they showed that latency as well as using a virtual mirror can alter gait behavior by adding 200 ms of delay to the system latency to one half of the participants' avatar [SPQ15].

To summarize, these findings show that the impact of latency differs in our four variables of interest — perceived latency, sense of agency, sense of ownership, and motor performance — for different setups and for different motor tasks.

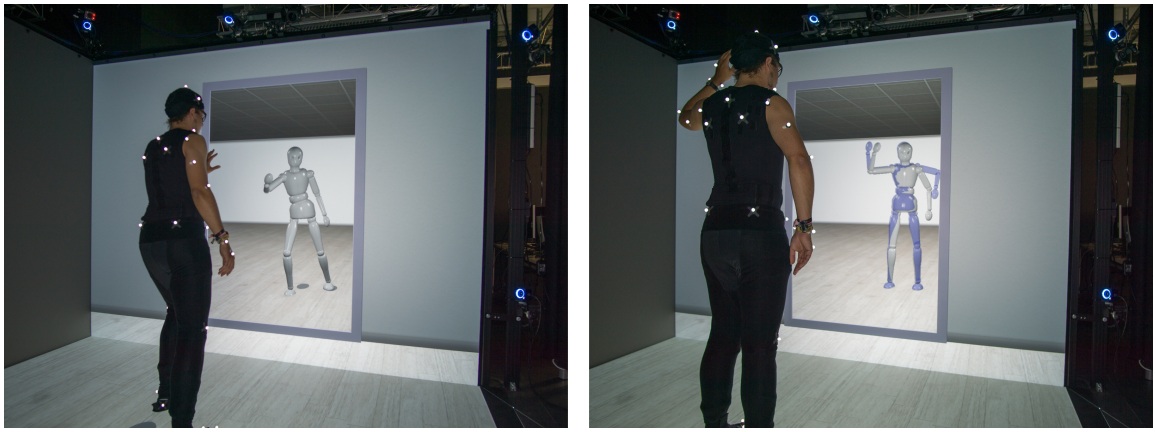


Figure 3.1: Left: The basic virtual mirror scenario consists of an empty room and a simplistic mirror avatar. Right: The extended scenario employed in the experiment, where the target movement is shown by a semi-transparent blue "ghost character".

3.4 Methods

In our experiment we confronted participants with their virtual mirror image and systematically induced five different levels of latency. Each participant performed 200 trials and after each trial we took perceptual judgments from the participants.
This section describes the used methods for this experiment. After starting off with the specifics of the employed apparatus, we go on with details on the procedure and stimulus used in the experiment. Then, we introduce key data on the participants that we recruited for the experiment. Finally, we provide details on the measures we took from the participants.

3.4.1 Apparatus

For this experiment we used the system as proposed in Chapter 2, except for the following modifications.
Participants were placed inside an empty virtual room in front of a virtual mirror. This mirror showed a reflection of the room itself as well as a mannequin-like virtual avatar. The engine rendered this scene at around 240 fps. This *mirror avatar* was scaled according to the participants' limb lengths and was animated according to the participants' motion in real time. Figure 3.1 (left) shows an example of this setup.
The end-to-end latency between the participants' movements and the corresponding movements of the mirror avatar was increased synthetically using a FIFO buffer holding back the motion data. This buffer was filled with incoming motion capture frames, and as soon as one of these frames was older than the desired latency offset, it was emitted and used to

animate the mirror avatar. If multiple frames satisfied this condition, the one closest to the desired latency was used. We only delayed the mapping of the participants' motion onto the mirror avatar. The movement of the tracked glasses used for perspective adaptation was not delayed.

Verifying latency

To determine the latency of our system itself and of our mechanism to induce additional latency, we conducted end-to-end latency measurements. To this end, we again used the latency measurement approach, as described in Section 2.5. For these measurements we were able to use a high-speed camera with higher temporal resolution as in Chapter 2 (Nikon 1 J4, 400 Hz). This allowed us to achieve more precise measurements. These measurements were performed for the base latency as well as for the latency offsets which we manually induced during the experiment. We were able to obtain a mean for the base latency of 44.9 ms (SD=6.1 ms). The latency induced manually by our system was identical to the desired latency.
The base latency of about 45 ms consists of the individual latency of the tracking cameras ($\sim$4 ms according to manufacturer), of the tracking software, the motion preprocessing ($\sim$2 ms), the network communication ($\sim$1 ms), as well as rendering ($\sim$4 ms at 240 fps), display synchronization, and display hardware ($\sim$19 ms according to manufacturer). The known latency values add up to $\sim$30 ms, thus the remaining $\sim$15 ms must be mainly due to display synchronization and tracking software.

3.4.2 Procedure and Stimulus

First of all, participants filled in questionnaires for demographic data, simulator sickness [KLBL93] and immersive tendencies [WS98]. Then they read the instructions for the experiment. Afterwards, they put on the marker suit and performed skeleton calibration in the CAVE. We did some refinement of marker positions if necessary. Next, participants were presented with the virtual mirror showing only the mirror avatar (cf. Figure 3.1, left). This scene was shown for about one minute to let participants familiarize themselves with the virtual mirror concept.
The following main part of the experiment consisted of multiple trials. In each trial, the mirror avatar was super-imposed with a second character (blue, semi-transparent), which was scaled the same way as the mirror avatar (cf. Figure 3.3(a)). This *ghost character* performed pre-recorded animations (cf. Figure 3.2). Participants were instructed to simultaneously mimic the movements of the ghost character, i.e., to move together with the ghost character. For all frames in which the difference in posture between both characters did not

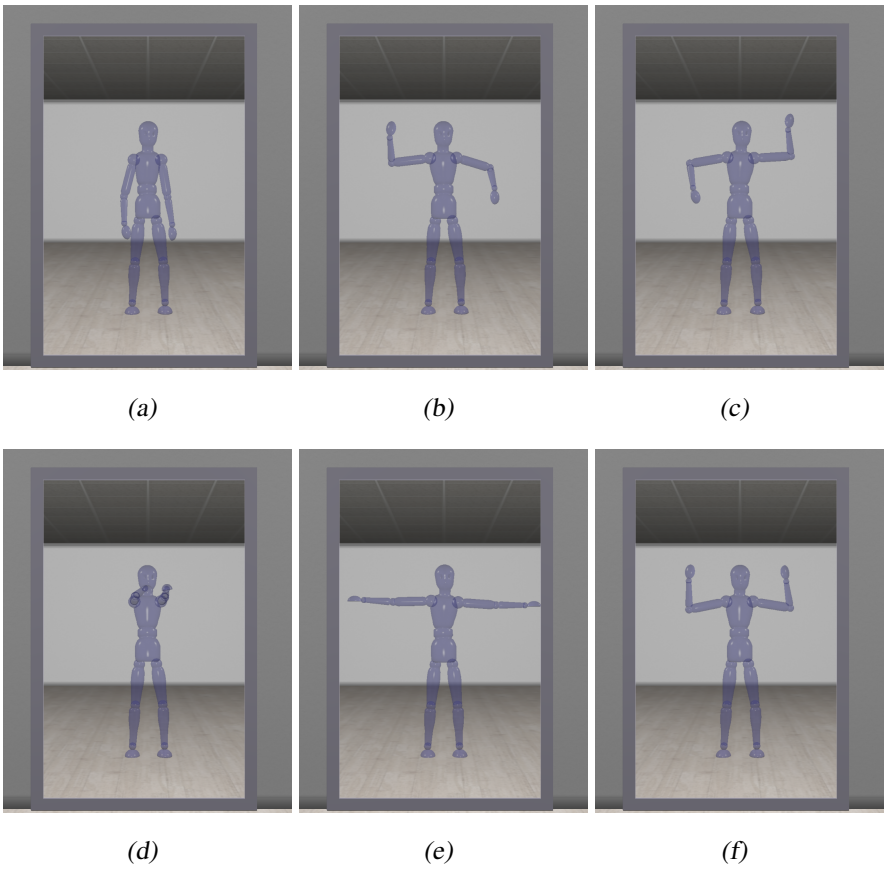


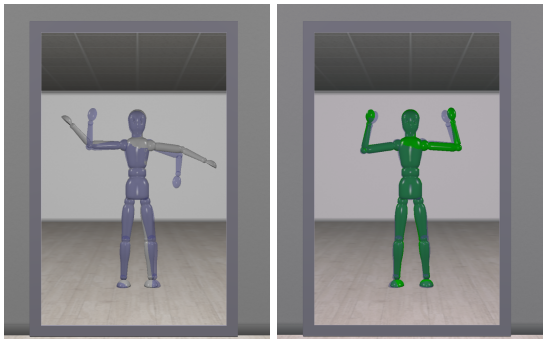
Figure 3.2: (a): Starting posture. (b)-(f): Turning point postures of the five animations used for the ghost character.

exceed a certain threshold (see Section 3.4.3), the mirror avatar was colored green to give feedback on the accuracy of the performance (cf. Figure 3.3(b)). During the trials, we systematically introduced a delay in the movements of the mirror avatar. This delay was varied between 45 and 350 ms on a logarithmic scale and randomly set per trial. After each trial the virtual mirror was disabled and turned white, and participants were asked three questions on perceptual judgments (cf. Section 3.4.3). The supplemental video shows an example of one trial. After the main part of the experiment, participants filled in the simulator sickness questionnaire again and took off the marker suit. In total the experiment took about two hours.

We used five different pre-recorded animation sequences to animate the ghost character (cf. Figure 3.2 and supplemental video). We limited these animations to upper body movements to keep the task simple and avoid fatigue. Pilot experiments had shown that large movements of the whole body including the legs (e.g., squat movement), are too exhausting for a long experiment as ours. Additionally, we noticed that lower body movements

that involve lifting one leg off the ground increase the risk of participants tumbling. The same movement was always presented twice in a row per trial. The movements varied in speed and trajectory and thus were not entirely predictable. This kind of movement was selected in order to prevent participants from recurring to unwanted strategies while performing the motor task. Indeed, individuals tend to modify their performance in order to compensate for feedback delays either by slowing down or by adopting a "move and wait" strategy [Fer65, PK99]. The latter consists of ignoring the visual feedback while executing fast movements, and then waiting to catch up with the visual feedback before continuing the task. By forcing participants to execute fast movements with variable velocity and non-linear paths, we aimed to prevent such strategies [RvDE14, RE16].

Each animation lasted 5 s and had the same starting posture (cf. Figure 3.2(a)). Before an animation started, the ghost character remained in the starting posture for 1.5 s to allow participants to take that posture. Thus, each trial lasted 6.5 s. As latency offsets we used 0, 30, 80, 165 and 305 ms. When added to the baseline latency of the system, these offsets resulted in end-to-end latency of 45, 75, 125, 210 and 350 ms, respectively. We decided to also take large latency ($>$ 125 ms) into account to identify the perceptual limits of the sense of agency and body ownership, as pilot experiments had revealed that participants tolerate larger delays when judging agency and ownership.



(a) *(b)*

Figure 3.3: The participants' avatar is grey (a), if it is not sufficiently close to the target posture of the ghost character (blue character) and green (b), if the posture is fitted sufficiently.

The experiment included 200 trials and ten practice trials. In the practice trials each animation sequence was shown two consecutive times with no additional latency. In the other 200 trials we randomized animations as well as latency offsets, while still making sure to equally distribute the combinations of animation sequences and latency offsets, so that each combination occurred equally often. Therefore, each animation as well as each latency was presented 40 times. Since this results in a long time for participants to stand and perform the movements (mean duration = 68 min (SD = 20 min)), we included a short break every 20 trials. These breaks were optional and participants could continue with the experiment whenever they wanted by touching a virtual box. Additionally, the mean accuracy of the motor performance since the last break was shown to motivate participants.

3.4.3 Dependent Variables

Perceptual Judgments

After each of the 200 trials participants were asked the following three questions in randomized order:

Agency judgment (AJ): *Was the motion of the grey avatar the visual feedback to the movement you just performed?*

Simultaneity judgment (SJ): *Was the motion of the grey avatar simultaneous to your movement?*

Ownership judgment (OJ): *Did you feel that the grey avatar belonged to you?*

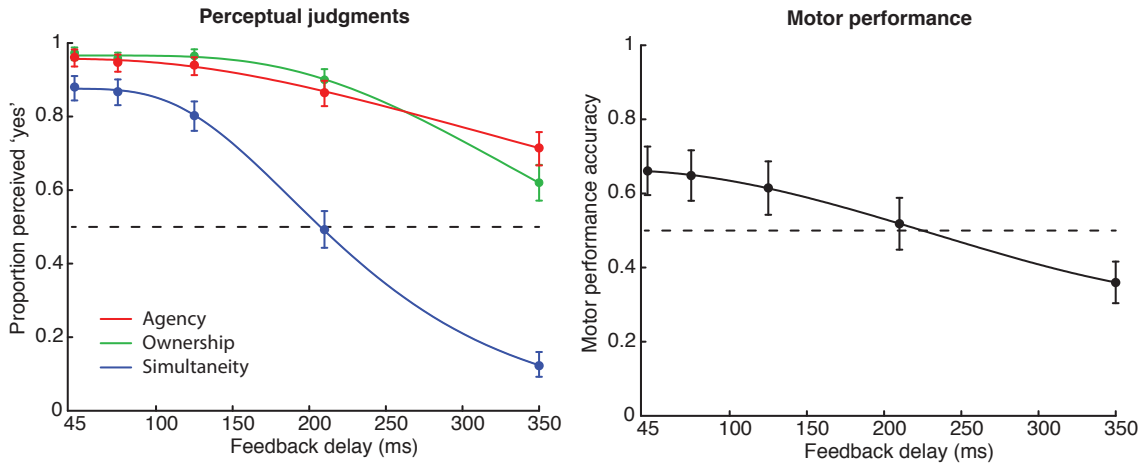
We asked participants to answer these questions according to how they *felt* and how they *perceived* the movements of their avatar. The questions were visualized as text inside the CAVE and were answered by touching virtual "Yes" or "No" boxes. The fact that AJ, OJ and SJ were all yes/no tasks allowed a direct comparison of the temporal windows of perceiving agency, ownership, and simultaneity [RSE14].
For the agency judgment we instructed participants that the motion performed by their avatar could either reflect their own movement or some pre-recorded movement previously performed by them or some previous participant. We introduced such a "cover story" to provide participants with a reasonable alternative when attributing the causality of the action [RE16]. To avoid that participants probe the system for agency (e.g., by making unexpected movements to test if the avatar follows), we instructed them explicitly against such probing.

Motor Performance

The motor performance was determined by checking the posture deviation of the participants' avatar with the ghost character. For each frame the distances of shoulder, elbow, and wrist joint of the mirror avatar to the corresponding joints of the ghost character were calculated and averaged. If that average distance was inside the error range of 12 cm, the participants' posture was counted as sufficiently close and successful. The number of successful frames divided by all rendered frames gave us the percentage of successful postures in each trial.

3.4.4 Participants

Ten participants, naïve towards the experiment, (4 males, mean age M = 23.2, standard deviation SD = 2.2, 9 right handed) with normal or corrected to normal vision took part in the



(a) *(b)*

Figure 3.4: Psychometric functions for all tasks, obtained by pulling together the data from all participants. a) Perceptual judgments (agency, ownership, simultaneity). b) Motor performance. Analyses were performed on the individual data from each participant, while aggregated data are shown for illustrative purposes only. Error bars represent confidence intervals set at 95 %.

study. Participants provided written informed consent and got paid for their participation. The study was conducted in accordance with the Declaration of Helsinki, and had ethical approval from our University's ethics committee.

3.5 Results

The first ten trials served as practice trials and were not included in the analyses. For each participant we calculated the percentage of "yes" responses in each perceptual task (i.e., AJ, OJ, SJ). The effect of feedback delay on perceptual judgments and motor performance was assessed separately for each task, by fitting the probability of "yes" responses and the accuracy in the motor performance with a generalized linear mixed model (GLMM, [Agr02, MML12]). In such a model, the overall variance is divided into a fixed and a random component [Agr02]. The fixed component tests the effect of the independent variable (manipulated in the experiment), while the random effect accounts for the heterogeneity among different participants. Thus, the GLMM allows the analysis of clustered categorical data, at the population and individual levels simultaneously [MML12]. Here, feedback delay (log-transformed) was the fixed effect, and subject identity was the random effect (Model 1). A probit link function was applied. The model parameters were estimated us-

Delay	Agency	Ownership	Simultaneity	Motor performance
45 ms	96 %	97.25 %	88 %	66.1 %
	(93.6–97.7 %)	(95.5–98.8 %)	(84.4–91 %)	(59.6–72.6 %)
75 ms	94.8 %	95.5 %	86.75 %	64.83 %
	(92–96.7 %)	(93–97.3 %)	(83–90 %)	(58–71.6 %)
125 ms	94 %	96.5 %	80.25 %	61.5 %
	(91.2–96.1 %)	(94–98 %)	(76–84 %)	(54.3–68.7 %)
210 ms	86.5 %	90 %	49.25 %	51.83 %
	(82.8–89.7 %)	(86.6–92.8 %)	(44.35–54.3 %)	(44.7–58.9 %)
350 ms	71.5 %	62 %	12.25 %	36 %
	(66.8–75.9 %)	(57–66.8 %)	(9.2–15.9 %)	(30.4–41.6 %)

Table 3.1: The table depicts the mean percentage of "yes" answers for agency, ownership, and simultaneity judgments as well as mean percentage of the successful postures for motor performance for each delay. The confidence intervals (lower limit–upper limit), set at 95 %, are listed beneath their corresponding mean value.

ing Maximum Likelihood Estimation (MLE). The data was analyzed with MATLAB. The GLMM analysis revealed a significant effect of feedback delay for all tasks (all $p < 0.001$). As shown in Figure 3.4, the higher the delay, the lower the probability of responding "yes" in the perceptual judgment tasks, and the lower the accuracy in motor performance (see also Table 3.1 for results for each tested delay). In other words, sense of agency, ownership, and perceived simultaneity decreased, and motor performance worsened, with increasing delay. Regarding the perceptual judgment tasks (Figure 3.4(a)), the temporal window of perceived simultaneity was narrower than those of perceived agency and ownership. Trials with the minimum system delay (45 ms) were perceived as simultaneous 88 % of the times on average across participants. Perceived simultaneity dramatically dropped with increasing delay: 49 % of trials with 210 ms delay and only 12 % of trials presenting 350 ms delay were perceived as simultaneous.

In contrast, although both perceived agency and ownership decreased significantly with increasing delay, they did not break down even at the highest tested delay (350 ms). Indeed, while for minimum delay participants reported sense of agency in 96 % of the trials and

sense of ownership in 97 % of them, sense of agency and ownership still occurred at 350 ms delay in 71.5 % and 62 % of the trials, respectively. Responses in the OJ and AJ tasks were comparable, as shown by overlapping confidence intervals (CI, set at 95 % confidence level) for all tested delays between the two tasks (see Figure 3.4(a) and Table 3.1). Motor performance gradually worsened with increasing delay, from 66 % of accuracy at 45 ms delay to 36 % at 350 ms (Figure 3.4(b) and Table 3.1). Overall, performance in each of the four tasks did not differ between 45 and 75 ms delays, as shown by overlapping CI in each task between those two delays. Motor performance significantly worsened between 75 ms (M = 64.8 %, CI = 58–71.6), and 125 ms (M = 61.4 %, CI = 54.3–68.7; Wilcoxon signed-rank test, p = 0.009). Similarly, perceived simultaneity showed a tendency to decrease between 75 ms (M = 87 %, CI = 83–90) and 125 ms (80 %, CI = 76–84, Wilcoxon signed-rank test, p = 0.07). Conversely, perceived agency and ownership started declining between 125 and 210 ms, while they did not significantly differ between 75 ms and 125 ms (agency, 75 ms: M = 94.8 %, CI = 92–96.7; 125 ms: M = 94 %, CI = 91.2–96.1, Wilcoxon signed-rank test, p = 1; ownership, 75 ms: M = 95.5 %, CI = 93–97.3; 125 ms: M = 96.5 %, CI = 94–98, Wilcoxon signed-rank test, p = 0.52).

With the minimum delay (i.e., 45 ms), motor performance was more accurate in trials reported as simultaneous (accuracy, M = 67.3 %, SD = 11.9) than in those perceived as non-simultaneous (M = 54.2 %, SD = 10.3), and in those for which participants perceived agency (M = 66.5 % SD = 12.3) and ownership (M = 66.5 %, SD = 23) as compared to those that did not elicit agency (M = 62 %, SD = 10.7) and ownership (M = 51.7 %, SD = 10.5) feelings.

Given the positive association between the quality of the motor performance and the successive perceptual judgements, we tested a second model to assess whether the influence of motor performance in the task was significant. We fitted the probability of "yes" responses in each task with a GLMM model with performance accuracy as fixed effect, and subject identity as random effect (Model 2). Results showed that motor performance accuracy predicts responses in all tasks (all $p < 0.001$). We compared the results of Model 2 (i.e., the one with performance accuracy as a fixed effect) with those of Model 1 (i.e., the one with feedback delay as a fixed effect). The two models employ the same functions, and differ in the predictors only. To select the model that better fits the data (i.e., the fixed effects that better explain the results), we compared the two models using the Akaike Information Criterion (AIC, [Aka73]). According to the AIC, Model 2 fitted the data better than Model 1. This result suggests that, although both delay and motor performance accuracy affect perceptual judgments, participants rely more on their own performance than on the delay itself when estimating feedback simultaneity, sense of agency, and ownership. However, we cannot exclude that the relationship between performance accuracy and subjective

judgments might be mediated by a third latent variable, and further studies are necessary to better understand this issue.

To make sure that participants were not influenced or biased because they were asked the same questions repeatedly in the AJ/SJ/OJ tasks, we ran further analyses. For each perceptual task we fitted the probability of "yes" responses with a GLMM with feedback delay, trial number, and their interaction as fixed effects, and subject identity as random effect (Model 3). The only significant fixed effect was that of the factor feedback delay (all $p < 0.001$), while trial number and the interaction of delay and trial number were not significant (all $p > 0.07$). Moreover, a model comparison according to the AIC favoured a model with a fixed effect only for the delay (Model 1) over more complex models with also fixed effects of trial number or the interaction between trial number and feedback delay (Model 3). These findings show that participants did not alter their response strategy during the time of the experimental session.

To assess whether individual differences in immersive tendencies are related to individual differences in perceptual and motor tasks, we calculated Pearson correlation between each item of the immersive tendencies questionnaire (ITQ) and the performance in each task with minimum delay (i.e., 45 ms). We found a significant correlation between the item of the ITQ ["Do you easily become deeply involved in movies or TV dramas?"] and sense of ownership ($r = 0.66$, $p = 0.038$), and between the same item and sense of agency ($r = 0.81$, $p = 0.005$). Thus, participants who report to be usually deeply involved in movies are the ones showing greater sense of agency and ownership in the experimental task. Moreover the item ["Do you ever become so involved in doing something that you lose all track of time?"] was negatively correlated with perceived simultaneity ($r = -0.73$, $p = 0.01$) and motor performance ($r = -0.65$, $p = 0.04$). Thus, participants reporting a tendency to lose track of time showed worse motor performance and a higher tendency to report trials at 45 ms delay as non-simultaneous.

Finally, to test for the presence of motion sickness induced by our system, we compared the responses to each item of the simulator sickness questionnaire between the first and the second presentation of the questionnaire using the Wilcoxon Signed rank test. Participants did not show motion sickness after taking part in the experiment (all $p > 0.38$).

3.6 Discussion

Previous research involving hand movement suggests that sense of ownership might be more affected by delay than sense of agency [IA15]. Here we found that even a delay as high as 350 ms elicits both sense of agency and ownership. This result is in line with recent findings suggesting that sense of agency toward a moving hand might drive sense of owner-

ship even toward the delayed image of the hand [Asa16]. Our findings suggest that movements which are more complex than a simple button press, as in the case of the full-body movements described in our study, induce a strong sense of ownership, even despite long delays in the visual feedback. Notably the temporal structure of such a feedback would be strongly correlated with participants' motor performance. Thus, participants might rely on correlation between the temporal structure of the motor and visual signals to infer a common cause (between motor actions and visual feedback), and a sense of agency, ownership, and simultaneity. A recent computational model for multisensory integration [PE16] suggests that people rely on such temporal correlation across different sensory signals when estimating simultaneity and common source.

Furthermore, results showed that motor performance accuracy predicts perceptual judgments even better than delays itself. Thus, when motor performance is poor (i.e., in case of a mismatch between the desired outcome and the actual performance), participants tend to perceive the feedback as non-simultaneous, and to lose sense of agency and ownership. In other words, a small delay tends to be perceived as non-simultaneous in case of high performance errors, and higher delays might go undetected if motor performance is accurate. This finding suggests that, in principle, the same delay in a VR system might or might not impair perceptual judgments, depending on the difficulty of the motor task. In case of a simple motor task, leading to high performance accuracy, the delay might go unnoticed. In case of more complex tasks, leading to higher chances of performance errors, delays will exert a stronger impact on perceived agency, ownership, and simultaneity. This might explain why in previous studies recurring to button press [RSE14], participants were unaware even of long delays: a simple motor performance requiring only a button press would not lead to error signals related to the motor performance, and this in turn reduces the probability of detecting the delay. For the same reason, trained users (for instance, experts in a certain sport), performing a motor task better than novices in a sport scenario, might be perceptually less affected by delays than novices. Previous studies have shown that people are not particularly accurate in detecting delays in sensorimotor tasks: delays up to around 200–250 ms can even go unnoticed, and in some cases participants might report agency also for events preceding the action [RSE14]. Here we show that the quality of the motor performance strongly influences perceptual decisions.

3.7 Conclusion

In the presented study, we used a novel paradigm employing VR and psychophysical procedures to investigate how delays between an action and its visual feedback affect perception and motor performance. In particular, we assessed the impact of latency on perceived si-

multaneity, sense of agency, ownership, as well as motor performance during the execution of full-body movements in front of a virtual mirror. Considering the large variability in the literature concerning a possible threshold for visual delays to be detected and affect behavior, we systematically assessed the effect of delays on motor performance and perceptual judgments in the same setup, and in the same experimental session. To this end, we parametrically varied the delay of the visual feedback between 45 ms (i.e., the latency intrinsic in the system) and 350 ms, while participants on each trial performed both a motor task and perceptual judgments on the delayed feedback. Results show that, although inducing delay increasingly affects perceptual judgments and motor performance, incrementing the delay from 45 to 75 ms had no significant impact on participants' performance in the four tasks. Delays above 75 ms worsen motor performance and influence simultaneity perception, while they do not affect perceived agency and ownership, which significantly start declining only later, between 125 ms and 210 ms. Interestingly, the temporal windows of perceived agency and ownership are much broader than the one of perceived simultaneity: while at the highest tested delay (i.e., 350 ms) visual feedback is hardly perceived as simultaneous, perceived agency and ownership are not disrupted, although significantly reduced. Thus, participants tolerated perceptible delays when reporting agency and ownership toward their avatar's movements. These findings are in line with studies investigating the temporal window of agency and simultaneity [FVH13, RSE14] when participants judged whether a visual event presented on a screen was caused by their button press and whether it was simultaneous. Here we show that this also holds true in the case of complex upper-body movements. Moreover, we found that individual differences in immersive tendencies modulate sensitivity to delays, sense of agency, and ownership, as well as motor performance.

In conclusion, our study extends our understanding of the effect of feedback delays on perceptual and motor tasks. While previous studies mainly involved manual tasks, we focused on more complex movements involving the entire upper body. Since we always tracked and visualized the whole body in an immersive environment employing a virtual mirror, we assume that our findings transfer from upper-body to full-body movements, but this has to be examined in future studies. Moreover, our study introduced a systematic investigation of the interplay between perceived simultaneity, agency, and ownership in virtual environments, while previous research has mainly focused on a subset of these phenomena. The present findings contribute to a deeper understanding of what information people rely on when making perceptual decisions about agency, ownership, and simultaneity. Furthermore, it contributes to fill a gap in the literature regarding which delay might be considered acceptable for VR systems for motor learning of complex tasks involving full-body

movements. While smaller delays might not appreciably affect perception, they might still impair motor performance.
In general, this work stresses the importance of latency for full-body interaction in virtual environments. While latency is often neglected or at least put in second place, we argue that the latency of a VR system used for a specific study should at least be reported. Moreover, the effect of latency should be taken into consideration as the source of possible side effects.

Up Next

While this chapter concentrates on latency and its impact on virtual embodiment effects as well as motor performance, the upcoming chapters will tackle the impact of personalized avatars and immersion level on virtual embodiment effects when interacting with the virtual mirror image.
To investigate the factor immersion, we only have to visualize the virtual mirror in different display devices (CAVE and HMD), which is a rather trivial task. In contrast, the technique to create the necessary personalized virtual avatars to investigate the factor personalization is a non-trivial task. Therefore, before we face the actual experiment and its results in Chapter 5, we will learn how to create personalized avatars in the next chapter.

4

Fast Generation of Virtual Humans

4.1 Introduction

Virtual characters are widely used for applications ranging from computer games, special effects in movies, virtual try-on, to medical surgery planning and virtual assistance. Virtual characters are especially important for Virtual Reality (VR) for both virtual agents simulated by artificial intelligence as well as avatars, the digital alter-egos of the users in the virtual worlds. Especially interesting in the context of this work are avatars in immersive embodied scenarios such as the virtual mirror. Here, they provide ample possibilities to study psychophysical effects caused by modifying avatar appearance and hence, e.g., altering self-perception or body ownership.

However, to create such virtual characters, a way to digitize humans is necessary. 3D-scanning of real humans is a prominent technique to generate virtual humans. Striving for realism and human-like appearance requires geometrically accurate meshes and detailed textures, and the application of the resulting models in interactive scenarios requires them to be animated: Their full-body posture, hand posture, eye gaze, and facial expressions have to be controllable through suitable skeletal rigs and blendshapes, respectively. To be widely employable, the resulting character models should be compatible with standard game engines or VR frameworks. Further, the overall avatar creation should ideally be fast enough to be performed right before empirical studies and thereby allow to create avatars for as many as possible participants to gain statistical more meaningful results.

However, despite the increasing availability of scanning technologies and the large body of research on 3D-scanning and mesh reconstruction in both computer vision and computer graphics, creating believable and ready-to-animate virtual humans in a short amount of time is still a challenging problem. Existing approaches reconstruct static full-body "selfies" [LVG+13] without animation controls, or full-body models without controls for hands or facial expressions [BRLB14, BBLR15], or head models for facial puppetry without a full-body [WBLP11, CHZ14]. Approaches for the fast generation of characters with all required animation controls are mostly lacking. In addition, many approaches focus on geometry reconstruction only, and neglect the generation of high quality textures from scanner input.

We present a complete character generation pipeline that is able to digitally clone a real person into a realistic high-quality virtual human, which can then be used for animation

and visualization in any standard graphics or VR engine. The whole reconstruction process requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC.

For 3D-scanning we employ a custom-built camera rig with 40 cameras for the body and 8 cameras for the face, and compute dense point clouds through multi-view stereo reconstruction. In order to robustly deal with noise and missing data, and to avoid character rigging in a post-process, we fit a generic human body model to the user's scanner data. In particular, we build upon the template model from Autodesk's Character Generator, which is already equipped with a detailed skeleton and skinning weights, a rich set of blendshapes, as well as eyes and teeth. This template model is further enriched by statistical data on human body shapes, which yields a prior for the template fitting process. By fitting the template geometry to the scanner data and transferring eyes, teeth, skeleton, and blendshapes to the morphed template, our reconstructed models are ready to be animated.

By construction, all our reconstructed characters share the tessellation of the template model. Hence they are in dense one-to-one correspondence, which allows to transfer properties between models. To keep our models simple and compatible to any standard rendering engine, and to enable highly efficient character animation, we represent our characters by a single-layer mesh and employ standard skinning and blendshapes for body and face animation, respectively.

Overall, our approach enables the generation of realistic and fully ready-to-animate virtual humans in just a couple of minutes, which makes them accessible to a wide range of VR experiments, where they can be used as avatars (see Chapter 5) or conversational agents.

My Contribution *The avatar creation pipeline was developed in cooperation with Jascha Achenbach. Jascha Achenbach developed the non-rigid fitting approach and used it to create realistic faces. Additionally, he implemented the texture processing techniques to generate homogeneous textures for the whole body. I extended the fitting pipeline so that the fitting of full bodies is possible. Further, I worked on speeding up the whole pipeline so that the fitting approach only takes a few minutes. Moreover, we worked together on the pipeline so that it works as automatic and reliable as possible. Finally, the full-body scanning rig was designed and built by both of us.*
Corresponding publication: Fast Generation of Realistic Virtual Humans, VRST 2017.

4.2 Related Work

Due to the increasing availability of 3D-scanning solutions and the growing demand for virtual human models, there is a huge body of literature on scanning, reconstructing, and animating virtual characters. For the sake of clarity we restrict to the approaches most relevant to ours, beginning with techniques for reconstructing full body models, followed by face capturing methods, and finally discussing approaches for reconstructing ready-to-animate VR characters.

4.2.1 Full-Body Reconstruction

Several methods employ affordable RGBD sensors (e.g., Kinect) for scanning and reconstructing human bodies [TZL+12, LVG+13, SBKC13, FSR+14]. However, due to the coarse and noisy data delivered by these sensors, their character reconstructions are bound to a rather low quality. Since our goal is reconstructing realistic high quality virtual humans, we instead base our framework on a multi-camera rig that can capture a subject in a fraction of a second. Using multi-view stereo we then reconstruct a dense point cloud from the camera data.

This point cloud could then be fed into a surface reconstruction method, followed by an auto-rigging process for embedding a control skeleton and defining skinning weights [BP07, FCS15]. However, the surface reconstruction might fail to faithfully capture delicate features (e.g., fingers), causing the auto-rigging to fail. We therefore use a fully-rigged template model that we fit to the scanner data using non-rigid registration.

Fitting a template model to a large amount of training data allows to build a statistical model, which can act as prior when fitting the template to scanner data. The SCAPE model [ASK+05] is one of the first, most prominent, and most frequently employed human body models. It has been extended in many ways [HLRB12, BRLB14, SBB07, SHRB12, PWH+17], which have been applied in different scenarios ranging from breathing animation [TMB14], over soft-tissue animation [LMB14], to estimation of shape and posture from either a single image [GWBB09] or from RGBD sequences [WHB11, BBLR15].

Many other statistical human body models have been proposed [ACP03, ACPH06, HSS+09, WPB+14, LMR+15], which can be roughly classified as triangle-based or vertex-based methods, depending on how they model posture articulation and fine-scale deformation. Triangle-based methods have to solve a linear Poisson system to compute the deformed vertex positions, and are therefore incompatible to standard graphics engines. In contrast, models based on per-vertex linear blend skinning, such as, e.g., SMPL [LMR+15] or S-SCAPE [PWH+17], can readily be used in such engines. We therefore also base our model on vertex-based linear blend skinning. However, in comparison to SMPL and S-SCAPE

our model has a higher geometric resolution and provides fine-scale details, such as fingers, eyes, and teeth. Furthermore, it is equipped with a more detailed skeleton and allows for hand and face animation.

In order to place the skeleton within the model shape, SMPL learns a joint regressor from a large amount of data, which then represents joint positions as a linear function of the model's shape. Since our skeleton is more detailed than that of SMPL and the training data is not available, we cannot use their regressor. Instead, we follow Feng et al. [FCS15] and represent the joint positions as generalized barycentric combinations of the template's vertex positions, which also is a linear function.

While the above methods work well for reconstructing the *geometry* of human bodies, they mostly neglect the texture reconstruction, which however is crucial for VR applications. In contrast, we reconstruct a high-quality texture from the reconstructed geometry and the individual camera images of our scanner.

4.2.2 Face Reconstruction

There is a lot of work dedicated to face reconstruction from images, video, RGBD data, laser scans, or multi-view stereo. The pioneering work of Blanz and Vetter [BV99] first proposed a PCA-based statistical face model for reconstructing face models from 3D scanner data or 2D photographs. Similar to body reconstruction, most approaches employ a statistical model as a regularizing prior.

Many approaches use an RGBD sensor to reconstruct face models [CWZ+14, LKS14] and/or to animate them based on captured performance data [ZMG+11, BWP13, HMYL15, TZN+15, TZS+18]. However, their face reconstructions suffer from low quality in geometry and texture, due to the inherent limitations of current RGBD sensors. High quality face reconstructions can be achieved through multi-camera rigs and multi-view stereo reconstruction [BBB+10, GFT+11, AZB15]. However, these approaches aim at a *static* high quality reconstruction and do not provide ready-to-animate models. Other works use video input to generate *dynamic* face models, which are subsequently animated based on the video stream [SWTC14, WBGB16, TZS+16a, GZC+16]. Ichim et al. [IBP15] proposed a method for creating a textured 3D face rig from picture and video input taken on a cellphone. For a thorough state of the art report on 3D face reconstruction, feel free to have a look at [ZTG+18].

Since we aim at high quality geometry and texture, while at the same time maintaining a short acquisition time, we employ multi-view face scanning based on [AZB15]. We take the deformed template model, which was previously fit to the full-body scan, and refine its face region by fitting it to the point cloud resulting from the face scan.

Dynamic facial animations are crucial for VR characters, e.g. for speech animation or emotional facial expressions. With the industry standard being linear blendshape models [LAR+14], the character generation pipeline also has to construct the required set of FACS blendshapes [EF78]. For high quality production without time constraints, these blendshapes are often created manually by artists or reconstructed by scanning real actors performing these expressions [ARL+09]. A faster process is enabled by example-based facial rigging [LWP10], which generates personalized facial blendshapes from a small set of example expressions. Since we want to keep acquisition and processing time low, we scan the actor in neutral expression only, and generate the full set of FACS blendshapes by adjusting the template's generic blendshapes to the deformed model using deformation transfer [SP04]. If acquisition and processing time is not that critical, reconstructing a few additional expressions and using example-based facial rigging would be a good compromise.

4.2.3 Avatar Reconstruction

While there are many approaches for reconstructing human body shapes *or* human faces *or* human hands, only few previous works aim at reconstructing a complete virtual human featuring animatable body, face, *and* hands.

Malleson et al. [MKK+17] present a single snapshot system for rapid acquisition of animatable, full-body avatars based on an RGBD sensors. While the total processing time is in the order of seconds, the body is a stylized astronaut character that roughly fits the body dimensions only. Albeit face shape and texture are also considered, the results are of rather low quality and lack facial details, as only a low-dimensional face space is considered for fitting.

Feng et al. [FRS17] present a system for generating virtual characters by scanning a human subject. Their model is equipped with a full-body skeleton rig and is capable of facial expressions and finger movements. In direct comparison, their reconstruction process takes about twice as long as ours and requires more manual effort. Blendshapes are generated by explicitly scanning the actor in five different expressions, restricting the model to a few, but nicely personalized blendshapes. In contrast, our method reconstructs the full set of FACS blendshapes from a single face scan in neutral pose, and thus is compatible with standard animation packages. On the downside, our blendshapes are more generic and not as actor-specific. The biggest drawback of Feng's method is that by construction each model has a different tessellation, which prevents statistical analysis and property transfer between models. In contrast, all our models share the tessellation of the initial template mesh.

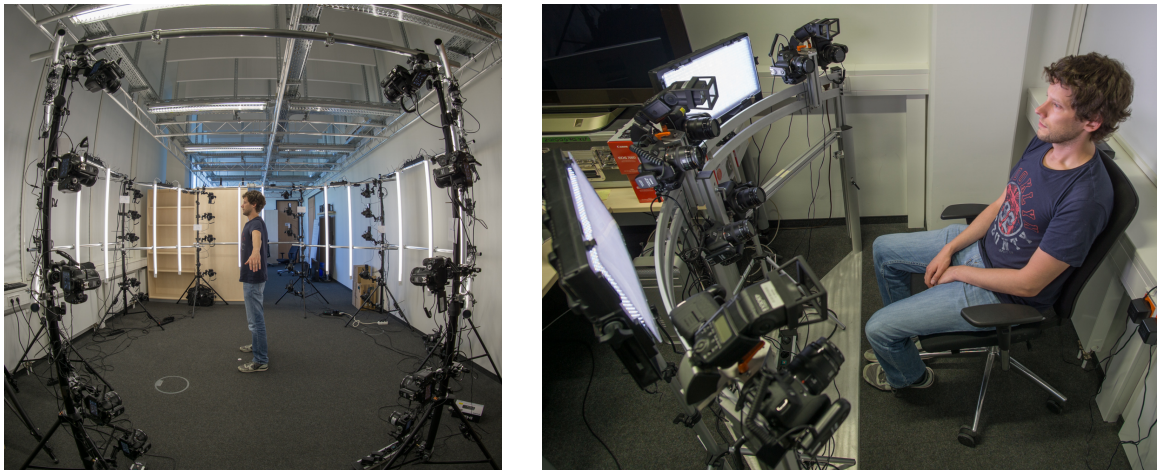


Figure 4.1: Our custom-built full-body scanner (left) and face scanner (right) are both based on multi-view stereo and consist of 40 and 8 DSLR cameras, respectively.

4.3 Input Data

There are several approaches available to create 3D scans of real people. Many of these employ active light to bring known information onto the to-be-scanned surface, e.g., laser or structured light scanners. However, this demands for special purpose hardware and often features a non-instant scan process. In contrast, passive stereo vision based methods like photogrammetry (also known as multi-view stereo reconstruction), are able to capture with a single instant shot using commodity hardware. Further, these methods readily adapt to different camera arrangements and number of cameras. Additionally, they are able to derive detailed high quality textures which are intrinsically registered with the captured 3D data. Moreover, current photogrammetry software solutions are already mature and provide fast and reliable acquisition of 3D point data as well as a lot of features like automatic self-calibration of camera positions.
Therefore, the scanners employed to create the virtual characters for this work are (custom-build) photogrammetry scanners. In order to provide significantly more detail in the important face region, we actually employ two scanners: A full-body scanner and a face scanner. The two scans are combined later during post processing of the scans as described later in the next section.

Full-body Scanner

Figure 4.1 (left) depicts the employed full-body scanner, which was build from 40 mid-range consumer DSLR cameras (Canon 700D) featuring a resolution of 18 Megapixels and 35 mm lenses. These cameras are mounted on eight stands, which are connected with each

other and arranged as an octagon. Each stand is therefore equipped with five cameras. The upper two and the lower two of these are placed as stereoscopic pairs picturing the same portion of the upper and lower part of the scanned subject's body, respectively. The other ones are positioned so that they picture the middle part of the subject's body to create additional information for the photogrammetry software, which eventually leads to higher scanning quality. In order to properly and uniformly light the scanned subject, bright LED tubes are mounted all around the octagon. We use LED tubes instead of flashes in this setup for two reasons. First, flashes would flash directly into the cameras on the other side of the rig, which would degrade overall image quality. Nevertheless, this could be compensated to some extend by using polarizing filters on the flashes as well as the camera lenses. Second, the triggering of all flashes at once, so that all cameras can see the light of the flash, would demand for a more complex and well calibrated setup. The LED lighting does not suffer from these issues, since it is more diffuse than a flash and shines continuously.

Face Scanner

The face scanner consists of eight mid-range consumer DSLR cameras (Canon 700D), each of which features 18 Megapixels and has a 50 mm lens attached. The cameras are set up pairwise so that two cameras are able to picture the same portion of the face from slightly different angles (Figure 4.1 right). In addition to the cameras the scanning rig also consists of three flashes, which are triggered in synchronization with each other and the cameras. We decided for a separate face scanner, in contrast to augmenting the full-body scanner with more cameras aiming at the face region, since otherwise the face cameras had to be manually adjusted to each subject's height in order to capture the face with the same amount of detail.

In order to minimize specular highlights on the scanned face, the cameras as well as the flashes are equipped with polarizing filters, which filter the polarized light of the flashes directly reflected from the skin's surface.

Instant Scan Process

The scan process itself must be instant and fast, to prevent erroneous scanning results. First, in order to take images that do not suffer from motion blur due to (small) movement or trembling of the subject, the cameras have to take pictures at a short exposure time (1/10 s). To this end the subject is illuminated by the LED tubes, for the full-body scanner, and the flashes, for the face scanner. Further, to avoid discrepancy between different shots, all cameras are triggered simultaneously.

Figure 4.2: Examples of the computed point clouds from our 40 camera full-body scanner (left) and 8 camera face scanner (right).

In order to do so, we use the cameras' built-in flash remote trigger input jacks, to trigger all cameras simultaneously. The input jacks are connected in parallel to a single wireless remote trigger receiver. By pressing the button of the remote trigger sender, the shutters of all cameras are released to shoot either 40 or 8 pictures at once. The three flashes of the face scanner are released simultaneously with the cameras so that the light of all flashes are visible in all resulting images. In the end we get sharp and well illuminated images of the subject.

Image Processing

These 40 and 8 images of the body cameras and the face cameras, respectively, are automatically passed to the commercial software Agisoft Photoscan Pro, which computes two high-resolution point sets $\mathcal{P}_B$ of the body and $\mathcal{P}_F$ of the face, as well as camera calibration data. The resulting face scans usually consist of about 1M points, while body scans feature about 3M points. Since the template mesh has a limited resolution of 21k vertices, we uniformly sub-sample the two point sets to 40k and 80k points, respectively. This sampling resolution is chosen such that the resulting point density is still about twice as high as the vertex density of the template mesh. This speeds up the fitting process significantly without noticeably sacrificing geometric fidelity. When it is clear from the context we omit the index B and F, and just write $\mathcal{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_N)$. Note that each point $\mathbf{p}_j$ is equipped with a normal vector $\mathbf{n}_j$ and RGB colors $\mathbf{c}_j$.

Since the bottom of the feet is not visible for the full-body scanner, these regions cannot be captured properly. The missing points below the feet can easily result in an erroneous fitting of the feet regions. In contrast, the floor around the feet is usually scanned quite well. We exploit this by detecting the floor plane and removing its points from the point cloud $\mathcal{P}_B$. We then we uniformly sample the detected floor plane underneath the feet region. This proved to be effective to capture the real extent of the feet and keep the feet on the floor during fitting without special treatment.

Statistical Template Model

As a template model we picked a character from Autodesk's Character Generator [Aut14], because these characters are already equipped with facial blendshapes, eyes and teeth, and a skeleton with corresponding skinning weights. However, any other template model with skeleton and blendshapes would work as well. The template mesh consists of $n \approx 21\text{k}$ vertices with positions $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$. A bar denotes vertex positions in the undeformed state: $\bar{\mathcal{X}} = (\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_n)$.

In order to incorporate prior knowledge on human body shapes into the reconstruction process, we integrate shapes from multiple data bases by fitting our template character to their registered body models. From the FAUST database [BRLB14] we used 10 scans of different subjects standing in A-pose, and we included 111 scans from [HSS+09]. Moreover, we added 82 synthetic models with different shapes from Autodesk's Character Generator. After fitting our template model to these models, they all share the same tessellation, allowing us to compute a ten-dimensional PCA subspace based on vertex positions of posture-normalized characters in T-pose. This PCA will act as initialization and regularization for the body fitting described in the next section.

4.4 Body Reconstruction

After computing and post-processing the point cloud $\mathcal{P}_B$ of the full-body scan, the next step is to align and fit the template model to this point set. As in most template fitting approaches, this fit is robustly performed in several steps: In the initialization phase, we optimize the alignment (scaling, rotation, translation), pose (skeleton joint angles), and PCA parameters for the ten-dimensional shape space. Afterwards, a fine-scale deformation fits the model to the data. Once the geometry fit is done, we have to compute texture, correct joint positions, and pose-normalize the model.

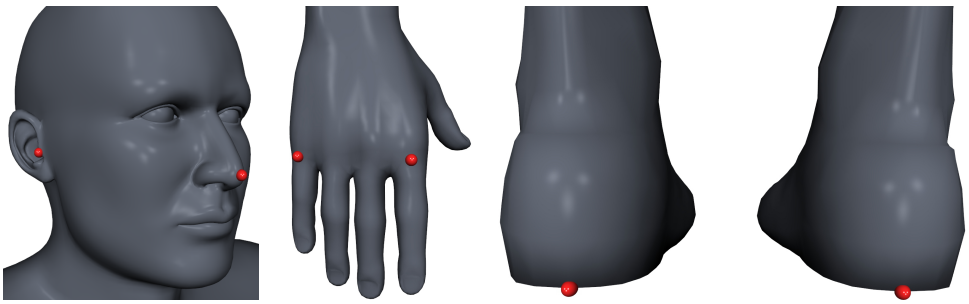


Figure 4.3: Nine landmarks are selected manually on the full-body point set, whose (pre-selected) counterpart vertices on the template model are shown here.

4.4.1 Initialization

Initially, the point set $\mathcal{P}_B$ and the template are in different coordinate systems and have different poses, since the template is in T-pose and the body scan is performed in A-pose. To bootstrap the template fitting procedure, we manually select nine easy to find landmarks $\mathcal{L}$ on the point-set $\mathcal{P}_B$, whose corresponding vertices on the template model have been pre-selected (see Figure 4.3). The landmarks have been chosen to ensure that important body parts like head, hands, and feet are fitted properly.

In the first step we optimize the alignment and pose of the template model in order to minimize the squared distances between these nine landmarks on the template model and their corresponding landmarks in the point set. To this end, we alternatingly compute (a) the optimal scaling, rotation, and translation [Hor87] and (b) optimize the joint angles using inverse kinematics [Bus04]. The template model is deformed based on linear blend skinning using the optimized joint angles. The procedure is iterated until the relative change of the squared distances falls below 0.05. This initialization process is depicted in Figure 4.4, (a) and (b).

The landmark-based fit gives us a good estimate of scaling, rotation, translation, and joint angles. We further optimize these variables by additionally taking closest point correspondences into account, which are computed by finding, for each point in $\mathcal{P}_B$, its closest point on the template. We prefer these scan-to-model correspondences over model-to-scan correspondences, since they were shown to yield more accurate fits [AZB15]. As usually done in ICP-based registrations, we prune unreliable correspondences based on distances and normal deviations. We employ the same alternating optimization as before to optimize alignment and pose, this time minimizing squared distances of landmarks and of correspondences (Figure 4.4(c)).

After convergence of the alignment and pose optimization, we add the PCA weights to the active variables and thereby optimize the geometric shape in the ten-dimensional PCA

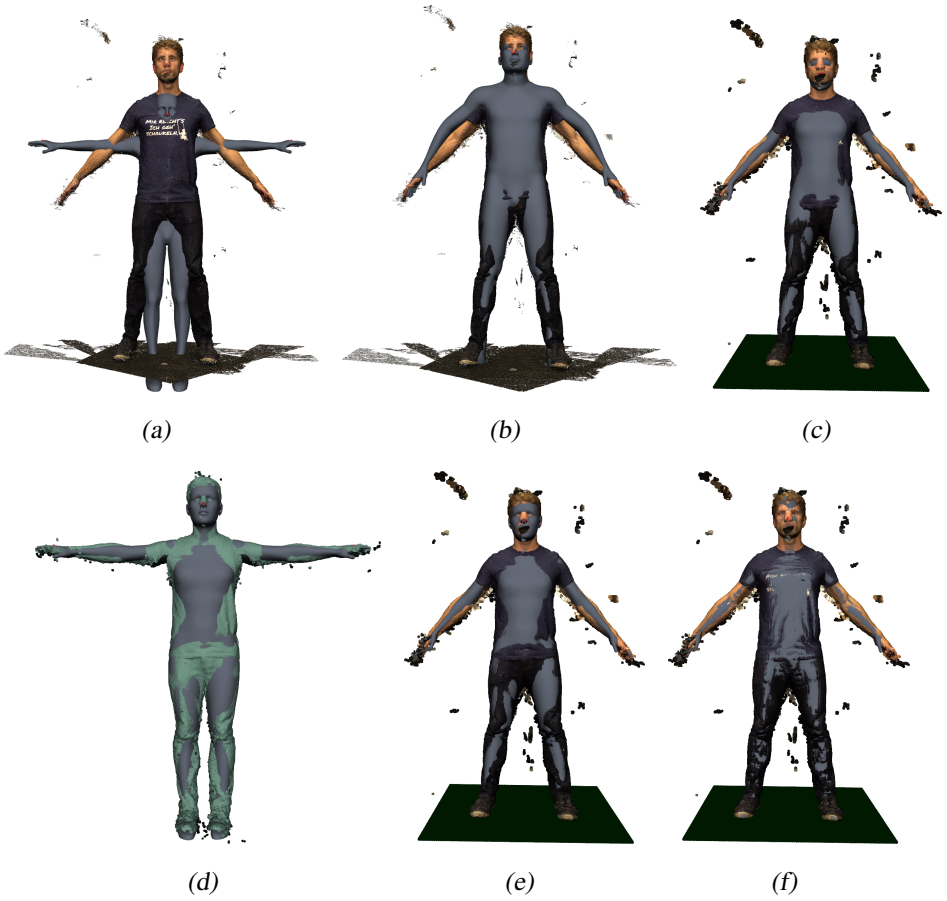


Figure 4.4: We first optimize alignment (scaling, rotation, translation: (a)) and pose (joint angles: (b)) based on nine manually selected landmarks. This fit is refined by incorporating closest point correspondences (c) and by alternating with PCA regularization in T-pose (d and e). After this initialization, we perform a fine-scale deformation to the point set (f).

space, again by minimizing squared distances between landmarks and correspondences. As our PCA model is pose-normalized in T-pose, the PCA-fitting is performed in T-pose (see Figure 4.4(d)), and is alternated with alignment and pose optimization. The shape change caused by adjusting PCA parameters requires adjusting the skeleton's joint positions. To this end, we represent joint positions by mean value coordinates [JSW05] with respect to the vertex positions of the template mesh. Joint positions are then a linear function of vertex positions, and hence also a linear function of PCA parameters. Two iterations of this procedure are usually sufficient for a good initial fit of shape and pose.

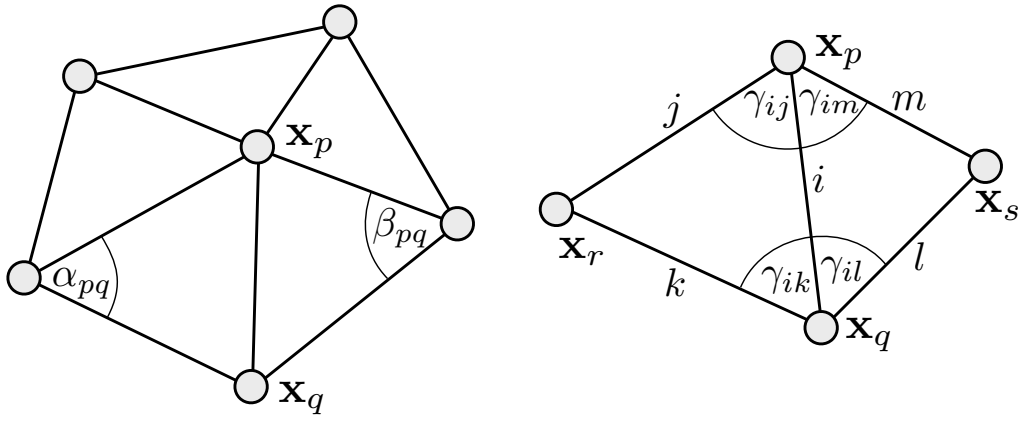


Figure 4.5: Notation for discrete Laplacians (Figure from [AZB15]).

4.4.2 Deformable Registration

With the point set and template model in good initial alignment we perform a fine-scale non-rigid registration, following the approach of [AZB15]. To this end, we minimize the energy

$$E_{\text{body}}(\mathcal{X}) = \lambda_{\text{lm}} E_{\text{lm}}(\mathcal{X}) + \lambda_{\text{fit}} E_{\text{fit}}(\mathcal{X}) + \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) \,, \tag{4.1}$$

where the three energy terms are explained below.

The *landmark term* E_{lm} penalizes the squared distance between the nine manually selected landmarks $\mathbf{p}_l$, $l \in \mathcal{L}$, in the point set and their counterpart vertices $\mathbf{x}_l$ on the template model

$$E_{\text{lm}}(\mathcal{X}) = \frac{1}{|\mathcal{L}|} \sum_{l \in \mathcal{L}} \|\mathbf{x}_l - \mathbf{p}_l\|^2 \,. \tag{4.2}$$

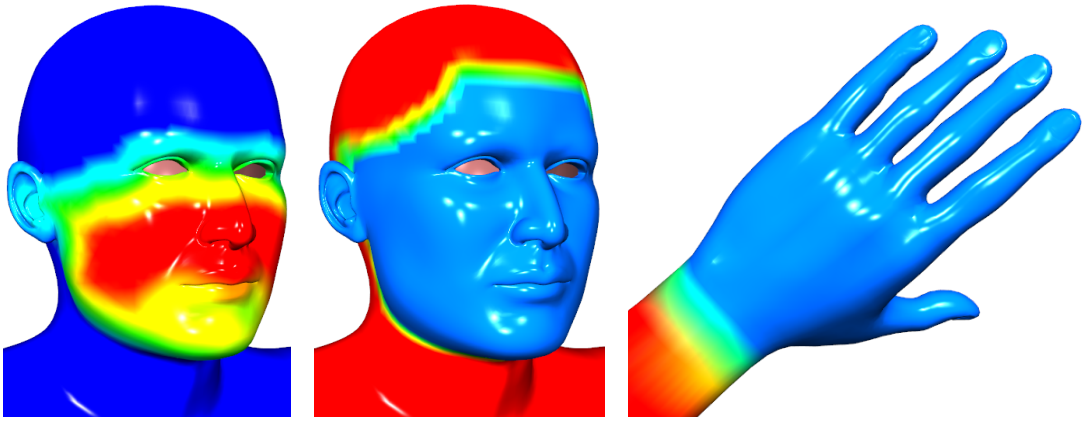
The *fitting term* E_{fit} penalizes the squared distance between corresponding points $\mathbf{x}_c$ and $\mathbf{p}_c$

$$E_{\text{fit}}(\mathcal{X}) = \frac{1}{\sum_{c \in \mathcal{C}} w_c} \sum_{c \in \mathcal{C}} w_c \|\mathbf{x}_c - \mathbf{p}_c\|^2 \,, \tag{4.3}$$

where $\mathcal{C}$ is the set of closest point correspondences and w_c are per-vertex weights as discussed below. The closest points $\mathbf{x}_c$ are expressed as barycentric combinations of the template vertices $\mathbf{x}_i$.

The *regularization term* E_{reg} penalizes the geometric distortion from the undeformed model $\bar{\mathcal{X}}$ (the result of the initialization phase of Section 4.4.1) to the deformed state $\mathcal{X}$, measured by the squared deviation of the per-edge Laplacians

$$E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) = \frac{1}{\sum_e A_e} \sum_{e \in \mathcal{E}} A_e \left\|\Delta^e \mathbf{x}(e) - \mathbf{R}_e \Delta^e \bar{\mathbf{x}}(e)\right\|^2 \,. \tag{4.4}$$



(a) (b) (c)

Figure 4.6: Per-vertex weights in the fitting energy allow to fit only the face region (a) or only the body (b), and to down-weight (typically poorly scanned) hands (c).

Here, A_e is the area associated to edge e, and $\mathbf{R}_e$ are per-edge rotations which cancel out local rigid transformations, such that the model can deal with large deformations. $\Delta^e\mathbf{x}(e)$ is the edge-based Laplacian for edge e and is defined as

$$\begin{aligned}\Delta^e\mathbf{x}(i) \;=\; &(\cot\gamma_{il} \;+\; \cot\gamma_{im})\,\mathbf{x}_s - (\cot\gamma_{ik} \;+\; \cot\gamma_{il})\,\mathbf{x}_p \;+\\ &(\cot\gamma_{ij} \;+\; \cot\gamma_{ik})\,\mathbf{x}_r - (\cot\gamma_{ij} \;+\; \cot\gamma_{im})\,\mathbf{x}_q,\end{aligned}$$

where $\mathbf{x}_*$ and γ_{i*} are as depicted in Figure 4.5 (more details in [AZB15]).

We prefer the edge-based Laplacian over the standard vertex-based Laplacian, since in our experiments it converges slightly faster to very similar results.

The three coefficients λ_{lm}, λ_{fit}, and λ_{reg} are used to guide the iterative fitting procedure, where the surface stiffness is controlled by λ_{reg}. In the beginning, only the manually specified (hence quite reliable) landmarks are taken into account, using $\lambda_{\text{reg}} = 1$, $\lambda_{\text{lm}} = 1$ and $\lambda_{\text{fit}} = 0$. We then gradually decrease λ_{reg} after each iteration until $\lambda_{\text{reg}} = 10^{-5}$. After these iterations, the template is sufficiently well aligned to yield reliable closest point correspondences. We therefore continue with $\lambda_{\text{reg}} = 10^{-5}$ and $\lambda_{\text{lm}} = 1$, but additionally set $\lambda_{\text{fit}} = 1$ to also consider E_{fit}. Then, both λ_{lm} and λ_{reg} are gradually decreased until $\lambda_{\text{reg}} = 10^{-9}$.

During the fitting procedure we weight down parts of the template using the per-vertex weights w_c in E_{fit} in order to prevent unreliably scanned regions from being fitted to strongly (see Figure 4.6). We weight down the hands, since they are usually not scanned well, and the face region to allow us to add more detail when combining with the face scan in Section 4.5.

The nonlinear objective function (4.1) is minimized by solving for vertex positions $\mathbf{x}_i$ and per-edge rotations $\mathbf{R}_e$ using alternating optimization (a.k.a. block coordinate descent) [BTP14, AZB15]. Figure 4.4(f) shows the final result of the body fitting procedure.

4.4.3 Texture Reconstruction

After the coarse scale initialization and the fine-scale non-rigid registration, the template has been accurately aligned and deformed to fit the point cloud of the body scan. We pass the deformed template model to Agisoft Photoscan Pro, which makes use of the existing texture layout from Autodesk's Character Generator and computes a high-quality 4k×4k texture based on the 40 camera images and their calibration data (see Figure 4.8(a)).
Since the camera images typically do not provide meaningful texture information for eyes and teeth, we use a pre-selected image mask to preserve the corresponding texture regions, i.e., to use eye and teeth texture from the generic template texture.
Due to occlusions and delicate geometric structures, scanning artifacts can easily occur for the fingers, which can result in an inaccurate template fit and then to misaligned textures for the fingers. We reconstruct a plausible hand texture by searching for the best-matching hand texture in Autodesk's Character Generator and using this hand texture instead. We identify the best-matching texture based on the Euclidean distance between RGB values of the back of both hands, the Autodesk texture and the one of the scanned subject (the latter is fitted reliable due to the manually selected landmark on the hands). Here, it turned out beneficial to distinguish between male and female hand textures. The found hand texture area is then seamlessly merged into the reconstructed full-body texture using Poisson image editing [PGB03].
Finally, the texture area below the armpits is typically corrupt as these are not sufficiently visible from our cameras. We smoothly fill these texture regions by harmonic color interpolation, which we compute by solving a sparse linear Laplace system with suitable Dirichlet color boundary constraints, similar in concept to Poisson image editing [PGB03].

4.4.4 Pose Normalization

Due to the non-rigid shape deformation the template's joints are not at their correct positions anymore. We again adjust the joint positions based on the precomputed mean value coordinates, this time representing the joint positions as a linear function of vertex positions. Employing mean value coordinates for this mapping ensures that joints are placed at meaningful positions even for strong shape deformations.
After mapping the skeleton to the deformed template (in scan pose), we use it to undo the pose fitting, i.e., to put the model into T-pose, as it is usually required by animation tools.

In particular for character animation via motion capturing this is an important step, since these systems usually rely on a standardized T-pose as initialization.
Finally, to make sure that both feet of the resulting character are standing exactly on the floor after pose-normalization, we first rigidly translate the model to put the (pre-selected) sole vertices onto the floor and then non-rigidly deform them onto the floor plane, while allowing only the feet to slightly deform, regularized by the Laplacian energy (4.4).

4.5 Face Reconstruction

After fitting the template model to the full-body scan $\mathcal{P}_B$, we now improve the geometry and texture of its facial region by fitting it to the face scan $\mathcal{P}_F$ and exploiting its eight close-up camera images. We closely follow the face reconstruction approach of [AZB15], but adjust it to the combined body-and-face reconstruction setup and extend it by blendshape reconstruction.

4.5.1 Initialization

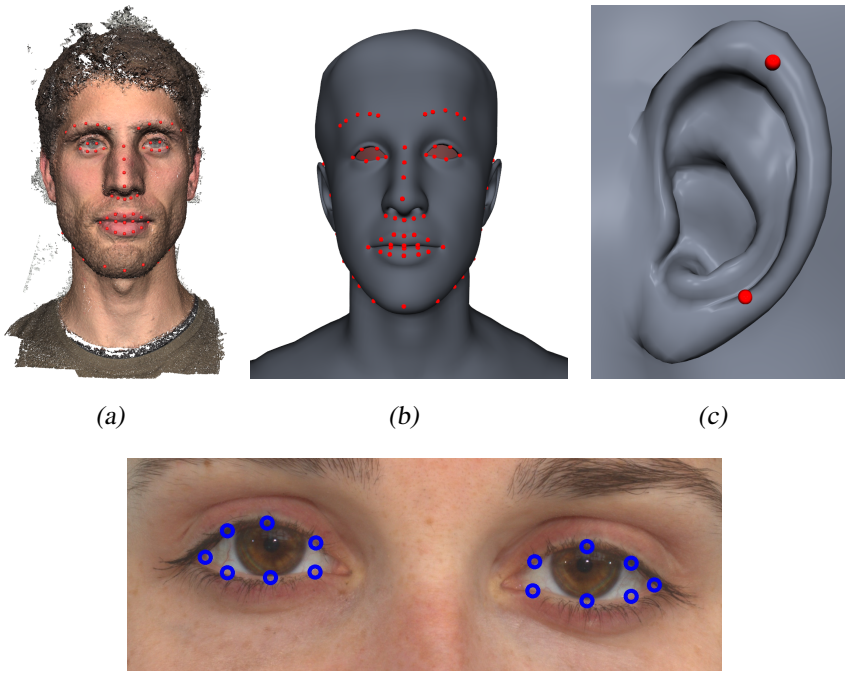
Since the face scan and the body scan are not aligned to each other, the template model is not aligned to the face scan either.
Following [AZB15], we automatically detect facial landmarks in the input camera images using [AZCP13], which are then mapped to 3D points in $\mathcal{P}_F$ using the camera calibration data (Figure 4.7(a)). We transform the whole face scan $\mathcal{P}_F$ to the template by finding optimal scale, rotation, and translation to minimize squared distances between the detected 3D facial landmarks and their (pre-selected) counterparts on the template model. Afterwards, we further refine scale, rotation, and translation by iteratively finding closest point correspondences and computing the optimal similarity transformation in the usual ICP manner [Hor87].

4.5.2 Deformable Registration

After the initialization the template model and the facial point set $\mathcal{P}_F$ are sufficiently well aligned to start the fine-scale non-rigid deformation. To this end we minimize the energy

$$\begin{aligned} E_{\text{face}}(\mathcal{X}) \; &= \lambda_{\text{lm}} E_{\text{lm}}(\mathcal{X}) + \lambda_{\text{fit}} E_{\text{fit}}(\mathcal{X}) + \\ &\quad \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) + \lambda_{\text{mouth}} E_{\text{mouth}}(\mathcal{X})\,. \end{aligned} \tag{4.5}$$

Here E_{fit} again represents closest point correspondences and is weighted by $\lambda_{\text{fit}} = 1$. We again employ per-vertex weighting in the fitting term E_{fit}, such that only the face and ear vertices are dragged toward the face scan (see Figure 4.6(a)).



(a) *(b)* *(c)*

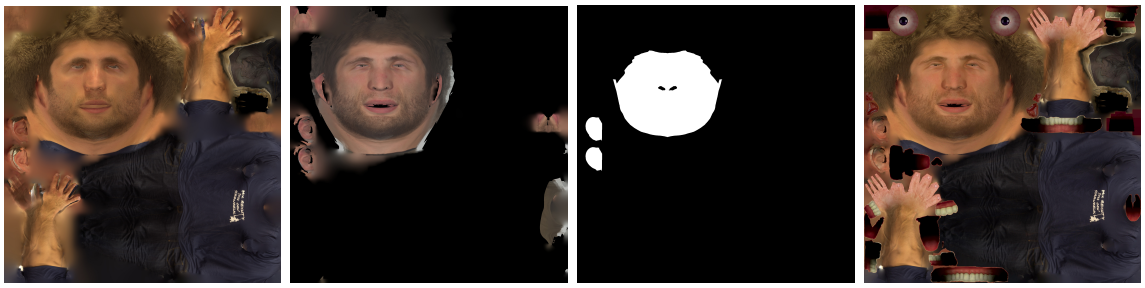
(d)

Figure 4.7: We automatically detect 66 facial features in the frontal 2D image and project them to our 3D point set (a). These facial features correspond to 66 pre-selected landmarks on the template mesh (b). To make sure the ears are properly fitted we select two landmarks on each ear (c). For proper eye lid reconstruction we select seven eye contour points for each eye (d).

E_{lm} represents a landmark term, weighted by $\lambda_{\text{lm}} = 1$, and includes three types of landmarks: Besides the automatically detected facial features (Figure 4.7(a)), we manually pick two landmarks on each ear to more precisely fit the ears (Figure 4.7(c)). Furthermore, we manually pick seven contour points for each eye in the frontal face picture and compute landmarks for eye lid reconstruction (Figure 4.7(d), see [AZB15] for more details).
The regularization term E_{reg} is the same as for the body fitting. It is initially weighted by $\lambda_{\text{reg}} = 1$ and is gradually decreased to $\lambda_{\text{reg}} = 10^{-9}$ during the iterative fitting procedure.
We observed that during fitting it is not guaranteed that the mouth stays closed, and therefore add an energy term preventing contour points on the upper/lower lip to diverge

$$E_{\text{mouth}}(\mathcal{X}) = \frac{1}{M}\sum_{i=1}^{M}\|\mathbf{x}_i^{(u)} - \mathbf{x}_i^{(l)}\|^2, \tag{4.6}$$

where $\left\{\mathbf{x}_i^{(u)}, \mathbf{x}_i^{(l)}\right\}$ are $M = 11$ pairs from upper and lower lip, respectively, which are pre-selected on the template mesh. This energy term is weighted by $\lambda_{\text{mouth}} = 0.5$.



(a) (b) (c) (d)

Figure 4.8: Textures computed from the camera images of the body scan (a) and the face scan (b). Since the face region is more accurately represented in the latter, it is extracted using a pre-computed image mask (c) and seamlessly copied into the body texture through Poisson image editing (d).

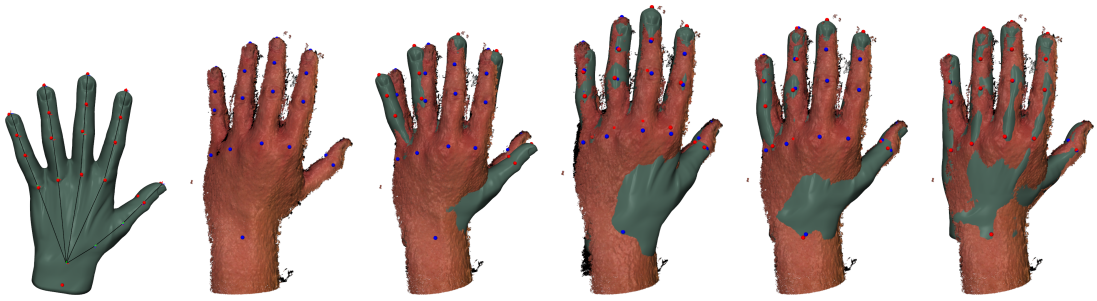
Note that at this stage we optimize the vertices of the head region only, while keeping all other vertices fixed by removing them from the linear systems. Analogous to the body fitting step we solve the nonlinear optimization using alternating optimization for vertex positions and edge rotations. Note that we do not employ the anisotropic bending model of [AZB15], since the template's face region is too coarse to benefit from the (computationally more expensive) anisotropic wrinkle reconstruction.

4.5.3 Facial Details and Blendshapes

Similar to [IBP15], we adjust the template's teeth by optimizing for anisotropic scaling, rotation, and translation, based on the deformation of the mouth region from the undeformed template to the deformed and fitted mesh. We also transform the eyes by optimizing for isotropic scaling, rotation, and translation for each eye individually, again based on the deformation of the individual eye region from the undeformed to the deformed mesh.

Face animation requires a suitable set of blendshapes, which represent the face in different expressions, typically consisting of the FACS blendshapes [EF78] and of visemes for speech animation. Since we only scan the actor in neutral facial expression, we have to "invent" a proper set of blendshapes. Since facial expression are similar across different individuals, we transfer all blendshapes from our generic template model to the fitted model using deformation transfer [SP04], similar to [WBLP11]. This transfers the deformation of each blendshape from the template model (generic neutral $\mapsto$ generic expression) to the target model.

Note that our blendshapes are rather generic, since they transfer the template's expression to the scanned person. In contrast, others reconstruct more personalized blendshape rigs by in-



(a) (b) (c) (d) (e) (f)

Figure 4.9: Hand fitting pipeline: we fit a template model (a) to the point cloud (b) by using landmarks (red and blue dots). For initial alignment and adaption of the posture we perform rigid ICP (c) alternating with inverse kinematic (d). Afterwards we fit the shape of the template model based on landmarks (e) and closest point correspondences (f).

corporating personalized expressions by video input [IBP15] or additional scans [LWP10]. Feng et al. [FRS17] also scan additional expressions and use those as (highly personalized) blendshapes, but they do not generate additional ones. Alternatively, Thies et al. [TZS+16a] use a sub-space deformation transfer technique using their parametric model to transfer person-specific facial expressions from a source to a target actor. A good compromise would be to add a small number of scanned example expressions to the deformation transfer process, as done by example-based facial rigging [LWP10]. However, all these approaches would increase the acquisition time.

4.5.4 Texture Reconstruction

Analogous to the body fitting step, we generate a 4k×4k texture from the eight camera images of the face scanning session using Agisoft Photoscan. This yields an accurate high-quality texture, but only for the face region, which we therefore extract using a pre-selected image mask and then seamlessly copy it into the full-body texture using Poisson image editing [PGB03] (see Figure 4.8). As mentioned before, we keep the texture for eyes and teeth from the original texture. The luminance of these regions are adjusted, such that their mean luminance coincides with the mean luminance of the face. This adapts the texture of teeth and eyes to the lighting conditions of the scan.

4.6 Reconstruction of Other Body Parts

Just like incorporating the separate face scan, we can add more detail to other regions by employing the same method. Therefore, we are able to include additional scans of other

Figure 4.10: Examples of characters which we are able to create with our pipeline. The resulting characters are ready to be animated through a skeletal rig and facial blendshapes, and are compatible with standard graphics and VR engines.

body parts, e.g., feet, hands, arms or legs. The only prerequisites are suitable per-vertex weightings for the template model (see Figure 4.6) and scanning rigs, which create more detailed scans of the specific regions.
For instance, the often erroneously scanned hands may be repaired by using per-vertex weights, like the ones we already have (compare Figure 4.6(c)), to prevent the hands from fitting the full-body scan's hands too closely. Then, after scanning the hands with a dedicated hand scanner, we can use the inverse of the per-vertex weighting to fit the hands only to the resulting point clouds of the hands.
In [SWM+17] we proposed a multi-view stereo hand scanner and a pipeline to fit a template model to point cloud data, to create personalized hand models (compare Figure 4.9). This method could be employed here as well by simply using the template characters hands as the template models and then compute and incorporate textures for the hands just like we did for the face region.
However, the overall pipeline would not be as fast and would demand additional user interaction, as the hand fitting requires the selection of additional landmarks.

4.7 Results

We tested our virtual human generation pipeline on a large set of subjects, and our approach reliably produced convincing results for all of them. A representative subset can be seen in Figure 4.10 and in the accompanying video.
The use of multi-view stereo reconstruction allows us to reconstruct both accurate geometries as well as high quality textures. As can be seen in Figure 4.11, additionally incorporating our dedicated face scanner significantly

Figure 4.11: Comparison of the face region reconstructed from the full-body scan only (left pair) and by additionally incorporating the dedicated face scanning (right pair).

improves the visual quality of the face region, since it was scanned at higher resolution. A comparison of a captured image from the body scanning session with the personalized virtual human is depicted in Figure 4.12.

Our reconstructed characters can readily be animated in any standard graphics or VR engine, since they feature a standard skeleton for full-body and hand animation as well as a standard set of blendshapes for face animation. The accompanying video demonstrates that our characters can efficiently be animated and rendered in a real-time scenario. Figure 4.13 and the accompanying video show one of our scanned characters used as a conversational virtual agent, where face and body animation are crucial to enable the agent to talk, perform gestures, and show facial expressions.

Process	**$\sim$ Time**
Face scanning	1/10 s
Image transfer face scanner	15 s
Full-body scanning	1/10 s
Images transfer body scanner	80 s
Compute face point set $\mathcal{P}_F$	15 s
Compute body point set $\mathcal{P}_B$	75 s
Manual landmarks selection	120 s
Facial feature detection	60 s
Fit face geometry	20 s
Fit body geometry	35 s
Compute face texture	45 s
Compute & merge body texture	100 s
Compute facial blendshapes	5 s
Overall	$\sim$ 10 min

Table 4.1: Time needed for the sub-processes of our pipeline.

Our method also has some limitations: Texture artifacts may still occur in regions that are not visible from more than one camera, as is the case for all photogrammetry approaches. The most critical areas are the armpits and the hands, but also the crotch and

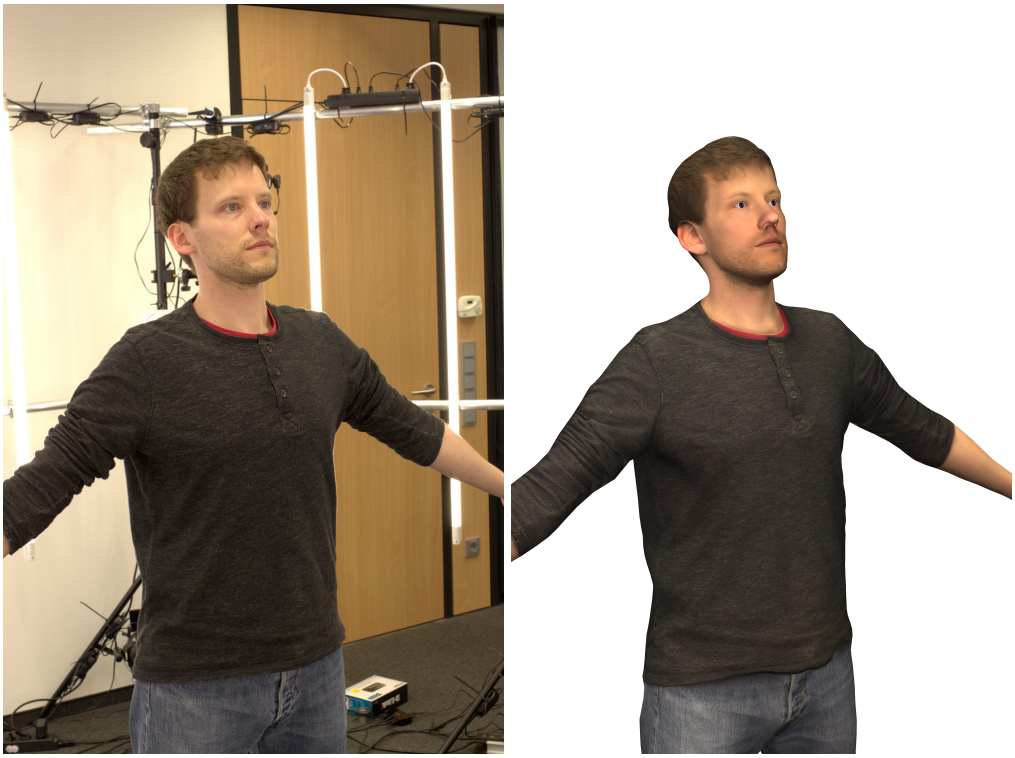


Figure 4.12: Comparison of a photo from the body scanning session with a rendering from the generated virtual human.

the inner parts of the arms can be problematic. These issues can be overcome by using more cameras, which will lead to a better coverage for texture data at the expense of longer computation times. Furthermore, we do not remove the scene lighting during scanning from the albedo textures, as done, e.g., in [BRLB14].

4.7.1 Performance

On average the processing of a single character takes about ten minutes from scan to a complete ready-to-animate avatar. See Table 4.1 for detailed information about computation times of our sub-processes, where the timings were taken on a desktop PC with Intel Xeon CPU (6×3.5 GHz) and a Nvidia GTX 980 GPU.

The computationally most expensive part of our template fitting procedure is the computation of the closest point correspondences in each fitting iteration. While this can be accelerated by using a kD-tree or a similar space partitioning technique, we found that a simple linear search implemented on the GPU provides a much higher speed-up for the model complexities in our application. In comparison to a CPU-based kD-tree, our straightforward GPU implementation of a brute-force search is about 12 times faster. A GPU-based

Figure 4.13: Our virtual humans can be directly used as expressive conversational agents, since they are able to gesture, talk, and to show facial expressions and emotions.

implementation of a spatial hierarchy would probably lead to an even higher speed-up, but would also require a considerably more complex implementation.

4.8 Conclusion

We presented a fast and reliable pipeline to digitally clone real persons into realistic virtual humans. For 3D-scanning we employ a custom-built camera rig with 40 cameras for the body and 8 cameras for the face, and compute dense point clouds through multi-view stereo reconstruction. In order to robustly deal with noise and missing data, we fit a generic human body model to the user's scanner data. By also transferring the skeleton, blendshapes, and eyes of the generic template to the model, our reconstructed virtual humans are ready-to-animate in standard game engines and VR frameworks.
Our character generation requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC. It is therefore fast enough to be performed at the beginning of each session in a VR experimental study.
While our pipeline produced convincing results with all tested subjects, some inherent limitations remain. Due to scanning subjects in A-pose, some areas are not visible from enough cameras and thus are not reconstructed well. While missing data can be compensated by template data during geometry reconstruction, these regions still suffer from texture artifacts.
As future work we plan on using the proposed pipeline to generate avatars for experiments with virtual mirrors like the one presented in Chapter 5. Further, we plan to employ our generated virtual humans in preference studies for personalized virtual agents. Another interesting direction for future work is the realistic modeling of clothing motion.

Up Next

This chapter explains in detail how to create realistic ready-to-animate virtual avatars of real people in less than 10 minutes.
Now, with this technique in hand we are able to investigate the impact of personalization and achieve interesting findings regarding the interaction with a personalized mirror image. Alongside with personalization we additionally investigate the impact of the immersion level on typical virtual embodiment effects in the virtual mirror scenario. The experiment and the results for both investigated factors are reported in the upcoming chapter.

5

Impact of Personalization and Immersion

5.1 Introduction

Our VR environment for motor learning introduced in Chapter 2 is an example for an embodied virtual environment: the user of the system is embodied as him-/herself in the virtual mirror. In such embodied virtual environments avatars are our embodied interfaces to and our proxy in the artificially generated environments. On the one hand, avatars provide a means of direct interaction with the environments based on the simulation of physical properties and cause an effect between virtual objects and the virtual bodies constituting the avatars in the virtual worlds. On the other hand, avatars are our proxies. They are the direct extension of ourselves into the virtual domain while they also constitute a close resemblance we only experience from our real physical bodies. That is, they are the digital representations tightly bound to our embodied self, our self-perception, and our personality.

As a result, avatar appearance, behavior, presentation, and control scheme cause a variety of psychophysical effects with users in control of the avatars as well as on other users sharing the same virtual worlds with our avatars. The acceptance of and identification with our virtual counterparts is called *illusion of virtual body ownership* (IVBO, see Section 3.2) [IdKH06, SPMESV08, LLL15a]. This identification can (temporarily) lead to a change of the user's behavior and self-image as described by the *Proteus* effect [YB07]. For example, the effect of avatar appearance on our behavior has been confirmed for a variety of properties including gender [SSSVB10], posture [DlPWL+10], figure [NGSS11], skin color [PSAS13], age and size [BGS13], or degree of realism and anthropomorphism [LLL15a, LLR16, RLG+16].

A typical method used for studying the psychophysical effects of avatars and their appearance and properties on the respective owning and controlling users is based on the virtual mirror metaphor. Virtual mirrors have been used and tested in fully immersive VR systems based on HMDs (e.g., [SSSVB10, SNB+14]) as well as in lesser immersive VR systems like our CAVE-based system for motor learning (see Chapter 2) or even in low immersive "fake mirror" displays [LLR16]. Notably, although the different virtual mirrors imply specific properties potentially affecting desired psychophysical effects, the impact of the degree of immersion has not been of particular interest so far.

Additionally, current advances in capturing individualized human bodies either by using depth cameras or photogrammetry methods like ours (see Chapter 4) or the approach

of Feng et al. [FRS17], motivated a closer look into the effects of realism and personalized avatars [LLL15b, LWBL15, RLG+16, LSG+16]. So far elaborate individualized ready-to-animate high quality virtual characters of users used to be a labor-intensive and time-consuming process, which only recently could be optimized to be applicable for prolonged and extensive embodiment studies by applying our approach or the one of Feng et al. [FRS17].
This chapter reports novel findings on two factors triggering or promoting embodiment effects in Virtual Reality based on human-like avatars. We investigated (1) the impact of avatar personalization and (2) the impact of the degree of immersion on virtual body ownership, presence, and emotional response as effects of embodied interfaces. The work combines recent advances in the optimization of a photogrammetry-based 3D scan process with two virtual mirror setups of different degrees of immersion. 32 participants could be tested with personalized avatars resembling their physical selves due to our optimized 3D scan workflow described in Chapter 4.
We found several significant and notable effects. First, personalized avatars significantly increase body ownership, presence, and dominance compared to generic counterparts, even if the latter were generated by the same photogrammetry process and hence could be valued as equal in terms of the degree of realism and graphical quality. Second, the degree of immersion significantly increases the body ownership, agency, as well as the feeling of presence.

My Contribution *The experiment was developed, conducted, and evaluated in cooperation with Dominik Gall, Daniel Roth, and Marc Erich Latoschik from the University of Würzburg. The experimental design was mainly developed by Dominik Gall, Daniel Roth, and Marc Erich Latoschik while I contributed ideas and evaluated possible designs. Further, I implemented the software and set up the hardware necessary for the experiment, which included the full-body scanning and the virtual mirror system. Additionally, I conducted the experiment. Finally, the creation of the questionnaires and the data analysis were done by Dominik Gall.*
Corresponding publication: The Impact of Avatar Personalization and Immersion on Virtual Body Ownership, Presence, and Emotional Response, IEEE VR 2018.

5.2 Related Work

Virtual embodiment describes the application of an artificial body as a virtual alter ego and proxy for the user's real physical body in an artificially generated Virtual Environment (VE). Initial work has been motivated by the classical rubber hand illusion [BC98].

Ijsselsteijn et al. [IdKH06] and Slater et al. [SPMESV08] confirmed BO to transfer to Virtual Reality and to artificially generated virtual worlds and stimuli. Similar to the real physical world BO, the respective illusion of virtual body ownership (IVBO) is triggered and promoted from artificial virtual stimuli. Most important, IVBO relies on (parts of) virtual bodies instead of physical replicas as visually perceivable anchors for IVBO to be effective. These virtual replicas are our avatars, our embodied interfaces in and to the artificially generated environments.

The IVBO promotes a variety of interesting psychophysical effects caused for the users controlling the avatars. Slater and Steed [SS00] confirmed that participants who had to interact with virtual objects through a virtual body had a higher sense of presence than those who interacted with a traditional user interface (pressing a button). Changing the visual and behavioral characteristic of a user's avatar will potentially also change the behavior [KBS13], attitude [PSAS13, BGS13], and emotional involvement [DlPWL$^+$10] of the user in control of the avatar. This *Proteus* effect [YB07] identifies a connection between our objective perception and a subjective interpretation and integration of the perceived information into our own cognitive models including expectations and preconception of role models. This effect has been explored for various dimensions, e.g., gender [SSSVB10], posture [DlPWL$^+$10], figure [NGSS11], skin color [PSAS13], age and size [BGS13], exertion [YS13], or degree of realism and anthropomorphism [LLL15a, LLR16, RLG$^+$16].

Similar to BO, IVBO is dependent on a convincing coherence between the real and the virtual body. For example, triggering the original rubber hand illusion relied on visuotactile stimulation of the real physical hand and the visual perception of the stimulus action performed on the artificial rubber proxy. This stimulation had to be synchronized in time and place to work effectively. Hence, the synchronized visuotactile stimulation acts like a promoter or even cause for inducing BO. Here, related work on IVBO and its promoters or triggers benefits from an extended design space: Virtual Reality technology allows to change virtual body appearance, behavior, and coverage with much less effort compared to physical setups where, e.g., the replication of a complete proxy body would only be possible with potentially complex and costly robotic tele-presence scenarios.

Related work on IVBO differentiates two types of relevant factors to promote or trigger the illusion: (1) bottom-up factors (e.g., synchronous visual, motor, and tactile sensory inputs) are thought to be related to the multi-sensory integration and (2) top-down factors (e.g., similarity of form and appearance) [TH05, LLL15b] are thought to be related to the conceptual interpretation of the observed virtual body parts.

Current results favor bottom-up factors such as first-person perspective, synchronous visuotactile stimulations, or synchronous visuomotor stimulations to be strong triggers for the IVBO effect [SNB$^+$14]. Sanchez-Vives et al. could induce IVBO by using just visuomo-

tor correlation without a visuotactile stimulus [SVSF+10]. These findings were confirmed by Kokkinara & Slater [KS14], although a disruption of visuotactile or visuomotor synchrony could equally lead to a break in the illusion. Debarba et al. [DMHB15] did not find differences between 1PP (first person perspective) and 3PP (third person perspective) and suggested that visuomotor synchrony dominates over perspective.

The impact of top-down factors, i.e., of anthropomorphism or realism, for IVBO is not as evident as the impact of the most important bottom-up factors. Lugrin et al. [LLL15b] found that IVBO even slightly decreased for avatars with a higher human resemblance compared to a robot and a cartoon-like figure. As one reason for this finding they hypothesized this to be caused by a potential uncanny valley effect [MMK12]. Latoschik et al. [LLR16] used a different low immersive setup, but included individualized avatar faces scanned with a 3D depth camera. They did not find any difference in IVBO.

Both types of factors, bottom-up as well as top-down factors, rely on the avatar's visibility to the user. For a 3PP this can easily be achieved with a variety of graphics and VR setups. As for the 1PP, this is not as straightforward. Fully immersive systems based on HMDs do block out potentially diverting stimuli from the real physical surrounding, i.e., the real physical body. Similarly, see-through Augmented Reality (AR) glasses can be used since they would also allow to graphically occlude the real physical body. But both would initially only allow to see parts of one's own body. These parts are mainly restricted to the hands and forearms and the front side of the torso and the legs. However, projection-based large-screen VR systems of a potentially lesser degree of immersion [Sla99], such as CAVEs [CNSD+92], L-shapes, power walls, and alike, by design cannot prevent visibility of users' real physical bodies at all when looking directly at themselves.

To overcome the virtual body visibility drawbacks caused by the different VR and AR systems, IVBO research usually applies a virtual mirror metaphor [BLB+02]. A virtual mirror works for most VR and AR display types, it allows to inspect almost the complete full avatar body including the face, while being a well-known everyday tool in the real world. Hence it fosters suspension of disbelief and does not result in breaks in presence. The virtual mirror metaphor has been used in fully immersive VR systems based on HMDs (e.g., [SSSVB10, SNB+14, LLL15b]) as well as in lesser immersive VR systems like CAVEs (see, e.g., Chapter 2), and even in low immersive "fake mirror" displays [LLR16]. Notably, although the displays used for these virtual mirrors significantly differ in the degree of immersion (as do some of the reported results from according studies), the potential impact of this factor on IVBO has not been investigated so far.

Virtual embodiment can cause a variety of interesting effects as has been confirmed by prior work. Potential applications of avatars include virtual therapy, entertainment, or social Virtual Reality [BBH+90, BRKE08, SS15, RWL+17, LRG+17] and many more. It would

be highly favorable to exactly know about the relevant triggers or promoters for IVBO and their respective relative effect strengths compared to each other. This would allow to, e.g., concentrate application design and development efforts on more important factors or to be able to manipulate and parameterize the to-be-caused target effects. The importance of visuomotor synchrony could repeatedly be confirmed. Findings for several other factors exist, but certainly would benefit from replication in different contexts. The context can have strong effects as could be shown in very recent work [LKK+18]. Relative effect strengths are only available for the apparently most important bottom-up factors.
Notably, we identified two factors which we would currently assess important, but whose impact on the IVBO and resulting embodiment effects is either missing or where the results are ambiguous or even contradictory at the moment. First, given the large variety of currently available VR and AR displays it is important to know the respective impact of the degree of immersion on the IVBO. Second, advances in capturing high quality 3D individualized human bodies by using photogrammetry methods enables a closer look into the impact of realism and personalized avatars as motivated by recent work [MS13, LLL15b, LWBL15, LSG+16, LLR16, LRG+17].
Unfortunately, until now, elaborated individualized high quality virtual characters of users, which are ready-to-animate, used to be a labor-intensive and time-consuming process, hence prior work either omitted personalized scans [RLG+16, LLL15b], reduced the scan quality (using only depth cameras [LLR16, LSG+16]), or reduced the scan coverage (only scanning, e.g., heads [LLR16]). The process to generate high quality 3D scans based on photogrammetry could just recently be optimized to be applicable for prolonged and extensive embodiment studies. For the work reported in this chapter we have utilized our avatar generation pipeline as described in Chapter 4.

5.3 Rationale, Hypotheses, and Design

As pointed out in the preceding discussion, we identified ambiguous and missing results on the impact of (1) avatar personalization and of (2) the degree of immersion on VBO. The exploration of said potential impact(s) defines the overall research goal for the work reported in this chapter. Hence, avatar personalization and degree of immersion define the two independent variables.
As potentially affected embodiment target effects we chose (E1) *virtual body ownership* (including agency), (E2) *presence*, and (E3) *emotional response* as our dependent variables. (E1) *virtual body ownership* is a frequently studied embodiment effect (see, e.g., [TH05, SVSF+10, SNB+14, LLL15b, DMHB15]) and hence is targeted here as the central embodiment effect for any potential impacts found. Similarly, (E2) the feeling of

Figure 5.1: The two VR setups used in the study. The participants were immersed in the same virtual room with a virtual mirror using an L-shape part of a CAVE (condition CM1, left) and a HMD (condition CM2, right). The participants wore a motion capture suit to track their full-body motion by a passive marker-based motion capturing system. Note that in the L-shape condition the participants had to wear 3D glasses for stereoscopic visualization (not shown in the image).

virtual *presence* is one of the most prominent psychophysical effects of Virtual Reality. Prior work on presence has reported that immersion [Sla99] as well as general embodiment [SS00, TNI15, JH16] do affect presence. Hence it is an appropriate measure to study the impact of personalization thought as a continuation of non-personalized general embodiment and to either confirm the reported impact of immersion or any unexpected deviation from prior results. Finally, we chose (E3) *emotional response*. Again, findings have reported emotion to be affected by immersion [BBA$^+$04, VTM10] as well as by embodiment [Nie07]. Hence, we chose *emotional response* as an additional dependent variable.
From the reported impact of general embodiment and immersion to increase presence and emotional response we bias our initial hypotheses as follows:

H1: Increased personalization increases the strengths of the target effects.

H2: Increased immersion increases the strengths of the target effects.

Figure 5.2: All twelve avatar types as used in the study. From left to right in 2×2 blocks: The synthetic generic female and male avatars created with Autodesk Character Generator [Aut14], the scanned non-individualized female and male avatars, and the personalized female and male avatars created from 3D scans of participants. The upper row represents the avatars as shown in the motion capture suit condition and the bottom row the avatars in the condition, in which participants were scanned in their own individual clothes. The face of the individualized female avatar is blurred for anonymization reasons.

For H1 we defined the independent variable *personalization* in terms of appearance similarity between a user's real physical body (including the face) and his/her avatar. We chose three conditions as levels for this variable:

CP1: Generic avatar created with Autodesk Character Generator [Aut14] (see left 2×2 block in Figure 5.2).

CP2: Generic avatar generated from 3D photogrammetry scan applying the technique from Chapter 4 (see center 2×2 block in Figure 5.2).

CP3: Individualized avatar generated from 3D photogrammetry scan applying the technique from Chapter 4 (see right 2×2 block in Figure 5.2).

As given for CP3 by design, avatars of the same sex as the respective participant were also chosen for CP1 and CP2.
Both visualization conditions, the low immersive L-shape (CM1) and the high immersive HMD (CM2), are depicted in Figure 5.1. For H2 we defined the independent variable *immersion* following the definition by Slater [Sla99] to mean *"the extent to which the actual*

system delivers a surrounding environment, one which shuts out sensations from the 'real world', which accommodates many sensory modalities, has rich representational capability, and so on".
The restriction of the L-shape part of the CAVE was purposely chosen to further limit the extend to which that system delivers a surrounding environment as a means to further reduce immersion of the L-shape condition in contrast to the HMD condition. As a result we chose two conditions (see Figure 5.1) as levels for this variable:

CM1: Less immersive medium, a two-screen L-shape part of a CAVE.

CM2: More immersive medium, a Head-Mounted-Display.

Notably, for both systems we used the same full body tracking system to provide a convincing visuomotor synchrony in addition to the same render engine to minimize potential confounds between systems. Also, we named this variable *medium* (and not only immersion) for the following reason: CM1 (L-shape) is inferior to CM2 (HMD) concerning the occlusion capability of the real body in the virtual mirror metaphor as described in the related work. In a CAVE-like environment, such as the employed L-shape, users will be able to see their own physical body when they look down on themselves. This is different for the HMD condition where participants will see just their artificial avatar when looking down on themselves.
Due to the required full body tracking, participants had to wear a tracking suit, which certainly would look different than their own cloth they were allowed to wear for the scan. This difference between the clothes worn during scanning and the motion capture suit worn during the trials could potentially impact our central hypotheses, which we investigated by including a third hypothesis:

H3: A difference in clothing between the mirrored avatar and the physical body negatively affects target effects.

Accordingly, we included *clothing* as an additional independent variable with the following two conditions as levels:

CC1: Participants scanned in their own clothes.

CC2: Participants scanned in motion capture suit.

CC1 and CC2 were tested between groups, whereas the six conditions resulting from the combination of CP1, CP2, and CP3 with CM1 and CM2 were tested randomized in-between subjects. An overview of the procedure is depicted in Figure 5.4, and a detailed description is given in the forthcoming sections. Figure 5.3 depicts an example of three conditions CP1-CC1, CP2-CC1, and CP3-CC1 as a combination of the three *personalization* CP1–CP3 levels with the CC1 condition as used in the experiment.

5.4 Apparatus

In our experiments the participants were immersed in a large, mostly empty room in order to minimize distraction (see Figures 5.1 left and 5.5). In this room there was a virtual mirror on one of the walls, which reflected the virtual world including the avatar of the participants. During the experiment the movements of the participants were tracked and mapped directly onto their avatar in real-time. In the following we describe the different devices and equipment used for the experiments.

5.4.1 Avatar Creation

In order to generate the individualized scanned avatars of the participants for condition CP3 we employed the avatar reconstruction pipeline as described in Chapter 4.
Since the whole scanning and avatar generation process takes only about ten minutes and requires minimal manual effort, it can easily and conveniently be used to scan participants right before the experiment. Using this approach, we were able to create convincing avatars of consistently high quality for all participants of our experiment.

5.4.2 Full-Body Motion Capturing

A convincing virtual mirror requires to robustly and rapidly capture the participants' motions and to map them onto their avatars in real time. We employed the motion capturing system as proposed in Section 2.5. Participants therefore had to wear a tight black

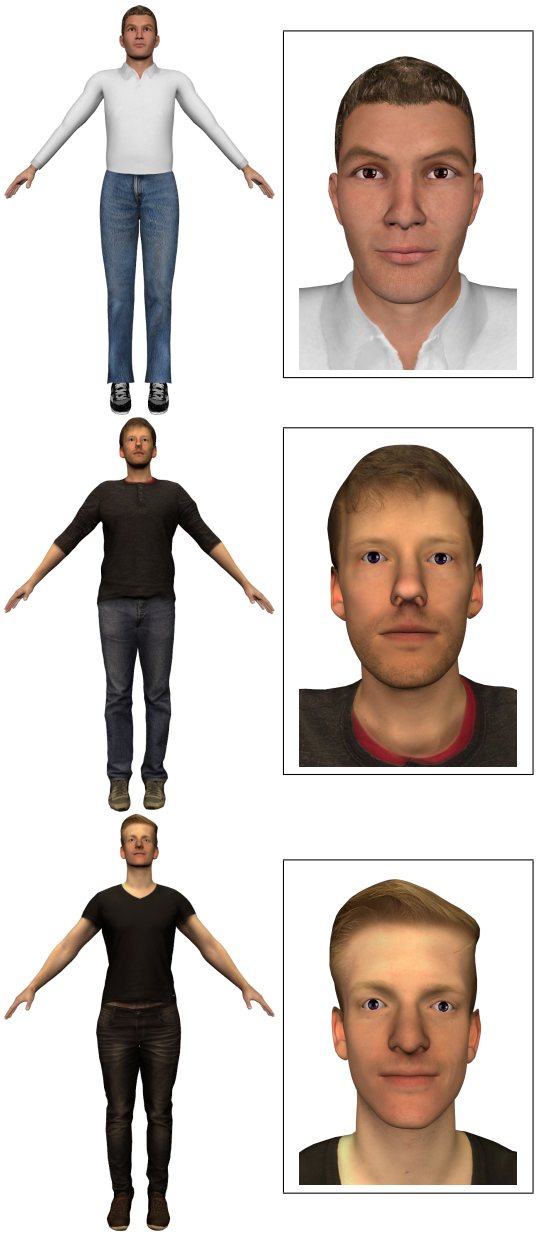


Figure 5.3: Example screenshots including face close-ups of the three different avatar types as used in the clothed (CC1) male condition: generic avatar (top), generic scanned avatar (middle), individualized scanned avatar (bottom) of the participant.

marker suit with overall 41 markers during the experiment (see Figure 5.1). Note that in the HMD condition the OptiTrack system was synchronized with the HMD's head tracking to avoid interference.

5.4.3 Visualization Systems

For the L-shape condition (CM1) we used our CAVE-based VR environment as introduced in Section 2.5. The HTC Vive was used for the HMD condition (CM2). It features a spatial resolution of 1080×1200 pixels per eye, provides a wide horizontal field of view of $110°$, has a refresh rate of 90 Hz, and a very robust and low-latency tracking of head position and orientation. The HMD was connected to a PC with Intel Xeon E5-1620 CPU with 4×3.5 GHz, 36 GB RAM, and a Nvidia GTX 1080 GPU, running Microsoft Windows 10. In order to minimize confounding factors, both systems were driven by the same custom-built render engine as proposed in Section 2.5. For the HMD condition, the HTC Vive was controlled using the OpenVR framework.
We again measured end-to-end motion-to-photon latency of the virtual mirror setup by following the technique described in Section 2.5. This time we used a high-speed camera with an even higher temporal resolution (Sony RX100 V, 500 Hz). For the L-shape setup we measured an average end-to-end latency of 62 ms using this technique.
We were not able to measure latency directly with this technique for the HMD, since the camera cannot record the real person and the HMD lenses simultaneously at a sufficient resolution. Instead, we recorded the real person and the desktop monitor showing the preview window of the HTC Vive. This measurement revealed an average latency of 67 ms. As HMDs are optimized for low latency, we expect a lower latency for the HMD itself. Note that these latency values are for the full-body tracking only. The head tracking of the HTC Vive is independent of the OptiTrack motion capturing.
With the reported end-to-end latency of 62 ms and 67 ms, respectively, both visualization setups performed below the critical thresholds for perceived sense of agency and sense of ownership as we reported for our virtual mirror experiment in Chapter 3.

5.5 Procedure and Stimulus

An overview of the overall experimental procedure is illustrated in Figure 5.4.

5.5.1 Participants

32 participants were recruited for this study. All performed preparation (including scanning) and the trials. Three of which had later to be excluded due to problems with data

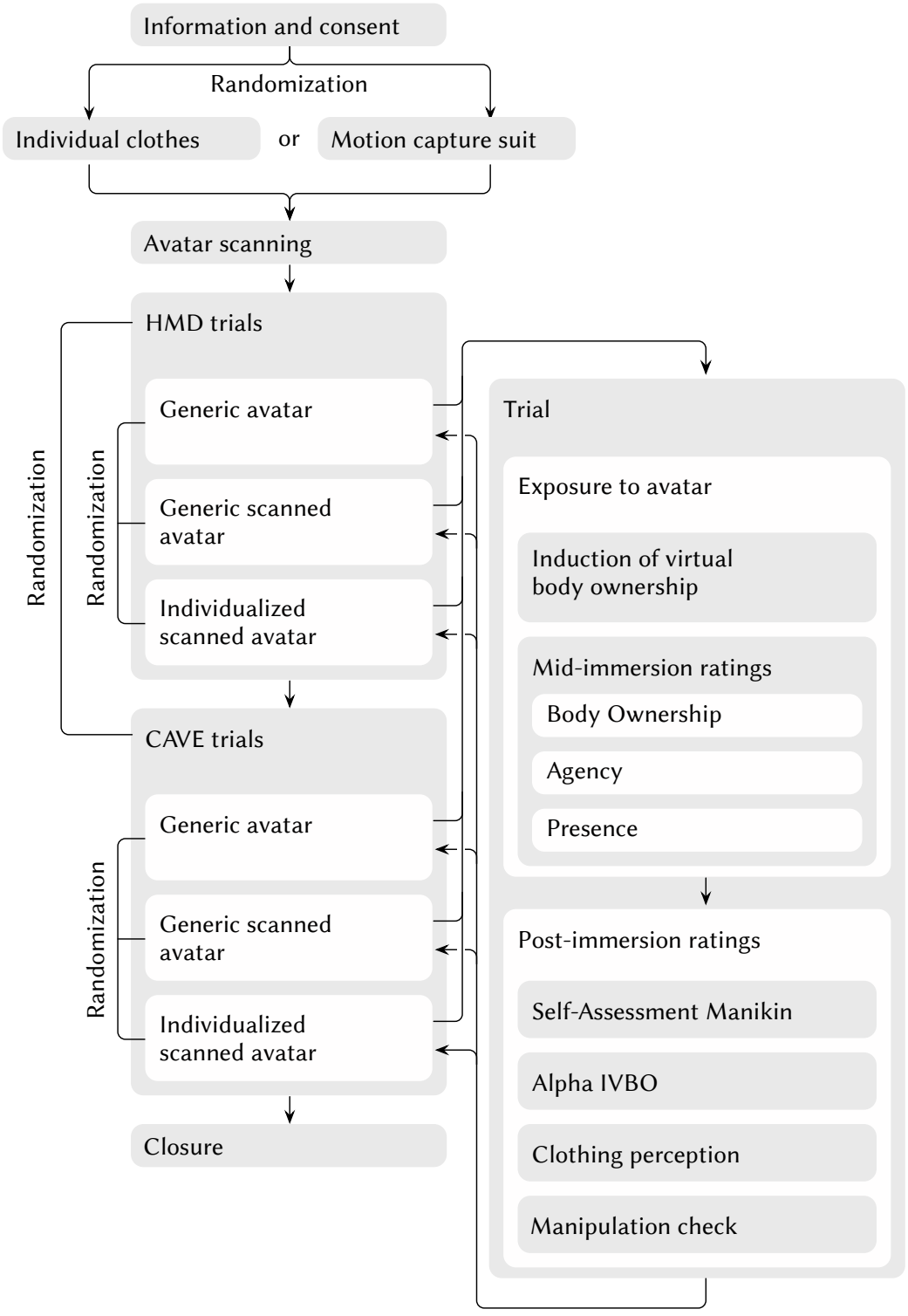


Figure 5.4: Illustration of the experimental procedure. For each participant, an avatar was generated either in individual clothes or with a motion capture suit. For each avatar appearance condition participants completed an experimental trial in the L-shape and HMD setup in randomized order. In a trial virtual body ownership was induced, then subjective mid and post immersion ratings were assessed.

recording. The final analyzed sample therefore consisted of 29 participants, 15 female and 14 male, with age ranging from 19 to 33 years ($M = 24$). None reported severe motor, auditive, or visual disabilities/disorders. All participants with visual impairment wore corrective glasses/lenses during the experiment.
All participants gave written informed consent and got paid for their participation. The study was conducted in accordance with the Declaration of Helsinki, and had ethical approval from the ethics committee of Bielefeld University.

5.5.2 Preparation

Participants first read general information about the devices and techniques used in the experiment and afterwards filled in and signed the consent form.
Depending on the clothing condition, participants then either got scanned in their own clothes (CC1) or put on the motion capture suit (without markers attached) and were scanned wearing the suit (CC2). We randomized this condition so that we scanned half of the participants in their own individual clothes and the other half in the motion capture suit. After the scan the participant's height was measured to scale the avatars of all conditions (CP1–CP3) to the correct height.
While the participants' avatars were computed, they filled in demographic and simulator sickness questionnaires, and put on the motion capture suit if they did not wear it already. Subsequently, the retro-reflective markers were attached, mostly to the motion capture suit, but some markers were also glued directly onto the skin to enable a more precise skeleton tracking (see Figure 5.1).

5.5.3 Experiment

After the initial preparation phase participants read the instructions of the main part of the experiment. Among other information, which is laid out in the following paragraphs, they were given the definition of presence in these instructions. Additionally, they were explicitly instructed to relax their hands as well as their face to minimize the effects of absent hand and face tracking.
The main part of the experiment took place in the same area of the L-shape for both *media* conditions: L-shape as well as HMD as illustrated in Figure 5.1. Each trial consisted of six conditions per participant: three personalization conditions (CP1–CP3) and two immersion conditions (CM1, CM2). Participants either started with the L-shape and continued with the HMD or vice-versa in a randomized manner. Accordingly they performed the three personalization conditions in randomized order for each *media* condition. Figure 5.4 illustrates this procedure. In the clothing condition CC2, where participants were scanned in

Figure 5.5: The virtual environment used for the experiment shown for the two stages of the trials: Before each trial the virtual mirror was turned off (left) and turned on as soon as the trial begun (right).

the motion capture suit, all avatars also wore the motion capture suit (top row in Figure 5.2) to factor out possible biases due to different clothes of the non-individualized avatars.
The virtual mirror was turned off before the trial to control the exposure time of the stimulus. The mirror was turned on and the avatar was shown to the participants as soon as the trial started. Both stages are illustrated in Figure 5.5. Subsequently, audio-instructions were played via loudspeaker. These instructions informed participants about which movements to perform and where to look at during the trial.
The movement-related audio-instructions were:

1. *"Lift your right arm and wave to your mirror image in a relaxed way."*
2. *"Now wave with your other hand."*
3. *"Now walk in place and lift your knees as high as your hips."*
4. *"Now stretch out both arms to the front and perform circular movements."*
5. *"Now stretch out your right arm to the side and perform circular movements."*
6. *"Now stretch out your left arm to the side and perform circular movements."*

Each of the given movement instruction was followed by an instruction to look back and forth at the movement in the mirror and at the own body (*"Look at the movement in the mirror – on your own body – in the mirror – on your own body."*). This approach served

two purposes: (a) all participants performed the same movements, and (b) participants were asked to constantly register the coherences between their body seen from 1PP body and their mirrored avatar to maximize potentially induced IVBO, specifically the visuo-motor synchrony between their movements. Depending on the immersion-related media conditions CM1 (L-shape) or CM2 (HMD) participants either saw their physical (CM1) or virtual (CM2) body from 1PP. Hence, while this was aiming at maximizing IVBO by taking advantage of a strong bottom-up IVBO trigger, it theoretically could also have negatively impacted IVBO in the CC1 condition where participants had been scanned in their own clothes. Hence, the rationale for the additional in-between groups factor *clothing*.
After the described instructions and while participants were still immersed in the virtual environment, they were asked the mid-immersion questions related to body ownership, agency, and presence. See upcoming Section 5.5.4 on Measures. Finally, the virtual mirror was turned off, and participants were asked to take off the 3D glasses or the HMD and to leave the area of the L-shape.
Following each trial (evaluating a particular avatar in a particular visualization setup) participants filled in the respective questionnaires for our dependent variables on a desktop computer. The next section gives the complete list of the measurements taken during and after the trials.
After all trials were done, participants took off the motion capture suit and got compensated for their participation.

5.5.4 Measures

While participants were still immersed in the virtual environment we took mid-immersion one-item measurements aiming at body ownership, agency, and presence. To this end, participants were asked to answer the following questions spontaneously on a scale from 0 (not at all) to 10 (totally):

1. Subjective body ownership, adapted from [KE12]: *"To what extent do you have the feeling as if the virtual body is your body?"*

2. Subjective agency, adapted from [KE12]: *"To what extent do you have the feeling that the virtual body moves just like you want it to, as if it is obeying your will?"*

3. Subjective presence, as proposed in [BSJRR08]: *"To what extent do you feel present in the virtual environment right now?"*

All participants were told in the instructions that *"Presence is defined as the subjective impression of really being there in the virtual environment."* These subjective one-item

measurements taken during immersion are accepted to have high sensitivity and reliability [Sla99, FAPI99, BSJRR08].

Self-Assessment Manikin (SAM) scales [BL94] were used for non-verbal pictorial assessment of self-reported affective experience directly after exposure to the virtual environment. This measure assumes the conceptualization of emotion as three independent dimensional-bipolar factors valence, arousal, and dominance. Validity and reliability of the SAM scales are confirmed [BL94] and have been supported by numerous studies [MR09]. In the underlying model, valence indicates the degree of pleasure or displeasure that the participant experiences during exposure. Arousal represents the experienced degree of physiological activation, whereas dominance signifies the perceived state of own social dominance or submission.

After exposure to the virtual environment and the virtual avatar the subjective sensation of virtual body ownership was assessed with the Alpha-IVBO scale [RLLH17], consisting of the three sub-scales acceptance, control, and change as dimensions linked to the virtual body ownership. The acceptance component refers to accepting the virtual body as the own body (e.g. "I felt as if the body I saw in the virtual mirror might be my body.", "The virtual body I saw was humanlike.", "I felt as if the body parts I looked upon were my body parts."). The control component relates to the concept of agency (e.g. "The movements I saw in the virtual mirror seemed to be my own movements", "I felt as if I was controlling the movement I saw in the virtual mirror"). The change component reflects changes in self-perception (e.g. "At a time during the experiment I felt as if my real body changed in its shape, and/or texture.", "I felt an after-effect as if my body had become lighter/heavier.", "During or after the task, I felt the need to check whether my body still looks like I remember it."), see [RLLH17] for the original questionnaire. The question order was randomized. Participants were asked to indicate spontaneously and intuitively how much they agree to each statement in a 7-point Likert style response format (0 – strongly disagree, 3 – neither agree or disagree, 6 – strongly agree), i.e., higher values would indicate a stronger illusion regarding each sub-scale. Cronbach's αs calculated for each within-factor measure (including both between-factor conditions) ranged between 6.79 and 9.34.

To determine perceptual changes in relation to the clothing manipulation, participants were asked

- *"To what extend did you have the feeling to wear different clothing from the clothes you were actually wearing?"*

on a scale of 0 (not at all) to 10 (totally), adapted from [SSSVB10].

Scale	Personalization†	Medium†	Clothing†
Mid-immersion Body Ownership	*** (.50)	*** (.40)	
Mid-immersion Agency		.010 (.22)	
Mid-immersion Presence	.002 (.21)	*** (.54)	
SAM Valence			.037 (.15)
SAM Dominance	*** (.27)		
IVBO Acceptance	*** (.48)	.001 (.35)	
IVBO Change		*** (.41)	
Clothing Perception	.023 (.14)		*** (.41)
Manipulation Check: Similarity	*** (.67)	.034 (.16)	

Note. † p (η_p^2); *** $p < .001$;

Table 5.1: Univariate main effects.

In order to assess if the personalization manipulation had been successful, participants were asked

- *"To what extent did you have the feeling that the virtual body was similar to your own?"*

on a scale of 0 (not at all) to 10 (totally).

5.6 Results

Each scale was analyzed by separately applying a 3-factorial mixed-design analysis of variance (split-plot ANOVA) with the within-factors *immersion/medium* and *personalization* and the between-factor *clothing*. When necessary, Huynh-Feldt corrections of degrees of freedom were applied. Post-hoc comparisons were realized using pairwise t-tests. A priori significance level was set at $p < .05$, two-tailed. Partial η^2 (η_p^2) is reported as a measure of effect size.

Scale	HMD†	L-shape†	p
Mid-immersion Body Ownership [a]	5.00 (± .31)	4.66 (± .41)	***
Mid-immersion Agency [a]	8.13 (± .27)	7.75 (± .27)	.010
Mid-immersion Presence [a]	6.77 (± .30)	4.56 (± .45)	***
IVBO Acceptance [c]	3.61 (± .21)	2.92 (± .23)	.001
IVBO Change [c]	1.76 (± .23)	1.23 (± .23)	***
Manipulation Check: Similarity [a]	4.80 (± .30)	4.26 (± .32)	.034

Note. † *Mean [± standard error of the mean (SEM)];* *** $p < .001$*;*

Likert scale range from low to high: [a] *0 – 10,* [c] *0 – 6;*

Table 5.2: Marginal means for the within-factor medium.

5.6.1 Medium

The univariate analysis showed significant main effects of the within-factor medium (HMD, L-shape) for the mid-immersion scales body ownership ($F_{1,27} = 17.66$, $p < .010$, $\eta_p^2 = .40$), agency ($F_{1,27} = 7.71$, $p = .010$, $\eta_p^2 = .22$), and presence ($F_{1,27} = 32.04$, $p < .001$, $\eta_p^2 = .54$). Here, we further observed significant main effects for the post-immersion Alpha-IVBO sub-scales acceptance ($F_{1,27} = 14.57$, $p = .001$, $\eta_p^2 = .35$) and change ($F_{1,27} = 18.78$, $p < .001$, $\eta_p^2 = .41$).

5.6.2 Personalization

Significant main effects of the within-factor personalization were found for the mid-immersion scales body ownership ($F_{2,54} = 27.43$, $p < .001$, $\eta_p^2 = .50$) and presence ($F_{2,54} = 32.04$, $p = .001$, $\eta_p^2 = .21$), as well as for the post-immersion scales SAM dominance ($F_{2,54} = 9.98$, $p < .001$, $\eta_p^2 = .27$) and the Alpha-IVBO sub-scale acceptance ($F_{2,54} = 25.16$, $p < .001$, $\eta_p^2 = .48$).

5.6.3 Clothing

For the between-factor clothing, a significant main effect for the scale SAM valence was found ($F_{1,27} = 4.80$, $p = .037$, $\eta_p^2 = .15$). The perception scale for clothing showed a

significant main effect for the between-factor clothing ($F_{1,27} = 18.83$, $p < .001$, $\eta_p^2 = .41$) and for the within-factor personalization ($F_{1.64,44.25} = 4.45$, $p = .023$, Huynh-Feldt-$\epsilon = .82$, $\eta_p^2 = .14$).

Scale	Motion capture suit†	Individual clothes†	p
SAM Valence [b]	6.63 (± .32)	7.56 (± .29)	.037
Clothing Perception [a]	1.96 (± .54)	5.10 (± .49)	***

Note. † *Mean (± SEM);* *** $p < .001$*;*

Likert scale range from low to high: [a] *0 – 10,* [b] *1 – 9;*

Table 5.3: Marginal means for the between-factor clothing.

The manipulation check scale for similarity showed a significant main effect for the within-factors personalization ($F_{2,54} = 55.45$, $p < .001$, $\eta_p^2 = .67$) and medium ($F_{1,27} = 5.00$, $p = .034$, $\eta_p^2 = .16$).
An overview of significant main effects and effect sizes is given in Table 5.1. Marginal means for significant main effects are listed in Table 5.2 for the within-factor medium, in Table 5.4 for the within-factor personalization, and in Table 5.3 for the between-factor clothing.

5.7 Discussion

H1: Personalization Impact

H1 assumed that increased personalization increases the strengths of the target effects. This could be confirmed particularly for the IVBO sub-scale *Acceptance* and was also strengthened by the mid-immersion BO results. Personalization also had a notable impact on increasing presence which, on the one hand confirms the known impact of general embodiment on presence, e.g., from [SS00, TNI15, JH16], but it also adds a novel finding concerning the specific appearance of the respective avatars. Finally, personalization also had a significant impact on increasing *SAM Dominance*. Hence, in general we found increased personalization to trigger a notable and significant increase of the strengths of the target effects for all three dependent variables (E1) body ownership, (E2) presence, and

Scale	(1) Generic hand-modeled avatar†	(2) Generic scanned avatar†	(3) Individualized scanned avatar†	p (1 to 2)§	p (1 to 3)§	p (2 to 3)§
Mid-immersion Body Ownership	4.42 (± .42)	4.75 (± .38)	6.82 (± .35)		***	***
Mid-immersion Presence	5.28 (± .35)	5.58 (± .36)	6.14 (± .34)		.002	.015
SAM Dominance	6.62 (± .28)	6.79 (± .26)	7.35 (± .24)		***	.004
IVBO Acceptance	2.66 (± .23)	3.11 (± .23)	4.02 (± .22)	.033	***	***
Clothing Perception	3.802 (± .49)	4.01 (± .43)	2.80 (± .41)			.016
Manipulation Check: Similarity	2.92 (± .45)	3.11 (± .46)	7.56 (± .30)		***	***

Note. † *Mean (± SEM);* § *pairwise comparison of indicated levels;* *** $p < .001$*;*

Likert scale range from low to high: [a] *0 – 10,* [b] *1 – 9,* [c] *0 – 6;*

Table 5.4: Marginal means for the within-factor personalization.

(E3) emotional response. The comparison of the marginal means also supports personalization to be the relevant factor here, since the main differences were recorded between the personalized avatar and the other two conditions and not between the other two non-personalized conditions alone.
Personalization did not affect all sub-scales in the respective measures, but it did have an impact on the measures thought to potentially be correlated to the participants' self-perception and identity, i.e., *SAM Dominance* and *IVBO Acceptance*. Our mid-immersion one-item presence measure did not include any sub-scales but was affected as a whole. Consistently, personalization did not have an impact neither on mid-immersion *Agency* nor on *IVBO Control*. Both can be thought to be much more affected by bottom-up visuomotor synchrony. This result is in line with the general assumption that the identification of similarity is a separate top-down factor for triggering the IVBO. The manipulation check for *Similarity* also confirmed that the personalized avatars were significantly identified to have a stronger resemblance to the respective participants. This validates the overall assumption that the scanned avatars do increase the resemblance to the participants' physical selves and it also confirms the quality of our apparatus and the applied scanning method.

H2: Immersion Impact

H2 assumed that increased immersion increases the strengths of the target effects. This hypothesis could partly be confirmed. We could identify an amplification impact particularly for the IVBO sub-scales *Acceptance* and *Change* and for all mid-immersion measures for IVBO, agency, and presence. The medium also revealed an impact on the similarity check, which is in line with and closely related to the IVBO *Acceptance*. The comparison of the marginal means between the HMD and the projection-based L-shape confirms the impact as to increase between the CM1 condition (the L-shape) thought to be of lesser immersion and the CM2 condition of higher immersion (the HMD) which is in line with the existing results on the impact of immersion on presence as expected from [Sla99].
The significant result for the *IVBO Change* sub-scale does support an impact that lately was indicated for a similar very low immersive virtual mirror [LLR16]. Since the *Change* factor seems to be an important factor for the Proteus effect, applications which rely on the latter could benefit from higher immersion. In contrast to the related work that did not find a difference of IVBO between 1PP and 3PP [DMHB15], degree of immersion does have an impact although one could assume a 3PP to, in general, have a lower immersion compared to a 1st person perspective. This is an interesting contradiction seeking for an explanation. Also, our media conditions did not reveal any impact of immersion on emotional response as potentially could have been expected [BBA$^+$04, VTM10]. An explanation for this might be twofold. On the one hand, our choice of task purposely did reduce any distracting addi-

tional stimuli as much as possible and was focusing on uniform body movements and their perception. Specifically, we tried to avoid power posing and alike. Hence, the identified impact of the personalization factor might have overshadowed any impact of immersion.

H3: Clothing Impact

H3 assumed that a difference in clothing between the mirrored avatar and the physical body negatively affects target effects. Besides the significant effect on the respective control measure (see below), there only was a minor effect of this factor on *Valence*, which could be attributed to a certain extent to participants to feel uncomfortable when asked to wear different clothes. It should be noted that a motion capture suit most certainly is inferior in both personally preferred comfort and appearance/look. The marginal means support the preference for the individual clothes. There was a significant impact of the clothing on the clothing perception as the according control measure, which confirms this factor to be clearly noticeable and hence potentially effective. But besides the minor effect on *Valence*, which in turn was not affected by any other of our independent variables, we could reject H3 and hence rule-out any suspected impact induced by the varying occlusion capability of the real body in the virtual mirror metaphor between CM1 (the L-shape) and CM2 (the HMD).

5.7.1 Limitations

We chose to induce the illusion in a most controlled way in order to prevent any third variable bias. The overall task to inspect one's own real/virtual body from 1PP and the respective reflection in the virtual mirror purposely aimed to avoid additional confounds from a more complex game application context, which was lately identified to potentially interfere with bottom-up factors for IVBO [LKK+18]. We suggest to carefully investigate potential generalization in cases of more complex stimuli such as immersive video games.

Also, facial expression is an important channel for social signals and as such is very prone to the detection of derivations and hence the potential provocation of eeriness. Nevertheless, in contrast to [LLR16], we had to avoid facial expressions due to the unreliable face detection in our setup of the HMD condition. Alternative sensing methods might overcome this problem in future work [TZS+16b]. Finally, with average end-to-end latency of 62 ms and 67 ms we also had to restrict movements to medium speeds and accelerations in order to not break bottom-up visuomotor synchrony.

5.8 Conclusion

In this chapter we reported novel findings on (1) the impact of avatar personalization and (2) the impact of the degree of immersion on virtual body ownership, presence, and emotional response as potential effects of embodied interfaces. We employed our developed 3D-scanning pipeline based on photogrammetry and template fitting which allows to capture personalized avatars in short time (about 10 minutes each) right before the experiment. This apparatus was used for the first time in a self-perception user study combining effects of personalization and immersion. In particular, this study greatly benefited from the optimized scanning and reconstruction process. So far, the generation of high quality personalized avatars was a labor-intensive process which required a lot of manual intervention and hence rendered similar undertakings time-consuming and costly. Hence, the impact of avatar personalization on body ownership, presence, and emotional response was – to our best knowledge – not known so far.
We found several significant and notable effects. First, personalized avatars significantly increase virtual body ownership, virtual presence, and dominance compared to generic counterparts, even if the latter were generated by the same photogrammetry process and hence could be valued as equal in terms of the degree of realism and graphical quality. Second, the degree of immersion significantly increases virtual body ownership and agency, and we could confirm its impact on the feeling of presence. As such, our findings add two important factors that impact the IVBO and virtual presence to the existing body of knowledge and they additionally contribute insight into the influence of our investigated factors on certain emotional responses.

5.8.1 Future Work

Given the number of confirmed factors which trigger IVBO there still are many open questions as to their mutual strengths and importance. Ideally, we imagine a correlation matrix that assigns every pair of impact factors some ordinal relation, hence a need for replicable studies filling this matrix, which in turn would greatly help developers to decide on which aspects to concentrate resources if the goal is a manipulation of embodiment effects. This matrix is a global goal would be an interesting endeavor for future work. Accordingly, this goal would include a closer examination of the relation between personalization, immersion (including perspective), emotion, and context, as motivated by the open questions resulting from the work reported in this chapter.
Related work gave some indication that an uncanny valley effect might also exist for avatars. We did not encounter such an effect here, but we also can neither confirm nor deny the existence of such an uncanny valley. We have not measured eeriness in our study

and cannot substantiate any claims about our avatars and where and on which side of the (potential) valley they would reside in terms of eeriness. We are planning to tackle this question in future work and are planning to include face tracking into the study design as motivated by [LLR16].

6

Conclusion

This chapter concludes this thesis by providing a short summary of all chapters and their results. Further, limitations and possible future directions are discussed.

6.1 Summary

This thesis thoroughly investigates how to create a virtual mirror for motor learning and embodiment experiments in VR. First, a basic system, which fulfills certain requirements with regard to motor learning in VR, was developed. Then, based on this system, important factors for the virtual mirror in general and motor learning in particular were empirically investigated. These factors are end-to-end latency, avatar personalization and level of immersion.

In order to build the presented virtual mirror system, hard- and software was evaluated with respect to requirements for motor learning in VR. It was argued that a CAVE environment is the proper choice as the visualization device for motor learning in VR. Further, marker-based motion tracking systems proved to be more robust, reliable and have lower latency compared to current marker-less systems and thus are the right choice for the virtual mirror system. Additionally, regarding software components this thesis proved that the used custom-build render engine running on a single PC has the edge over the usually employed cluster-based CAVE rendering approaches with regard to end-to-end latency.

With the virtual mirror system featuring low latency, we were able to investigate the impact of latency on motor performance as well as the perception of simultaneity, body ownership and agency. Here, results show that motor performance and simultaneity perception significantly drop at 75 ms, while body ownership and agency are still well present even for higher latency as they only start to significantly decrease for latency higher than 125 ms. The latter two are actually still quite present for latency as high as 350 ms. However, even our lowest latency used for the experiment of 45 ms was perceived as non-simultaneous in some trials of the experiment, which might indicate that an even lower latency is desirable for virtual mirrors.

In order to investigate the effects of the virtual mirror actually reflecting the appearance of the user, a technique to create avatars from real people in a short amount of time was necessary. Therefor, we introduced a 3D scanning pipeline, which creates a holistic ready-to-animate virtual clone of a real person at a decent quality level in less than 10 minutes

involving only minimal and simple manual effort. This is fast and simple enough to create avatars of participants right before an experiment. Additionally to having a skeleton as well as a blendshape rig to fully animate the body and face of the avatar, the so created virtual characters feature high quality in geometry as well as texture. Therefore, experiments, which need ready-to-animate personalized characters and a larger number of participants, are feasible by using our pipeline.

With the ability to create these avatars in hand, a follow-up experiment was conducted to investigate how avatar personalization influences the most interesting target effects: body ownership, agency, presence and emotional response. The results show, that all target effects are increased with the use of personalized avatars. Thus, the experiment substantiates the use of such personalized avatars for virtual mirrors and other applications where strong effects of virtual embodiment are desired. Further, the experiment outcome confirmed the benefit of the applied scanning pipeline's results as well as its applicability for virtual embodiment experiments.

Moreover, it was shown that immersion has a significant impact on VR- and embodiment-related effects (body ownership, agency, and presence) and that these are increased with higher immersion. Hence, even if the more immersive HMD is not suitable for motor learning in VR, it is the device of choice when stronger presence as well as more sense of body ownership is desired.

Summarized, this thesis thoroughly expounds how to create a virtual mirror which features a personalized mirror image with a low amount of latency. Further, it empirically demonstrates that the factors, latency, personalization, and immersion, are substantial to generate a virtual mirror image, so that it is accepted well by users as *their* mirror image. Concluding, the virtual mirror introduced in this work is suitable to provide a sophisticated and realistic virtual mirror image, which is empirically proven to provide the desired features (low latency and personalization). Therefore, the so created virtual mirror is very well applicable to use it for motor learning in VR and experiments in this field or experiments regarding virtual embodiment.

Finally, in the introduction of this thesis we ask whether a virtual mirror is accepted by us humans despite of its unavoidable non-realistic properties of having a comparably high latency and a non-personalized mirror image.

Now we can answer this question:

- Yes, we accept the virtual mirror image despite of its much higher latency compared to a real mirror, but with some restrictions: Too high latency leads to significantly worse motor performance and perceivable delay ($>$75 ms) as well as to significantly less sense of body ownership and sense of agency ($>$125 ms).

- Yes, we accept the virtual mirror even with a virtual mirror avatar which looks completely different from us (anthropomorphism provided). But by creating personalized avatars from 3D scans and testing these against generic non-personalized ones, we were able to show that personalized avatars elicit significantly more body ownership than non-personalized avatars. Thus, while a virtual mirror with non-personalized avatars is accepted to some degree, a virtual mirror which actually "reflects" the visual appearance of its user is accepted even more.

6.2 Limitations & Future Work

There are still factors influencing the virtual mirror which were not considered in this work and may be part of future work. Further, the character creation pipeline still has room for potential improvement concerning quality, automatism as well as speed. Thus, possible future directions may involve both the virtual mirror as well as the avatar creation pipeline. As the virtual mirror is a perfect tool to evoke virtual body ownership in VR by just employing visual cues, the use of the system in various follow-up studies concerning virtual embodiment and its effects are an interesting future direction.

Our lowest possible latency with the hardware of the proposed virtual mirror system rendering the scene as depicted in Figure 3.1 was about 45 ms (at 240 fps). However, with more recent and thus more powerful hardware (Two Nvidia Quadro P6000 GPUs and Two Intel Xeon E5-2643 v4 CPUs) we were actually able to further reduce latency to about $30\,ms$, while rendering the same scene at about 800 fps. Unfortunately, we were not able to use this hardware for our experiments. Therefore, a follow-up experiment further probing whether this even lower latency is low enough to be always perceived as simultaneous might lead to further interesting findings.

Similar, a user study investigating the effects of latency jitter on the virtual mirror may be of interest, as latency jitter, similar to latency itself, usually occurs in visualization of closed loop interaction such as the virtual mirror and may has a negative impact on motor performance as well as virtual embodiment factors.

Further, the impact of manipulated weight, height or other body properties of personalized avatars in the virtual mirror on a potentially stronger (compared to non-personalized avatars) Proteus effect and/or other corresponding aftereffects may result in interesting findings.

Additionally, visual quality of the avatars and the overall shading quality presumably have severe effects on the virtual mirror and corresponding virtual embodiment effects. Therefore, a systematic manipulation of those factors in an experiment will provide interesting insights. This is especially interesting concerning a potential uncanny valley.

Furthermore, as the animation of the mirror avatars is currently driven by simple yet fast geometry-based animation techniques, another promising research direction is a thorough investigation whether more sophisticated (physical) simulation during animation is necessary and how this influences the typical psychophysical virtual embodiment factors.
Also, in order to generate virtual characters even faster and fully automatic, the manual selection of landmarks has to be avoided. This imposes a whole new challenge to robustly, reliably and precisely detect these landmarks or to fit the template model without the need of landmarks. Still, this is an important and promising future direction as this would create an even faster and easier-to-use pipeline to create virtual characters that does not need any manual user interaction. Especially the latter would make the approach even more applicable for other research or even consumer applications.
Moreover, the endeavor to further increase the overall quality of the resulting avatars may be accomplished by the following measures.
First, and probably the most simple one, is to add more cameras to the scanner setup. However, while this will increase the texture as well as geometry quality, it will also decrease the speed of the whole pipeline as processing of more images will take more time.
Second, the resolution of the geometry is currently limited in order to achieve a fast fitting process and high rendering performance of the characters. But, since the tessellation of all created characters is equal, a normal map at least for the face could be employed by plugging in a pre-generated generic normal map. Similar, a normal map could be derived from image data.
Another interesting application for the proposed ready-to-animate characters are experiments in telepresence or Wizard-of-Oz scenarios, in which an operator remotely controls his/her character in another virtual environment to perform a certain task, e.g., coaching or collaborative tasks. Here, especially experiments comparing virtual agents against remote controlled characters may be an interesting research area, but also sophisticated social interactions are a promising scenario.
Furthermore, the proposed character scanning pipeline allows for experiments employing personalized virtual agents. Hence, one could actually get instructions from or interact with a virtual agent that looks like him-/herself, a relative or a friend. Then, in another study the influence of these personalized virtual agents on the behavior of participants may be investigated.
Concluding, both, the low latency virtual mirror and the fast character scanning, open the door to a lot of research opportunities and make many applications possible, which (so far) were only feasible with a whole lot of effort and involving tedious manual work.

Bibliography

[ACP03] Brett Allen, Brian Curless, and Zoran Popović. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Transactions on Graphics*, 22(3):587–594, 2003.

[ACPH06] Brett Allen, Brian Curless, Zoran Popović, and Aaron Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proc. of Eurographics Symposium on Computer Animation*, pages 147–156, 2006.

[Agr02] A. Agresti. *Categorical Data Analysis*. Wiley Series in Probability and Statistics. Wiley, 2002.

[Aka73] H Akaike. Information theory and an extension of the maximum likelihood principle. In *Proc. of Second International Symposium on Information Theory*, pages 267–281, 1973.

[ARL+09] Oleg Alexander, Mike Rogers, William Lambeth, Matt Chiang, and Paul Debevec. The digital emily project: Photoreal facial modeling and animation. In *SIGGRAPH 2009 Courses*, pages 1–15. ACM, 2009.

[Asa16] Tomohisa Asai. Agency elicits body-ownership: proprioceptive drift toward a synchronously acting external proxy. *Experimental brain research*, 234(5):1163–1174, 2016.

[ASK+05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: Shape completion and animation of people. *ACM Transactions on Graphics*, 24(3):408–416, 2005.

[Aut14] Autodesk. Character generator. `https://charactergenerator.autodesk.com/`, 2014.

[AZB15] Jascha Achenbach, Eduard Zell, and Mario Botsch. Accurate face reconstruction through anisotropic fitting and eye correction. In *Proc. of Vision, Modeling & Visualization*, pages 1–8, 2015.

[AZCP13] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Robust discriminative response map fitting with constrained local models. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3444–3451, 2013.

[BBA+04] Rosa María Baños, Cristina Botella, Mariano Alcañiz, Víctor Liaño, Belén Guerrero, and Beatriz Rey. Immersion and emotion: their impact on the sense of presence. *CyberPsychology & Behavior*, 7(6):734–741, 2004.

[BBB+10] Thabo Beeler, Bernd Bickel, Paul Beardsley, Bob Sumner, and Markus Gross. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics*, 29(4):1–9, 2010.

[BBH+90] Chuck Blanchard, Scott Burgess, Young Harvill, Jaron Lanier, Ann Lasko, Mark Oberman, and Mike Teitel. Reality built for two: A virtual reality tool. *ACM SIGGRAPH Comput. Graph.*, 24(2):35–36, 1990.

[BBLR15] Federica Bogo, Michael J Black, Matthew Loper, and Javier Romero. Detailed full-body reconstructions of moving people from monocular rgb-d sequences. In *Proc. of IEEE International Conference on Computer Vision*, pages 2300–2308, 2015.

[BC98] Matthew Botvinick and Jonathan Cohen. Rubber hands 'feel' touch that eyes see. *Nature*, 391(6669):756–756, 1998.

[BGS13] Domna Banakou, Raphaela Groten, and Mel Slater. Illusory ownership of a virtual child body causes overestimation of object sizes and implicit attitude changes. *Proceedings of the National Academy of Sciences*, 110(31):12846–12851, 2013.

[Bie00] Allen Douglas Bierbaum. *VR Juggler: A Virtual Platform for Virtual Reality Application Development*. PhD thesis, Iowa State University, 2000.

[BL94] Margaret M Bradley and Peter J Lang. Measuring emotion: The self-assessment manikin and the semantic differential. *Journal of Behavior Therapy and Experimental Psychiatry*, 25(1):49–59, 1994.

[BLB+02] Jim Blascovich, Jack Loomis, Andrew C Beall, Kimberly R Swinth, Crystal L Hoyt, and Jeremy N Bailenson. Immersive virtual environment technology as a methodological tool for social psychology. *Psychological Inquiry*, 13(2):103–124, 2002.

[BP07] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Transactions on Graphics*, 26(3), 2007.

[BRKE08] Gary Bente, Sabine Rüggenberg, Nicole C. Krämer, and Felix Eschenburg. Avatar-Mediated Networking: Increasing Social Presence and Interpersonal Trust in Net-Based Collaborations. *Human Communication Research*, 34(2):287–318, 2008.

[BRLB14] Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. Faust: Dataset and evaluation for 3d mesh registration. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3794–3801, 2014.

[BSJRR08] Stéphane Bouchard, Julie St-Jacques, Geneviève Robillard, and Patrice Renaud. Anxiety increases the feeling of presence in virtual reality. *Presence: Teleoperators and Virtual Environments*, 17(4):376–391, 2008.

[BTP14] Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Dynamic 2D/3D registration. In *Eurographics Tutorials*, 2014.

[Bus04] Samuel R Buss. Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17:1–19, 2004.

[BV99] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proc. of SIGGRAPH*, pages 187–194, 1999.

[BWP13] Sofien Bouaziz, Yangang Wang, and Mark Pauly. Online modeling for realtime facial animation. *ACM Transactions on Graphics*, 32(4):1–10, 2013.

[CCD+03] Philo Tan Chua, R. Crivella, B. Daly, Ning Hu, R. Schaaf, D. Ventura, T. Camill, J. Hodgins, and R. Pausch. Training for physical tasks in virtual environments: Tai Chi. In *Proc. of IEEE Virtual Reality*, pages 87–94, 2003.

[CHZ14] Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics*, 33(4):1–10, 2014.

[CNSD+92] Carolina Cruz-Neira, Daniel J. Sandin, Thomas A. DeFanti, Robert V. Kenyon, and John C. Hart. The CAVE: audio visual experience automatic virtual environment. *Communications of the ACM*, 35(6):64–72, 1992.

[COM14] Alexandra Covaci, Anne-Hélène Olivier, and Franck Multon. Third person view and guidance for more natural motor behaviour in immersive basketball playing. In *Proc. of ACM Symposium on Virtual Reality Software and Technology*, pages 55–64, 2014.

[CS99] KM Chung and Richard HY So. Effects of hand movement lag on discrete manual control tasks in virtual environements. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 43, pages 1210–1213. SAGE Publications, 1999.

[CWZ+14] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. FaceWarehouse: A 3D facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2014.

[DlPWL+10] Nonny De la Peña, Peggy Weil, Joan Llobera, Elias Giannopoulos, Ausiàs Pomés, Bernhard Spanlang, Doron Friedman, Maria V Sanchez-Vives, and Mel Slater. Immersive journalism: immersive virtual reality for the first-person experience of news. *Presence: Teleoperators and Virtual Environments*, 19(4):291–301, 2010.

[DMHB15] Henrique G Debarba, Eray Molla, Bruno Herbelin, and Ronan Boulic. Characterizing embodied interaction in first and third person perspective viewpoints. In *Proceedings of IEEE Symposium on 3D User Interfaces*, pages 67–72, 2015.

[DSPB12] Eugen Dyck, Holger Schmidt, Martina Piefke, and Mario Botsch. Octavis: Optimization techniques for multi-gpu multi-view rendering. *JVRB - Journal of Virtual Reality and Broadcasting*, 9(2012)(6), 2012.

[EF78] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psychologists Press, 1978.

[EMAH04] Stephen R Ellis, Katerina Mania, Bernard D Adelstein, and Michael I Hill. Generalizeability of latency detection in a variety of virtual environments. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 48, pages 2632–2636. SAGE Publications, 2004.

[EMP09] Stefan Eilemann, Maxim Makhinya, and Renato Pajarola. Equalizer: A scalable parallel rendering framework. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):436–452, 2009.

[FAPI99] Jonathan Freeman, Steve E Avons, Don E Pearson, and Wijnand A IJsselsteijn. Effects of sensory information and prior experience on direct subjective ratings of presence. *Presence*, 8(1):1–13, 1999.

[FBW00] Christopher D. Frith, Blakemore, and Daniel M. Wolpert. Abnormalities in the awareness and control of action. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 355(1404):1771–1788, 2000.

[FCS15] Andrew Feng, Dan Casas, and Ari Shapiro. Avatar reshaping and automatic rigging using a deformable model. In *Proc. of ACM Motion in Games*, pages 57–64, 2015.

[Fer65] William R Ferrell. Remote manipulation with transmission delay. *IEEE Transactions on Human Factors in Electronics*, 6(1):24–32, 1965.

[FF02] C. Farrer and C.D. Frith. Experiencing oneself vs another person as being the cause of an action: The neural correlates of the experience of agency. *NeuroImage*, 15(3):596–603, 2002.

[FFG+01] Nicolas Franck, Chlöé Farrer, Nicolas Georgieff, Michel Marie-Cardine, Jean Daléry, Thierry d'Amato, and Marc Jeannerod. Defective recognition of one's own actions in patients with schizophrenia. *American Journal of Psychiatry*, 158(3):454–459, 2001.

[FKS16] Sebastian Friston, Per Karlström, and Anthony Steed. The effects of low latency on pointing and steering tasks. *IEEE transactions on visualization and computer graphics*, 22(5):1605–1615, 2016.

[FRS17] Andrew Feng, Evan Suma Rosenberg, and Ari Shapiro. Just-in-time, viable, 3-d avatars from scans. *Computer Animation and Virtual Worlds*, 28:3–4, 2017.

[FS14] Sebastian Friston and Anthony Steed. Measuring latency in virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 20(4):616–625, 2014.

[FSR+14] Andrew Feng, Ari Shapiro, Wang Ruizhe, Mark Bolas, Gerard Medioni, and Evan Suma. Rapid avatar capture and simulation using commodity depth sensors. In *SIGGRAPH 2014 Talks*. ACM, 2014.

[FVH13] C. Farrer, G. Valentin, and J.M. Hupé. The time windows of the sense of agency. *Consciousness and Cognition*, 22(4):1431–1441, 2013.

[Gal00] Shaun Gallagher. Philosophical conceptions of the self: implications for cognitive science. *Trends in Cognitive Sciences*, 4(1):14–21, 2000.

[GFT+11] Abhijeet Ghosh, Graham Fyffe, Borom Tunwattanapong, Jay Busch, Xueming Yu, and Paul Debevec. Multiview face capture using polarized spherical gradient illumination. *ACM Transactions on Graphics*, 30(6):1–10, 2011.

[Gut01] Carl Gutwin. The effects of network delays on group work in real-time groupware. In *Proc. Seventh European Conference on Computer-Supported Cooperative Work*, pages 299–318. Springer, 2001.

[GWBB09] P. Guan, A. Weiss, A. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *Proc. of International Conference on Computer Vision*, pages 1381–1388, 2009.

[GZC+16] Pablo Garrido, Michael Zollhöfer, Dan Casas, Levi Valgaerts, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. Reconstruction of personalized 3d face rigs from monocular video. *ACM Transactions on Graphics*, 35(3):1–15, 2016.

[Häm04] Perttu Hämäläinen. Interactive video mirrors for sports training. In *Proc. of the third Nordic conference on Human-computer interaction*, pages 199–202. ACM, 2004.

[HD91] Richard Held and Nathaniel Durlach. Telepresence, time delay and adaptation. In *Pictorial communication in virtual and real environments*, pages 232–246. ACM, 1991.

[HLRB12] David A. Hirshberg, Matthew Loper, Eric Rachlin, and Michael J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *Proc. of European Conference on Computer Vision*, pages 242–255, 2012.

[HMST13] Thomas Helten, Meinard Muller, Hans-Peter Seidel, and Christian Theobalt. Real-time body tracking with one depth camera and inertial sensors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1105–1112, 2013.

[HMYL15] Pei-Lun Hsieh, Chongyang Ma, Jihun Yu, and Hao Li. Unconstrained realtime facial performance capture. In *Proc. of Computer Vision and Pattern Recognition*, pages 1675–1683, 2015.

[Hor87] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[HSS+09] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009.

[IA15] Shu Imaizumi and Tomohisa Asai. Dissociation of agency and body ownership following visuomotor temporal recalibration. *Frontiers in Integrative Neuroscience*, 9(35), 2015.

[IBP15] Alexandru Eugen Ichim, Sofien Bouaziz, and Mark Pauly. Dynamic 3d avatar creation from hand-held video input. *ACM Transactions on Graphics*, 34(4):1–14, 2015.

[IdKH06] Wijnand A IJsselsteijn, Yvonne A W de Kort, and Antal Haans. Is this my hand I see before me? The rubber hand illusion in reality, virtual reality, and mixed reality. *Presence: Teleoperators and Virtual Environments*, 15(4):455–464, 2006.

[JDKL14] Alec Jacobson, Zhigang Deng, Ladislav Kavan, and JP Lewis. Skinning: Real-time shape deformation. In *ACM SIGGRAPH Course Notes*, 2014.

[JH16] Sungchul Jung and Charles E. Hughes. The effects of indirect real body cues of irrelevant parts on virtual body ownership and presence. In D. Reiners, D. Iwai, and F. Steinicke, editors, *International Conference on Artificial Reality and Telexistence Eurographics Symposium on Virtual Environments (ICAT/EGVE)*, 2016.

[JNDW13] Ricardo Jota, Albert Ng, Paul Dietz, and Daniel Wigdor. How fast is fast enough? a study of the effects of latency in direct-touch pointing tasks. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 2291–2300, 2013.

[JNS12] Sophie Jörg, Aline Normoyle, and Alla Safonova. How responsiveness affects players' perception in digital games. In *Proceedings of the ACM Symposium on Applied Perception*, pages 33–38, 2012.

[JSW05] Tao Ju, Scott Schaefer, and Joe Warren. Mean value coordinates for closed triangular meshes. *ACM Transactions on Graphics*, 24(3):561–566, 2005.

[KBS13] K. Kilteni, I. Bergstrom, and M. Slater. Drumming in immersive virtual reality: The body shapes the way we play. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):597–605, 2013.

[KE12] Andreas Kalckert and H Henrik Ehrsson. Moving a rubber hand that feels like your own: a dissociation of ownership and agency. *Frontiers in human neuroscience*, 6, 2012.

[KLBL93] Robert S Kennedy, Norman E Lane, Kevin S Berbaum, and Michael G Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The international journal of aviation psychology*, 3(3):203–220, 1993.

[KRN13] Takahiro Kawabe, Warrick Roseboom, and Shin'ya Nishida. The sense of agency is action–effect causality perception based on cross-modal grouping. *Proceedings of the Royal Society of London B: Biological Sciences*, 280(1763), 2013.

[KS14] Elena Kokkianra and Mel Slater. Measuring the effects through time of the influence of visuomotor and visuotactile synchronous stimulation on a virtual body ownership illusion. *Perception*, 43(1):67–72, 2014.

[LAR+14] J. P. Lewis, Ken Anjyo, Taehyun Rhee, Mengjie Zhang, Fred Pighin, and Zhigang Deng. Practice and theory of blendshape facial models. In *Eurographics 2014 - State of the Art Reports*, 2014.

[LCC+12] Jean-Luc Lugrin, Fred Charles, Marc Cavazza, Marc Le Renard, Jonathan Freeman, and Jane Lessiter. CaveUDK: a VR game engine middleware. In *Proc. of ACM symposium on Virtual reality software and technology*, pages 137–144, 2012.

[LH09] Matthew R Longo and Patrick Haggard. Sense of agency primes manual motor responses. *Perception*, 38(1):69–78, 2009.

[LKK+18] Jean-Luc Lugrin, Philipp Krop, Richard Kluepfel, Bianca Weisz, Sebastian Stiersdorfer, Maximilian Rueck, Johann Schmitt, Nina Schmidt, Maximilian Ertl, and Marc Erich Latoschik. Any "Body" There? - Avatar Visibility Effects in a Virtual Reality Game. In *Proceedings of the 25th IEEE Virtual Reality Conference (IEEE VR)*, 2018. Under review.

[LKS14] Shu Liang, Ira Kemelmacher-Shlizerman, and Linda G. Shapiro. 3D face hallucination from a single depth frame. In *Proc. of International Conference on 3D Vision*, pages 31–38, 2014.

[LLL15a] Jean-Luc Lugrin, Johanna Latt, and Marc Erich Latoschik. Anthropomorphism and illusion of virtual body ownership. In *Proceedings of the 25th*

International Conference on Artificial Reality and Telexistence and 20th Eurographics Symposium on Virtual Environments, pages 1–8, 2015.

[LLL15b] Jean-Luc Lugrin, Johanna Latt, and Marc Erich Latoschik. Avatar anthropomorphism and illusion of body ownership in VR. In *Proceedings of IEEE Virtual Reality*, pages 229–230, 2015.

[LLR16] Marc Erich Latoschik, Jean-Luc Lugrin, and Daniel Roth. FakeMi: A fake mirror system for avatar embodiment studies. In *Proceeding of the 22nd ACM Symposium on Virtual Reality Software and Technology (VRST)*, pages 73–76, 2016.

[LMB14] Matthew Loper, Naureen Mahmood, and Michael J. Black. MoSh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics*, 33(6), 2014.

[LMR+15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6), 2015.

[LRG+17] Marc Erich Latoschik, Daniel Roth, Dominik Gall, Jascha Achenbach, Thomas Waltemate, and Mario Botsch. The effect of avatar realism in immersive social virtual realities. In *23rd ACM Symposium on Virtual Reality Software and Technology (VRST)*, pages 39:1–39:10, 2017.

[LSG91] Jiandong Liang, Chris Shaw, and Mark Green. On temporal-spatial realism in the virtual reality environment. In *Proc. of ACM symposium on User interface software and technology*, pages 19–25, 1991.

[LSG+16] Gale M. Lucas, Evan Szablowski, Jonathan Gratch, Andrew Feng, Tiffany Huang, Jill Boberg, and Ari Shapiro. Do avatars that look like their users improve performance in a simulation? In *Proceedings of International Conference on Intelligent Virtual Agents*, pages 351–354, 2016.

[LVG+13] Hao Li, Etienne Vouga, Anton Gudym, Linjie Luo, Jonathan T. Barron, and Gleb Gusev. 3d self-portraits. *ACM Transactions on Graphics*, 32(6), 2013.

[LWBL15] Jean-Luc Lugrin, Maximilian Wiedemann, Daniel Bieberstein, and Marc Erich Latoschik. Influence of avatar realism on stressfull situation in VR. In *Proceedings of IEEE Virtual Reality*, pages 227–228, 2015.

[LWP10] Hao Li, Thibaut Weise, and Mark Pauly. Example-based facial rigging. *ACM Transactions on Graphics*, 29(4), 2010.

[MAEH04] Katerina Mania, Bernard D. Adelstein, Stephen R. Ellis, and Michael I. Hill. Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. In *Proc. of the 1st Symposium on Applied Perception in Graphics and Visualization*, pages 39–47. ACM, 2004.

[MKK+17] C. Malleson, M. Kosek, M. Klaudiny, I. Huerta, J. C. Bazin, A. Sorkine-Hornung, M. Mine, and K. Mitchell. Rapid one-shot acquisition of dynamic vr avatars. In *Proc. of IEEE Virtual Reality Conference*, pages 131–140, 2017.

[MMK12] M. Mori, K. F. MacDorman, and N. Kageki. The uncanny valley [from the field]. *IEEE Robotics & Automation Magazine*, 19(2):98–100, 2012.

[MML12] Alessandro Moscatelli, Maura Mezzetti, and Francesco Lacquaniti. Modeling psychophysical data at the population-level: the generalized linear mixed model. *Journal of Vision*, 12(11):26–26, 2012.

[MR09] Iris B Mauss and Michael D Robinson. Measures of emotion: A review. *Cognition and emotion*, 23(2):209–237, 2009.

[MRWBJ03] Michael Meehan, Sharif Razzaque, Mary C Whitton, and Frederick P Brooks Jr. Effect of latency on presence in stressful virtual environments. In *Proc. of IEEE Virtual Reality*, pages 141–148, 2003.

[MS13] Antonella Maselli and Mel Slater. The building blocks of the full body ownership illusion. *Frontiers in human neuroscience*, 7:83, 2013.

[MSS+17] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017.

[MVHE04] Martin Mauve, Jürgen Vogel, Volker Hilt, and Wolfgang Effelsberg. Local-lag and timewarp: providing consistency for replicated continuous applications. *IEEE Transactions on Multimedia*, 6(1):47–57, 2004.

[MW93] I Scott MacKenzie and Colin Ware. Lag as a determinant of human performance in interactive systems. In *Proc. of the ACM INTERACT'93 and*

CHI'93 conference on Human factors in computing systems, pages 488–493, 1993.

[NGSS11] Jean-Marie Normand, Elias Giannopoulos, Bernhard Spanlang, and Mel Slater. Multisensory stimulation can induce an illusion of larger belly size in immersive virtual reality. *PloS one*, 6(1):e16128, 2011.

[Nie07] Paula M Niedenthal. Embodying emotion. *Science*, 316(5827):1002–1005, 2007.

[PE16] Cesare V Parise and Marc O Ernst. Correlation detection as a general mechanism for multisensory integration. *Nature communications*, 7, 2016.

[PGB03] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Transactions on Graphics*, 22(3):313–318, 2003.

[PK99] Kyoung Shin Park and Robert V Kenyon. Effects of network characteristics on human performance in a collaborative virtual environment. In *Proc. of IEEE Virtual Reality*, pages 104–111, 1999.

[PS11] Andriy Pavlovych and Wolfgang Stuerzlinger. Target following performance in the presence of latency, jitter, and signal dropouts. In *Proceedings of Graphics Interface*, pages 33–40, 2011.

[PSAS13] Tabitha C Peck, Sofia Seinfeld, Salvatore M Aglioti, and Mel Slater. Putting yourself in the skin of a black avatar reduces implicit racial bias. *Consciousness and cognition*, 22(3):779–787, 2013.

[PWH+17] Leonid Pishchulin, Stefanie Wuhrer, Thomas Helten, Christian Theobalt, and Bernt Schiele. Building statistical shape spaces for 3d human modeling. *Pattern Recognition*, pages 276–286, 2017.

[RBK13] Kyle Rector, Cynthia L Bennett, and Julie A Kientz. Eyes-free yoga: an exergame using depth cameras for blind & low vision exercise. In *Proc. of International ACM SIGACCESS Conference on Computers and Accessibility*, pages 12:1–12:8, 2013.

[RE16] Marieke Rohde and Marc O Ernst. Time, agency, and sensory feedback delays during action. *Current Opinion in Behavioral Sciences*, 8:193–199, 2016.

[REG14] Kjetil Raaen, Ragnhild Eg, and Carsten Griwodz. Can gamers detect cloud delay? In *Proceedings of the 13th Annual Workshop on Network and Systems Support for Games*. IEEE Press, 2014.

[RK05] Albert Rizzo and Gerard Kim. A SWOT analysis of the field of virtual reality rehabilitation and therapy. *Presence*, 14(2):119–146, 2005.

[RLG+16] Daniel Roth, Jean-Luc Lugrin, Dmitri Galakhov, Arvid Hofmann, Gary Bente, Marc Erich Latoschik, and Arnulph Fuhrmann. Avatar realism and social interaction quality in virtual reality. In *Proceedings of IEEE Virtual Reality*, pages 277–278, 2016.

[RLLH17] Daniel Roth, Jean-Luc Lugrin, Marc Erich Latoschik, and Stephan Huber. Alpha IVBO – construction of a scale to measure the illusion of virtual body ownership. In *Proceedings of CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2875–2883. ACM, 2017.

[RSE14] Marieke Rohde, Meike Scheller, and Marc O Ernst. Effects can precede their cause in the sense of agency. *Neuropsychologia*, 65:191–196, 2014.

[RvDE14] Marieke Rohde, Loes CJ van Dam, and Marc O Ernst. Predictability is necessary for closed-loop visual feedback delay adaptation. *Journal of Vision*, 14(3):1–23, 2014.

[RWL+17] Daniel Roth, Kristoffer Waldow, Marc Erich Latoschik, Arnulph Fuhrmann, and Gary Bente. Socially immersive avatar-based communication. In *Proceedings of IEEE Virtual Reality*, pages 259–260, 2017.

[SBB07] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Proc. of International Conference on Neural Information Processing Systems*, pages 1337–1344, 2007.

[SBK+14] Thomas Schack, M. Bertollo, Dirk Koester, Jonathan Maycock, and Kai Essig. *Technological advancements in sport psychology*, pages 953–965. Routledge Companion to Sport and Exercise Psychology: Global perspectives and fundamental concepts. Routledge, 2014.

[SBKC13] Jürgen Sturm, Erik Bylow, Fredrik Kahl, and Daniel Cremers. Copyme3d: Scanning and printing persons in 3d. In *Proc. of German Conference on Pattern Recognition*, pages 405–414, 2013.

[SGQ13] Gayani Samaraweera, Rongkai Guo, and John Quarles. Latency and avatars in virtual environments and the effects on gait for persons with mobility impairments. In *IEEE Symposium on 3D User Interfaces*, pages 23–30. IEEE, 2013.

[SHG+11] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *International Conference on Computer Vision (ICCV)*, pages 951–958, 2011.

[SHRB12] Matthias Straka, Stefan Hauswiesner, Matthias Ruther, and Horst Bischof. Rapid skin: Estimating the 3d human pose and shape in real-time. In *Proc. of International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 41–48, 2012.

[Sla99] Mel Slater. Measuring presence: A response to the Witmer and Singer presence questionnaire. *Presence: Teleoperators and Virtual Environments*, 8(5):560–565, 1999.

[SNB+14] Bernhard Spanlang, Jean-Marie Normand, David Borland, Konstantina Kilteni, Elias Giannopoulos, Ausiàs Pomés, Mar González-Franco, Daniel Perez-Marcos, Jorge Arroyo-Palacios, Xavi Navarro Muncunill, and Mel Slater. How to build an embodiment lab: Achieving body representation illusions in virtual reality. *Frontiers in Robotics and AI*, 1:9, 2014.

[SNL16] J. P. Stauffert, F. Niebling, and M. E. Latoschik. Reducing application-stage latencies of interprocess communication techniques for real-time interactive systems. In *Proceedings of IEEE Virtual Reality*, pages 287–288, 2016.

[SP04] Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on Graphics*, 23(3):399–405, 2004.

[SPMESV08] Mel Slater, Daniel Perez-Marcos, Henrik Ehrsson, and María Victoria Sánchez-Vives. Towards a digital body: The virtual arm illusion. *Frontiers in Human Neuroscience*, 2(6), 2008.

[SPQ15] Gayani Samaraweera, Alex Perdomo, and John Quarles. Applying latency to half of a self-avatar's body to change real walking patterns. In *Proceedings of IEEE Virtual Reality*, pages 89–96, 2015.

[SRMC+14] Roland Sigrist, Georg Rauter, Laura Marchal-Crespo, Robert Riener, and Peter Wolf. Sonification and haptic feedback in addition to visual feedback

enhances complex motor task learning. *Experimental brain research*, 233:1–17, 2014.

[SS00] Mel Slater and Anthony Steed. A virtual presence counter. *Presence*, 9(5):413–434, 2000.

[SS15] Anthony Steed and Ralph Schroeder. Collaboration in Immersive and Non-immersive Virtual Environments. In Matthew Lombard, Frank Biocca, Jonathan Freeman, Wijnand IJsselsteijn, and Rachel J. Schaevitz, editors, *Immersed in Media*, pages 263–282. Springer, 2015.

[SSSVB10] Mel Slater, Bernhard Spanlang, Maria V Sanchez-Vives, and Olaf Blanke. First person experience of body transfer in virtual reality. *PloS one*, 5(5):e10564, 2010.

[Ste08] Anthony Steed. A simple method for estimating the latency of interactive, real-time graphics simulations. In *Proc. of ACM symposium on Virtual reality software and technology*, pages 123–129, 2008.

[SVHM14] Jan David Smeddinck, Jens Voges, Marc Herrlich, and Rainer Malaka. Comparing modalities for kinesiatric exercise instruction. In *ACM CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pages 2377–2382, 2014.

[SVSF+10] Maria V Sanchez-Vives, Bernhard Spanlang, Antonio Frisoli, Massimo Bergamasco, and Mel Slater. Virtual hand illusion induced by visuomotor correlations. *PloS one*, 5(4):e10381, 2010.

[SWM+17] Matthias Schröder, Thomas Waltemate, Jonathan Maycock, Tobias Röhlig, Helge Ritter, and Mario Botsch. Design and Evaluation of Reduced Marker Layouts for Hand Motion Capture. *Computer Animation and Virtual Worlds*, page e1751, 2017.

[SWTC14] Fuhao Shi, Hsiang-Tao Wu, Xin Tong, and Jinxiang Chai. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Transactions on Graphics*, 33(6), 2014.

[TH05] Manos Tsakiris and Patrick Haggard. The rubber hand illusion revisited: visuotactile integration and self-attribution. *Journal of Experimental Psychology: Human Perception and Performance*, 31(1):80, 2005.

[TLH10] Manos Tsakiris, Matthew R Longo, and Patrick Haggard. Having a body versus moving your body: neural signatures of agency and body-ownership. *Neuropsychologia*, 48(9):2740–2749, 2010.

[TMB14] Aggeliki Tsoli, Naureen Mahmood, and Michael J. Black. Breathing life into shape: Capturing, modeling and animating 3d human breathing. *ACM Transactions on Graphics*, 33(4), 2014.

[TNI15] Kazuaki Tanaka, Hideyuki Nakanishi, and Hiroshi Ishiguro. Physical embodiment can produce robot operator's pseudo presence. *Frontiers in ICT*, 2:8, 2015.

[TPH06] Manos Tsakiris, Gita Prabhu, and Patrick Haggard. Having a body versus moving your body: How agency structures body-ownership. *Consciousness and cognition*, 15(2):423–432, 2006.

[TPSH14] Marco Tarini, Daniele Panozzo, and Olga Sorkine-Hornung. Accurate and efficient lighting for skinned models. *Computer Graphics Forum (proceedings of EUROGRAPHICS issue)*, 33(2):421–428, 2014.

[TPSM09] Robert J Teather, Andriy Pavlovych, Wolfgang Stuerzlinger, and I Scott MacKenzie. Effects of tracking technology, latency, and spatial jitter on object movement. In *Proceedings of IEEE Symposium on 3D User Interfaces*, pages 43–50, 2009.

[TZL+12] 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, 2012.

[TZN+15] Justus Thies, Michael Zollhöfer, Matthias Nießner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM Transactions on Graphics*, 34(6), 2015.

[TZS+16a] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 2387–2395, 2016.

[TZS+16b] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Facevr: Real-time facial reenactment and eye gaze control in virtual reality. *arXiv preprint arXiv:1610.03151*, 2016.

[TZS+18] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Headon: Real-time reenactment of human portrait videos. *ACM Transactions on Graphics 2018 (TOG)*, 2018.

[VTM10] Valentijn T Visch, Ed S Tan, and Dylan Molenaar. The emotional and cognitive effect of immersion in film viewing. *Cognition and Emotion*, 24(8):1439–1445, 2010.

[WB94] Colin Ware and Ravin Balakrishnan. Reaching for objects in VR displays: lag and frame rate. *ACM Transactions on Computer-Human Interaction*, 1(4):331–356, 1994.

[WBGB16] Chenglei Wu, Derek Bradley, Markus Gross, and Thabo Beeler. An anatomically-constrained local deformation model for monocular face capture. *ACM Transactions on Graphics*, 35(4), 2016.

[WBLP11] Thibaut Weise, Sofien Bouaziz, Hao Li, and Mark Pauly. Realtime performance-based facial animation. *ACM Transactions on Graphics*, 30(4), 2011.

[WGJ95] Daniel M Wolpert, Zoubin Ghahramani, and Michael I Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, 1995.

[WHB11] Alexander Weiss, David Hirshberg, and Michael J. Black. Home 3d body scans from noisy image and range data. In *Proc. of IEEE International Conference on Computer Vision*, pages 1951–1958, 2011.

[WPB+14] Stefanie Wuhrer, Leonid Pishchulin, Alan Brunton, Chang Shu, and Jochen Lang. Estimation of human body shape and posture under clothing. *Computer Vision and Image Understanding*, 127:31–42, 2014.

[WS98] Bob G Witmer and Michael J Singer. Measuring presence in virtual environments: A presence questionnaire. *Presence: Teleoperators and virtual environments*, 7(3):225–240, 1998.

[YB07] Nick Yee and Jeremy Bailenson. The Proteus effect: The effect of transformed self-representation on behavior. *Human communication research*, 33(3):271–290, 2007.

[YS13] Sangseok You and S. Shyam Sundar. I feel for my avatar: Embodied perception in VEs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3135–3138. ACM, 2013.

[YSSRA13] Kielan Yarrow, Ingvild Sverdrup-Stueland, Warrick Roseboom, and Derek H Arnold. Sensorimotor temporal recalibration within and across limbs. *Journal of Experimental Psychology: Human Perception and Performance*, 39(6):1678–1689, 2013.

[ZMG+11] Michael Zollhöfer, Michael Martinek, Günther Greiner, Marc Stamminger, and Jochen Süßmuth. Automatic reconstruction of personalized avatars from 3d face scans. *Computer Animation and Virtual Worlds*, 22(2-3):195–202, 2011.

[ZTG+18] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. *Computer Graphics Forum (Eurographics State of the Art Reports 2018)*, 37(2), 2018.