TECHNISCHE
UNIVERSITÄT
DARMSTADT

FULL-BODY MOTION TRACKING *in* IMMERSIVE VIRTUAL REALITY

Full-Body Motion Reconstruction and Recognition for Immersive Multiplayer Serious Games

Dem Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

POLONA CASERMAN, M.SC.

Geboren am 14. März 1990 in Ljubljana, Slowenien

Vorsitz: Prof. Dr. rer. nat. Andy Schürr
Referent: Prof. Dr.-Ing. Ralf Steinmetz
Korreferent: Prof. Dr.-Ing. Uwe Rüppel
Korreferent: PD. Dr.-Ing. Stefan Göbel

Tag der Einreichung: 16. März 2021
Tag der Disputation: 22. Juni 2021

Hochschulkennziffer D17
Darmstadt 2021

Polona Caserman, M.Sc.: *Full-Body Motion Tracking in Immersive Virtual Reality,*
Full-Body Motion Reconstruction and Recognition for Immersive Multiplayer Serious Games
Darmstadt, Technische Universität Darmstadt

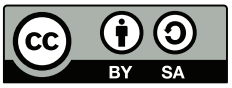
Jahr der Veröffentlichung der Dissertation auf TUprints: 2021
Tag der mündlichen Prüfung: 22. Juni 2021

Dieses Dokument wird bereitgestellt von: This document is provided by:

tuprints, E-Publishing-Service der Technischen Universität Darmstadt.
http://tuprints.ulb.tu-darmstadt.de
tuprints@ulb.tu-darmstadt.de

Bitte zitieren Sie dieses Dokument als: Please cite this document as:

URN: urn:nbn:de:tuda-tuprints-175720
URL: https://tuprints.ulb.tu-darmstadt.de/id/eprint/17572



ABSTRACT

The release of consumer-grade virtual reality head-mounted displays contributed to the development of immersive applications that convey an illusion of being present in the virtual environment. This great potential of virtual reality is promising not only for the entertainment industry but also for education and health.

However, the head-mounted display obstructs the players' view of the real environment, causing them to see neither the real environment nor their bodies or those of their teammates and opponents. Therefore, full-body motion reconstruction is essential to improve the sense of presence and interaction among users. Nevertheless, due to the lack of users' motion data, many popular virtual reality games focus solely on upper-body movements and show only controllers or floating hands. Moreover, full-body motion recognition is crucial to ensure that users perform desired physical activities correctly, either to improve health outcomes or to lower the risk of injury.

The contributions in this thesis include the reconstruction and recognition of full-body movements using off-the-shelf virtual reality devices. However, such a motion tracking system requires many sensors to be attached to the body, making it difficult to set up and uncomfortable to wear. Therefore, as the *first contribution*, the number of sensors is reduced to not restrict the user's movements. A reduction in sensors is also required in health-based applications as patients with physical limitations often cannot hold or wear additional devices. To this end, inverse kinematics methods are explored and their parameters are optimized to estimate the full-body pose with high accuracy and low latency. Because high latency between the user's movements and the corresponding visual feedback on the head-mounted display causes cybersickness, the effect of increased end-to-end latency on user experience and performance is investigated as the *second contribution*. Here, an end-to-end latency threshold that elicits significant cybersickness and causes users to need significantly more time to complete a task is identified. As the *third contribution*, machine learning algorithms are employed to identify suitable sensor positions for reliable full-body motion recognition. Thereby, the entire movement is analyzed and potential activity execution errors are identified.

The elaborated model on full-body motion reconstruction and recognition is prototypically implemented and validated in the context of two serious games: (1) an exergame designed to motivate players to train specific movements and (2) a multiplayer training simulation for police forces to enable training of stressful situations. In the exergame, the system's capability has been demonstrated to recognize the activity execution errors and provide appropriate feedback so that players can improve their movements. By means of the training simulation, statistical significance and effect sizes have been analyzed to examine the impact of full-body avatars in contrast to an abstract representation with head and hands on stress level. Thereby, an empirical study with police forces showed the added value of full-body avatars, which improve the feeling of presence and enable communication via body language and gestures.

KURZFASSUNG

Die Virtual Reality Technologie gewinnt immer mehr an Bedeutung, nicht nur in der Unterhaltungsindustrie, sondern auch beispielsweise für das Training im Bildungs- oder Gesundheitsbereich. Seitdem erschwingliche Head-Mounted Displays auf dem Markt verfügbar sind, können viele Endverbraucher diese Technologie nutzen und so in immersive virtuelle Umgebungen eintauchen.

Beim Tragen der Head-Mounted Displays können die Nutzer jedoch weder die reale Umgebung, noch ihren eigenen Körper oder den ihrer Teamkollegen und Gegner sehen. Daher ist die Rekonstruktion von Ganzkörper-Bewegungen essentiell, um das Gefühl der Präsenz und die Interaktion zwischen den Nutzern zu verbessern. Aufgrund fehlender Bewegungsdaten der Benutzer befassen sich viele beliebte Virtual Reality Spiele ausschließlich mit den Bewegungen des Oberkörpers und zeigen nur die Controller oder Hände. Darüber hinaus ist die Erkennung von Ganzkörper-Bewegungen maßgeblich um sicherzustellen, dass Benutzer beispielsweise körperliche Übungen korrekt ausführen, um die gewünschten Gesundheitseffekte zu erzielen oder das Verletzungsrisiko zu senken.

Die Beiträge in dieser Arbeit umfassen die Rekonstruktion und Erkennung von Ganzkörper-Bewegungen mit handelsüblichen Virtual Reality Geräten. Derartige Bewegungserfassungssysteme erfordern jedoch, dass viele Sensoren am Körper angebracht werden, was den Einrichtungsaufwand erhöht und den Nutzungskomfort senkt. Daher wurde im *ersten Beitrag* dieser Arbeit die Anzahl der Sensoren reduziert, um die Benutzer nicht einzuschränken. Eine kleine Anzahl von Sensoren ist beispielsweise bei Anwendungen im Gesundheitsbereich erforderlich, da Patienten mit körperlichen Einschränkungen häufig keine zusätzlichen Geräte halten oder tragen können. Zu diesem Zweck wurden Methoden der inversen Kinematik untersucht und deren Parameter optimiert, um die Ganzkörperbewegungen mit hoher Genauigkeit und geringer Latenz zu rekonstruieren. Da eine hohe Latenz zwischen den realen Bewegungen des Benutzers und dem entsprechenden visuellen Feedback auf dem Head-Mounted Display Cybersickness verursachen kann wurden anschließend, als *zweiten Beitrag* der Arbeit, die Auswirkungen einer erhöhten End-to-End-Latenz auf die Benutzererfahrung und -leistung untersucht. Hierzu wurde ein End-to-End-Latenz Schwellwert identifiziert, der signifikante Cybersickness-Symptome hervorruft und dazu führt, dass Benutzer erheblich mehr Zeit benötigen um eine Aufgabe abzuschließen. Als *dritter Beitrag* dieser Arbeit wurden Algorithmen des maschinellen Lernens angewandt, um geeignete Sensorpositionen für eine zuverlässige Ganzkörper-Bewegungserkennung zu identifizieren. Dabei wurden die gesamte Bewegungsausführung analysiert und mögliche Ausführungsfehler der Übung identifiziert.

Das erarbeitete Modell zur Rekonstruktion und Erkennung von Ganzkörperbewegungen wurde im Kontext von zwei Serious Games prototypisch realisiert und validiert: (1) ein Exergame, in dem die Spieler zum Trainieren bestimmter Bewegungen

motiviert werden sollen und (2) eine Multiplayer-Trainingsumgebung für Polizeikräfte, in der Stress-Situationen simuliert werden können. Im Exergame wurde gezeigt, dass der vorgestellte Ansatz die Bewegungen und Ausführungsfehler der Spieler erkennt und es konnte ein direktes Feedback an die Nutzer gegeben werden. Mittels der Trainingssimulation wurde die statistische Signifikanz und Effektstärke analysiert, um den Einfluss von Ganzkörper-Avataren auf das Stressniveau zu untersuchen. Dabei zeigte eine empirische Studie mit Polizeikräften den Mehrwert von Ganzkörper-Avataren im Vergleich zu einer abstrakten Darstellung mit Kopf und Händen, wodurch das Gefühl der Präsenz verbessert und die Kommunikation durch Körpersprache und Gesten ermöglicht wurde.

CONTENTS

PREVIOUSLY PUBLISHED MATERIAL

This thesis includes material that has been previously published in scientific journals and conferences. Table 1 summarizes the relevant publications for the content of this thesis. No text in this thesis has been copied directly from these publications. Figures and tables, particularly those containing evaluation data have been replicated, restructured, and their content adapted. A list with all scientific publications of the author of this thesis is available in Appendix B.

Scientific work is usually the result of teamwork. In the context of this thesis, we address an interdisciplinary research area tackling computer science and information technology in the healthcare (exergames) as well as vocational (training simulations) context. Hence, all research publications (see Table 1 and Appendix B) are the results of the collaborative work of computer and sports scientists, mathematicians, police officers, and game designers.

Table 1: List of publications in peer-reviewed journals and conferences related to this thesis.

	Publications
Chapter 2: BACKGROUND AND RELATED WORK	
Section 2.1.3: Cybersickness	Caserman et al. [41]
Section 2.2: Full-Body Motion Tracking	Caserman et al. [42]
Chapter 4: FULL-BODY MOTION RECONSTRUCTION	
Section 4.1.2: Parameters Optimization for Jacobian Inverse Methods	Caserman et al. [39]
Section 4.2: Avatar Animation	Caserman et al. [43]
Section 4.4: Effect of End-to-End Latency	Caserman et al. [48], Garcia-Agundez et al. [96]
Chapter 5: FULL-BODY MOTION RECOGNITION	
Section 5.2.3: Hidden Markov Models	Caserman et al. [47, 50]
Section 5.3.1: Appropriate Feedback	Caserman et al. [45]
Chapter 6: VALIDATION OF FULL-BODY MOTION RECONSTRUCTION AND RECOGNITION	
Section 6.1: Virtual Reality Training Simulation	Caserman et al. [40, 49, 51]
Section 6.2: Virtual Reality-Based Exergame	Caserman et al. [47]

Therefore, we will disclose the contributions of all co-authors, including their affiliations. Whenever no dedicated affiliation is provided, the respective person is or has been a colleague at the Multimedia Communications Lab of the Technical University of Darmstadt. The pronoun "I" will be used exclusively in this chapter to describe the specific contributions of the author of this thesis. For the remainder of this thesis, the pronoun "we" will be used instead to refer to the contribution of all co-authors of the respective publication.

Chapter 2, BACKGROUND AND RELATED WORK, presents the results of the relevant related work on Virtual Reality (VR). I conducted a systematic literature review, receiving assistance from Dr.-Ing. Augusto Garcia-Agundez and PD Dr.-Ing. Stefan Göbel regarding methodology and choice of inclusion and exclusion criteria. The aim of the literature review was to identify the research gap and to provide the requirements analysis for this thesis. I gathered and critically analyzed recent advances about the potential of full-body motion reconstruction in VR-based applications. The results of the survey were published in [42]. The analysis indicated the importance of full-body motion reconstruction to enhance the sense of presence within VR. The analysis further yielded valuable results and revealed the advantages and disadvantages of different motion capture systems regarding accuracy, robustness, latency, and setup effort.

Furthermore, because cybersickness is still present in many VR-based applications, I investigated its causes in collaboration with Dr.-Ing. Augusto Garcia-Agundez (as an equally collaborating co-author), Alvar Gámes Zerban, and PD Dr.-Ing. Stefan Göbel. We conducted a meta-analysis and presented results by combining and comparing data from multiple scientific studies. The goal of this meta-analysis was to investigate the differences in the intensity and patterns of cybersickness among different VR Head-Mounted Displays (HMDs), e. g., Oculus Rift vs. HTC Vive. Additionally, we explored the effects of different locomotion techniques that cause a sensory mismatch. The results of the analysis were published in [41].

Moreover, I conducted a literature review on full-body motion recognition with assistance from Philipp Niklas Müller. However, the results of this review have not been published separately to this thesis. I identified the research gap showing that only a few studies tackle the recognition of full-body movements in VR-based applications. In particular, most studies lack accurate full-body motion recognition and focus only on detecting the movements of one body part, usually upper-body movements or only hand gestures.

Chapter 4, FULL-BODY MOTION RECONSTRUCTION, introduces an approach for full-body motion reconstruction and highlights evaluation results on quality. The initial idea of using off-the-shelf VR devices for full-body motion reconstruction emerged during the master thesis in 2017 from a discussion with Robert Konrad. Based on the results of the master thesis, I proposed in [43] a preliminary approach to reconstruct movements of a single limb using one off-the-shelf VR sensor, namely the HTC Vive tracker. The results of a user study demonstrated the feasibility of movement reconstruction based on the desired rotation and position of a sensor. Subsequently, in

collaboration with Philipp Achenbach, I elaborated an approach for full-body motion reconstruction using HTC Vive HMD, two controllers, and three trackers. The results on the quality of relevant inverse kinematics methods, which were published in [39], show that full-body movements are reconstructed with reasonable accuracy and low end-to-latency.

Furthermore, because solving the inverse kinematics problem can cause high computation time and a decrease in frame rate, we explored the effects of end-to-end latency on user experience and performance. A high end-to-end latency between the users' movements and the corresponding visual feedback can induce negative effects, such as cybersickness. We explored the possibility of different extrapolation and filtering techniques to predict head movements accurately to reduce the impact of end-to-end latency. The results of a study conducted by Dr.-Ing. Augusto Garcia-Agundez were published in [96].

However, although it is well known that a mismatch or a delay between real and virtual head movements causes cybersickness, it is still unclear when an increased end-to-end latency indeed triggers significant cybersickness symptoms. Consequently, I conducted a user study, together with Michelle Martinussen, to investigate the effects of cybersickness and to identify an end-to-end latency threshold that causes significant symptoms in users. Dr.-Ing. Augusto Garcia-Agundez helped with the design and ethical considerations. The results of the study, which were published in [48], show that end-to-end latency higher than 56.82 ms elicits significant cybersickness symptoms, although much lower latency is required to abolish cybersickness. The results additionally show that an end-to-end latency above 69 ms causes users to need significantly more time to complete a task.

Chapter 5, FULL-BODY MOTION RECOGNITION, introduces an approach for full-body motion recognition and highlights evaluation results on quality. The primary goal of full-body motion recognition is to analyze the entire movement execution, recognize activity recognition errors, and to provide appropriate feedback. The importance of appropriate feedback and other quality criteria for serious games were identified during the *WTT Serious Games* project (IWB-EFRE-Program of the State of Hesse) as an interdisciplinary work of computer and sports scientists, mathematicians, and game designers. PD Dr.-Ing. Stefan Göbel, as the main researcher, initiator, and coordinator of the *WTT Serious Games* project, supervised the overall approach and contributed to the identification of the quality criteria. In particular, in interdisciplinary work with Prof. Dr. rer. medic. Josef Wiemeyer and Katrin Hoffmann (both researchers are with the Institute of Sport Science, Technical University of Darmstadt), we identified quality criteria for serious games for health. Additionally, together with Prof. Dr. Regina Bruder and Marcel Schaub (both researchers are with the Research Group Didactics of Mathematics, Technical University of Darmstadt), we identified the quality criteria for educational games. Furthermore, together with Philipp Niklas Müller and Katharina Straßburg, we identified the quality criteria for the balance between the serious and the game part. A summarized text providing guidelines for high-quality serious games was published in [45], with all co-authors contributing to the manuscript's writing.

Because high-quality serious games should provide players with appropriate feedback on their performance and progress, I elaborated an approach, together with Marco Fendrich, Moritz Kolvenbach, and Markus Stabel, to identify activity execution errors. From an initial implementation published in [50], I implemented feedback to let users know if they accomplished the movement as required. Concerning motion recognition, Dr.-Ing. Thomas Tregel and Philipp Niklas Müller provided valuable feedback and helped me to verify the validity of the machine learning approach. Subsequently, I improved recognition performance in collaboration with Shule Liu. The results of the extended version were published in [47].

To provide appropriate feedback, we further investigated where on the body the sensors should best be attached to recognize full-body movements reliably. In collaboration with Roman Uhlig, as part of his master thesis [219], we employed machine learning algorithms and developed a system that analyzes the recognition performance of relevant sensor subsets. I assisted and supervised his work. Built upon the collaboration with Roman Uhlig, I trained and tested new machine learning models using my own database. Thereby, I investigated which sensor subset is most suitable for reliable full-body motion recognition. Dr.-Ing. Thomas Tregel and Philipp Niklas Müller contributed with valuable ideas and suggestions for improvement.

Chapter 6, VALIDATION OF FULL-BODY MOTION RECONSTRUCTION AND RECOGNITION, implements and validates the approach on full-body motion reconstruction and recognition through the example of two immersive training simulations. Thereby, I investigate the added value of the full-body avatars in multiplayer immersive training simulation for police officers. An initial idea to use VR technology to enable police forces to train scenarios that are otherwise too expensive, require high effort for setup, or are too dangerous to be trained in the real world emerged in discussion with Jonas Zinnäcker (Hessian Police Headquarters for Technology). Subsequently, Miriam Cornel and Michelle Dieter elaborated in close cooperation with police officers a VR training simulation for police control. I supervised the work and helped with the design. The results of the initial user study, published in [40], revealed the importance of full-body motion tracking and intelligent virtual agents. I then improved the gameplay with support from Hongtao Zhang. In particular, we proposed a novel directed teleport function to show points of interest to players and to guide them. The results of the study, which were published in [51], indicate that the directed teleport function is very effective and easy to use.

The importance of full-body motion reconstruction within the VR training simulations was further identified during the research project *SimuLab* (project number 2019 FO-09) and driven by the continuing cooperation with Prof. Dr.-Ing. Thorsten Göbel and André Kecke (both are with the Hessian University of Applied Sciences for Police and Public Administration). To investigate the effects of full-body avatars in VR, Philip Schmidt conducted a user study with police officers at the Hessian University of Police and Public Administration with my help. I then performed the data analysis. The results of the study, which were submitted for publication in [49], showed a significant

increase in stress level and threat during the immersion with a full-body avatar in contrast to the head and hands-only representation.

Furthermore, based on the initial idea by Marco Fendrich, Moritz Kolvenbach, and Markus Stabel, I developed an exergame designed to recognize full-body movements and thereby to motivate players to train and learn certain poses. The game design for the exergame was developed by Katharina Straßburg, whereas I implemented it and integrated the feedback on users' performance. I also received technical support from Robert Konrad. After the implementation of the game design, I elaborated and conducted a user study. The purpose of this user study was to evaluate the preliminary game design and to gain further insights into player preferences. Because cybersickness is still present in many immersive VR games, we also evaluated if the current game elicits any symptoms. The results of this user study on game experience and cybersickness were published in [47]. Subsequently, I elaborated in collaboration with Jan Thorsten Neitzel, as part of his bachelor thesis [158], an approach to improve player performance. I assisted and supervised his work. The goal of this approach was to adapt the difficulty level in the game to challenge novice and expert players. Therefore, we extended the existing exergame to support various players by introducing an additional character (i. e., a virtual trainer) who shows correct executions of poses and draws attention to the misplaced body parts.

Moreover, in cooperation with SoftwareAG, I explored the interaction techniques in multiplayer virtual environments within the framework of the Software Campus project *TargetVR*, which was funded by German Federal Ministry of Education and Research (BMBF). The aim of the *TargetVR* project was to design and develop a multiplayer VR serious game in order to convey approaches of agile software development in IT companies. The game design and the results of initial user testings of the early state serious game were published in [44].

1 INTRODUCTION

Immersive Virtual Reality (VR) has become increasingly popular in recent years, both in the entertainment industry and in academic research. In contrast to desktop applications, immersive VR refers to a computer-generated virtual world, presented to users through a Head-Mounted Display (HMD) [197]. Due to technological advancements, i. e., the increase in graphics and computing power, HMDs with high resolution and wide field of view became broadly available for consumers. The availability of affordable HMDs opened unique possibilities to create immersive VR-based applications and games that give users the illusion of being there in the virtual environment.

Although VR is a relatively new industry, it has grown significantly over the past years. The growth of the VR market is reflected in an increasing number of users [207] and revenue [209]. Recent reports show that the global VR gaming market reached a value of USD 10.3 billion in 2018 and forecasts show the value will increase to USD 40.2 billion by 2024 [182]. This evidence shows an enormous boom of the VR gaming segment as a promising and prospering sector of the overall games market.

While VR technology is currently mostly used for entertainment, i. e., video games [208], VR devices also offer opportunities to be used in the field of education [10, 206], e. g., training simulations. Thus, VR technology is beneficial for various purposes, including serious games. Serious games are digital games with an additional goal beyond entertainment, e. g., they intend to improve learning or training outcomes [70, 97].

A recent report on serious games shows that, in particular, the usage of training simulations is steadily growing, e. g., enabling trainees to undertake high-risk tasks within a safe environment without dangerous real-world implications [181]. The New York Police Department recently demonstrated the use of VR technology for efficient police training [139]. Similarly, in the logistics industry, DHL utilizes VR technology to educate and onboard employees [67]. Further examples include training simulations for medical personnel [218] and firefighters [76]. Hence, VR has great potential and is promising for entertainment games as well as serious games, especially in the context of training simulations.

1.1 Motivation for full-body motion reconstruction and recognition

Several studies agree that a virtual body is a significant contributor to induce the sense of being in the virtual environment [200, 221]. Therefore, to enhance the sense of embodiment, immersive VR applications should reconstruct full-body motions in real-time, employ the first-person perspective, and ensure sensory correlation between the user's body and the avatar's body [118].

Solutions such as IKINEMA Orion[1] or VRchat[2] support full-body motion tracking with six to eight sensors. Nevertheless, many popular VR games on Steam [61] or PlayStation Store [78], such as Beat Saber [92], Half-Life: Alyx [60], and Hot Dogs, Horseshoes and Hand Grenades [141], still focus solely on tracking upper-body movements. These games either show controllers or floating hands and neglect reliable reconstruction of the lower body. Similarly, only a minority of multiplayer VR games such as Star Trek: Bridge Crew [77] and Population: One [20] show full-body avatars. However, because they only track the players' head and hand movements, merely the upper-body movements are synchronized.

The lack of an accurate full-body representation is already apparent in singleplayer and especially in multiplayer VR experiences. As VR players cannot see their bodies and their teammates or opponents, interaction in the VR environment becomes cumbersome. For example, full-body avatars in training simulations for police forces are necessary to enable non-verbal communication via body language and gestures. Without full-body tracking, the VR training simulations restrict the players' ability to interact naturally and are not realistic in terms of actions. Hence, avatars improve communication within shared virtual environments and provide information that would otherwise be difficult to mediate [79]. Moreover, the degree of presence in training simulation is important to enable users to behave in the virtual world similarly to how they behave in the real world to learn and develop skills and tactics, which they can then transfer to the real world.

Likewise, in the field of motion recognition, many studies focus only on one body part and usually recognize hand gestures with controllers such as Leap Motion [83] or a Myo armband [168]. However, especially in exergames that aim to get players physically active, it is crucial to track all relevant body parts to ensure that players perform the exercises as required by the game. One prominent example is the Nintendo Wii Fit, which allows players to remain seated on the couch and make minimal movements with Wii Remote controllers to play the game successfully [126]. A previous study has shown that games played with Nintendo Wii controllers require less physical activity than the same games played with the Microsoft Kinect [147]. To overcome this problem, the Nintendo recently released a game Ring Fit Adventure with a Ring-Con and a Leg Strap to also track the leg movements [163].

Apart from the Nintendo Wii Fit, many VR games lack lower-body movement tracking. VR games such as Beat Saber [92] or Audioshield [11] provide playful fitness experiences; however, they track only the player's arms or upper-body movements. Indeed, such VR games do not necessarily have to track lower-body movements. However, in games for health, appropriate tracking technology and accurate recognition are necessary to ensure that players perform the exercises correctly. Unreliable recognition of physical activity can lead to diminished health improvements and injuries. Above all, unreliable recognition of in-game activities can also decrease players' motivation and ultimately deter them from additional training. Faric et al. [81] show that VR players prefer games that are intuitive in terms of body movements and akin to

[1] `https://ikinema.com/`, last accessed on March 3, 2021

[2] `https://docs.vrchat.com/docs/full-body-tracking`, last accessed on March 3, 2021

real-world sport or activities. These findings indicate that the inclusion of accurate motion recognition is beneficial for motivating users to exercise.

Thus, there is currently a gap in the literature regarding the simultaneous reconstruction and recognition of full-body movements to develop immersive VR-based applications. Full-body motion reconstruction is required to enhance the sense of presence and provide more realistic experiences, whereas full-body motion recognition is crucial to improve health outcomes and ensure that players perform activities as instructed. Our approach addresses the emerging challenges, which we explain in the following.

1.2 RESEARCH CHALLENGES

Accurate reconstruction and reliable recognition of full-body movements in immersive VR experiences pose two main challenges.

Challenge: *Reconstructing full-body movements accurately and in real-time.*

Existing research underlines the need for a full-body motion reconstruction to improve the illusion of being in the virtual environment and to induce a sense of control over the virtual body [118, 200]. However, an accurate reconstruction of full-body movements is challenging, especially in room-scale VR experiences that enable users to move freely. Although full-body motion estimation has been possible for quite some time, the potential of motion capture systems for VR could not be fully exploited. In contrast to motion capture systems, which require users to wear tight suits or markers, the Microsoft Kinect device's main advantage is that users do not need to wear any sensors [101, 134]. However, a single Kinect camera suffers from occlusion problems [103, 134], high latency [29, 191], and insufficient data [167, 223].

Other researchers propose using more precise motion capture technologies, relying on different sensors attached to the user's body, e. g., inertial sensors [87, 99] or motion capture suits [156]. The above-mentioned New York Police Department's training simulation uses such motion capture technology to reconstruct police officers' full-body avatars [139]. However, motion capture systems that require users to wear tight suits are expensive, intrusive, and uncomfortable to wear [101, 134]. Depending on the application scenario, existing motion capture solutions may not always be suitable.

For this reason, we explore the potential of off-the-shelf VR devices to reconstruct full-body movements accurately and in real-time. Current generation VR systems provide built-in tracking and offer great potential for full-body motion reconstruction. For example, the HTC Vive not only tracks head and hand movements. Special trackers, initially developed to bring real-world objects into the virtual environment, can also be attached to the body to enable full-body tracking. For this purpose, however, appropriate methods and the number of required sensors have yet to be explored.

Challenge: *Recognizing full-body movements reliably.*

Recent research has added a growing body of evidence pointing out the need to track relevant body parts to ensure that users get the desired physical activity [126, 147].

For example, for activities involving lower-body movements, it is insufficient to track only the upper body. Hence, precise tracking of full-body movements is required to ensure that physical activities are performed as intended to improve health outcomes or to minimize injury risks [45].

Thereby, the main challenge is to analyze the entire movement execution, recognize if the desired activity has been performed correctly, and identify potential activity execution errors. Especially in exergames for rehabilitation purposes, it is crucial to recognize movements reliably to assist players, i. e., provide appropriate feedback so that players can correct activity execution mistakes and improve their movements.

1.3 RESEARCH GOALS AND CONTRIBUTIONS

According to the main challenges addressed above, the overall research goal is to reconstruct and recognize full-body movements in VR accurately. The objective of this thesis is divided into the following primary research goals, with the first two research goals tackling the first challenge and the third research goal focusing on the second challenge.

Research Goal 1: *Reducing the number of sensors for accurate full-body motion reconstruction.*

For accurate full-body motion tracking, we would have to attach one sensor to the upper and lower limbs, i. e., arm and forearm as well as thigh and calf. Such a setup would be intrusive, expensive, and uncomfortable to wear. Therefore, we aim to reduce the number of sensors the users need to wear to not restrict their movements. A reduction in sensors is also crucial in health applications, where patients with physical limitations often cannot hold or wear additional devices.

To this end, we explore relevant inverse kinematics methods to estimate the full-body pose of a user based upon the position and rotation of specific limbs, such as hands and feet, which are tracked by off-the-shelf VR devices. After implementing an initial approach, published in [43], we subsequently improve the existing approach by optimizing parameters for relevant inverse kinematics methods to determine the method that performs best regarding high accuracy and low computation time. The results of the study, published in [39], reveal that the full-body avatars are reconstructed with reasonable accuracy and low latency. We furthermore examine the statistical significance and effect sizes and show the added value of full-body motion reconstruction in an immersive multiplayer training scenario for police forces. The results of a user study, submitted for publication in [49], demonstrate the advantages of a full-body avatar in contrast to a head and hands-only representation.

Research Goal 2: *Investigating the effects of increased end-to-end latency on user experience and performance.*

The time required to reconstruct full-body movements depends primarily on the used motion capture system. Without a system that tracks human movements in real-time, users will perceive an increased end-to-end latency between their movements and the

corresponding visual feedback on the HMD. A high end-to-end latency negatively impacts the sense of control over the avatar [222] and also causes cybersickness [69]. Cybersickness is a term used to refer to symptoms such as nausea, headache, and dizziness, which users experience during or after being immersed in VR [131]. Recent research has already identified that cybersickness is crucial when developing VR-based exergames [93].

Although several researchers have investigated the cause for cybersickness [63, 131, 179], it is still unclear from what point the end-to-end latency triggers cybersickness. Therefore, in this thesis, we investigate the effects of increased end-to-end latency by examining the statistical significance and effect sizes to identify the latency threshold that induces cybersickness. The results of our user study published in [48] show that increased latency between users' movement and visual feedback on the HMD has a significant impact on user experience and elicits significant cybersickness symptoms. Furthermore, increased latency also influences user performance, causing them to need statistically significantly more time to complete a task.

Research Goal 3: *Identifying suitable sensor positions for full-body motion recognition.*
To identify execution errors, we need to attach sensors to suitable places on the body. We do not intend to distinguish among different movements but instead analyze the entire motion execution and provide appropriate feedback so that users can improve their movements. For this purpose, we employ common machine learning approaches to train and test models for relevant sensor subsets, i. e., we vary the number of sensors and the sensor positions on the body. We then investigate these models' recognition performance to determine where the sensors should best be attached to recognize full-body movements reliably. The results show that we achieve the best results when we combine a sensor on the right hand with a sensor on the left foot to recognize the upper- and lower-body movements.

To verify our approach, we developed an exergame that motivates players to train specific movements. The results on user performance, published in [47], show that we can identify player's execution errors and provide appropriate feedback due to full-body motion recognition. Such an exergame could be used to monitor the patient's improvement and to motivate them to repeat specific exercises regularly.

1.4 STRUCTURE OF THE THESIS

After this short introduction to this thesis, we describe the relevant related work regarding full-body motion tracking in Chapter 2. In Chapter 3, we analyze the requirements and introduce an overall approach. We then present an approach for full-body motion reconstruction and recognition, together with the results on quality in Chapter 4 and Chapter 5, respectively. The model on full-body motion reconstruction and recognition is then implemented and validated in the context of two serious games in Chapter 6. We conclude this thesis with a brief summary of the core contributions in Chapter 7. Finally, we provide an outlook on potential future work.

2 BACKGROUND AND RELATED WORK

This chapter provides background information about Virtual Reality (VR) and current related work regarding full-body motion tracking. Firstly, we provide an insight into VR technology and define some of the most important VR characteristics in Section 2.1. Secondly, we investigate different motion capture systems for full-body tracking and summarize related work on motion reconstruction and recognition in Section 2.2. Finally, we describe the current research gap in Section 2.3.

2.1 VIRTUAL REALITY

The definition of VR by Slater [197] includes a computer-generated virtual world with an additional fundamental element of head tracking. In other words, VR refers to virtual environments in which a turn of the head results in a corresponding change to the display [200]. Other definitions of VR include only computer-generated virtual worlds [165]; however, such a VR should instead be referred to as desktop VR [197]. Immersive VR systems should include a Head-Mounted Display (HMD) with a wide field of view, high-resolution, real-time head tracking, and should additionally include full-body motion tracking [197]. Additional studies suggest that immersive experiences also require correspondence between vision and touch [30, 234].

2.1.1 *Virtual Reality Technology*

In recent years, HMDs have become more accessible and better in quality, e. g., higher resolution, faster frame rate, and a wider field of view. While the first consumer version of the Oculus Rift provides a resolution of 2160×1200 pixels (1080×1200 per eye), the newer generation HMDs, such as the Oculus Rift S, provides a single display with a higher resolution of 2560×1440 pixels [102]. VR technology also became more popular due to the release of affordable hardware for consumers. For example, the PlayStation VR offers an HMD for consoles, with a potentially low price for an entire operating system and presumably making it accessible to a broader audience. Furthermore, even simpler and cheaper solutions, such as the Google Cardboard, can turn any smartphone with a sufficient resolution into an HMD. A comprehensive review on VR technology conducted by Anthes et al. [6] provides a detailed overview of further HMDs and input devices.

Moreover, in the past few years, VR systems with accurate head tracking and full-body tracking have been released, e. g., the HTC Vive system with controllers and additional trackers [58, 59]. In addition to the HTC Vive, the Oculus Rift CV1 also supports room-scale tracking and enables users to naturally explore and navigate within real-size virtual environments. Such modern VR systems deliver an adequate

tracking area and perform well in terms of accuracy and jitter [25, 160]. Previous studies have already shown that room-scale VR experiences (in contrast to seated VR experiences) lead to higher immersion [193]. Additional solutions work entirely independently, without annoying cables and a PC or smartphone, and do not require external tracking sensors, e. g., the Oculus Quest [109].

2.1.2 *Immersion and Presence*

The terms immersion and the sense of presence are often used interchangeably; however, they are clearly distinct. Immersion refers exclusively to the technical capabilities of a system to provide an illusion of reality [198]. Immersive VR experiences should integrate a wide range of appropriate synchronized sensory modalities. Desurvire and Wiberg [65] suggest including sound effects and haptic feedback to immerse players more deeply in the game. Hence, a truly immersive experience can be achieved by stimulating different human senses such as sound, touch, force, taste, smell, and vision using noise-canceling headphones, data gloves with force feedback, smell dispenser, and HMDs.

The sense of presence is concomitant with immersion [200]. Immersion contributes to the feeling of presence. While immersion requires a self-representation in the virtual environment, i. e., a virtual body, the sense of presence requires that users identify with this virtual body [200]. That is to say, a higher sense of presence can be achieved through a full-body avatar that mimics the user's movements in real-time. Previous studies have already reported the advantages of full-body avatars to enhance the sense of presence and to create more immersive VR applications (see Section 2.2.2).

2.1.3 *Cybersickness*

Cybersickness is similar to motion sickness and refers to negative effects users experience during or after immersion into VR [131]. Cybersickness is not restricted to VR experiences but also occurs in other visual display systems, such as large screens, curved screens, and CAVEs that consist of three or more walls on which the virtual content is projected [179]. Although its symptoms are similar to classical motion sickness, cybersickness is commonly associated with vection (a deceptive impression of self-motion) [131]. The most frequent symptoms are nausea, headache, and dizziness, albeit other symptoms such as sweating and eyestrain can also occur [113].

Causes for Cybersickness

There exist many conflicting theories aiming to explain the cause of cybersickness. The *sensory conflict theory* is the most widely accepted theory and is based on the mismatch or conflict between sensory inputs [178]. Additionally, the *postural instability theory* describes the physiological response to the inability to maintain postural stability [184]. An alternative theory is the *eye movement theory* [71]. Although several studies support these theories, there exist competing hypotheses for the cause of cy-

bersickness. For example, Bos [26] identified negative correlations between postural instability and cybersickness, whereas Lubeck et al. [142] contend that postural sway is not necessarily increased by visual motion. Another theory, also known as *subjective vertical conflict theory*, describes a conflict between subjective and sensed vertical and causes motion sickness in general [22] and cybersickness in particular [27]. Furthermore, Treisman [217] defines the *poison theory* as a possible cause for cybersickness; however, this theory is substantially different from the other mentioned before and difficult to verify [131].

Previous research carefully examined factors that have been shown to cause cybersickness [63, 131, 179]. These factors can be grouped into personal, environmental, and hardware factors and are detailed in Table 20 in Appendix A.1. The results of our meta-analysis show that current-generation VR HMDs (HTC Vive) seem to be significantly less susceptible to cybersickness compared to older HMDs (Oculus Rift DK1 and DK2) [41]. The results further show that the nature of movements is still the leading cause. Especially continuous locomotion techniques, such as walking in place or joystick-based movements, cause mismatched stimuli and provoke cybersickness. On the contrary, discrete locomotion techniques like teleportation or room-scale natural walking are less susceptible to cybersickness.

Quantifying Cybersickness

A commonly used method to quantify cybersickness is the *Simulator Sickness Questionnaire*, consisting of 16 possible symptoms (see Table 21 in Appendix A.1), which are grouped into three blocks: nausea, oculomotor, and disorientation [113]. Prior research has thoroughly investigated this questionnaire's suitability for VR environments [33], in particular, considering the differences between cybersickness and simulator sickness [205]. Additionally, the *Motion Sickness Susceptibility Questionnaire* aims to predict susceptibility to motion sickness through questions related to previous experiences [98, 177, 178]. Furthermore, the *Fast Motion Sickness Scale* has the advantage to quickly detect cybersickness during exposure without interrupting the stimuli [117]. However, the *Fast Motion Sickness Scale* mainly refers to nausea and does not measure other symptoms, such as dizziness and fatigue. Moreover, cybersickness can be detected with biosignal analysis [64, 94], e. g., the heart rate variability [95].

2.2 FULL-BODY MOTION TRACKING

Recent advances in motion capture systems for full-body tracking of a single user or even multiple users instigated interest across the VR community, i. e., in the research and game industry. Many studies track full-body movements in different means to accurately transfer them onto virtual avatars. We conducted a systematic review to investigate the potential of motion capture systems for full-body motion reconstruction in VR. The review results, published in [42], show that markerless motion capture systems, such as the Microsoft Kinect, are frequently used for singleplayer experiences. However, there is a trend of using marker-based systems for multiplayer experiences as they are more robust and do not suffer from occlusion problems. We will further

discuss the advantages and disadvantages of motion capture systems in Section 2.2.1. The review furthermore confirmed that full-body avatars can enhance the sense of presence within VR. The added value of full-body tracking will further be examined in Section 2.2.2. Additionally, relevant related work on full-body motion recognition will be investigated in Section 2.2.3.

2.2.1 *Motion Capture Systems*

There exist different motion capture systems that can accurately track full-body movements. We classify them into two groups depending on whether users need to wear a sensor or a marker on their body (marker-based motion capture systems) or not (markerless motion capture systems). The following section summarizes the relevant research results on motion capture systems with regard to accuracy, latency, and setup effort. A summary of the results based on our survey [42] is provided in Table 2.

Markerless Motion Capture Systems

Many research publications use markerless motion capture systems, such as the Microsoft Kinect, to reconstruct the full-body movements of a single user [122, 191] or multiple users [134, 204]. However, research publications often report restrictions, such as inferior estimation of human poses when a person is partially occluded [103, 196]. Recent studies suggest using multiple Microsoft Kinect devices [134] or additional inertial sensors [105] to overcome occlusion problems. Furthermore, studies report large error rates, especially for lower-body tracking [223]. In general, the Kinect can successfully track gross movements but performs poorly for fine movements [91]. Moreover, because the Kinect operates at only 30 Hz [103], it also causes high delays. Nevertheless, as such a system is inexpensive (in contrast to motion capture suits) [101, 106, 191], simple to set up [101, 191], and does not require additional sensors to be attached to the user's body [101, 134], it can still be suitable for many application scenarios.

Marker-based Motion Capture Systems

More precise motion capture systems are based on multiple markers attached to the body that are tracked by several cameras. For example, OptiTrack or Vicon, are very accurate and capable of capturing movements at high sampling frequencies [156]. However, such systems are often intrusive and uncomfortable because they usually require wearing tight suits [101, 134]. Alternatively, full-body movements can be tracked with inertial sensors, typically consisting of a gyroscope, an accelerometer, and a magnetometer, e. g., Xsens and Perception Neuron [87, 99]. However, inertial sensors drift over time due to double integration of the accelerometer measurements [121].

Furthermore, to enable full-body tracking, different off-the-shelf VR devices can be attached to the user's body, e. g., hands and feet. However, when the tracking devices are attached to specific body parts, it is necessary to use appropriate methods to estimate the full-body motions, i. e., inverse kinematics methods [8]. Previous studies

Table 2: Summarized review results on motion capture systems for full-body motion tracking.

Motion Capture System	**Advantages**	**Disadvantages**	**Possible Application Scenario**
Marker-less	• No additional sensors need to be attached to the body [101, 134] • Inexpensive [101, 106, 191] • Simple setup [101, 191]	• Occlusion problems [103, 134, 196] • High latency [29, 191] • Imprecise [167, 223], inconsistent tracking, jitter problems [134, 196]	Suitable for VR experiences when users are always directed toward the sensor.
Marker-based	• Low latency [52, 156] • High accuracy [52, 134, 156]	• Expensive [101, 134] • Intrusive, uncomfortable [134] • Complicated setup [101]	Suitable to support room-scale VR setups with multiple users.

have demonstrated the usage of VR controllers for upper-body reconstruction [12, 110, 166] or additional sensors attached to feet for full-body reconstruction [24].

2.2.2 *Full-Body Motion Reconstruction*

Immersive VR experiences require full-body motion reconstruction to give users a feeling of owning, controlling, and being inside a virtual body. This sensation towards a virtual body is also known as the sense of embodiment [118]. The following section investigates the effects of full-body motion reconstruction on the sense of control over the avatar (synchronous vs. asynchronous movements), the visual appearance of avatars (partial-body vs. full-body avatars and minimal vs. realistic avatars), and the role of user perspective (first- vs. third-person perspective). The findings of studies on motion reconstruction are detailed in Table 3.

Sense of Control Towards Virtual Avatar

The sense of control over the virtual body, also known as the sense of agency, is provided when the user's motions are mapped onto the virtual body [118]. Prior research shows that users prefer full-body control over partial control [87], although already partial control of the head movements can contribute to the sense of agency [90]. Prior research furthermore investigated the effect of delay on the sense of control. The results show that a low delay between the actual body movements and the corresponding perceived movements in VR is crucial to ensure the sense of agency [190]. Recent evidence suggests that the sense of agency and body ownership decline at latency higher

than 125 ms, whereas motor performance is affected by lower latencies, already at 75 ms and above [222].

Avatar's Visual Appearance

Previous studies have shown that realistic avatars are preferred over abstract avatars [87] and can significantly increase the sense of body ownership [130, 221]. The sense of body ownership is defined as an illusion when users perceive the virtual body as their own body [118]. Although personalized avatars can strengthen ownership [118], hyper-realistic avatars may not always be necessary [87]. In general, the avatar's appearance is less important than the degree of control or the user's perspective [87].

One of the most known methods to examine the sense of body ownership is the *rubber hand illusion*, where tactile stimulation is applied simultaneously to the real and the virtual hand. The experiments with the rubber hand show that after a few minutes, the users indeed perceive the rubber hand as their hand [9, 73]. Additional work shows that by including a physical threat to the virtual hand within immersive VR, the users experience similar responses to threat as those in comparable *rubber hand illusion* experiments [234].

Moreover, studies argue that an avatar representation is beneficial in enabling communication in collaborative virtual environments [79] and is crucial for more realistic interactions [74]. Above all, research publications show that full-body avatars increase the sense of presence in contrast to no avatars [191] or floating hands [233]. However, there is some contradictory evidence. According to Kondo et al. [122], the visual hands and feet are sufficient to induce the sense of body ownership. Nevertheless, the researchers have drawn attention to the fact that the avatar's movements must match the user's movements. Similarly, Bodenheimer and Fu [23] found no significant differences among different full-body avatars. However, the researchers confirm that the representation of a full-body avatar (compared with no avatar at all) is important.

User's Perspective

Recent studies suggest that the user's perspective, particularly the first-person perspective, is an essential factor for the sense of embodiment [118]. A study conducted by Fribourg et al. [87] shows that users also prefer a first- over a third-person perspective. However, the sense of ownership can be achieved in a first- and third-person, although it is, in general, stronger in a first-person perspective [90]. An additional study shows that the sense of agency is present regardless of the user perspective [99].

Furthermore, the third-person perspective elicits significantly different motor behavior in contrast to the first-person perspective; however, it is unclear if the difference arises due to user perspective (first- or third-person perspective) or the display type (HMD or 3D television) [212]. Additionally, the work of Born et al. [24] shows that the first-person perspective enables more accurate interactions and better performance, whereas the third-person perspective enables greater spatial awareness. Other studies on the role of user's perspective found that physiological response to a threat to a virtual body is more significant for first- compared with a third-person perspective [199].

Table 3: Summary of studies on motion reconstruction.

Authors	# of participants	Age (M ± SD, range)	Head-Mounted Display	Motion capture system	Alternated conditions	Major findings
Young et al. [233]	16 (8 female)	19 – 39	Oculus Rift DK2	Vicon	Full-body vs. partial-body	Full-body avatars provide significant improvements on the sense of presence.
Schäfer et al. [191]	42 (15 female)	19 – 79	Oculus Rift DK1	Microsoft Kinect v1	Full-body avatar vs. no avatar	Full-body avatars significantly improve the sense of presence.
Bodenheimer and Fu [23]	18 (7 female)	18 – 32	nVisor SX	Vicon	Different full-body avatars vs. no avatar	No significant effect among the different full-body avatars (realistic avatar vs. stick figure) could be found.
Latoschik et al. [130]	21 (11 female)	20.25 ± 1.21, 19 – 24	Oculus Rift CV1	OptiTrack	Different full-body avatars	Realistic avatars significantly improve the sense of body ownership in contrast to a wooden mannequin.
Yan et al. [230]	8 (5 female)	24.50 ± 4.28, 19 – 32	Oculus Rift DK1	Microsoft Kinect v1	Different full-body avatars	Users prefer a real human shape over a skeleton model.
Waltemate et al. [221]	29 (15 female)	24.00, 19 – 33	HTC Vive, CAVE	OptiTrack	CAVE vs. HMD, different realistic full-body avatars	Personalized avatars viewed through HMDs (instead of CAVE) significantly increase the sense of body ownership and presence.
Thomas et al. [212]	17 (8 female)	18 – 35	Oculus Rift DK2	Vicon	First- vs. third-person	The user's perspective influences motor behavior.
Gorisse et al. [99]	28 (all male)	22.46 ± 1.53, 20 – 26	HTC Vive	Perception Neuron	First- vs. third-person	The sense of embodiment is higher in a first-person perspective. The sense of agency is independent of perspective.

Continued on next page

Table 3: Summary of studies on motion reconstruction (continued).

Authors	# of participants	Age (M ± SD, range)	Head-Mounted Display	Motion capture system	Alternated conditions	Major findings
Born et al. [24]	44 (32 female)	21.42 ± 1.41, 18 – 24	HTC Vive	HTC Vive tracker, Microsoft Kinect v2	First- vs. third-person	The sense of body ownership significantly increases in a first-person perspective. The sense of agency is high for both perspectives.
Waltemate et al. [222]	10 (6 female)	23.20 ± 2.20	CAVE	OptiTrack	Synchronous vs. asynchronous (increasing delay)	A higher latency has a negative impact on the sense of control over the avatar.
Kondo et al. [122]	40 (all male)	22.23 ± 1.14	Oculus Rift DK2	Microsoft Kinect v2	Full-body vs. partial-body; synchronous vs. asynchronous	A partial-body avatar induces the same body ownership illusion as a full-body avatar. Synchronous movements enhance the sense of body ownership.
Debarba et al. [90]	48 (8 female)	22.60, 19 – 30	Oculus Rift DK2	Phasespace Impulse X2	Synchronous vs. asynchronous movements; first- vs. third-person	Body ownership can be achieved regardless of the user's perspectives. In the first-person perspective, the sense of agency can be induced even without full control over the virtual body.
Fribourg et al. [87]	40 (20 female)	32.50 ± 10.10	HTC Vive Pro	Xsens	Full-body vs. partial-body; synchronous vs. asynchronous; first- vs. third-person	Users prefer full-body control over the virtual body, first-person over third-person perspective, and realistic full-body over abstract avatars.

2.2.3 *Motion Recognition in Virtual Reality*

In the field of motion recognition, many studies focus on tracking one body part, usually arm or hand gestures. However, only a few studies tackle the recognition of full-body movements in VR-based applications. A summary of selected studies on motion recognition in VR is detailed in Table 4.

Gesture and Activity Recognition

The advancement of tracking technologies, such as wearable sensors, makes it possible to recognize activities and gestures. Various studies on hand gesture recognition in VR exist, e. g., using a Leap Motion [83, 106], a Myo armband [66, 104, 168], or a smartwatch [187]. Activity recognition is also quite popular in many other fields, independent of VR, e. g., to detect daily activities [56, 129, 194] or physical exercises [136, 146]. Additional scenarios include patient health monitoring [155], mobility detection [80, 215, 216], and activity recognition in disaster scenarios [135].

However, only a few studies tackle gesture or activity recognition in immersive VR. Jiang et al. [110] proposed a generic motion recognition framework based on neural networks, using only the position and orientation of the head and hands retrieved from HTC Vive HMD and corresponding controllers. The results showed that different activities could be recognized with an accuracy of 86.81%, without any false positives. Similarly, Bates et al. [12] utilize the HTC Vive controllers to train robots in household tasks, e. g., washing dishes. Using a simple decision tree, the results of the experiments revealed an accuracy of 92%. Furthermore, several studies propose locomotion techniques, such as walking in place, to enable users to explore virtual environments that are usually larger than the available real-world tracking space. Walking or jogging in place are often recognized by analyzing the sensor data of an HMD [3, 46, 133, 232] or by detecting steps from tracking data of sensors attached to the user's legs [162, 225].

Further studies aim to recognize movements without the utilization of machine learning. Various studies suggest interesting approaches for activity recognition without the usage of common machine learning algorithms. For example, different dance learning tools make use of motion matching approaches and simply compare movements between a trainer and a trainee [53, 107]. Furthermore, in the game developed by Born et al. [24], players need to fit through the hole in the wall without colliding with it. If any player's body part collides with the wall, the corresponding cubes will break apart, indicating that the player did not hold the correct posture. Although such a recognition indeed works precisely, it is only suitable to detect static poses.

Similarly, in the exergame Astrojumper, developed by Finkelstein et al. [84], movement recognition relies on body colliders. However, although this exergame provides a playful full-body fitness experience, it tracks only the player's arms. Moreover, in the exergame ExerCube, developed by Martin-Niedecken et al. [150], the researchers apply predetermined target points (determined at the calibration stage) to recognize different exercises, such as burpees, lunges, and punches.

Table 4: Summary of selected studies on motion recognition.

Authors	**# of participants**	**Age (M ± SD, range)**	**Methodology**	**Activities**	**Sensors**	**Accuracy**
Born et al. [24]	44 (32 female)	$21.42 \pm 1.41, 18-24$	Fit a shape in the wall	Different static poses	HTC Vive HMD + trackers (7) or Microsoft Kinect v2	N/A
Martin-Niedecken et al. [150]	40 (19 female)	$34.45 \pm 8.7, 16-62$	Target points	Different sports exercises	HTC Vive tracker (2)	N/A
Hoang et al. [107]	23 (10 female)	$23 \pm 3.8, 17-32$	Motion matching	Any kind of movements	(Oculus Rift DK2) + Microsoft Kinect v2	N/A
Jiang et al. [110]	3	N/A	Neutral networks	Arm gestures (e. g., come here)	HTC Vive HMD + controllers (2)	86.81%
Bates et al. [12]	12	$23-44$	Decision tree	Arm gestures (washing dishes)	(HTC Vive HMD) + controllers	92.00%
Ferracani et al. [83]	19 (8 female)	$26.4 \pm 5.8, 21-39$	Microsoft Visual Gesture Builder and Leap Motion SDK	Gestures for locomotion (e. g., walking in place)	Microsoft Kinect v2, Leap Motion, (Oculus Rift DK2)	N/A
Lee et al. [133]	9 (2 female)	$28.56 \pm 2.96, 24-33$	Peak detection	Walking in place	HTC Vive HMD	99.32%
Caserman et al. [46]	2 (1 female)	N/A	Peak detection	Step detection (left/right)	Oculus Rift DK2	$>$ 92% for walking, $>$ 60% for jogging
Ali et al. [3]	30	$20-40$	Inertial sensor of a smartphone	Walking, running, jogging	Samsung Gear VR + (PS4 controllers)	82.46%

Value of Accurate Motion Detection

Prior research publications have added a growing base of evidence supporting the value of accurate full-body recognition to improve user performance [126, 127, 147]. Especially exergames that intend to motivate players to engage in a physical activity must use appropriate interaction technology [45]. The importance of accurate physical activity detection and appropriate feedback on performance was already discussed by Lange et al. [127]. In their work, the researchers expose the limitations of the commercial video game motion sensing technologies, such as the Nintendo Wii Fit, and suggest to alternatively use a full-body motion capture system, such as PrimeSense.

Similarly, Laamarti et al. [126] point out that games need to ensure that players move as required. When players' movements are not accurately tracked, they can easily exploit this gap and not perform desired physical exercises. For example, the Nintendo Wii Fit allows players to sit on a couch and only move the Wii Remote controllers to successfully play the game without getting the desired physical exercise [126]. This gap was also investigated in a study by Marks et al. [147], showing that games that track only hands with Nintendo Wii controllers require less physical activity than the same games that track full-body movements with the Microsoft Kinect. Our work on identifying the quality criteria for serious games further supports the importance of accurate motion detection to provide appropriate feedback on performance and progress [45]. Additionally, accurate motion tracking is also crucial to ensure that exercises are performed correctly to avoid accidents or even prevent players from injuring themselves.

2.3 SUMMARY AND IDENTIFIED RESEARCH GAP

In the field of full-body reconstruction, studies have shown that full-body avatars enhance the sense of presence and communication among users in VR. Although full-body avatars, in contrast to hands-only representation, are beneficial, many research studies and popular VR games still focus on the reconstruction of upper-body movements and either show only controllers or floating hands. However, they neglect reliable reconstruction of the lower-body movements.

Furthermore, in the field of motion recognition, previous work has shown promising results in detecting daily activities. Studies that aim to recognize the users' movements in VR often focus on only one body part, e. g., hand gestures. Other studies propose recognition algorithms for well-established locomotion techniques, such as walking or jogging in place. However, only a few studies tackle the recognition of full-body movements in VR-based applications.

Moreover, little research has been conducted to reconstruct and recognize full-body movements in immersive VR simultaneously. Here, appropriate tracking technology is necessary to provide an enjoyable experience. To enhance the sense of presence, one needs to reconstruct full-body movements to give users the feeling of owning and controlling the virtual body. Additionally, accurate full-body motion recognition is crucial to ensure that users perform desired physical activities correctly, either to improve health outcomes or to lower the risk of injury.

3

OVERALL CONCEPT AND APPROACH

The gap identified in the related work makes the necessity of full-body motion reconstruction and recognition in immersive Virtual Reality (VR) experiences clear. Therefore, we analyze the requirements for full-body motion reconstruction and recognition in Section 3.1. Based on the requirement analysis, we propose our overall approach for simultaneous reconstruction and recognition of full-body movements in Section 3.2.

3.1 REQUIREMENTS ANALYSIS FOR THE FULL-BODY MOTION CAPTURE SYSTEM

In addition to a suitable Head-Mounted Display (HMD) (see Section 3.1.1), the reconstruction and recognition of full-body motion require an appropriate full-body motion capture system (see Section 3.1.2). We furthermore define requirements for precise hand tracking (see Section 3.1.3).

3.1.1 *Requirements for Head Tracking*

Full-body motion reconstruction and recognition within an immersive VR environment require a current-generation HMD. An HMD enables users to view the virtual environment from the first-person perspective. Based on our findings in the related work (see Section 2.2.2), user perspective is an essential factor to induce the sense of embodiment. However, immersive VR experiences require accurate head tracking with a high update rate and a high frame rate of an HMD to ensure low end-to-end latency. Otherwise, if the images on the HMD are displayed with high end-to-end latency, and the perceived and real movements do not match, then cybersickness can occur.

3.1.2 *Requirements for Full-Body Motion Tracking*

An appropriate motion capture system is necessary to create room-scale immersive VR experiences. As already mentioned in Section 2.1.2, immersion requires reconstruction of full-body movements, whereas the sense of presence requires that players also identify with this virtual body.

Different motion capture systems described in Section 2.2.1 have their advantages and disadvantages regarding accuracy, robustness, and latency. For example, the Microsoft Kinect is indeed very feasible for many application scenarios because the users do not need to wear any additional sensors. Such a markerless motion capture system is inexpensive and easy to set up. However, depending on the sensor position and field

of view, the Microsoft Kinect cannot always accurately track all body parts, and thus suffers from occlusion problems. Previous studies suggest using multiple Kinect cameras or additional tracking devices to overcome these occlusion problems [105, 134]. However, even without occlusion problems, the Kinect operates with only 30 Hz [103] and causes high delays in reconstructed movements. Compared with marker-based motion capture systems, such as OptiTrack and Vicon, which enable tracking at frame rates of 100 Hz and above [156], the Kinect's tracking frame rate is below these state-of-the-art devices.

In contrast to markerless motion capture systems, marker-based motion capture systems are more precise and capable of accurate full-body tracking. However, they are uncomfortable to wear and costly, making them impractical for home-based scenarios. Motion capture systems relying on multiple markers tracked by several cameras generally require time-consuming installation and appropriate arrangement of cameras. Similarly, motion capture systems based on only inertial units are often not suitable for full-body motion reconstruction or other purposes that rely on accurate positional data because they are subject to drift over time [121].

VR systems with off-the-shelf devices provide another possibility for full-body motion tracking. Current-generation VR systems with built-in-tracking offer excellent potential for full-body motion reconstruction and recognition. For example, the HTC Vive performs well in terms of accuracy and latency [25, 124, 160]. In addition to the tracking of the user's head with an HMD and hands with corresponding controllers, the HTC Vive system features special trackers initially developed to bring any real-world object, such as rackets, into the virtual environment [59]. These HTC Vive trackers can also be attached to the user's body to enable full-body motion tracking.

3.1.3 *Additional Requirements for Hand Tracking*

The HTC Vive system with the associated controllers is also suitable because it enables precise hand tracking. Alternatively, a Leap Motion controller could be used, as it is able to track individual fingers. However, Leap Motion requires keeping the hands within the sensor's field of view; otherwise, the hand gestures cannot be accurately tracked. For gesture recognition, the Leap Motion is commonly attached to the HMD. Thereby, users need to look continually at their hands to maintain tracking of hands. This disadvantage restricts the user's freedom, as one of the main features of VR is allowing users to freely look around and not force them to watch their hands. A more accurate solution for hand gesture reconstruction was recently presented by Smith et al. [201]. The researchers proposed a vision-based tracking algorithm, which relies on multi-view cameras and overcomes the occlusion problems of existing hand tracking methods.

Lost hand tracking is indeed not crucial for motion reconstruction. For example, when users do not look at their fingers, they usually do not need to be accurately reconstructed. However, lost tracking might pose a problem for motion recognition, e. g., when hand gestures should be reliably detected, even when users are not looking at their hands. Alternatively, data gloves show promising results for accurate individ-

ual finger tracking; however, they often do not provide accurate global position and orientation data of the hands. For this purpose, Noitom Hi5 VR Glove employs an HTC Vive tracker attached to the wrist to enable accurate tracking in a room-scale setup [140].

3.2 APPROACH

Based on the requirements analysis in Section 3.1, we focus on tracking full-body movements using off-the-shelf VR devices, i. e., HTC Vive HMD, controllers, and trackers. This setup is particularly suitable due to room-scale tracking of up to $10\,\mathrm{m} \times 10\,\mathrm{m}$ [58], high accuracy [25, 160, 203], and low latency [124]. The HTC Vive system is also suitable because it enables full-body tracking regardless of the user orientation. Due to its tracking system with two base stations (also called Lighthouses), which are mounted diagonally at opposite corners of tracking space, such a system suffers much less from occlusion than the Microsoft Kinect. The HTC Vive system with room-scale tracking also provides accurate yaw, pitch, roll angle, and spatial position for the HMD, controllers, and trackers.

Alternatively, we could employ newer generation HMD with inside-out tracking, such as the HTC Vive Cosmos [57]. With six camera sensors on the HMD, such an approach does not require base stations. However, because the VR devices, i. e., controllers and trackers, can only be tracked when they are in the HMD's field of view, tracking of sensors attached to the lower body might not always be possible. Nevertheless, in singleplayer experiences, the reconstruction of body parts that are out of the field of view must not be necessarily reconstructed. To still use such an approach in single and multiplayer experiences, body movements could be animated (e. g., using idle avatar animations) when the sensors cannot be tracked by the HMD and reconstructed when the sensors are in the field of view.

Consequently, based on the identified gap described in Section 2.3, we propose an approach to simultaneously reconstruct and recognize full-body movements. As shown in Figure 1, our model consists of two components: (i) full-body motion reconstruction and (ii) full-body motion recognition. On the one hand, we use off-the-shelf VR devices to reconstruct full-body avatars. Here we examine the inverse kinematics methods and optimize their parameters to reconstruct the full-body avatar with only a small number of sensors (see Section 3.2.1). Furthermore, because a high delay between the user's movements and the corresponding visual feedback causes cybersickness, we investigate the effects of increased end-to-end latency on user experience and performance (see Section 3.2.2). Therefore, the overall goal of full-body motion reconstruction is to reconstruct the avatars with high accuracy and low latency.

On the other hand, we use off-the-shelf VR devices also to recognize full-body movements using machine learning algorithms. The aim of full-body motion recognition is to ensure that players perform activities correctly and as required by the game to get the desired physical exercise. Thereby, we need to identify suitable sensor positions to recognize activity execution errors and provide appropriate feedback on the user's performance (see Section 3.2.3).

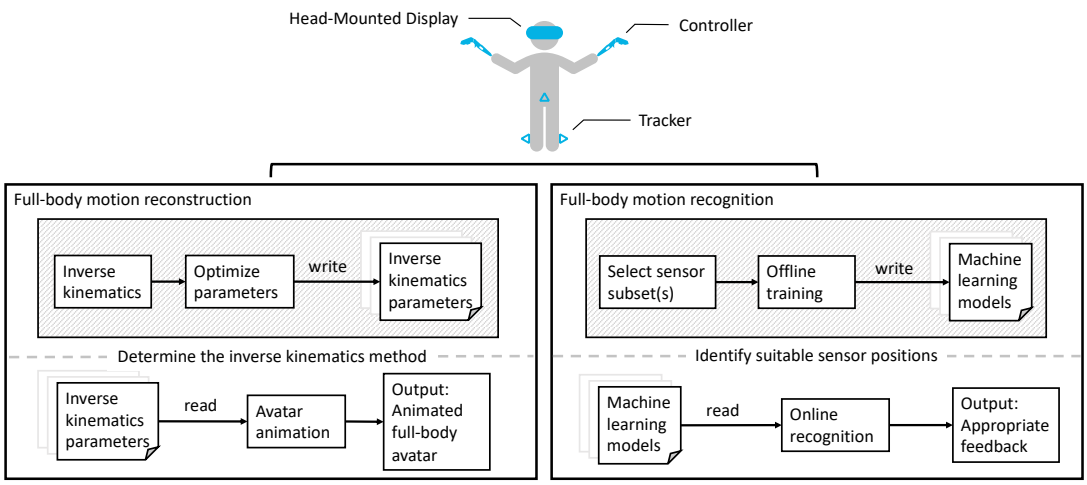


Figure 1: Model for full-body motion reconstruction and recognition.

3.2.1 *Reducing the Number of Sensors for Full-Body Motion Reconstruction*

The main goal of full-body motion reconstruction is to reduce the number of sensors users have to wear and still reconstruct full-body movements accurately. Accurate full-body motion reconstruction technically requires tracking the movements of several body parts, e. g., arm, forearm, thigh, calf, hips, and head. However, if a user would need to wear a sensor on each upper and lower limb, such a setup would be too costly. At the time this thesis is written, each HTC Vive tracker costs 119.99 EUR [59]. Additionally, due to a high number of sensors, such a setup would also be uncomfortable to wear or would restrict users' movements. Furthermore, in exergames and particularly in games for rehabilitation purposes, reducing the number of sensors is also required because patients with physical limitations often cannot hold or wear additional devices [93].

Therefore, to reduce the number of sensors, we do not explicitly track every upper and lower limb movement. Instead, we track only the position and rotation of particular body parts, i. e., head, hips, hands, and feet. However, to estimate the full-body movements accurately without knowing the exact position and rotation of every limb is challenging. To determine the position and rotation of the remaining limbs, i. e., elbow and knee, we need to solve the inverse kinematics problem. Inverse kinematics is often used in computer graphics, e. g., to animate and control highly articulated characters. Similar to character animations in games, we take advantage of inverse kinematics to transfer the users' movements onto the virtual avatars.

Character animations require a convenient model for humans, i. e., a skeleton that consists of multiple chains of links (bones) connected by joints. Figure 2 shows a skeleton, which is defined as a tree structure of bones. In general, by solving the inverse kinematics problem, we determine the rotation of every joint in a kinematic chain, e. g., the elbow or knee of the character, based only on the position and rotation of the chain's end, e. g., the hand or foot of the character. These end joints are also

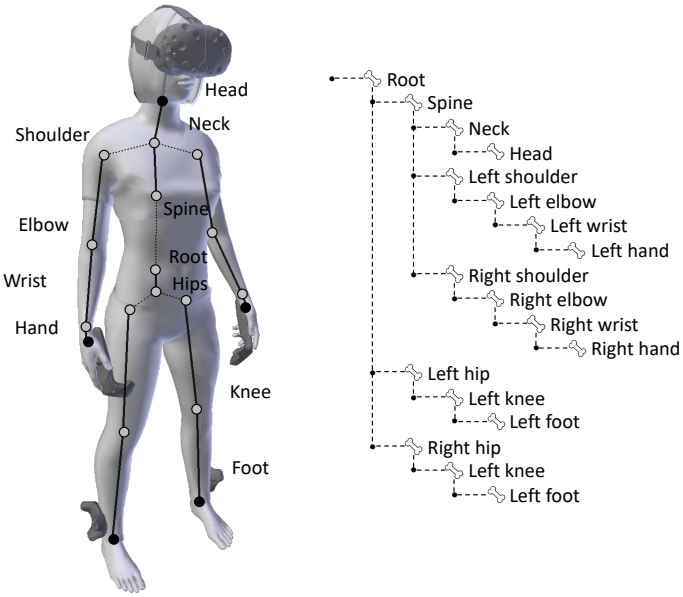


Figure 2: Hierarchically defined skeleton structure, consisting of links connected by joints. The end-effectors are highlighted in black.

commonly referred to as end-effectors and are, in our case, manipulated by off-the-shelf VR sensors, i. e., HTC Vive HMD, controllers, and trackers. Hence, due to the tracking of multiple end-effectors, i. e., hands and feet, we then determine the position and orientation of all remaining joints and reconstruct the full-body character.

However, inverse kinematics is a highly non-linear problem. An inverse kinematics solution should meet the following four requirements, which we adapted from [8]:

1. *Accuracy:* The full-body movements must be reconstructed as accurately as possible.

2. *Efficiency:* The full-body movements must be reconstructed with low latency.

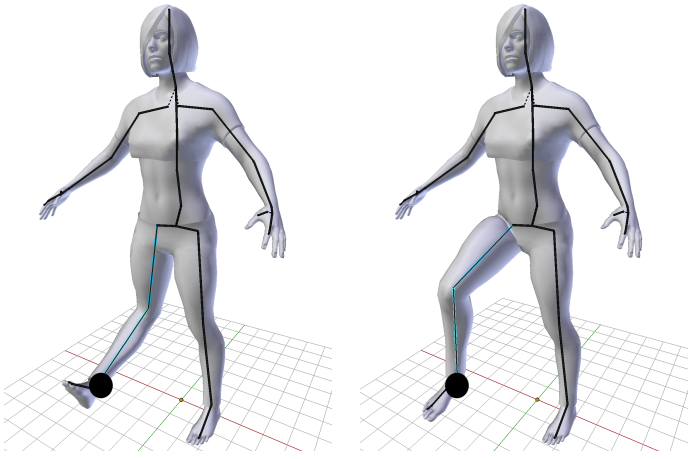
3. *Smoothness:* The inverse kinematics solver must reconstruct full-body movements without oscillations and should produce smooth movements.

4. *Scalability:* The inverse kinematics solver should provide a solution for articulated bodies with a high number of degrees of freedom.

The first and the second requirement, to reconstruct full-body movements as accurately and as fast as possible, are crucial and depend on the quality of the implemented inverse kinematics method. A method with high computational costs causes delays between the virtual and real user's movements. If full-body motions are not reconstructed precisely and involve high delays, this breaks the feeling of owning and controlling the virtual body. Ideally, the users should not perceive any delays between their and the virtual movements. Accuracy and efficiency are also generally interdependent, i. e., a higher accuracy requires more computational time and causes higher delays.

The third requirement ensures that the full-body avatars are animated smoothly. Hence, a slight change in position or rotation of the end-effector should result in a small change in the kinematic chain. Many numerical inverse kinematics methods require an appropriate damping constant to remove oscillations and jerky movements. Because visual smoothness depends on the chosen method and its parameters, we need to optimize them. As detailed in Figure 1, we first optimize the parameters of the inverse kinematics methods and then determine an appropriate inverse kinematics method that performs best regarding high accuracy, low computation time, and smooth movements. Subsequently, we utilize these optimized parameters to animate full-body avatars in real-time.

The last requirement ensures that the inverse kinematics solver provides natural full-body poses, even when the articulated body has a high number of degrees of freedom. Especially for articulated bodies with a high number of degrees of freedom, the inverse kinematics may have multiple solutions, no solution, or a unique solution. For example, already a kinematic chain with two links can cause at least two possible solutions. As shown in Figure 3, the end-effector (the right foot) can reach the desired target in multiple ways, resulting in many different poses. Some of these solutions also result in unnatural poses. Note that Figure 3 only demonstrates the problem of unnatural poses and that in a 3D environment even more possible solutions exist. In the case that multiple solutions exist, we need to ensure that the reconstructed pose looks natural. Hence, we need to define joint constraints, i. e., the rotation axes with respective minimal and maximal possible values. The full-body motion reconstruction approach, including joint constraints and parameter optimization, will be further discussed in Chapter 4.



(a) Unnatural pose (b) Natural pose

Figure 3: The inverse kinematics provides multiple solutions. In both cases, the end-effector (the character's foot) reaches the desired target, marked by the black dot. However, only the pose on the right-hand side looks natural.

3.2.2 *Investigating the Effects of Increased End-to-End Latency*

The end-to-end latency is primarily dependent on the used motion capture system. Without a high tracking update rate and a high refresh rate of the HMD, the end-to-end latency between users' movements and the corresponding visual feedback increases. As already discussed in Section 2.2.2, studies show that a high end-to-end latency negatively impacts the sense of control over the avatar. Furthermore, although it is well known that a mismatch between real and virtual head movements leads to cybersickness (see Section 2.1.3), it is still unclear when an increased end-to-end latency indeed triggers significant symptoms.

Many recent studies suggest that end-to-end latency should not be higher than 20 ms [69, 89]. However, previous work has already shown that the end-to-end latency of the HTC Vive systems is already higher, i. e., about 22 ms [160]. Other research publications conclude that a higher total system latency between 50 ms and 70 ms could be tolerated [2]. Because these findings seem to be contradictory, we intend to investigate the effects of increased end-to-end latency on user experience (especially cybersickness) and user performance. We aim to specify an end-to-end latency threshold that causes significant symptoms in users. To this end, we artificially increase a delay and quantify cybersickness using the *Simulator Sickness Questionnaire*. This questionnaire is one of the most commonly used methods for assessing the subjective severity of cybersickness (see Section 2.1.3). To analyze the effect of different end-to-end conditions, we explore the statistical significance and effect size of the total cybersickness scores.

3.2.3 *Identifying Suitable Sensor Positions for Full-Body Motion Recognition*

The main goal of full-body motion recognition is to identify suitable sensor positions to identify activity execution errors and provide appropriate feedback. Especially in exergames, physical activities need to be recognized reliably to prevent users from injuring themselves because they carry out activities incorrectly. Furthermore, without explicit tracking of relevant body parts, players can quickly learn how to cheat and will eventually fail to perform the required physical exercises as intended.

Therefore, in offline training (see Figure 1), we aim to identify where the sensors should be attached to the body to recognize full-body movements reliably. To this end, we attach several sensors to the body and build machine learning models based on relevant sensor subsets. We then analyze the model's performance to determine a suitable sensor subset. After identifying a suitable sensor subset, we can use the trained machine learning models for online recognition. Online recognition happens during run-time, while the users are immersed in VR. The goal of online recognition is to provide appropriate feedback, i. e., show users which body parts were not moved as intended by the game so that they can improve the movements. The full-body motion recognition approach, including offline and online training, will be further discussed in Chapter 5.

4

FULL-BODY MOTION RECONSTRUCTION

The full-body motion reconstruction's objective is to transfer users' movements onto virtual avatars. As already discussed in related work regarding motion capture systems (see Section 2.2.1), common devices, such as the Microsoft Kinect, implement body tracking to a certain degree. However, they are often limited by either high latency, insufficient accuracy, or occlusion problems. Other motion capture systems are either too expensive, uncomfortable to wear, or difficult to set up. The requirements analysis in Section 3.1 has shown that off-the-shelf Virtual Reality (VR) devices of the HTC Vive system are best suited to our purpose as they perform well in terms of accuracy and latency, provide global position and rotation, and enable room-scale tracking.

We track the head and hands movements with the HTC Vive Head-Mounted Display (HMD) and two controllers, whereas the movements of the hips and feet are tracked with three associated trackers. Because we aim to reconstruct full-body avatars in VR-based applications by tracking the movements of only particular body parts, we need to solve the inverse kinematics problem. An overview of the inverse kinematics problem is provided in Appendix A.2. The inverse kinematics solution then estimates the full-body motions based upon the movements of these particular body parts. In contrast to related work, often considering only the desired target position [7, 152, 220], we also include the desired target rotation to reconstruct full-body movements more

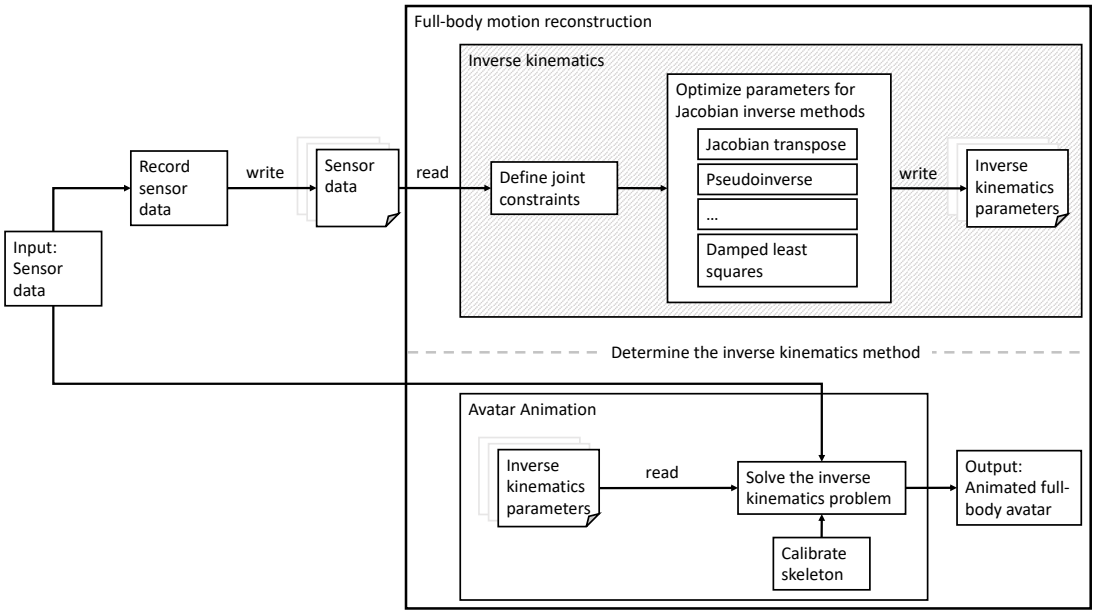


Figure 4: Full-body motion reconstruction model.

precisely. In [43], we proposed a preliminary approach to reconstruct movements of a single limb using one sensor. Subsequently, we extended our approach to reconstruct full-body movements using six sensors in [39].

As shown in Figure 4, we employ inverse kinematics methods to reconstruct full-body movements accurately with only a small number of sensors. We first define joint constraints and optimize the parameters for relevant inverse kinematics methods in Section 4.1. We describe the algorithm for full-body motion reconstruction, including skeleton calibration in Section 4.2. We then analyze the quality of the inverse kinematics methods by comparing the estimated pose with the ground truth and determine a method that performs best in terms of high accuracy and low latency in Section 4.3. After the optimal parameters and appropriate inverse kinematics method have been obtained, we use sensor data to animate the full-body avatar while users are immersed in VR. Finally, we explore the effect of increased end-to-end latency in Section 4.4.

4.1 INVERSE KINEMATICS

The goal of the inverse kinematics solver is to reconstruct full-body avatars reliably and in real-time. As already discussed in Section 3.2.1, we need to address mainly four challenges when reconstructing full-body movements: (1) *accuracy*, (2) *efficiency*, (3) *smoothness*, and (4) *scalability*. Firstly, we need to ensure that the full-body movements are reconstructed accurately and efficiently. Secondly, we need to ensure that the reconstructed full-body poses are stable and that movements are smooth. Lastly, the inverse kinematics method should always provide a solution, even for articulated bodies with high degrees of freedom. This requirement ensures that the inverse kinematics solution always provides natural poses. To prevent unnatural poses, we define joint constraints, which we discuss in Section 4.1.1. Moreover, because the accuracy, efficiency, and smoothness depend on the inverse kinematics solver, we optimize their parameters in Section 4.1.2.

4.1.1 *Joint Constraints*

Because an articulated model consists of several links connected with joints, such a model has a high number of degrees of freedom. Depending on the degrees of freedom and the end-effector target, the inverse kinematics can have several possible solutions, some of which can result in unnatural poses, as depicted in Figure 3.

To overcome this challenge, we define joint constraints and restrict possible rotations around the x, y, and z-axis. We first design an articulated character with an appropriate number of joints. Different tools, such as MakeHuman[1], allow us to easily design personalized human models with skeletons made up of different numbers of bones, e. g., a skeleton with 53 bones for game engines or an optimized skeleton for motion capture with 31 bones. However, a skeleton with 31 bones would still have a too high number of degrees of freedom in order to reconstruct full-body movements smoothly

[1] `http://www.makehuman.org`, last accessed on March 3, 2021

and without unnatural poses. Therefore, we further reduce the number of bones and finally consider an avatar model with 13 joints (see Figure 5).

All joints, except for the root, are rotational. In contrast, the root joint can rotate and translate, and thus has six degrees of freedom. The remaining joints are restricted by the rotation around the x, y, and z-axis. The articulated model consists of five body parts influenced by end-effectors. The head movements are controlled by two joints, neck and spine, with 5 degrees of freedom. Furthermore, the arms movements are controlled by three joints, wrist, elbow, and shoulder, with 2×7 degrees of freedom. Moreover, the leg movements are controlled by two joints, knee and hips, resulting in 2×4 degrees of freedom. Thus, each body part contains unambiguous joints that are independent of each other and are considered separately. Such an approach makes the inverse kinematics solver simpler and less ambiguous. Altogether, the skeleton is defined through 33 degrees of freedom.

Once a body model has been defined, we specify the upper and lower boundary (min and max angles) for each joint. Table 22 in Appendix A.3 summarizes the joints' ranges of motions, adapted from the study by Luttgens et al. [143]. We further add a tolerance of $\pm 15°$ to allow a wider range of motions. We specify a rotation vector $\mathbf{a} \in \mathbb{R}^{3\times 1}$ for each joint and set the angular limits for each axis. The local frames for individual joints are depicted in Figure 37 in Appendix A.3. For example, for the arms in the T-pose, the z-axis always points forward, whereas the y-axis is horizontal with the positive direction pointing along the bone. Consequently, the x-axis is obtained with the cross product $x = y \times z$. The constraint for the elbow is then defined as $\mathbf{a}_{elbow} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$ with angular limits $\min^x_{elbow} = 0$ and $\max^x_{elbow} = 155°$. Thus, the elbow is only allowed to rotate around the x-axis and can bend forward, but not backward. In a similar manner, we specify joint constraints for the knees and the remaining joints to prevent unnatural poses.

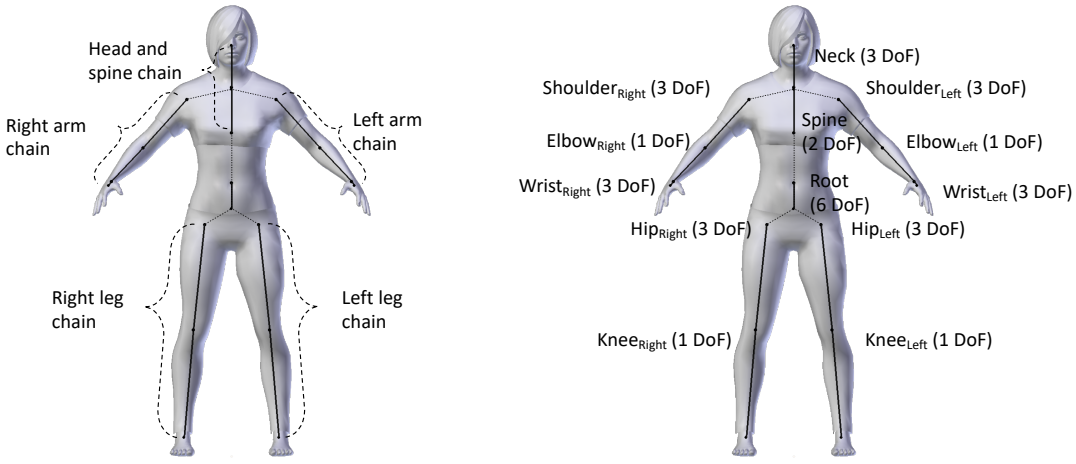


Figure 5: Articulated avatar model with 13 joints, resulting in 33 Degrees of Freedom (DoF). The bones represented by a thick line are controlled by the respective end-effector. The remaining bones (represented by a dashed line) are rigid.

4.1.2 *Parameters Optimization for Jacobian Inverse Methods*

Because the accuracy and latency of the full-body motion reconstruction depend on the chosen inverse kinematics solver, we aim to explore relevant Jacobian solutions. We optimize parameters for several popular Jacobian-based methods, such as *Jacobian transpose*, *pseudoinverse*, *damped least squares*, *damped least squares with singular value decomposition*, and *selectively damped least squares*. Jacobian-based methods offer a linear approximation of the inverse kinematics problem and usually provide very accurate results (see Appendix A.2).

However, a large Jacobian matrix leads to higher computational costs. We aim to determine a method that solves the inverse kinematics problem with low computational time and as accurately as possible. Low computational time is especially crucial in immersive VR-based applications in order to ensure high frame rates. Otherwise, if an inverse kinematics solver requires high computational time, causing high latencies between the user's movements and the corresponding visual feedback, it would eventually elicit cybersickness. We will investigate the effect of the increased latency in Section 4.4.

Defining Searching Ranges for Parameters

Depending on the method, we need to choose a constant λ carefully. The value for λ specifies a damping constant for *damped least squares* and *damped least squares with singular value decomposition*, whereas it specifies a "maximum permissible change in any joint angle in a single step" for *selectively damped least squares* [36]. For *damped least squares*, a larger value for λ makes the solution for $\Delta\theta$ (see Equation 13) well behaved near singularities; however, if the value is too high, it will lower the convergence rate and the accuracy in reaching the targets [36].

Furthermore, we need to define how close the end-effector needs to approach the target, i. e., the maximum error in position e_{maxPos} and rotation e_{maxRot}. With the appropriate e_{maxPos} and e_{maxRot}, the inverse kinematics solver then terminates when the end-effector is close enough to the desired target or when there is no significant change between current and desired rotation. However, because the end-effector may not always be able to reach the desired target, we further need to specify the maximal number of iterations $\mathrm{it}_{\mathrm{max}}$. The desired targets are not reachable when they are too far away or when the joint constraints permit a certain movement. Hence, we need to specify appropriate values so that the algorithm terminates when either the end-effector is close enough to the target or when the maximum number of iterations has been reached.

We define search intervals with values adapted from the study by Buss and Kim [36] as follows:

SEARCH RANGE FOR λ: For *damped least squares*, Buss and Kim [36] suggest $\lambda = 1.1$. A higher value makes the solution more stable; however, it lowers the convergence rate. For this reason, we first try a smaller value for λ and gradually increase it until we remove oscillations and jerky movements. Therefore, for *damped least*

squares and *damped least squares with singular value decomposition*, we search for the optimal damping constant in the range of $0.05 \leqslant \lambda \leqslant 1.5$ and increase the value by 0.05 in each step.

Furthermore, for *selectively damped least squares*, the results of the study by Buss and Kim [36] show that $\lambda = \pi/4$ (i. e., 45°) is a robust choice. However, because the tracking update rate of the HTC Vive is high, i. e., 120 Hz [124], the difference in position or rotation between two frames is relatively small. Consequently, we search for the optimal value in the range of $\pi/120 \leqslant \lambda \leqslant \pi/4$ (i. e., $1.5° \leqslant \lambda \leqslant 45°$) and gradually increase the value by $\pi/120$ (i. e., 1.5°).

SEARCH RANGE FOR e_{maxPos} AND e_{maxRot}: We search for the optimal maximum error in the range of 0.0001 and 0.1 and increase values exponentially in each step, i. e., $e_{maxPos}, e_{maxRot} \in \{0.0001, 0.001, 0.01, 0.1\}$. The values are given in meters and radians, respectively.

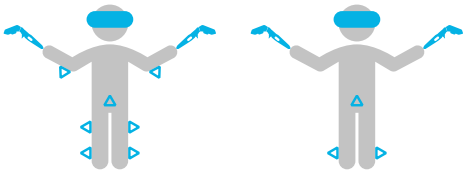
SEARCH RANGE FOR it_{max}: The results obtained by Buss and Kim [36] show that all methods, on average, need 192 iterations to solve the inverse kinematics problem. Consequently, we search for the optimal it_{max} in the range of 1 and 200 and increase the value by 1 in each step.

Optimizing Parameter Values

The goal of the parameter optimization is to choose appropriate values for the constant λ, the maximum error in position and rotation ($e_{maxPosition}$ and $e_{maxRotation}$), and the maximum number of iterations (it_{max}). To optimize parameter values for each Jacobian method, we create a target database containing raw sensor data, i. e., the position vector and rotation vector represented as a quaternion. We attach several HTC Vive trackers to the body, i. e., elbow, knee, feet, and hips, to track the movements of the upper and lower limbs (see Figure 6a). The head movements are captured with the HTC Vive HMD, whereas the arm movements are captured with two controllers. Note that for the full-body motion reconstruction, only six sensors are required (see Figure 6b) and additional trackers are only necessary for the quality assessment. The sensor data were collected at a constant frame rate (90 frames per second).

The target data set for training consists of different tasks, adapted from the related work [91, 167]: *shoulder forward flexion*, *shoulder abduction*, *shoulder horizontal abduction*, *rotation with arms at side* and *with arms in abduction*, *hand pronation* as well as *elbow*, *knee*, and *hip flexion*. Each task was repeated ten times for each side. The movement description is provided in Table 23 in Appendix A.4. We chose these tasks deliberately as they specifically affect the body parts that are reconstructed by solving the inverse kinematics problem.

To find an optimal solution in terms of low error and latency, we need to compare the parameter values with each other. Because measures are on a different scale, we use the *z-score* to standardize the data before further analysis [188]: $z = (X - \bar{X})/s$, where X is the original data value, $\bar{X}$ the sample mean, and s the standard deviation. The mean of the *z-score* is always zero, while the variance is one. The *z-score* enables us



(a) Ten sensors (b) Six sensors

Figure 6: Sensor position for parameter optimization (left). Note that the inverse kinematics approach uses only six sensors to estimate a full-body pose (right).

to compare two (or more) parameters with different units. Finally, for a multi-criteria optimization with regard to the number of iterations as well as the error in position and rotation, we perform regression and minimize the root mean square error.

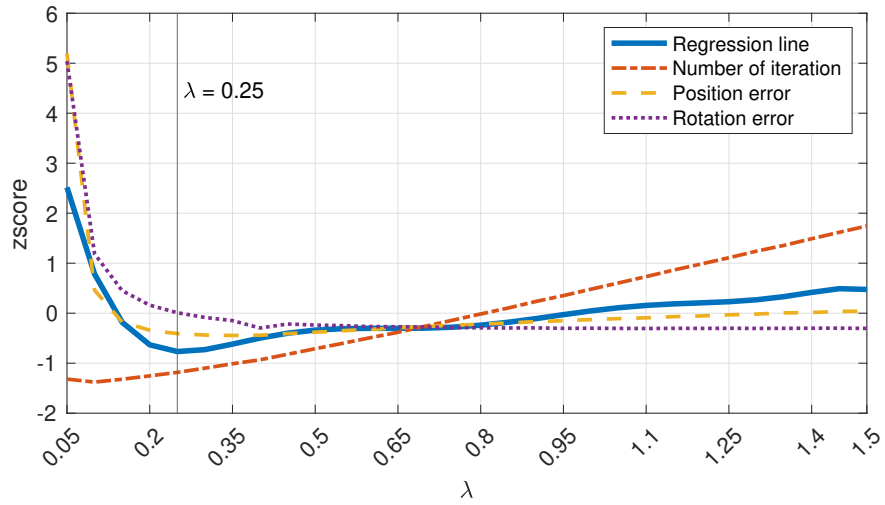
OPTIMIZING λ: To optimize λ, we choose the maximum number of iterations, i. e., $it_{max} = 200$ and the highest accuracy degree, i. e., $e_{maxPos} = e_{maxRot} = 0.0001$. While these parameters are fixed, we gradually increase λ.

The results in Figure 7 show how the value for damping constant λ influences the error in position and rotation as we as the number of iterations. Note that the *Jacobian transpose* and *pseudoinverse* do not require a damping constant and are therefore excluded from Figure 7.

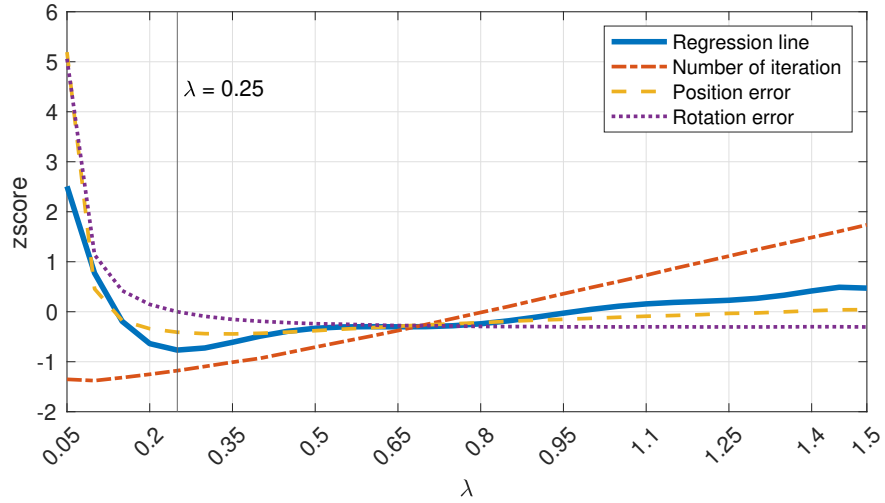
As shown in Figures 7a-7b, the error in position and rotation decrease substantially with larger λ for *damped least squares* with and without *singular value decomposition*, while the number of iterations increases. In other words, shaking movements and oscillations occur with a too small λ. In contrast, with a too large λ, latency appears between the actual and the reconstructed movements. Both methods perform best with a damping constant $\lambda = 0.25$. The pattern for both methods is nearly the same, and indeed both methods also perform similarly in terms of error and performance time (see Section 4.3.1).

Furthermore, as shown in Figure 7c, the position error decreases with larger λ for *selectively damped least squares*, while the rotation error increases. The method performs best with $\lambda = \pi/12$. Alternatively, if we would optimize the parameters more in terms of the position error rather than position and rotation error, we could choose a larger λ.

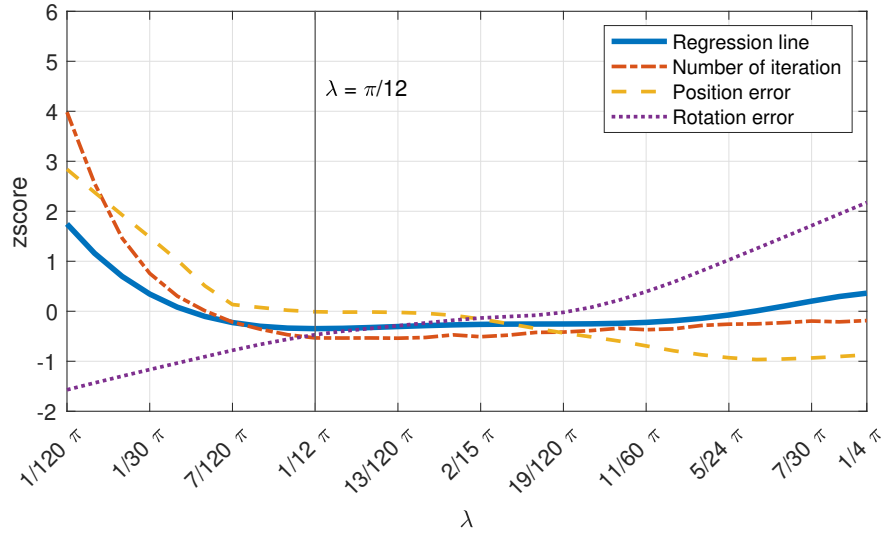
Compared to parameter values from Buss and Kim [36] with $\lambda = 1.1$ for *damped least squares (with singular value decomposition)*, the optimization led to a smaller number of iterations and thus also a lower latency, which is a crucial factor for VR-based applications. Similarly, compared to parameter values from Buss and Kim [36] with $\lambda = \pi/4$ for *selectively damped least squares*, the optimization resulted in a better balance between position and rotation error.



(a) *Damped least squares*



(b) *Damped least squares with singular value decomposition*



(c) *Selectively damped least squares*

Figure 7: Position and rotation error as well as the number of iterations, depending on the λ.

Table 5: Number of iterations it_{num} depending on the chosen degree of accuracy.

	0.0001	0.001	0.01	0.1
JT	25.83	21.80	11.14	1.97
JPI	198.44	196.35	182.52	125.48
DLS	6.24	5.41	3.64	1.28
SVD-DLS	6.25	5.43	3.64	1.28
SDLS	4.07	3.94	3.00	1.27

Abbreviations: *Jacobian Transpose* (JT), *Jacobian Pseudoinverse* (JPI), *Damped Least Squares* (DLS), *Damped Least Squares with Singular Value Decomposition* (SVD-DLS), and *Selectively Damped Least Squares* (SDLS).

OPTIMIZING e_{maxPos} AND e_{maxRot}: To optimize the e_{maxPos} and e_{maxRot}, we again use the maximum number of iterations, i. e., $it_{max} = 200$. However, because the optimal λ is already known, we use $\lambda = 0.25$ for *damped least squares* and *damped least squares with singular value decomposition* and $\lambda = \pi/12$ for *selectively damped least squares*. With these fixed parameters, we gradually increase the degree of accuracy.

In general, a smaller maximum error in position e_{maxPos} and rotation e_{maxRot} provide more accurate results. However, a smaller requested accuracy affects the number of iterations and the total time needed to provide a solution. The data in Table 5 show that the number of iterations increases with higher chosen accuracy. Comparing the *damped least squares* and *damped least squares with singular value decomposition*, we observe that both methods need a similar number of iterations. However, as shown in Figure 8, the latter method is more computationally intensive due to the *singular value decomposition* computation.

Furthermore, the results show that although the *selectively damped least squares* method needs the lowest number of iterations, it needs for almost all accuracies

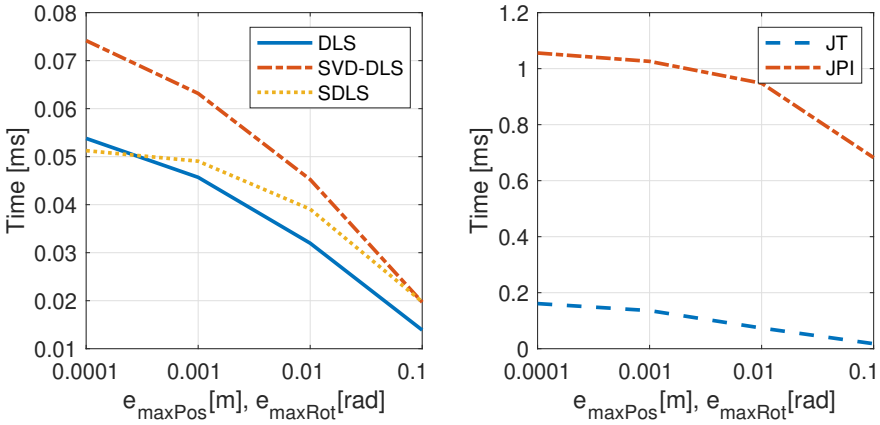


Figure 8: Average total time t_{total} to reach the target goal for the requested degree of accuracy.

more time than the *damped least squares* method. Additionally, the results for *Jacobian transpose* show that the method requires, for almost all accuracy degrees, significantly more iterations than all other methods and, with the exception of *pseudoinverse*, it is also the slowest method (see Figure 8). The *pseudoinverse* is not robust and is unable to provide a stable solution without oscillations.

OPTIMIZING $\mathtt{it}_{max}$: To optimize the maximum number of iterations $\mathtt{it}_{max}$, we choose the optimal λ values for each method and the highest accuracy degree, i. e., $e_{maxPos} = e_{maxRot} = 0.0001$. While these parameters are fixed, we gradually increase $\mathtt{it}_{max}$. Because the *pseudoinverse* is unstable and does not provide an accurate solution even with up to 200 iterations, we discarded it from the parameter optimization.

The maximum number of iterations affects the overall error. As expected, with a higher number of iterations, the error in position and rotation decrease. Figure 9 depicts the number of iterations needed to reduce the error between the target and end-effector. The results show that the *Jacobian transpose* converges much slower than other methods and needs more than 10 iterations to reach the target rotation, while it needs at least 30 iterations to reach the target position.

The *damped least squares* and *damped least squares with singular value decomposition* methods perform similarly and do not need more than 10 iterations to attain the target position and rotation. In fact, the *damped least squares with singular value decomposition* method does not have any qualitative improvements compared to

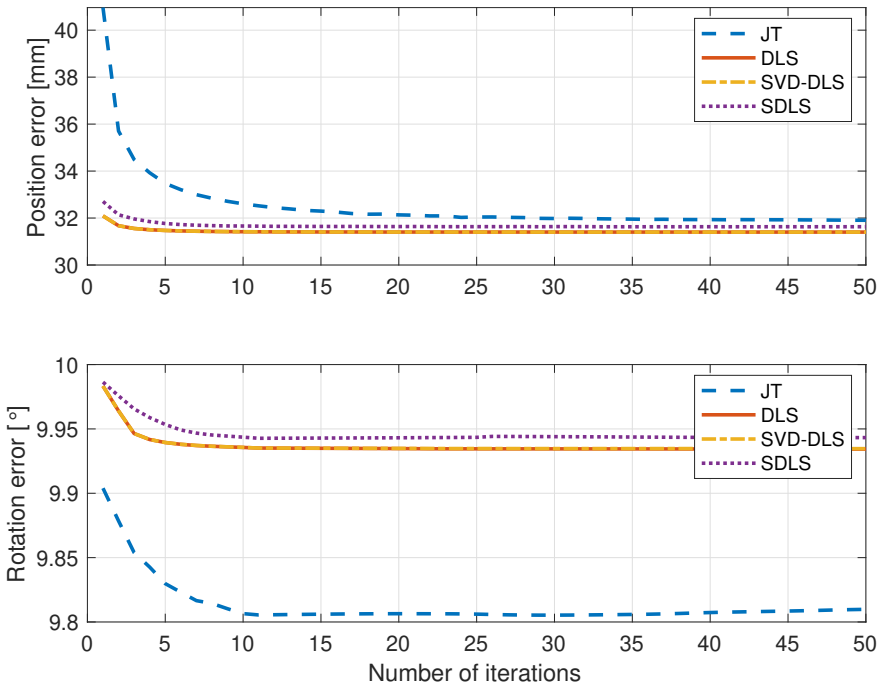


Figure 9: The number of iterations needed to reach the target goal against the position and rotation error.

the *damped least squares* method without *singular value decomposition* and is also significantly slower. Similarly, in terms of position error, the *selectively damped least squares* method converges with less than 10 iterations; however, in terms of rotation error, the method needs more iterations than *damped least squares* and *damped least squares with singular value decomposition*.

The optimal parameters for each Jacobian method are summarized in Table 6. The results revealed that the *damped least squares* and *damped least squares with singular value decomposition* methods perform best regarding the error in position and rotation as well as the number of iterations with $\lambda = 0.25$, whereas the *selectively damped least squares* method performs best with $\lambda = \pi/12$. Furthermore, all methods, except for *pseudoinverse*, converge with less than 30 or even 20 iterations. Moreover, parameter optimization revealed that regardless of the degree of accuracy, all methods solve the inverse kinematics problem in real-time, i. e., we choose $e_{\mathrm{maxPos}} = e_{\mathrm{maxRot}} = 0.0001$.

4.2 AVATAR ANIMATION

With the optimal parameters and through the efficient implementation of the iterative method for solving the inverse kinematics problem, we animate an articulated avatar model and synchronize the virtual body's movements with the user's movements. Consequently, we enable users wearing an HMD to see an avatar that moves in the same way as they do, either from the first-person perspective or in a virtual mirror. We use the HTC Vive HMD and two controllers held in the hands to track the upper-body movements. Additionally, we use HTC Vive trackers attached to the hips and ankles to track the lower-body movements. Figure 6b shows the positions of the attached sensors.

This section includes the implementation of the inverse kinematics method to animate avatars, which we initially proposed in [43]. Because the avatar animation should work regardless of the user's height and the position as well as rotation of the attached devices, we first describe the process of the skeleton calibration in Section 4.2.1. Afterward, we present an inverse kinematics solver with optimal parameter values in Section 4.2.2.

Table 6: An overview of the optimized parameter values for each Jacobian method.

	JT	JPI	DLS	SVD-DLS	SDLS
λ	/	/	0.25	0.25	$\pi/12$
e_{maxPos} [m]	0.0001	0.1	0.0001	0.0001	0.0001
e_{maxRot} [rad]	0.0001	0.1	0.0001	0.0001	0.0001
$\mathrm{it}_{\mathrm{max}}$	30	200	20	20	20

Abbreviations: *Jacobian Transpose* (JT), *Jacobian Pseudoinverse* (JPI), *Damped Least Squares* (DLS), *Damped Least Squares with Singular Value Decomposition* (SVD-DLS), and *Selectively Damped Least Squares* (SDLS).

4.2.1 *Skeleton Calibration*

Skeleton calibration is required so that full-body motion reconstruction works regardless of the user's height. In particular, with the skeleton calibration, we ensure that the motion reconstruction for the arms and legs works regardless of the position and rotation of the HTC Vive sensors.

To calibrate the skeleton, the user has to stand in a T-pose, as shown in Figure 10. We first transform, rotate, and scale the character so that it is in the same position, oriented in the same direction, and has the same height as the user. The initial transformation matrix $\mathbf{T}_{init} \in \mathbb{R}^{4\times1}$ of a character is given by:

$$\mathbf{T}_{init} = \mathrm{Trans}(\mathbf{p}_{init}) \cdot \mathrm{Rot}(\mathbf{q}_{init}) \cdot \mathrm{Scale}(s), \tag{1}$$

where $\mathbf{p}_{init} \in \mathbb{R}^{4\times1}$ is the position, $\mathbf{q}_{init} \in \mathbb{R}^{4\times1}$ the rotation, and s the scale factor of the character. Quaternions ($\mathbf{q}$) represent rotations and are utilized because they are very simple to use, efficient, robust, and do not suffer from Gimbal lock, i. e., "the loss of one degree of rotational freedom" [195]. The scale factor can be determined as $s = h_{hmd}/h_{head}$, where h_{hmd} is the actual user's height of the eye level and h_{head} the height of the character's head bone.

Before we calibrate the skeleton, we need to ensure that all coordinate values are in the same reference frame. Figure 10 shows the two reference frames, along with the world reference frame. The avatar's reference frame is attached to the floor and is a right-handed reference frame with the x-axis pointing to the left, the y-axis pointing backward, and the z-axis pointing upward. However, the sensor measurements of the

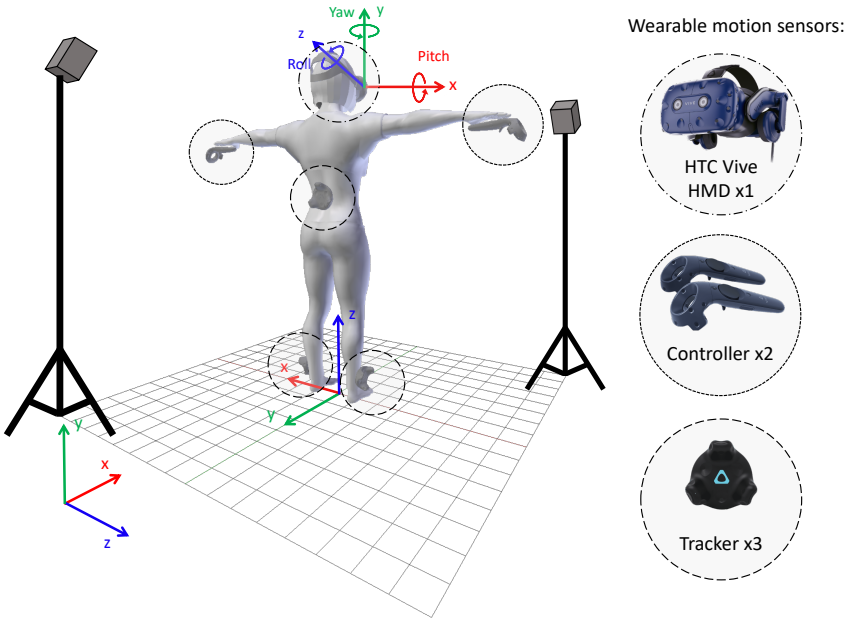


Figure 10: Calibration pose: A user wearing an HMD needs to stand in a T-pose, holding two controllers, while three trackers are attached to the feet and hips.[2]

[2]Image partially adapted from `https://www.vive.com/`

HTC Vive devices are provided in another reference frame. For example, the sensor measurements of the HMD are provided in a head-fixed reference frame, where the x-axis points to the right, the y-axis points upwards, and the z-axis points backward. Because the sensor data of the HTC Vive devices are not presented in the same reference frame as that of the character, we first need to transform the coordinate values between the two reference frames.

Let $\mathbf{q}_{raw}[t] \in \mathbb{R}^{4\times1}$ and $\mathbf{p}_{raw}[t] \in \mathbb{R}^{4\times1}$ be a quaternion and a vector representing the raw sensor rotation and position at time step t. To transfer the sensor measurements to the character reference frame, we first multiply the inverse initial quaternion $\mathbf{q}_{init}^{-1}$ by the raw quaternion $\mathbf{q}_{raw}$ (see Equation 2). Similarly, we also multiply the inverse transformation matrix $\mathbf{T}_{init}^{-1}$ by the raw position vector $\mathbf{p}_{raw}$ (see Equation 3). Thus, $\mathbf{q}_{des}[t] \in \mathbb{R}^{4\times1}$ and $\mathbf{p}_{des}[t] \in \mathbb{R}^{4\times1}$ represent the desired rotation and position at time t with respect to the character reference frame:

$$\mathbf{q}_{des}[t] = (\mathbf{q}_{init})^{-1} \cdot \mathbf{q}_{raw}[t], \tag{2}$$

$$\mathbf{p}_{des}[t] = (\mathbf{T}_{init})^{-1} \cdot \mathbf{p}_{raw}[t] \tag{3}$$

After the sensor measurements have been transformed into the avatar's reference frame, we need to determine the offset position and rotation for each end-effector. The determination of the offset values is necessary so that the full-body motion reconstruction works independently of the position and rotation of the attached sensors. Therefore, we calculate the distance between the desired (the user's joints, i. e., sensors) and the actual (character's joints) rotation and position at the time of calibration, thus at time step $t = 0$:

$$\mathbf{q}_{offset} = (\mathbf{q}_{des}[0])^{-1} \cdot \mathbf{q}_{act}[0], \tag{4}$$

$$\mathbf{p}_{offset} = (\mathrm{Trans}(\mathbf{p}_{des}[0]) \cdot \mathrm{Rot}(\mathbf{q}_{act}[0]))^{-1} \cdot \mathbf{p}_{act}[0] \tag{5}$$

where $\mathbf{q}_{des}, \mathbf{p}_{des} \in \mathbb{R}^{4\times1}$ represent the desired and $\mathbf{q}_{act}, \mathbf{p}_{act} \in \mathbb{R}^{4\times1}$ actual rotation and position, respectively. Note that offset quaternion and position are calculated only once and remain the same during the runtime of the application.

Finally, after the offset values have been determined, we apply the offset and calculate the final quaternion $\mathbf{q}_{final} \in \mathbb{R}^{4\times1}$ and position $\mathbf{p}_{final} \in \mathbb{R}^{4\times1}$:

$$\mathbf{q}_{final}[t] = \mathbf{q}_{des}[t] \cdot \mathbf{q}_{offset}, \tag{6}$$

$$\mathbf{p}_{final}[t] = \mathrm{Trans}(\mathbf{p}_{des}[t]) \cdot \mathrm{Rot}(\mathbf{q}_{final}[t]) \cdot \mathbf{p}_{offset}. \tag{7}$$

4.2.2 *Solving the Inverse Kinematics Problem*

Solving the inverse kinematics problem is especially challenging when the virtual character contains many joints. If we need to consider a high degree of freedom, the computational costs for inverting the Jacobian method will increase. Therefore, as described in Section 4.1.1, we reduce the number of bones in an articulated avatar model and consider for each end-effector only certain joints to minimize the computational effort. Furthermore, because we perform the inverse kinematics on individual limbs,

i. e., arms and legs, they do not affect each other. Such a Jacobian matrix makes the inverse kinematics solver simpler and less ambiguous. An overview of the algorithm is presented in Figure 11. The algorithm was written in C++, using low-level game API and hardware abstraction library Kinc [123]. The source code and short videos are available at [38].

The algorithm finds an inverse kinematics solution in several iterations and works as follows:

1. Calculate the Jacobian matrix
2. Invert the Jacobian matrix
3. Calculate joint angles
4. Update joint angles considering the joint constraints
5. Update new positions
6. Check for convergence

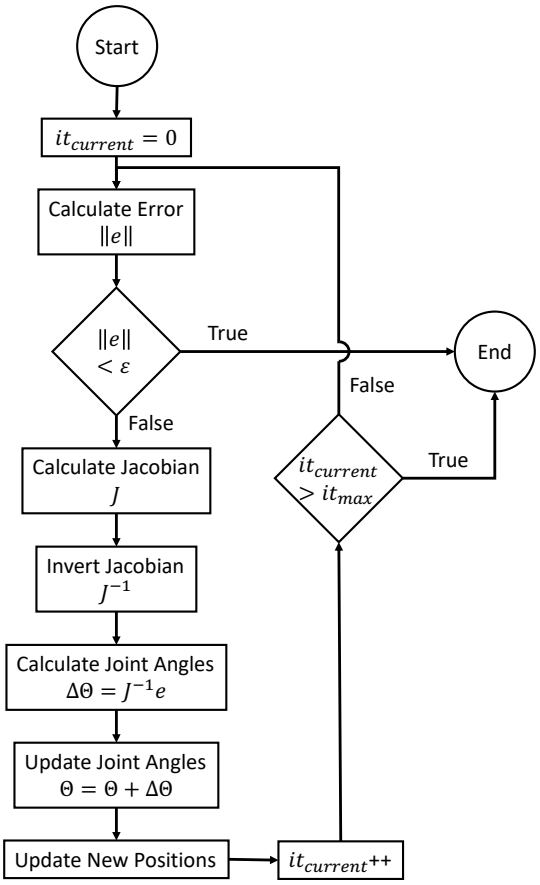


Figure 11: Flowchart for solving the inverse kinematics problem.

STEP 1: In the first step, we calculate the Jacobian matrix for the i^{th} end-effector using Equation 11. Because the character skeleton consists only of revolute joints (rotational), the entries in the Jacobian matrix are determined as [34]:

$$\mathbf{J}(\theta)_{ij} = \frac{\partial \mathbf{s}_i}{\partial \theta_j} = \begin{pmatrix} \mathbf{v}_j \times (\mathbf{s}_i - \mathbf{p}_j) \\ \mathbf{v}_j \end{pmatrix}, \text{ for } j = 1, \ldots, n. \tag{8}$$

where $\mathbf{v}_j \in \mathbb{R}^{3\times1}$ is the unit vector pointing along the rotation axis of the j^{th} joint, $\mathbf{s}_i \in \mathbb{R}^{3\times1}$ is the position of the i^{th} end-effector, and $\mathbf{p}_j \in \mathbb{R}^{3\times1}$ is the current position of the j^{th} joint.

Depending on the number of degrees of freedom n, the Jacobian matrix $\mathbf{J}$ will result in a $6 \times n$ matrix. Due to our modified skeleton with 13 joints, we reduced the overall number of degrees of freedom. As described in Section 4.1.1, we consider two joints with a total of five degrees of freedom for the head. Furthermore, we consider three joints with a total of seven degrees of freedom for the arms. Additionally, we consider two joints with a total of four degrees of freedom for the feet. Hence, the Jacobian matrix for the head contains 6×5 entries, for the arms 6×7 entries, or for the legs 6×4 entries. Such a smaller matrix is easier to invert, and thus reduces the computational costs.

STEP 2: In the second step, we adapt the existing algorithms from [35] to invert the Jacobian matrix. Thus, as the Jacobian matrix is a non-square matrix, we approximate its inverse by using the methods described in Appendix A.2, i. e., *Jacobian transpose*, *pseudoinverse*, *damped least squares*, *damped least squares with singular value decomposition*, or *selectively damped least squares*.

STEP 3: In the third step, we apply Equation 13 to obtain the joint angles $\Delta\theta$. Therefore, we multiply the inverse of the Jacobian matrix $\mathbf{J}^{-1}$ by the difference between the desired and current position as well as rotation of the end-effector $\mathbf{e}$.

STEP 4: In the fourth step, we apply the new rotation $\Delta\theta$ to the joints. However, before new rotations are applied, we need to ensure that the joint constraints are respected, as described in Table 22 in Appendix A.3. Otherwise, the character hand or foot will reach the desired goal; however, the individual joints will eventually cause unnatural movements. The joint rotations are enforced through clamping between a lower bound (LB) and an upper bound (UP):

$$\theta = \begin{cases} LB & \text{if } \theta + \Delta\theta < LB \\ UB & \text{if } \theta + \Delta\theta > UB \\ \theta + \Delta\theta & \text{otherwise} \end{cases} \tag{9}$$

STEP 5: In the fifth step, we apply forward kinematics to update the new end-effector positions.

STEP 6: Finally, in the last iteration step, we check for convergence. We examine if the end-effector has already reached the desired position and rotation. We compare the error $\mathbf{e} = [\mathbf{e}_{\mathrm{pos}}; \mathbf{e}_{\mathrm{rot}}]$ with the threshold $\|\mathbf{e}\| < \varepsilon$, where $\varepsilon = [e_{\mathrm{maxPosition}}, e_{\mathrm{maxRotation}}]$. Hence, the algorithm terminates when either the end-effector is close enough to the desired target or when there is no significant change between the current and desired rotation.

Furthermore, because the end-effector might not be able to reach the target (when the desired target is out of range), the algorithm will terminate when the maximal iteration number it_{max} has been reached. The optimal parameters for the maximum error in position and rotation ($e_{\mathrm{maxPosition}}$ and $e_{\mathrm{maxRotation}}$) as well as the maximum number of iterations (it_{max}) were identified in the parameter optimization process in Section 4.1.2.

4.3 QUALITY OF FULL-BODY MOTION RECONSTRUCTION

The quality of full-body motion reconstruction is determined by accuracy and performance time. The accuracy is defined as the error between the desired target and actual end-effector position and rotation. The performance time depends on the number of iterations and time per iteration. We evaluate the accuracy of the full-body motion reconstruction by comparing the estimated pose (position and rotation of the reconstructed limbs of an avatar) with the ground truth in Section 4.3.1. Additionally, we evaluate the performance time by calculating the number of iterations, the time per iteration, and the total time needed to reach the desired target as accurately as possible. Subsequently, we assess the subjective quality of the reconstructed full-body avatar in Section 4.3.2.

4.3.1 *Accuracy and Performance Time of Full-Body Motion Reconstruction*

Previously, we obtained the optimal parameter values for each Jacobian method in Section 4.1 (see Table 6). We now use these optimized parameters and calculate the error and performance time. To determine the recognition accuracy of the Jacobian inverse methods, we compare the estimated pose against a baseline. To this end, we create a target database containing raw sensor data. Similar to the parameter optimization, we attach several HTC Vive trackers to the body, including elbows and knees (see Figure 6a). Altogether, we compare the ground truth of ten joints, i. e., head, left and right hand, left and right elbow, left and right knee, left and right foot, as well as hips.

The target data set for testing consists of different tasks, similar to those used in related work [91, 220, 223]: *standing*, *walking*, *punching*, and *kicking*, as well as doing *squats* and *lunges*. Each task was either repeated ten times (for each side) or lasted 30 seconds. The movement description is provided in Table 24 in Appendix A.4.

Comparison of Jacobian Inverse Methods in Terms of Error and Performance Time

The position (e_{pos}) and rotation (e_{rot}) error define the accuracy of the motion reconstruction. Additionally, the performance or total time (t_{total}) depends on the number of iterations (it_{num}) needed to reach the target and the time per iteration (t_{it}). Results regarding accuracy and performance time are detailed in Table 7. Runtimes are specified in milliseconds (ms) and were measured on Intel Core i5 CPU 2.9 GHz with 8 RAM while rendering at constant 90 frames per second.

In general, we observe for all Jacobian inverse methods, except for *pseudoinverse*, a small error in position and rotation error. The *damped least squares* and *damped least squares with singular value decomposition* methods perform best in terms of position error $e_{pos} = 37.23\,mm$, whereas the *Jacobian transpose* performs best in terms of rotation error $e_{rot} = 13.62°$. However, the quality of the Jacobian method does not only depend on the position and rotation error. It also depends on the total time t_{total}, i. e., the number of iterations it_{num} and the time per iterations t_{it}. Regarding only the total time, the *damped least squares* method again is the most efficient method with $t_{total} = 0.07\,ms$. The second fastest method is *selectively damped least squares* with $t_{total} = 0.08\,ms$.

To determine which method performs best regarding the accuracy and performance time, we calculate weighted sum, which is one of the most known and simplest multi-criteria decision-making methods [148]. Therefore, we first normalize the data and calculate the weighted sum as: $U = w_{pos} \cdot e_{pos} + w_{rot} \cdot e_{rot} + w_{time} \cdot t_{total}$ for each inverse kinematics method. We need to choose appropriate weights, with all weights summing to one. Because we focus on the position and rotation error, we choose for e_{pos} and e_{rot} an equal weight of $w_{pos} = w_{rot} = 0.4$. As all methods solve the inverse kinematics problem in less than 11.11 ms, and thus satisfy the refresh rate of the HTC Vive HMD, which is 90 Hz, we choose a smaller weight for the total time, i. e., $w_{time} = 0.2$.

Table 7: Comparison in terms of error and performance time.

Method	**Error in position**	**Error in rotation**	**Number of iterations**	**Time per iteration**	**Total time**
	e_{pos} [mm]	e_{rot} [°]	it_{num}	t_{it} [ms]	t_{total} [ms]
JT	38.33	13.62	18.79	0.009	0.17
JPI	454.11	92.16	197.35	0.006	1.27
DLS	37.23	13.69	7.46	0.01	0.07
SVD-DLS	37.23	13.69	7.47	0.016	0.12
SDLS	37.62	13.70	4.33	0.018	0.08

Abbreviations: *Jacobian Transpose* (JT), *Jacobian Pseudoinverse* (JPI), *Damped Least Squares* (DLS), *Damped Least Squares with Singular Value Decomposition* (SVD-DLS), and *Selectively Damped Least Squares* (SDLS).

According to the weighted sum, the *damped least squares* outperforms the other methods. It performs best in terms of position error and solves the inverse kinematics problem in a short computational time. The second best method is the *selectively damped least squares*, followed by the *damped least squares with singular value decomposition*, and *Jacobian transpose*. The *pseudoinverse* is unstable and shows worse performance.

Joint Position and Rotation Accuracy

Table 8 shows the overall accuracy of individual joints for *damped least squares*. The results of other methods are provided in Appendix A.5. The results show that the high errors in position and rotation arise due to inaccuracies in the elbow and knee. Thus, although the end-effectors (i. e., head, hands, and feet) attain the desired target with a small error in position ($e_{pos} = 22.53\,mm$), the error in position for the remaining joints (i. e., elbow and knee) is, as expected, higher ($e_{pos} = 64.12\,mm$). Similarly, the results show that not only the error in position, but also in rotation is significantly smaller for end-effectors ($e_{rot} = 0.64°$) compared to the remaining joints ($e_{rot} = 33.32°$). In particular, the end-effectors reach the target rotation with an accuracy of less than 2°. Increased accuracy for hands and feet follows from the fact that the inverse kinematics solver only minimizes the position and rotation error for the end-effectors, whereas the position and rotation of the remaining joints are only estimated.

Accuracy of Upper and Lower Body

The position and the rotational error for the upper and lower body are further highlighted in Table 9. A comparison regarding the error in position shows that the lower-

Table 8: Overall error in joint position and rotation for the *damped least squares* method.

Joint	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]
Head	14.75 ± 8.39	0.001 ± 0.01
Hips	3.13 ± 2.20	0.44 ± 0.34
Left hand	21.23 ± 19.60	0.34 ± 0.33
Right hand	17.49 ± 15.93	0.28 ± 0.27
Left elbow	68.56 ± 24.25	43.08 ± 10.25
Right elbow	67.32 ± 29.87	41.45 ± 8.83
Left foot	31.06 ± 18.68	1.38 ± 0.87
Right foot	28.15 ± 16.69	1.22 ± 0.67
Left knee	59.74 ± 33.60	23.56 ± 16.12
Right knee	60.86 ± 71.76	25.20 ± 16.09
Average	37.23 ± 24.10	13.69 ± 5.38

Table 9: Upper and lower body accuracy for the *damped least squares* method.

Joint	**Error in position**	**Error in rotation**
	e_{pos} [mm]	e_{rot} [°]
Upper body	37.87 ± 19.61	17.03 ± 3.93
Lower body	36.59 ± 28.59	10.36 ± 6.82

The upper body includes movements of the head, elbows, and hands. The lower body includes movements of the hips, knees, and feet.

body movements ($e_{pos} = 36.59$ mm) are reconstructed slightly more accurately than the upper-body movements ($e_{pos} = 37.87$ mm). Similarly, regarding the error in rotation, the data indicate more accurate results for the lower body ($e_{rot} = 10.36°$) than the upper body ($e_{rot} = 17.03°$).

The main reason for this is that the upper body has a higher degree of freedom than the lower body. As shown in Table 8, the error in position and rotation is higher for elbows than knees. To further minimize position and rotation error for arms, we would need to adapt the joint constraints for the upper and lower limbs. Alternatively, we could attach additional sensors (HTC Vive trackers) to the forearms to minimize the position and rotation error of the elbows. In this case, we would reduce the number of joints and consequently also the degrees of freedom for arms. However, as already discussed in Section 3.2.1, additional trackers would make such a system intrusive, uncomfortable to wear, and more difficult to set up. Furthermore, additional trackers would also restrict the user's movements.

Accuracy Based on Activities

We further compare the accuracy and performance time of the Jacobian inverse methods concerning the activities. Figure 12 depicts the reconstructed full-body avatars for some specific activities. Furthermore, Table 10 shows the error in position and rotation as well as the required number of iterations and the total time. Based on these results, *squats* show the highest error in position ($e_{pos} = 53.66$ mm). Regarding rotation, *lunges*

(a) *Kicks* (b) *Lunges* (c) *Squats*

Figure 12: Examples of full-body avatar reconstruction for specific activities.

Table 10: Reconstruction comparison of the *damped least squares* method based on different activities.

Movement	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]	**Number of iterations** it_{num}	**Time per iteration** t_{it} [ms]	**Total time** t_{total} [ms]
Standing	37.97	12.01	4.83	0.01	0.04
Walking	34.38	10.73	7.55	0.01	0.07
Punching	36.60	11.22	6.98	0.01	0.07
Kicking	27.04	15.13	7.92	0.01	0.08
Squats	53.66	15.14	8.71	0.01	0.09
Lunges	33.73	17.94	8.80	0.01	0.09
Average	37.23	13.69	7.46	0.01	0.07

perform worse ($e_{rot} = 17.94°$). In contrast, *kicking* shows the smallest error in position ($e_{pos} = 27.04$ mm) and *walking* the smallest error in rotation ($e_{rot} = 10.73$ mm). These results demonstrate that the inverse kinematics solver can generally reconstruct different motions with reasonable accuracy and low latency.

Discussion and Comparison With Related Work

To validate the quality of the full-body motion reconstruction, we compare our results with related work. A direct comparison is hardly possible as we do not have the same target database as the one used in related work. Because we explicitly aim to investigate the suitability of the off-the-shelf VR devices for motion recognition purposes, we use the tracking data of the HTC Vive system. In contrast, related studies usually use publicly available motion capture data [144, 166]. The following comparison, therefore, serves only to identify trends. Table 11 summarizes the accuracy and performance time of related work on inverse kinematics.

We optimized the parameter values of relevant Jacobian inverse methods to reduce the error in position and rotation for end-effectors. Comparing our results with related work, we obtain a smaller position error for all methods, except for the *pseudoinverse*. For example, after the parameter optimization, the *damped least squares* outperforms all other methods and reconstructs full-body motions with an average position error of 37.23 mm. Hence, we obtain smaller errors in position than Aristidou and Lasenby [7], with the median position error of 58.68 mm. Furthermore, we also achieve smaller errors in positions compared to Malleson et al. [144], with a position error of 62 mm (for 13 inertial sensors) or 91 mm (for six inertial sensors). Parger et al. [166] reconstructs only upper-body movements and report an overall root mean square error of 108.5 mm for neck, shoulder, and elbow. Unfortunately, we cannot compare our results with the results provided by Unzueta et al. [220] because the authors specified only the normalized values.

Table 11: Comparison with relevant related work.

Reference	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]	**Number of iterations** it_{num}	**Time per iteration** t_{it} [ms]	**Total time** t_{total} [ms]
Aristidou and Lasenby [7]	58.68	/	65	0.0246	1.6
Malleson et al. [144]	91	12.5	/	/	/
Unzueta et al. [220]	/	16.85	/	/	0.44
Parger et al. [166]	108.5	/	/	/	/
Our optimized method	37.23	13.69	7.46	0.01	0.07

[7] uses a heuristic method, namely *forward and backward reaching inverse kinematics*. [144] uses an own implementation to estimate the pose based on six (or 13) inertial units. [220] uses a hybrid method, namely the *sequential inverse kinematics method*. [166] reconstructs upper-body movements with an analytic inverse kinematics solver. Our optimized method includes results of the *damped least squares* with parameters as defined in Table 6.

Regarding the rotation error, our solution shows similar results. After the parameter optimization, the *damped least squares* method achieves an average rotation error of 13.69°. Compared to Unzueta et al. [220], with a rotation error of 16.85°, our results are slightly better. However, compared to Malleson et al. [144] with a rotation error of 7.8° (using 13 inertial sensors) or 12.5° (using six inertial sensors), our results are slightly inferior.

Additional studies explicitly evaluated the tracking accuracy of the Microsoft Kinect device. For example, Wang et al. [223] report a position error of 100.78 mm, whereas Pfister et al. [167] show that the angular displacement of the Kinect is always greater than 5°. The results of the study by Wang et al. [223] further show higher error rates for the lower body. Other studies aim to optimize incomplete and noisy postures captured by the Kinect and achieve better results (an average error of 39 mm) [196]. However, as already discussed in the requirements analysis in Section 3.1.2, even with optimization algorithms, the Kinect device is not sufficient for full-body motion reconstruction in immersive VR due to its low sampling frequency of 30 Hz [103].

4.3.2 *Subjective Quality of the Reconstructed Full-Body Avatars*

We conducted a user study with ten participants (average age 27.1 $\pm$ 2.42 years, four female) to evaluate the subjective quality of the reconstructed full-body avatars. To

reconstruct full-body movements, we used the *damped least squares* method with the optimized parameters, which we previously identified in [39].

The aim of the study was to evaluate the sense of agency, body ownership, and the perception of the end-to-end latency. To this end, we adapted the sense of embodiment questionnaire from Kilteni et al. [118]. The questionnaire results are summarized in Figure 13 and further detailed in Table 36 in Appendix A.9. The participants could assess the questions on a five-point Likert scale from zero (strongly disagree) to four (strongly agree).

At the beginning of the evaluation, we presented the participants the HTC Vive HMD and bound the trackers to the body. Then, we explained the calibration process. After the avatar was calibrated, the participants could see their avatar while looking down towards their real body and in a virtual mirror.

Results

The results on the sense of body ownership show that 70% of the participants felt that the avatar belonged to them. We believe that some of the participants disagreed with this statement because the arm movements were less precise than the leg movements. We further investigate if there is a significant increase in the sense of ownership for legs compared to arms. Because the data, according to the Anderson-Darling test, do not follow a normal distribution, we employ a one-tiled Wilcoxon signed rank test. The results confirm that the body ownership for legs is significantly higher than for the arms ($p = 0.008$). Hence, we believe that the wide range of the response on the overall sense of body ownership arises due to the fact that the reconstruction of the upper body sometimes fails, whereas the reconstruction of the lower body is very accurate.

The results on the sense of agency further show that 80% of the participants felt they were controlling the avatar through their movements. High responses on the sense of agency are also supported by low responses on full-body disconnection. These results confirm that participants did not perceive the avatar's movements as another person's movement. Furthermore, the results show that the avatar size matched the real one. Moreover, regarding end-to-end latency, the results show that the participants perceived only a small delay between the avatar's movements and their own.

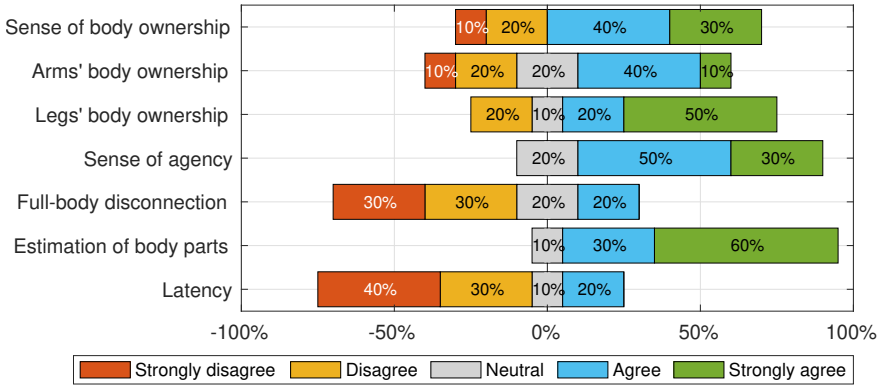


Figure 13: Sense of embodiment questionnaire responses.

Conclusions and Limitations

The current experiment was limited by the small sample size. Nevertheless, the findings of this study suggest that a full-body motion reconstruction using off-the-shelf VR devices is feasible. Between 70% and 80% of the participants had the feeling of owning and controlling the virtual body through their movements. The results further point out that the arm reconstruction needs to be improved. As already discussed in Section 4.3.1, an inaccurate reconstruction of the upper-body movements is not unexpected, as the avatar's arms have a greater degree of freedom than the avatar's legs. To overcome this problem, we could attach additional sensors to the elbow. Above all, such an approach would improve the elbow reconstruction and would contribute to a higher sense of body ownership.

4.4 EFFECT OF END-TO-END LATENCY

Because solving the inverse kinematics problem can cause high computation time and, therefore, a decrease in frame rate, we investigate the effects of end-to-end latency on user experience and performance. End-to-end latency in VR refers to the time delay between a user's action until this action response is visible on the HMD. To this end, we conducted a user study to examine the end-to-end latency threshold at which the user experience and performance decrease. In particular, we aim to identify the end-to-end latency threshold that causes significant cybersickness symptoms in users.

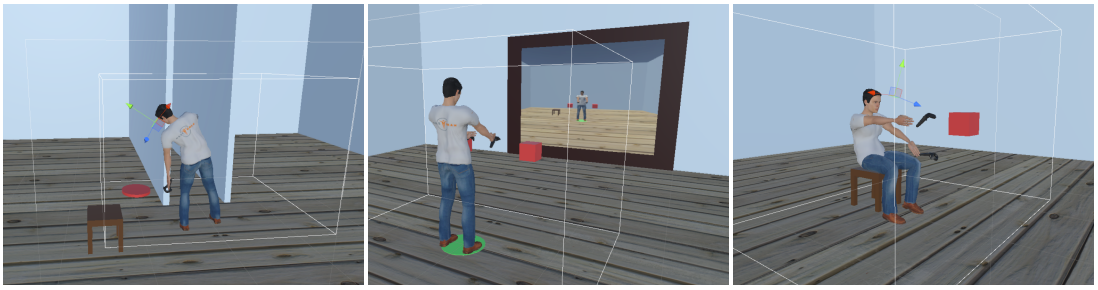
Cybersickness refers to adverse effects that occur while wearing an HMD and can, among others, cause nausea, headache, and dizziness (see Table 21 in Appendix A.1). As already mentioned in Section 2.1.3, although several researchers have investigated the cause for cybersickness, the reason for it is still the subject of research. Similar to the *sensory conflict theory*, indicating that cybersickness occurs when the two primary senses (visual and vestibular) do not match [131], also an increased end-to-end latency causes a mismatch between the perceived and real movements, and therefore elicits cybersickness. However, it is not clear which end-to-end latencies induce significant symptoms or which latencies affect the performance of the users.

4.4.1 *Experimental Design*

To examine the effect of end-to-end latency on user experience and performance, we elaborated three tasks, i. e., the *searching*, *reaching*, and *embodiment task* (see Figure 14). The results of the study were published in [48].

Tasks

We design the *searching task* to quantify cybersickness using the Kennedy's *Simulator Sickness Questionnaire* [113]. Alternative questionnaires, as already described in Section 2.1.3, would also be possible. As proposed by Stauffert et al. [210], we created a virtual environment where users need to search for a virtual object hidden in one of three corridors. As soon as the user finds the hidden virtual object, a new one ran-



(a) *Searching task* (b) *Embodiment task* (c) *Reaching task*

Figure 14: Example of virtual environments with three tasks [48].

domly spawns in another corridor. During the *searching task*, we intentionally increase the tracking delay of the HMD and the movements of the virtual avatar up to 100 ms.

Secondly, we design the *embodiment task* to evaluate the user perception on mismatched virtual movements. As proposed by Samaraweera et al. [189], we delay only one side of the virtual body (either the left arm and leg or right arm and leg). We again intentionally increase the tracking delay of the HTC Vive devices and controllers up to 200 ms. The aim of this task is to identify the impaired side while observing the movements in a virtual mirror.

Finally, we design the *reaching task* to measure user performance, i. e., how much time a user needs to complete a task while being exposed to different latencies. As proposed by Ware and Balakrishnan [224], we created a virtual environment where users need to reach virtual targets with their left or right hand as fast as possible while sitting on a stool. As soon as the user reaches the virtual target, a new one spawns randomly in their field of view. Similar to the *searching task*, we increase the tracking delay of the HTC Vive devices up to 100 ms.

Methods

The experiment was conducted using a within-subject design. Hence, all participants tested all tasks and all conditions. In total, 21 participants ($M = 28.6 \pm 8.79$, six female) took part in the study. Because the measured base end-to-end latency of our system was approximately 50 ms (measured using the technique proposed by Friston and Steed [88]), we could investigate the effects only above this value. Depending on the task, we artificially increased the delay up to 100 ms or even 200 ms. The task order and the latencies conditions were randomized in order to ensure that the participants do not adapt to the latency during the evaluation.

According to the Anderson-Darling test, the data do not follow a normal distribution. Therefore, we use Friedman's test to evaluate statistical significance. Furthermore, we apply pairwise comparisons using the Conover post-hoc test with a Bonferroni correlation to investigate if there are any significant differences among the end-to-end latency conditions. Moreover, for all latency conditions, we calculate the effect size using a Hedge's g [186]. The values are interpreted using the Cohen's d rule of thumb,

where a value of 0.2 indicates a small, 0.5 a medium, and 0.8 a large effect size [55]. The statistical results on the effect of end-to-end latency are detailed in Appendix A.6.

4.4.2 *Results on User Experience and Performance*

The results show that increased end-to-end latency elicits significant cybersickness symptoms. Additionally, increased end-to-end latency also causes users to need more time to complete tasks. In the following, we will discuss the main findings of the user study and their implications.

Effect of End-to-End Latency on User Experience

The aim of the *searching task* is to identify at which end-to-end latency threshold users experience significant cybersickness symptoms. Therefore, we calculate the total score from the Kennedy's *Simulator Sickness Questionnaire* [113] at each latency condition. A detailed description of how the score is calculated is provided in Appendix A.1.

The results in Figure 15 show that higher end-to-end latency tends to correspond to higher total scores on the *Simulator Sickness Questionnaire*. The Friedman's test reveals a statistically significant difference between the end-to-end latency conditions ($\chi^2(11) = 100.59$, $p < 0.001$, Kendall's $W = 0.6$), with a high internal reliability (Cronbach's $\alpha = 0.92$). A pairwise comparison shows a significant difference between base end-to-end latency (50 ms) and end-to-end latency above 75 ms ($p < 0.001$, Hedge's $g = 0.65$). Statistical results for all latency conditions on user experience are detailed in Table 30 in Appendix A.6.

These results show that users experience significant cybersickness symptoms at the end-to-end latency above 75 ms. As can be seen in Figure 15, at this point, the final score is already above 20. These results are complemented with the results from Stanney et al. [205], pointing out that scores above 20 indicate that the virtual environment is considered a "bad simulator." Moreover, according to Stanney et al. [205], final

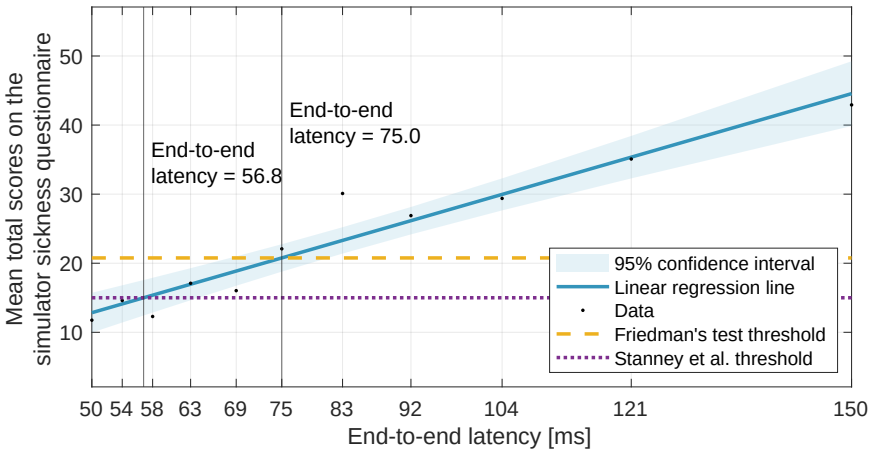


Figure 15: Results of the *searching task*. The data is fitted with a least-squares regression line ($y = 0.31713x, r^2 = 0.93, p < 0.001$).

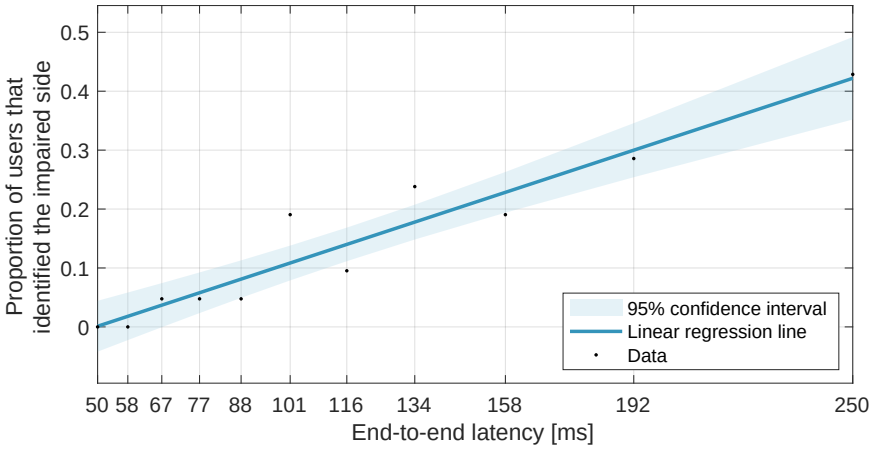


Figure 16: Results of the *embodiment task*. The data is fitted with a least-squares regression line ($y = 0.044187x, r^2 = 0.92, p < 0.001$).

scores higher than 15 on *Simulator Sickness Questionnaire* already cause sufficient discomfort. Hence, the end-to-end latency of 56.82 ms is below the threshold for eliciting cybersickness, although even lower latency is required to abolish cybersickness.

Furthermore, the aim of the *embodiment task* is to investigate the effect of end-to-end latency on simultaneity perception. The results in Figure 16 show that only a minority of the participants (42%) noticed a 250 ms delay. Nevertheless, with the increased end-to-end latency, more and more participants could specify the impaired side.

Effect of End-to-End Latency on User Performance

The aim of the *reaching task* is to identify a latency threshold at which the user performance decreases. Figure 17 shows the relationship between end-to-end latency and time. Overall, the results indicate that higher latency values tend to reduce task performance and, in particular, require more time to complete the task.

Results of the Friedman's test show a statistically significant difference in user performance depending on the end-to-end latency ($\chi^2(10) = 68.71$, $p < 0.001$, Kendall's $W = 0.66$), with a good internal reliability (Cronbach's $\alpha = 0.87$). Pairwise comparisons show a significant difference between the base end-to-end latency (50 ms) and end-to-end latency above 69 ms ($p = 0.01$, Hedge's $g = 0.55$). Thus, participants need significantly more time to complete the task at the 69 ms latency conditions compared to the base latency of 50 ms. Statistical results for all latency conditions on user performance are detailed in Table 31 in Appendix A.6.

Conclusions and Limitations

Currently, it is generally accepted that the end-to-end latency should be kept below 20 ms [69, 89], whereas others suggest that a higher total system latency between 50 and 70 ms could be tolerated [2]. However, the base end-to-end latency of many HMDs, such as Oculus Rift DK1 and DK2 as well as smartphones, do not meet these requirements [172]. Similarly, the end-to-end latency of the HTC Vive HMD with

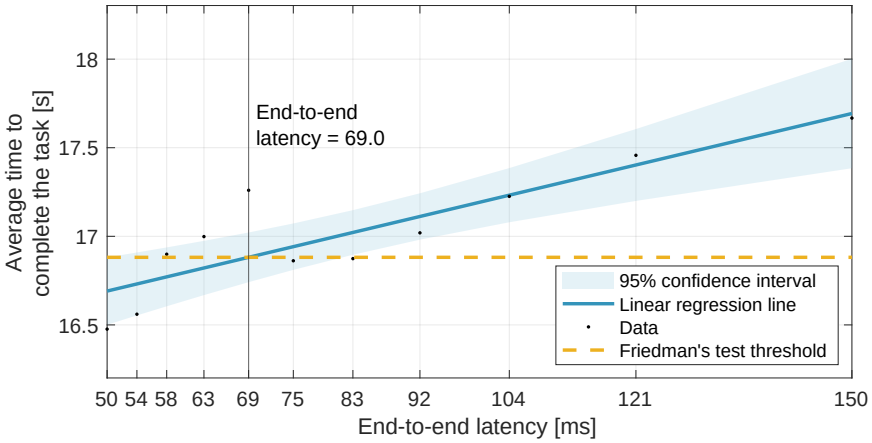


Figure 17: Results of the *reaching task*. The data is fitted with a least-squares regression line ($y = 0.010022x, r^2 = 0.76, p < 0.001$).

20 ms [160] is higher as expected. Studies reporting the base latency of their system usually report higher values, between 31 and 36 ms [2, 210], whereas other studies using full-body motion capturing report even higher values between 45 and 80 ms [122, 189, 222]. According to a study by Garcia-Agundez et al. [96], extrapolation and filtering techniques predicting head movements could reduce the impact of end-to-end latency until future VR systems can track the head movements at a higher update rate and render the head movements with a higher refresh rate.

The user study with 21 participants shows statistically significant results and medium effect sizes. The findings indicate that end-to-end latency above 56.82 ms evokes significant cybersickness symptoms with final scores higher than 15 on the *Simulator Sickness Questionnaire*. In experiments conducted by Dizio and Lackner [69], the researcher found that delays above 39 ms evoke mild malaise, whereas severe malaise is first identified at the 355 ms latency threshold. Thus, to abolish cybersickness, much lower end-to-end latency is required.

Furthermore, similar to [189] and [210], in our experiments, less than half of the participants were able to identify the impaired side with an end-to-end latency of 250 ms. Waltemate et al. [222] further conclude that the sense of agency and body ownership decrease at latency higher than 125 ms, whereas simultaneity perception is affected by lower latency at above 75 ms. However, in their study, the experiments were conducted inside a CAVE environment.

Moreover, our results show that the user performance significantly decreases at end-to-end latency above 69 ms. In comparison with related work, we determine a lower end-to-end latency that affects user performance. According to Waltemate et al. [222], motor performance decrease at end-to-end latency above 75 ms. Furthermore, Becher et al. [14] conclude that a latency below 230 ms will not affect the user performance and is more likely not even noticed. However, it is important to note that the researchers investigated user performance in a collaborative virtual environment. In contrast, in our study, user performance was only measured in a singleplayer mode.

5

FULL-BODY MOTION RECOGNITION

Accurate full-body motion recognition is essential to identify motion execution errors and provide appropriate feedback so that players can improve their movements. As already analyzed in Section 2.2.3, several studies agree that, especially in exergames, accurate detection of physical activities is crucial to ensure that users perform desired physical activities as intended [126, 127, 147]. Additionally, accurate motion tracking is required to ensure that players perform the exercises correctly to minimize the risk of injuries and to improve health outcomes [45]. For example, exergames can utilize full-body motion recognition to support patients, monitor their improvements, and motivate them to repeat specific exercises regularly.

Therefore, in this thesis, we employ machine learning algorithms to analyze the entire movement execution and identify potential activity execution errors in the context of Virtual Reality (VR). We intend to recognize whether the desired activity was performed correctly rather than distinguishing among different movements. For this purpose, we first identify a suitable set of sensor positions. As shown in Figure 18, we vary the number of sensors and sensor positions to train and test machine learning models in offline training. We then analyze the recognition performance of these models to identify the sensor subset that performs best. Afterward, these models are used for online recognition to recognize activity execution errors and provide appropriate feedback to the players.

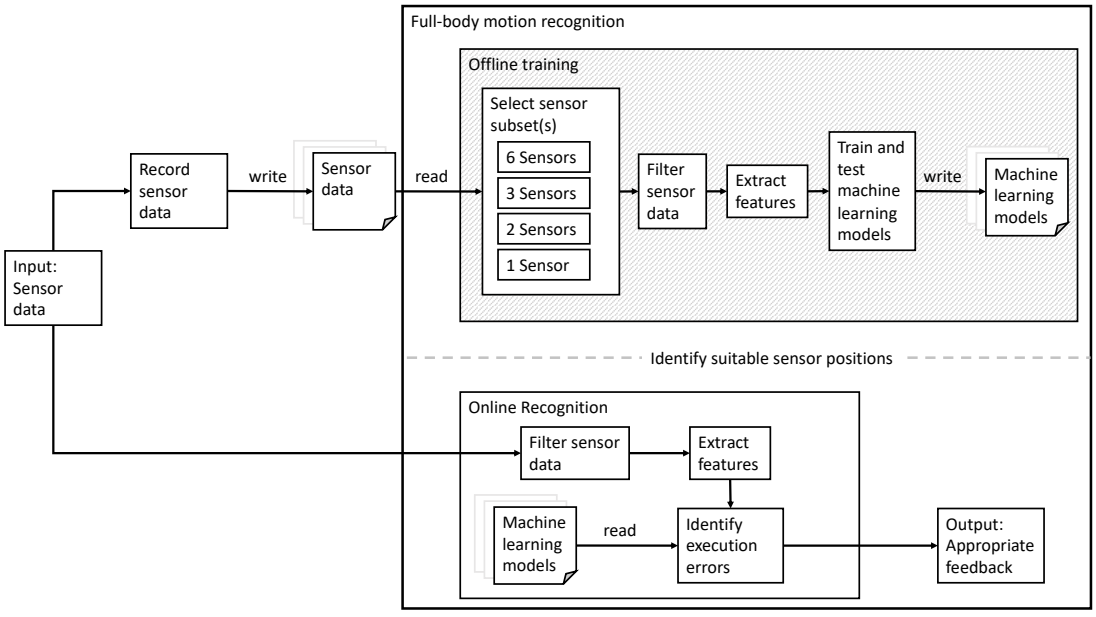


Figure 18: Full-body motion recognition model.

Similar to the full-body motion reconstruction (see Chapter 4), we utilize only off-the-shelf VR devices to track the movements of particular body parts. However, the retrieved tracking data still need to be correctly interpreted to recognize different full-body movements. We first define sensor subsets, present the feature extraction process, and describe training and testing methods based on different sensor subsets in Section 5.1. We then investigate the effect of different feature types and analyze the machine learning models' performance of relevant sensor subsets to identify suitable sensor positions in Section 5.2. Finally, we discuss relevant alternatives on how recognition models can detect full-body movements in real-time to provide appropriate feedback in Section 5.3.

In this thesis, we focus on investigating the suitability of the off-the-shelf VR devices for motion recognition purposes using existing machine learning algorithms. However, we do not perform research on individual classifiers, as plenty of research is already available in this field. Previous research already provides a detailed review on human activity recognition using wearable sensors, including accelerometer, gyroscope, and magnetometer sensors [56, 129, 155]. In contrast to the previous studies, we further use sensors that provide global position and rotation data.

5.1 OFFLINE TRAINING

The objective of offline training is to train classifiers for relevant sensor subsets and identify suitable sensor positions on the body. As detailed in Figure 18, we first create a database and collect sensor data of ten participants (27.3 ± 2.26 years, between 24 and 31 years, five females) performing three yoga poses, *warrior I*, *warrior II*, and *extended side angle* (see Figure 19). We deliberately select activities for which the movements of the upper and lower body are substantial. A detailed movement description is provided in Table 25 in Appendix A.4. Afterward, we use this database to select sensor subsets, filter invalid sensor measurements, calculate features, and finally, train and test models.

Because all participants were beginners in yoga, they first watched a correct execution of the yoga pose in a video and were allowed to practice it before being recorded. Once the participants have learned the correct execution of the movements, we captured the sensor data of the VR devices at a constant frame rate (90 frames per second). Each participant performed each yoga pose at least ten times. We gathered only the samples where the yoga pose was performed as instructed and deleted the samples where a participant made a mistake. In total, the database consists of 100 samples for each yoga pose, i. e., ten samples per each of ten participants. Previously it has been shown that between 75 and 100 samples are sufficient to train classification models with good performance [15]. However, the researchers refer to independent samples and also point out that generally, even larger test sample sizes are required for higher sensitivity (true positive rate).

In order to be able to recognize movements that are carried out at different speeds during online recognition, we record them individually to obtain a start and an end sequence for each execution. The start sequence is defined by a base pose in which

(a) *Warrior I* (b) *Warrior II* (c) *Extended side angle*

Figure 19: Yoga poses used for the training and test data set.

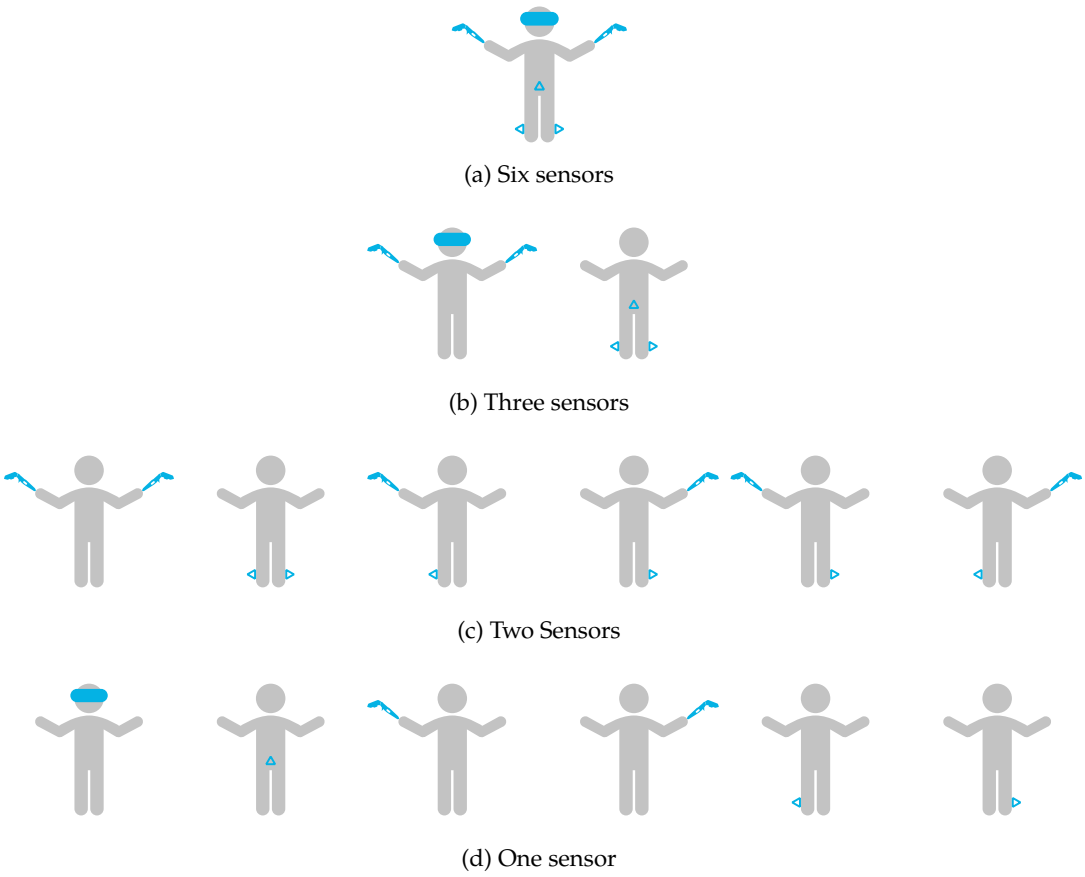
users stand upright with their arms lowered and looking forward. The end sequence is defined as the final pose, as shown in Figure 19. The start and end sequence enable us then to analyze the entire execution of the yoga movement regardless of the execution speed. In contrast to continuous movements, such as walking or jogging in place, which are performed over a certain time window, such an approach, with a start and end sequence, increases the probability of recognizing a gesture. Without a start and end sequence, we would need to define an appropriate (sliding) window size in which the movement should be recognized.

After the database has been created, we select relevant sensor subsets of the captured sensor data in Section 5.1.1. Then we identify and remove faulty sensor measurements and extract features for the selected sensors in Section 5.1.2. Finally, we train and test models for each sensor subset and compare their respective performances to identify suitable sensor positions in Section 5.1.3.

5.1.1 *Sensor Subsets*

To identify a suitable sensor set, we attach several sensors to the body. Similarly to the full-body motion reconstruction, we track the upper-body movements with the HTC Vive HMD and two controllers, which are held in the hands. Additionally, we attach three HTC Vive trackers to track the lower-body movements, i. e., both feet and the hips. Figure 20 shows several possible sensor configurations, with either all six sensors or a sensor subset. We do not aim to determine the best possible classification result for each sensor subset. Instead, we intend to assess the general suitability of the currently available off-the-shelf VR devices for motion recognition purposes.

We purposefully selected these six positions for the sensors, as they are nevertheless necessary for full-body motion reconstruction. Additional sensors (e. g., on the knee or upper arm) would also be possible. However, because we want to reduce the number of sensors required for full-body motion reconstruction (see Section 3.2.1), we also want to recognize the full-body movements with only these six sensors instead of



(a) Six sensors

(b) Three sensors

(c) Two Sensors

(d) One sensor

Figure 20: Possible sets of sensor positions.

using additional sensors. Thus, no additional sensors should be added to those used for full-body motion reconstruction.

We split the six sensors into relevant subsets to investigate which sensor subset is most suitable for recognizing full-body movements. When using three sensors, we track either the upper- or lower-body movements (see Figure 20b). Upper-body tracking includes an HTC Vive HMD and two controllers, whereas lower-body tracking includes three HTC Vive trackers attached to the hips and both feet. Apart from three sensors, we use two sensors to investigate which combination of arms or feet is required for accurate full-body motion recognition (see Figure 20c). Moreover, we examine the recognition performance of full-body movements based on models trained on individual sensors (see Figure 20d). Recognition with a single sensor is especially crucial to provide feedback to the users (see Section 5.3.1).

Alternative combinations of two sensors are also possible, e. g., hands and head or hands and hips. However, such a combination does not take the leg movements into account and has no added value in recognizing full-body movements. Additionally, a sensor combination with feet and head or feet and hips could be possible. However, such a combination does not take the hand movements into account and again has no added value in recognizing full-body movements.

5.1.2 *Feature Extraction*

The recognition success rate depends on multiple aspects, including appropriate features. We calculate features using the database that contains measurements of tracked sensors, i. e., a position vector and a quaternion representing spatial rotations at each frame. HTC Vive system determines the position and rotation of sensors by the base stations emitting infrared light, received by infrared photodiodes on the sensors. The system uses, in addition to optical sensors, also built-in inertial measurement units to increase reliability. For this purpose, it applies widely-used sensor fusion algorithms such as the Kalman filter [124]. Thus, the HTC Vive system provides for all tracked devices linear and angular velocity as well.[1] However, because the HTC Vive does not explicitly provide acceleration measurements, we calculate them from the provided velocity as $a = \Delta v / \Delta t$.

Although the HTC Vive generally provides accurate sensor measurements, they might suffer from jitter or drifting problems [25, 160]. Additionally, the system can lose track when the line of sight between a sensor and the base station is blocked, e. g., when users walk out of the tracking area or cover the sensors [160]. In this case, when the sensor position drifts away from its actual position, we filter the data and remove sensor measurements with large offsets in position. To identify drifting sensors, we define individual thresholds and compare sensor position differences between frames. Alternatively, we could apply filters to detect outliers or smooth the data.

Furthermore, we transform the sensor measurements into the character reference frame to ensure that the recognition works independently of the player's orientation. Without sensor measurement transformation, the users would need to always look in a specific direction while performing activities. Sensor measurement transformation is already done during the character calibration (see Section 4.2.1). It ensures that full-body motion reconstruction works independently of the player's orientation or the attached sensors' position and rotation. With the character calibration, we also ensure that full-body motion recognition works independently of the player's position and orientation. Therefore, we extract features of the transformed sensor measurements. However, if the user does not properly calibrate the character, the full-body reconstruction and recognition will be faulty.

The data in Table 12 provide an overview of utilized features for the classification problem. We calculate these features for each sample between the start and end sequence. In addition to these features, we also use time-series data. As the name implies, time-series features give values associated with time. Thus, each feature value represents a different data point in time. We will further discuss time-series data in Section 5.2.3.

Features Based on a Single Sensor

Previous studies have shown promising results using features extracted from accelerometer data to recognize various activities [135, 151]. As already mentioned

[1] `https://github.com/ValveSoftware/openvr/blob/4c85abcb7f7f1f02adaf3812018c99fc593bc341/headers/openvr.h#L259-L260`, last accessed on March 3, 2021

Table 12: Initial set of features.

Sensors	Feature type	Axis	Features			
			Mean, SD, MAD	IQR, Q_{25}, Q_{75}	Min, Max, Range	MCR
Single	Position	x, y, z	×	×	×	×
	Rotation	x, y, z, w	×	×	×	×
	Velocity and acceleration	Magnitude(x, y, z)	×	×	×	×
Dual sensor	Distance in position	x, y, z	×	×	×	×
	Difference in velocity	Magnitude(x, y, z)	×	×	×	×

Abbreviations: *Standard Deviation* (SD), *Mean Absolute Deviation* (MAD), *Interquartile Range* (IQR), *Lower and Upper Quartiles* (Q_{25} and Q_{75}), and *Mean Crossing Rate* (MCR).

above, the HTC Vive system provides linear and angular velocity for all tracked devices, whereas the acceleration is derived from the velocity. Due to the tracking system of the HTC Vive with external sensors (base stations, also called Lighthouses), we also obtain position and rotation data in a tracking area of up to $10\,\mathrm{m} \times 10\,\mathrm{m}$ [58]. Thus, in addition to recognition systems based on only inertial sensors (typically consisting of a gyroscope, an accelerometer, and a magnetometer), which usually suffer from drifting problems [121] and cannot provide reliable global positions, we further use features based on position and rotation data.

One possibility to design a set of features is to use the position data relative to the user: x (left and right movements), y (backward and forward movements), and z-axis (up and down movements). However, because the positional features are dependent on the user's height, we scale them by an individual scale factor that is directly related to the user's range of motion. This scale factor is already determined during the character calibration (see Section 4.2.1). Furthermore, we reduce the dependency on the user's orientation when using features based on velocity and acceleration.

Overall, to build a set of features, we calculate: mean, standard deviation, mean absolute deviation, lower- and upper-quartiles (Q_{25} and Q_{75}), interquartile range ($Q_{75} - Q_{25}$), minimum and maximum value, as well as mean crossing rate. We calculate these values for the individual axis or a combination of all three axes between the start and end sequence. Additionally to the mean, we calculate the standard deviation and mean absolute deviation to gauge variability. The range ($\max_{x,y,z} - \min_{x,y,z}$) is used in order to detect in which direction the user is moving. Furthermore, we calculate the mean crossing rate, which is, as the name implies, a rate at which a signal

crosses the mean, normalized by the number of data points. Moreover, we calculate the magnitude of all three axes to remove the orientation dependence of the velocity and acceleration-based features.

Features Based on Two Sensors

Apart from the features based on only a single sensor, we also extract features from the supplied data of two sensors, i. e., the distance in position and difference in velocity. The distance feature provides details on body movement during an exercise. For example, depending on the exercise, the movement of two limbs remains constant or they move apart. During the execution of *warrior I* (see Figure 19a), both arms are raised over the head while the distance between the arms remains constant. However, during the execution of *warrior II* (see Figure 19b), both arms are raised to the side at shoulder height so that they are parallel to the floor. In this case, the distance between the arms increases over time.

Furthermore, we calculate the difference in velocity to analyze the similarity between the movement of two sensors. The difference in velocity directly indicates whether two limbs were moved simultaneously or independently.

5.1.3 *Training and Testing of Machine Learning Models*

We employ several machine learning algorithms provided by the Weka framework[2], which showed promising results in related work to recognize gestures and activities in many fields, independent of VR. We evaluate and compare the recognition results of seven base classifiers: *J48*, *JRip*, *REP-Tree* (*Reduced Error Pruning Tree*), *Random Forest*, *Naïve Bayes*, *Bayesian Net*, and *SMO* (*Sequential Minimal Optimization* algorithm, which is utilized for the training of *Support Vector Machines* [170]). The chosen hyperparameters are detailed in Table 32 in Appendix A.8.

Additionally, we use two baseline methods, namely, *OneR* and *ZeroR*, to calculate the baseline performance. *OneR* predicts an activity using a single feature, whereas *ZeroR* always predicts the majority class. Furthermore, we apply two meta-learning algorithms, *AdaBoostM1* and *Bagging*. The meta-learning algorithms' objective is to improve the classification rule of the base classifier. Both meta-learning algorithms are used with all seven base classifiers. A detailed description of the machine learning algorithms used in this thesis is provided in [228].

For the time-series classification, we implement *hidden Markov models* based on the tutorial by Rabiner [173]. *Hidden Markov models* are especially suitable to recognize sequential activities based on time-series [119, 235]. Therefore, on the one hand, we use the classifiers to identify suitable sensor positions for full-body motion recognition. On the other hand, we employ *hidden Markov models* to analyze the entire activity execution based on time-series and recognize potential activity execution errors by evaluating the models.

[2] `https://www.cs.waikato.ac.nz/ml/weka/index.html`, last accessed on March 3, 2021

Model Type

The training and testing data sets contain recorded sensor measurements of three yoga poses: *warrior I*, *warrior II*, and *extended side angle* (see Figure 19). We use these data sets to train and test models for relevant sensor subsets. Depending on the application scenario, we can use either personal, generic (impersonal), or hybrid models [137].

As the name implies, personal models use only one user's data to train and test models and are thus tailored to one user. In contrast, generic models use training data of multiple users and evaluate it by using the testing data of different users. Hence, generic models are expected to recognize the activities of any user, even when no data belonging to them was used for training. Previous work has shown that personal models can outperform generic models because they can more effectively learn user-specific differences [138]. The researchers further point out that the performance of generic models can further be improved by using hybrid models, thus by combining personal and generic models.

For well-defined motions, such as yoga poses or Tai Chi, which are usually always performed in the same way, it may be sufficient to train models using only samples of an expert. For other sports exercises, which are usually performed differently, it is necessary to collect training samples from multiple users. For example, while performing squats, users might keep their arms outstretched or flexed. To recognize both exercise variants, we need to collect data of users performing the movements slightly differently and train generic models.

In this thesis, we focus on generic models, as our goal is to build an activity recognition model that can be used in a game scenario, e. g., to motivate players to practice yoga (see Chapter 6). As we do not want to re-train a model for each player, we use data from different users for training and testing. Such an approach enables us a better prediction of the classifier's performance in a game scenario.

Validation Protocol

A standard way to predict the classifier's error rate is to apply k-fold cross-validation [228], whereby the data are split into k parts of similar size. Applying cross-validation also counteracts overfitting, which occurs when a classifier fits the training data too tightly and performs poorly to previously unseen data. Although the training and testing data sets in k-fold cross-validation are different, both data sets can contain the data of the same user. However, as we intend to build generic models, we therefore apply the leave-one-subject-out cross-validation to measure the error rate of the learned model. Leave-one-subject-out cross-validation also offers a chance to get the most out of a small data set and get the most accurate estimate possible [228].

To do so, we train for each of the given n users a separate model using the data set of the $n-1$ other users. Afterward, we test the trained model using the remaining data, i. e., data related to one user, previously not used for training. Because each user performed each pose ten times, we obtained in total 100 samples for each of the three poses. Consequently, each validation step consists of 9 users $\cdot$ 3 poses $\cdot$ 10 samples $=$ 270 samples for training and 1 user $\cdot$ 3 poses $\cdot$ 10 samples $=$ 30 samples for testing.

Evaluation Metrics

We evaluate the success rate of a recognition model by reporting the F_1 score (see Equation 14 in Appendix A.8), which is defined as the harmonic mean of precision and recall [100]. Precision is referred to as the positive predictive value (the ratio between true positives and the sum of true positives and false positives), whereas recall is referred to as the true positive rate (the ratio between true positives and the sum of true positives and false negatives). The optimal F_1 score is one (perfect precision and recall), with the worst being zero.

In contrast to overall accuracy, which measures a model's performance based on the correctly predicted samples among all classified instances, the F_1 score gives a better measure of the incorrectly classified samples. As we have a multi-class classification problem, we calculate the macro-averaged F_1 scores because it weights all classes equally [169]. We select the balanced F_1 score because we do not have a particular emphasis on precision or recall. Conversely, we could use different metrics that focus either on precision or recall.

5.2 PERFORMANCE OF FULL-BODY MOTION RECOGNITION

The performance of machine learning algorithms depends, among others, on the selected features and sensor subset. However, a machine learning approach that achieves outstanding F_1 scores may also lead to overfitting problems. To counteract overfitting, we identify relevant features in Section 5.2.1. We investigate the effect of different feature types to select features that are not redundant and contribute to better overall performance. Next, we train machine learning classifiers and analyze the recognition performance of relevant sensor subsets in Section 5.2.2. Thereby, we intend to identify a suitable position for sensors to recognize full-body movements accurately. Finally, we present an alternative approach to recognize full-body movements using *hidden Markov models* in Section 5.2.3.

5.2.1 *Comparison of Feature Types*

We analyze different feature types to identify the features that are most likely to predict the class best. As mentioned in Section 5.1.2, our database contains five feature types: features based on position, rotation, velocity, acceleration, and dual-sensor features. To identify the best performing feature type, we start with an empty set of features and apply a top-down approach. Hence, in each step, we add a single feature type and observe the information gain of this feature by calculating the F_1 score. This approach is also known as *forward estimation* and is particularly suitable to detect when redundant features are added [228]. Alternatively, the bottom-up approach (*backward elimination*) would be possible where we then start with a full set of features and delete one in each step. Both approaches are suitable to identify inappropriate features that might deteriorate the classifiers' performance.

Table 13: F_1 scores for *Naïve Bayes* depending on the feature subset.

Feature type	F_1 **score**	**Feature type**	F_1 **score**	**Feature type**	F_1 **score**
PosRotDual	0.86	*RotAccVelDual*	0.84	*Pos*	0.73
PosRotVelDual	0.86	*PosRotAcc*	0.83	*PosVel*	0.73
PosRotAccDual	0.86	*PosRot*	0.82	*RotAccVel*	0.73
PosDual	0.85	*PosRotVel*	0.82	*PosAcc*	0.72
PosVelDual	0.85	*PosRotAccVel*	0.82	*AccVelDual*	0.62
PosAccDual	0.85	*RotAcc*	0.81	*VelDual*	0.59
PosRotAccVelDual	0.85	*RotVel*	0.80	*AccVel*	0.53
RotVelDual	0.84	*PosAccVel*	0.80	*Vel*	0.52
PosAccVelDual	0.84	*Rot*	0.79	*Acc*	0.50

Abbreviations: *Acceleration* (Acc), *Velocity* (Vel), *Position* (Pos), *Rotation* (Rot), and *Dual-Sensor Features* (Dual).

Feature selection can be computationally intensive. As shown in Figure 38 in Appendix A.7, we need to train and test 31 models. Thus, the number of possible feature subsets increases exponentially with the number of features. With m features, we need to train and test $2^m - 1$ models. If we want to evaluate the performance of individual features, the evaluation time would significantly increase. In this case, we could accelerate the search process by stopping the evaluation as soon as the estimated performance of the model does not improve.

We employ leave-one-subject-out cross-validation and calculate the macro-averaged F_1 scores for all sensor subsets using *Naïve Bayes*. *Naïve Bayes* generally performs well with random features; however, it can be misled when dependencies among features exist and primarily when redundant features are used [228]. The results of the power set (excluding the empty subset) of feature types in Table 13 show that the recognition performance of rotation-based features is already high ($F_1 = 0.79$). The features based on position and rotation achieve an F_1 score of 0.82, whereas the dual-sensor features further increase the F_1 score to 0.86. By adding the velocity or acceleration data to the position and rotation, the overall F_1 score remains the same. Moreover, using all five feature types slightly reduces the F_1 score from 0.86 to 0.85. As velocity and acceleration data are redundant and do not contribute to better performance, we use only position and rotation-based features with dual-sensor features.

5.2.2 *Analysis of Sensor Subsets*

We compare the performance of trained classifiers for different sensor subsets with the baseline methods *OneR* and *ZeroR*. Because we have the same number of samples for all three poses, the baseline F_1 score is always 0.17. The remaining classifiers should always perform better than *ZeroR* or *OneR*.

The comparison of feature types in Section 5.2.1 has shown that features based on position, rotation, and dual-sensor features contribute to better performance. We use these features to train and test the classifiers. Figure 21 provides an overview of F_1 scores of base classifiers for the different number of sensors. Among the base classifiers, *Random Forest* and *SMO* achieve good results with $F_1 > 0.88$ for almost all sensor subsets, except for the single sensor condition. Similarly, *Bayesian Net* and *REP-Tree* achieve F_1 scores above 0.8 when more than one sensor is available.

As expected, the performance of the classifier algorithms decreases with fewer sensors. Note that the classifiers' performance for the condition with three sensors is sometimes lower than with two sensors because we did not calculate F_1 scores for all possible sensor combinations. The subset with three sensors includes one condition with sensors attached only to the upper body and one condition with sensors attached only to the lower body (see Section 5.1.1). As we will discuss later in this thesis, the lower-body recognition is significantly inferior to that of the upper body. Therefore, when we calculate a mean F_1 score for three sensors, the value decreases due to the low recognition performance of the lower-body movements. In contrast, the subset with two sensors includes several conditions, combining sensors attached to the upper and lower body. Because five out of six sensor combinations include at least one hand sensor, the recognition of two sensors is generally higher than that of three sensors.

Table 14 further provides macro-averaged F_1 scores for relevant sensor subsets, as defined in Section 5.1.1. The results of meta-learners are provided in Appendix A.8. The results indicate an increase in all classifiers' performance when sensors are worn on the upper and lower body. This improvement is due to the nature of the selected yoga poses. Upper-body movements differ among different poses, whereas the lower-body movements are nearly the same. Hence, for reliable recognition, we need to combine sensors on the upper and lower body.

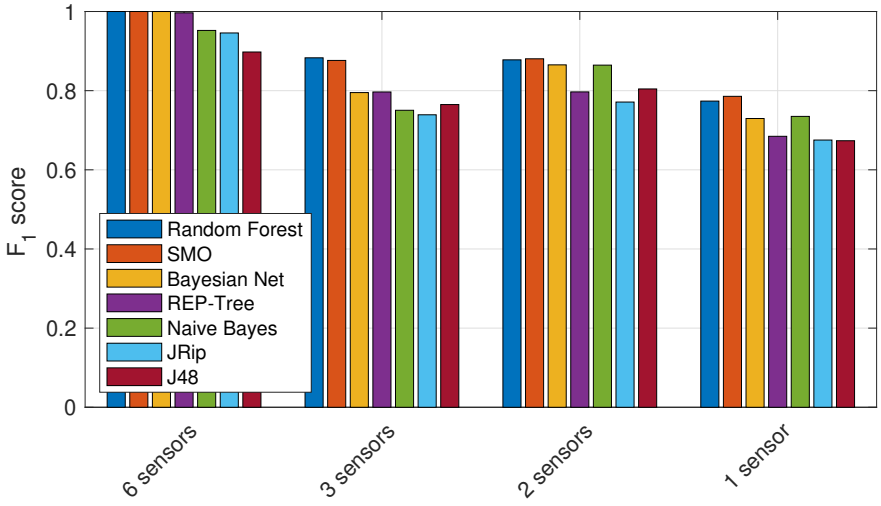


Figure 21: Mean F_1 scores of the base classifiers with respect to the number of sensors.

Table 14: F_1 scores for relevant sensor subsets and machine learning algorithms.

Classifier	6 sensors	3 sensors		2 sensors						1 sensor					
	Full-body	Upper-body	Lower-body	lHand-rHand	lFoot-rFoot	lHand-lFoot	rHand-lFoot	rHand-rFoot	lHand-rFoot	Head	rHand	lHand	Hips	rFoot	lFoot
Random Forest	1.00	1.00	0.77	0.99	0.51	0.90	0.98	0.98	0.90	0.99	0.98	0.91	0.79	0.44	0.53
SMO	1.00	1.00	0.75	0.99	0.57	0.96	0.92	0.93	0.91	1.00	0.94	0.95	0.88	0.46	0.49
REP-Tree	1.00	1.00	0.60	0.90	0.44	0.78	0.88	0.92	0.84	1.00	0.92	0.84	0.60	0.33	0.40
Bayes Net	1.00	1.00	0.59	1.00	0.50	0.94	0.91	0.89	0.94	0.91	0.95	0.89	0.64	0.39	0.58
Naïve Bayes	0.95	0.95	0.55	0.99	0.55	0.93	0.91	0.91	0.90	0.90	0.94	0.93	0.55	0.53	0.57
JRip	0.95	0.93	0.55	0.94	0.49	0.83	0.86	0.71	0.80	0.98	0.78	0.82	0.58	0.51	0.38
J48	0.90	0.94	0.59	0.98	0.48	0.79	0.89	0.90	0.79	0.95	0.90	0.79	0.62	0.34	0.45
OneR	0.92	0.92	0.50	0.92	0.35	0.85	0.67	0.67	0.85	0.83	0.67	0.85	0.50	0.27	0.41
ZeroR	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17

Abbreviations: *Left Hand* (lHand), *Right Hand* (rHand), *Left Foot* (lFoot), and *Right Foot* (rFoot). The *full-body* configuration includes all six sensors. The *upper-body* configuration includes the HMD and two controllers held in the hands, whereas the *lower-body* configuration includes sensors attached to the hips and both feet.

One sensor

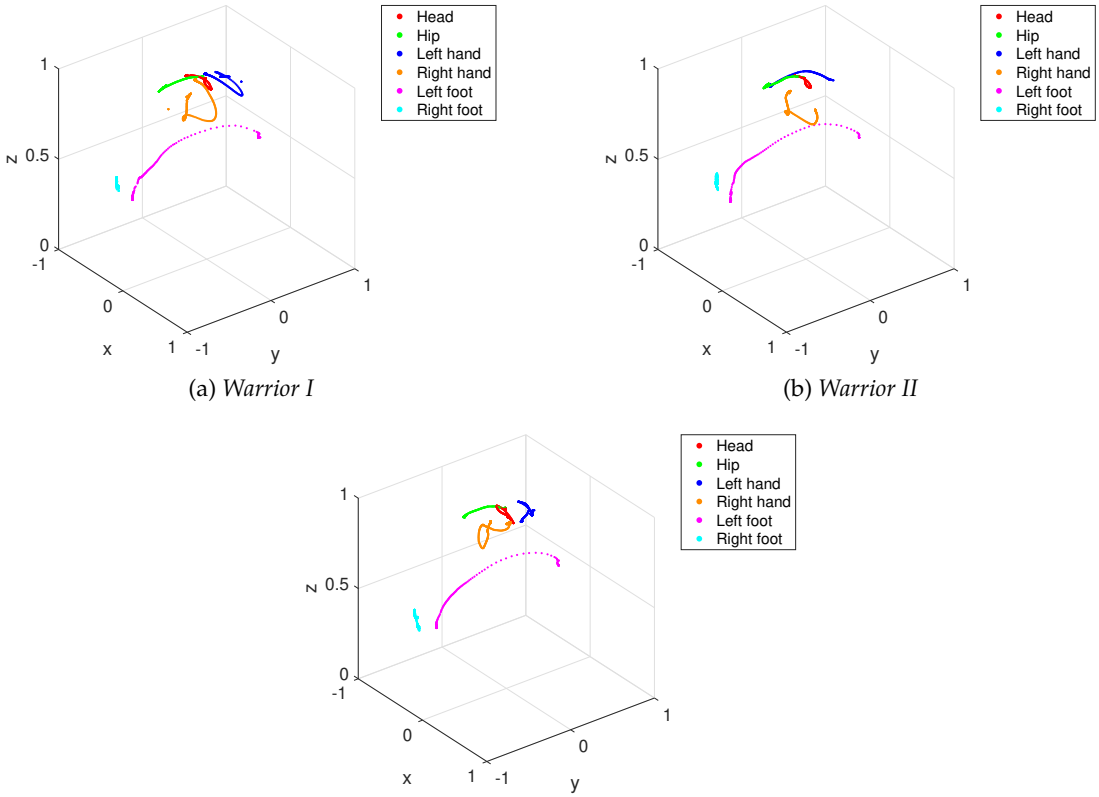
The one-sensor approach combines several models that are trained on individual sensors to recognize full-body movements. Thus, we analyze movements of individual body parts separately to finally recognize if full-body movements are performed as intended. The results in Table 14 show that *Random Forest*, *SMO*, *REP-Tree*, *JRip*, and *J48* recognize the head movements with $F_1 > 0.95$. As shown in the confusion matrix in Table 35a in Appendix A.8, *Random Forest* almost perfectly classifies the yoga poses based on the sensor measurements of the HMD ($F_1 = 0.99$). The analysis of the tree- and rule-based classifiers has shown that classifiers mainly rely on rotation-based features to determine the execution errors in the head movements. Similarly, *OneR* uses a single rotation-based feature (the 75^{th} percentile of the rotation about the *z*-axis) and achieves an F_1 score of 0.83. Overall, using the tracking data of the HMD, we can recognize if the head movement was performed correctly with an F_1 score above 0.95.

Nevertheless, to employ a one-sensor approach for full-body motion recognition, we also need to recognize the movements of the remaining body parts accurately. For the left- and right-hand movements, we achieve F_1 above 0.9, either using *Random Forest*, *SMO*, or *Naïve Bayes*. With meta-learning algorithms, only marginal improvements could be achieved. The F_1 score for the right hand using *JRip* is improved with *AdaBoostM1* or *Baggining* from 0.78 to 0.87 and 0.89, respectively (see Tables 33-34 in Appendix A.8). Similarly, the F_1 score for the left hand using *J48* is improved with *Bagging* from 0.79 to 0.87. Altogether, we achieve the highest accuracy for the left-hand movements using *SMO*, with an F_1 score equal to 0.95. The right-hand movements are recognized with an F_1 score equal to 0.98 using *Random Forest*.

However, the F_1 scores are low for the remaining sensors, especially the left and right foot. As already mentioned, the foot movements are similar for all three yoga poses (see Figure 22). That is why machine learning algorithms cannot distinguish among the poses based solely on feet sensor data. The results in Table 14 further indicate that the hip movements are also very similar among the three poses. The confusion matrices (see Table 35 in Appendix A.8) point out the extent of the misclassification for individual sensors, e. g., the foot movements in the three poses are often confused.

Because the foot movements are similar for all three yoga poses, we expect an accuracy comparable with the baseline ($F_1 = 0.17$). However, the results in Table 14 show that the performance for all machine learning algorithms is higher than the baseline. We believe this is due to the fact that the *extended side angle* requires more balance than the remaining two poses. During the evaluation, we observed that because the users need to bring the elbow down to the knee and look to the ceiling, they lost balance and took a small step to the side. Thus, users did not keep the right foot in one place throughout the execution. In contrast, *warrior I* and *warrior II* are more straightforward, and the players could keep the right foot in one place throughout the execution. This observation could also explain why the right-foot movements for *extended side angle* are not often confused with the *warrior I* and *warrior II* (see Table 35f in Appendix A.8).

Overall, the results indicate that we can recognize upper-body movements using models trained by single sensors. For example, we achieve an F_1 score of 0.94 and



(a) *Warrior I*

(b) *Warrior II*

(c) *Extended side angle*

Figure 22: Position data for different yoga poses.

above with *SMO* for head and hand movements. However, because with a one-sensor approach, we cannot reliably recognize hip and foot movements, we need to explore alternative approaches, e. g., *hidden Markov models*, which we will discuss in Section 5.2.3. Furthermore, we can employ rule-based decision-makers, which we will discuss in Section 5.3.1.

Two sensors

Overall, the results in Table 14 show that two sensors attached to the upper and lower body are sufficient to recognize the movements with an F_1 score up to 0.98. Merely, the recognition based on feet alone fails again, as the lower-body movements are very similar for all three poses. *SMO* performs best and recognizes the foot movements with an F_1 score equal to 0.57, although such a high score indicates overfitting. As already previously discussed, due to balance problems in users, they made small steps to the side, causing the classifiers to learn these execution errors. If we were to use such a model for online recognition, the movements of an expert perfectly performing a yoga pose would not be recognized. In contrast, the movements of an inexperienced user would be recognized. Thus, two sensors attached to both feet are not suitable for detecting movements within our database.

Nevertheless, a two-sensor approach can be suitable for sensor subsets combining upper and lower body. When we use more than one sensor, we need to ensure that a classifier uses at least one feature from each sensor. For example, tree-based classifiers such as *J48* with pruning or *REP-Tree* automatically choose the best features from the available feature subset. Pruning can result in a classifier using only features based on one sensor and ignoring the remaining features. Hence, a two-sensor approach with classifiers that do not incorporate at least one feature from each sensor is not always suitable to recognize movements based on both sensors. Nevertheless, the possibility of tree- and rule-based classifiers for full-body recognition will further be discussed in Section 5.3.1.

As we want to ensure that all classifiers with two sensors also use features that rely on both sensors, we further analyze the classifier's performance using only dual-sensor features, namely, distance in position and difference in velocity. Such an approach, where one sensor is attached to an arm and one to a foot, enables verification that users move the upper and lower body as desired. The data in Table 15 show that with a sensor attached to the right hand and left foot, we recognize movements with $F_1 = 0.91$, followed by $F_1 = 0.90$ for sensors attached to the left hand and left foot. Other sensor subsets tracking upper- and lower-body movements, e. g., left hand and right foot ($F_1 = 0.71$), or right hand and right foot ($F_1 = 0.67$), show inferior performance.

Table 15: Recognition performance for two sensors using only dual-sensor features.

Classifier	2 sensors			
	lHand-lFoot	rHand-lFoot	rHand-rFoot	lHand-rFoot
SMO	0.90	0.91	0.63	0.69
Naïve Bayes	0.85	0.85	0.63	0.62
J48	0.70	0.85	0.64	0.50
Random Forest	0.82	0.83	0.67	0.71
REP-Tree	0.74	0.73	0.65	0.67
JRip	0.68	0.73	0.63	0.61
Bayes Net	0.83	0.70	0.60	0.60
OneR	0.75	0.52	0.50	0.36
ZeroR	0.17	0.17	0.17	0.17

Abbreviations: *Left Hand* (lHand), *Right Hand* (rHand), *Left Foot* (lFoot), and *Right Foot* (rFoot).

Meta-learning algorithms do not significantly influence the overall result. Overall, within our data set, we achieve the best performance if we attach two sensors to the right hand and the left foot ($F_1 = 0.91$) or the left hand and the left foot ($F_1 = 0.90$).

Three sensors

An approach with three sensors for full-body motion recognition is only suitable if the recognition model of the three sensors attached to the upper body and the recognition model of the three sensors attached to the lower body is reliable. Such an approach enables us to provide feedback explicitly for the upper or lower body. In other words, depending on the results of the two classifiers, we can identify whether the movements of the upper and lower body are performed as desired. The results in Table 14 show that three sensors are sufficient to recognize the upper-body movements with an F_1 score above 0.93, using basis classifiers. Due to the basis classifiers' high accuracy, additional meta-learners do not lead to any further improvements.

However, as expected, with sensors attached only to the lower body, no machine learning algorithm can differentiate among the three yoga poses. The results show that the recognition with the three sensors attached to the lower body performs significantly worse than with the three sensors attached to the upper body.[3] Nevertheless, the recognition with three sensors on the lower body performs significantly better than with only two sensors attached to feet.[4] Thus, we can significantly improve the recognition performance if we additionally attach a sensor to the hips. We believe that the classifiers' performance is higher because the hip rotation slightly varies depending on the yoga pose. For example, when performing the *extended side angle*, users need to place the right elbow on the right thigh while the left arm should reach over the head. This movement causes users to rotate the hips. Examination of *OneR*, which uses a single feature to recognize movements, shows that the classifier indeed relies mainly on the hips' rotation.

Overall, although we recognize upper-body movements reliably with an F_1 up to 1.0, the approach with three sensors is not suitable for our purposes as the recognition of lower-body movements is unreliable. To improve the recognition of lower-body movements, we could integrate a subset with three sensors where at least one sensor is attached to the upper body.

Six sensors

All base classifiers with six sensors achieve an F_1 score above 0.9. Comparing the classifiers' performance between six and three sensors shows that the recognition mainly relies on the sensors attached to the upper body. In other words, when we use all six sensors, we achieve very similar results as if we would use only three sensors on the upper body or even only two sensors on both hands. These results indicate that the approach with six sensors is again not suitable for recognizing full-body movements

[3] One-tailed Wilcoxon signed rank test (lower-body vs. upper-body): $p = 0.002$

[4] One-tailed Wilcoxon signed rank test (lower-body vs. lFoot-rFoot): $p = 0.0059$

because the poses within our database can be entirely recognized based on upper-body movement. To overcome this issue, we need to include lower-body execution errors or poses which differ in their lower-body movements.

Discussion and Comparison with Related Work

As identified in Section 2.3, only a few research contributions tackle the recognition of full-body movements in VR-based applications. For example, exergames such as Astrojumper [84] and ExerCube [150], played inside a three-sided CAVE environment, indeed provide playful full-body exercise. However, Astrojumper [84] tracks only the upper-body movements. Nevertheless, the interdisciplinary team of ExerCube identified the importance of accurate lower-body tracking and expanded their tracking by additional HTC Vive trackers attached to the ankles [149].

Other studies presented in related work (see Section 2.2.3) often focus on recognizing upper-body movements and neglect accurate recognition of lower-body movements. In contrast, we recognize full-body movements by explicitly tracking the head, hands, hips, and feet. Our analysis based on different sensor subsets revealed that the approach based on two sensors with dual-sensor features performs best. In particular, the results show that we achieve an F_1 score equal to 0.91 if we use two sensors attached to the right hand and left foot.

Furthermore, although Jiang et al. [110] introduce a full-body motion recognition method, their action recognition algorithm relies only on the sensor data of the head and both hands. Additionally, Born et al. [24] propose an innovative approach for full-body motion recognition by requiring players to fit the defined shape made out of cubes. Thus, the number of cubes that fall out of the wall when a player collides with it indicates the player's performance. However, such an approach is only suitable to detect static poses. On the contrary, we recognize continuous movements by analyzing the entire movement execution and not only the final yoga pose.

5.2.3 *Hidden Markov Models*

Analyzing full-body movements to identify potential execution errors is challenging. The evaluation on sensor subsets in Section 5.2.2 showed that we recognize full-body movements with an F_1 score equal to 0.91 when we attach one sensor on the right hand in addition to one sensor on the left foot. However, with only two sensors, we cannot provide feedback on the movement of each limb.

For this reason, we further employ *hidden Markov models* because they can model time-series and are suitable for recognizing sequential activities [119, 235]. Previous studies have already demonstrated that *hidden Markov models* can be successfully applied to recognize hand and arm gestures [31, 231]. Alternatively, we could use other machine learning algorithms, such as *artificial neural networks*. However, such an approach requires a larger sample size.

Generally, a *hidden Markov model* is characterized by a set of parameters $\lambda = (N, M, \mathbf{A}, \mathbf{B}, \pi)$ [173]:

- N, the number of hidden states in the model,
- M, the number of distinct observation symbols per state,
- $\mathbf{A} \in \mathbb{R}^{N \times N}$, a matrix with transition probabilities between states,
- $\mathbf{B} \in \mathbb{R}^{N \times M}$, an observation symbol probability matrix, and
- $\pi \in \mathbb{R}^{N \times N}$, the initial probability distribution over all states.

Given an observation sequence $\mathbf{O} = (\mathbf{o}_1, ..., \mathbf{o}_T)$, we need to adjust the parameters to maximize the likelihood $P(\mathbf{O} \mid \lambda)$ [231]. We use raw time-series data to train models. In our case, an observation sequence consists of a position vector and a quaternion for each sensor at time step t. We use the *Baum-Welch* algorithm [13] to adjust the model parameters and maximize the likelihood. Afterward, we evaluate the trained *hidden Markov model* using the *Forward-Backward* algorithm, as described by Rabiner [173].

The full-body motion recognition approach's primary goal with *hidden Markov models* is to analyze the entire execution of a specific movement. The parameters for *hidden Markov models* are determined by observing the activity sequences over time. By evaluating individual models for each sensor, we identify the player's execution errors and provide appropriate feedback to let players know if they accomplished a movement correctly. Thus, we analyze individual sensor's motion execution and draw attention to the body part that did not move as intended. Such a system can then be used in exergames designed for rehabilitation purposes to assist patients by providing appropriate feedback or monitoring their recovery.

Training of Hidden Markov Models

Hidden Markov models have multiple hyperparameters that need to be selected, e. g., the number of hidden states N and the number of observation symbols M. Additionally, we need to define the topology to specify how states can be reached. In a fully connected model, every state can be reached from every other state [173]. Another possibility is to define that states proceed only from left to right, i. e., as time increases, the state index increases or stays the same.

In our initial work [50], we trained and tested *hidden Markov models* for a single yoga pose. The results revealed the importance of appropriate hyperparameters in order to increase the recognition performance. We found that we achieve better results with six to eight hidden states and eight to twelve observation symbols [50]. A training of *hidden Markov models* with six hidden states and eight observation symbols resulted in an accuracy of 88.75%. However, in this initial study, we used only the training data of a single user. As previously described in Section 5.1.3, personal models are tailored to one user and usually work only well for this user. Therefore, we subsequently collected training data of multiple users for three different yoga poses to build generic models that are expected to perform better for multiple users. The results on the recognition performance were publication in [47].

We perform a random search to find optimal hyperparameters because it is more efficient than a grid search [16]. The performance of hyperparameter combinations is then evaluated on the test data set. Based on our preliminary results [50], we search for the optimal hyperparameters as follows:

- number of hidden states $N \in \{6, ..., 16\}$,
- number of observation symbols $M \in \{3, ..., 100\}$, and
- topology $L \in \{0, 1, 2, 3\}$, where 0 indicates a fully connected model and $1 \leqslant L \leqslant 3$ indicates how many states can be transferred from left to right.

Figure 23 shows the F_1 scores of the best 12 parameter combinations. The analysis revealed that a fully connected ($L = 0$) *hidden Markov model* with six hidden states ($N = 6$) and seven distinct observation symbols ($M = 7$) achieves the highest F_1 score of 0.79 (see the rightmost column in Figure 23).

Furthermore, the result of *hidden Markov models* using only the position-based features reduces the recognition performance. As already discussed in Section 5.2.1, classifiers require features based on position and rotation. Thus, removing the rotation-based features generally reduces the overall performance of machine learning models.

Recognition Performance of Hidden Markov Models

Subsequently, we train final models for all yoga poses with the identified optimal hyperparameters, i. e., $N = 6$, $M = 7$, and $L = 0$. The standard process to train *hidden Markov models* is to use positive samples, i. e., samples belonging to a class [28, 72, 226]. In contrast, other studies propose methods that additionally also consider negative samples [145]. Consequently, following the standard process, we use only positive samples for training; however, we also use negative samples for testing.

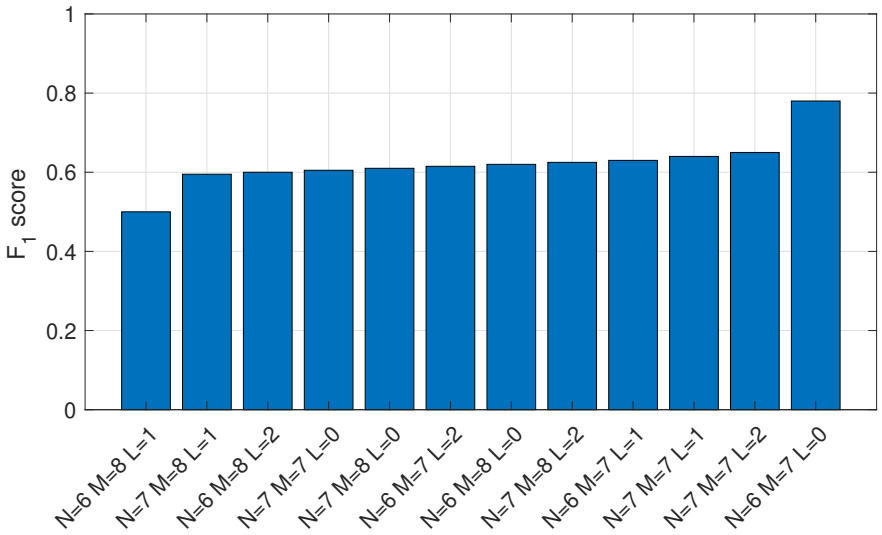


Figure 23: The results of the hyperparameter search.

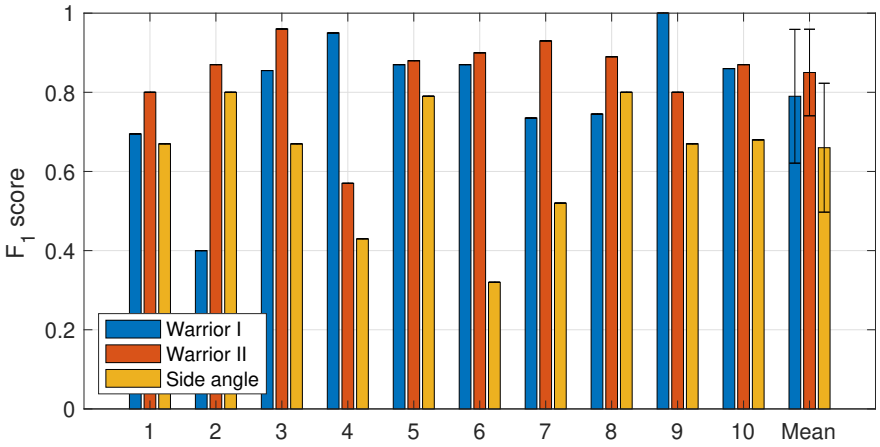


Figure 24: Performance evaluation of the *hidden Markov models* for the three poses.

We apply leave-one-subject-out cross-validation to assess the recognition performance. The cross-validation results in Figure 24 highlight the F_1 scores for each *hidden Markov model* (one model for each participant). The results show that we achieve an F_1 score of 0.79 for *warrior I*, 0.85 for *warrior II*, and 0.66 for *extended side angle*. Examining the F_1 scores for all three yoga poses, we assume that the difficulty level of movements influences the overall recognition performance. During the collection of training data, we observed that especially unpracticed participants had balance problems while bringing their arms up towards the ceiling. As a result, they made small steps to the side to gain balance. In this case, the final pose was correct; however, because the movement itself was not smooth, the *hidden Markov model* cannot recognize this movement correctly. We believe that this is the main reason for the lower F_1 score for the *extended side angle*, compared to the higher scores for the *warrior I* and *II*.

Limitations

The current recognition approach is limited by only three poses and one execution possibility for each yoga pose. For example, the current *hidden Markov models* only correctly recognize a yoga pose if a user takes a step backward to perform a lunge. Because we normalize the sensor data, the recognition is independent of the user's height and the step size. However, if the user takes a step forward, the *hidden Markov model* detects an execution error for the leg movement, although the final pose is correct. Hence, the current *hidden Markov models* recognize a pose only for one movement execution. To recognize several possible executions, we would need to train additional *hidden Markov models* for each execution possibility.

Furthermore, the current recognition based on *hidden Markov models* is limited by the execution time of a yoga pose. To overcome this limitation and recognize the movements accurately even when they are performed at different speeds, we could employ *dynamic time warping* or *derivative dynamic time warping* [116]. Alternatively, we could utilize *conversive hidden non-Markovian models*, as they have recently been developed to explicitly incorporate information about the speed of a movement [68].

With such an approach, it would be possible to detect movements that differ by their execution time even more precisely.

Discussion and Comparison with Related Work

Several studies have aimed to recognize human activities using *hidden Markov models*. For example, Karaman et al. [112] use a single camera attached to the user's shoulder to detect daily living activities. Results for all 32 activities show an accuracy of 42% and a higher accuracy of 60% for specific activities. Additionally, Trabelsi et al. [214] use acceleration data from inertial sensors attached to the chest, right thigh, and left ankle to recognize activities such as walking, sitting, and standing. The results show a classification rate of 91.4% when all three sensors are used and a lower accuracy between 83.9% and 86.2% when only two sensors are used.

Furthermore, Liu et al. [136] recognize basketball play actions using a smartphone with an accuracy of up to 69.44%. Zhu and Sheng [235] recognize eight daily activities, such as sitting, standing, and lying, with an overall accuracy of 85% using an inertial sensor attached to the right thigh. In a study by Brand et al. [31], the researchers recognize Tai Chi gestures involving only arm movements from a video with an accuracy of 94.2%.

Compared with related work, our recognition performance for *warrior I* with $F_1 = 0.85$ is similar, although the recognition performances for the *warrior II* and *extended side angle* with $F_1 = 0.77$ and $F_1 = 66$, respectively, are inferior. To improve the recognition performance, we would need to further tune the hyperparameters, particularly for the *extended side angle*. To achieve higher F_1 scores, we would also need a more extensive training and testing data set and, in particular, additional poses that differ more strongly in their lower-body movements. Furthermore, as proposed by Lockhart and Weiss [138], we could also integrate hybrid methods to improve the performance of generic models. Another possibility would be to employ *artificial neural networks*, as previous work has shown that they are well suitable for continuous gesture recognition [159].

5.3 ONLINE RECOGNITION

Accurate motion recognition in real-time is necessary to ensure that players perform movements as instructed. Additionally, in games, appropriate feedback on the player's performance and progress is also crucial. Thus, the primary goal of online recognition is to analyze the motion executions and draw attention to the body parts that were not moved as intended by the game. In Section 5.3.1, we discuss alternative approaches to identify execution errors in order to provide appropriate feedback.

5.3.1 *Appropriate Feedback*

Prior research has already thoroughly investigated the quality criteria for (serious) games. However, they often focus on game design [65, 211] or are specific to an

application field [5, 154, 176]. We adapt these existing principles and requirements from game-related literature to propose quality criteria for serious games that are effective and attractive. In particular, we present essential quality criteria for serious games that equally focus on (1) serious aspects, (2) game aspects, and (3) balance between them in [45]. Among others, high-quality serious games must support players in reaching the intended goal and require appropriate feedback on player's progress. Additional elaborated quality criteria for serious games are detailed in [45].

Generally, in exergames, appropriate feedback is crucial to avoid accidents or even to prevent players from injuring themselves because they perform an activity incorrectly. Feedback is also necessary to support players to work towards achieving the characterizing goal. For this reason, the intended effects (e. g., training effects) must be visible and recognizable. For example, players should always know what they need to do to complete a task. In addition to in-game feedback, post-game feedback can be beneficial to improve learning [176]. Game statistics indicate that an overview of players' progress at the end of the level or game is not only highly valuable for the players themselves. Particularly in games for health, such an overview of the player's progress can be useful for therapists, e. g., to monitor the recovery of a patient [180].

Therefore, we employ different machine learning algorithms to recognize full-body movements, identify execution errors, and provide appropriate feedback to support players so that they can then improve their movements. As discussed in Section 5.1.1, different sensor subsets are suitable to recognize full-body movements. However, when a user does not perform the desired pose correctly, it is challenging to identify why the classification failed and, in particular, which body part was moved incorrectly. In the following, we will discuss several possibilities that address this challenge.

Explicit Training with Execution Errors

In this thesis, we focus on correct executions of yoga poses. Alternatively, we could explicitly collect training data, including movements that are often performed incorrectly. For example, when performing a lunge, it is crucial that the front knee is bent to a right angle and is not allowed to exceed the foot. To identify such specific execution errors, we could explicitly collect samples that were intentionally performed incorrectly. In the training phase, we would then build models with the data of possible activity execution errors. Although such an approach is indeed promising to detect severe execution errors, particularly when activity execution errors are known, we would need to collect every possible incorrect execution. Furthermore, a requirement of larger training data set increases the costs and time of data collection.

Identifying Activity Execution Errors Based on Decision Trees

One possibility to identify execution errors is to use a decision tree and traverse backward to identify at which point the classification failed. However, when we build a decision tree using multiple sensors, we cannot ensure that the machine learning algorithm uses the features of all these sensors. Due to pruning, the Weka framework ensures that classifiers based on decision trees include only features that contribute to

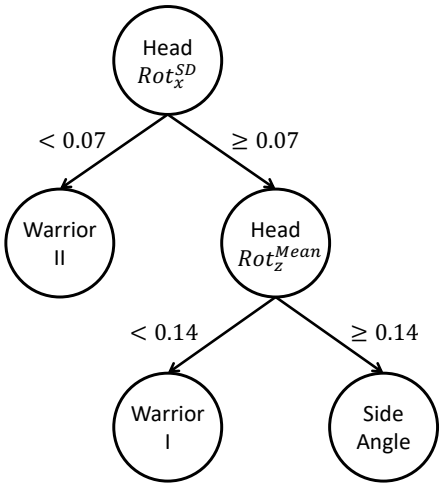


Figure 25: Example of a decision tree.

better overall performance. Including all features would otherwise reduce the performance or would even cause overfitting problems.

The extent of pruning is already apparent in Figure 25. Although the decision tree (*REP-Tree*) was trained on six sensors, the nodes contain only head sensor features (standard deviation and mean rotation about x- and z-axis, respectively). As previously discussed in Section 5.2.2, we achieve high performance with a single sensor on the head. As a result, a decision tree trained on all six sensors selects only head sensor features because additional features would otherwise decrease the overall performance. Thus, from the decision tree in Figure 25, we cannot verify all limb movements. When a player performs *warrior II* incorrectly and we traverse backward from the respective leaf, we only identify that the head rotation was incorrect. However, because the decision tree does not include features regarding the hands, hips, or feet, we cannot verify if the movements of these particular limbs were correct. Thus, as we cannot ensure that each branch from the leaf to the root includes at least one feature for each sensor, decision trees are generally not suitable to identify all execution errors.

Identifying Activity Execution Errors Using Individual Sensors

A model based on all six sensors is suitable for classifying poses; however, such a model is not suitable for identifying execution errors. Similarly, with three sensors, we only ensure that either the upper or lower body are moved as intended. However, to provide appropriate feedback and assist users, we need to analyze the movements of individual body parts. Therefore, we need to explicitly describe the movements of specific body parts to identify execution errors of individual limbs.

However, the lower-body movements within our database are very similar. To nevertheless identify activity execution errors using a one-sensor approach, we could define a rule-based decision-maker manually. With other words, while classifiers are used to recognize upper-body movements, we determine whether the lower-body movements are performed correctly by utilizing feature analysis. For example, we could examine

the variance in movements for the right foot. If the variance remains small, it means users kept the right foot in place. Furthermore, because users should take a large step backward until the right knee is bent to a right angle (see Table 25 in Appendix A.4 for movement description), the maximum value in the y-axis should be higher than zero (see Figure 10 for reference frames). Depending on the threshold, we could define how big a lunge should be. A small threshold would accept movement even if a user only takes a small step backward, whereas a large threshold would only accept movement for very flexible users that can perform a deep lunge.

Because the classifiers in Section 5.2.2 cannot differentiate foot movements, we alternatively employed *hidden Markov models*. To this end, we used time-series data and trained a separate model for each sensor to describe a sequence of each yoga pose separately. Through evaluation of the trained *hidden Markov models*, we are able to identify the user's activity execution errors and provide appropriate feedback on their performance. However, the results presented in Section 5.2.3 show only low performance of the trained *hidden Markov models*: F_1 score of 0.79 for *warrior I*, 0.85 for *warrior II*, and 0.66 for *extended side angle*. These results point out the necessity for training models with bigger sample size and further tuning the hyperparameters to improve the performance of the *hidden Markov models*.

Recognizing Full-Body Movements Using Dual-Sensor Features

Another possibility to ensure that full-body movements are performed as instructed is to use at least two sensors. Thereby, one sensor needs to be attached to the hand and another one to the feet. Such an approach, including upper- and lower-body movements, enables us to ensure that full-body movements are performed as intended; however, it does not allow us to identify activity execution errors of individual body parts. Furthermore, when execution errors need to be detected based on two sensors, we need to ensure that the classifiers use the features of both sensors. For example, *Naïve Bayes* assumes that all attributes are equally important and statistically independent [228]. However, other machine learning algorithms, especially rule- or tree-based classifiers with pruning, usually do not use all features. Instead, they automatically choose the best features from the available feature subset and ignore the remaining features that do not contribute to better performance.

To overcome this issue, we employed dual-sensor features. With dual-sensor features, we ensure that the machine learning algorithms equally rely on both sensors. By attaching one sensor to the lower body and one sensor to the upper body, such an approach makes it possible to recognize full-body movements. The analysis on sensor subsets in Section 5.2.2 revealed that we achieve the best results within our data set with F_1 equal to 0.91 and 0.90 if we attach one sensor to the left foot and one to the right or left hand, respectively.

6

VALIDATION OF FULL-BODY MOTION RECONSTRUCTION AND RECOGNITION

This chapter implements and validates elaborated models on full-body motion reconstruction and recognition in the context of two Virtual Reality (VR)-based serious games, namely (1) immersive training simulation for police forces and (2) exergame to train and practice particular movements. In Section 6.1, we explore the added value of full-body motion reconstruction in multiplayer VR training simulation for police forces. Full-body motion reconstruction thereby enables users of a shared virtual environment to communicate through body language and gestures. The results of the user study with police officers confirmed the importance of full-body avatars to induce a more threatening situation and provide a realistic training environment.

Subsequently, we evaluate the capability of the full-body motion recognition model to identify motion execution errors and provide appropriate feedback in Section 6.2. To this end, we develop an engaging VR-based exergame to assist players in training and practicing yoga. The results of a user study show that the full-body motion recognition model identifies the player's execution errors and provides appropriate feedback so that players can then improve their movements.

6.1 VIRTUAL REALITY TRAINING SIMULATION

VR-based training simulations are a valuable tool for police forces to practice specific skills within a safe, controlled, and monitored environment. Virtual simulations can be used to train novice police officers, e. g., to decrease response time in threatening, dangerous, and stressful situations. Thus, such a training simulation can supplement traditional training and offers an opportunity to train dangerous and stressful situations that are otherwise too expensive or complicated to rebuild in the real world.

We briefly present the concept for a singleplayer VR training simulation in Section 6.1.1. Subsequently, we discuss the added value of full-body motion reconstruction in multiplayer VR training simulation in Section 6.1.2. For example, full-body motion reconstruction offers an opportunity to visualize full-body avatars for all players, thus enabling communication among team members in shared virtual environments. Furthermore, such an approach can be used to record the movements of the police officers during the training and evaluate their behavior afterward in a debriefing phase.

6.1.1 *Singleplayer Virtual Reality Training Simulation*

In cooperation with ten police officers (eight men, average age 27), we raised requirements and elaborated a concept for a singleplayer VR training simulation as a tool to learn particular behavior and strategies. In traditional training, police officers usually

learn and train specific behaviors in role-playing games. Such training requires at least two police officers and must be additionally assessed by an expert. Furthermore, frequent training of different situations in the real world is often lacking due to high risk, setup effort, and costs. Therefore, traditional training can be complemented by virtual training simulations to offer a safe training environment, where users can undertake high-risk tasks without dangerous real-world implications. Recent studies already provided significant evidence that virtual training can be as efficient as traditional training [18, 153].

We have identified various relevant scenarios where the police officers have to follow a certain procedure, e. g., during a police control. Consequently, we prototypically implemented a virtual training environment, where police officers, among others, need to inspect the driver's license and the registration papers (see Figure 26). The training environment was designed in the Unity game engine[1] and was displayed on an HTC Vive HMD to enable a first-person view of the virtual environment. To make training more effective, the inspected items vary so that the police officers can frequently practice certain scenarios and make mistakes in order to gain experience that will help them avoid bad decisions in the future. The findings of the participatory design of our training simulation were published in [40].

The initial evaluation of the first prototype with three police officers (all male, aged between 27 and 34 years old) revealed the necessity of full-body motion tracking. The importance of full-body avatars to enhance the sense of presence was also already identified in Section 2.2.2. In particular, the participants found it irritating to see only HTC Vive controllers floating in the air. Therefore, we subsequently improved the training scenario and integrated full-body avatars, which move synchronously with the players and further increase the sense of presence (see Figure 27). Furthermore, to enhance the suspense, we integrated additional scenarios with alternative endings. Thereby, the story changes, depending on how the user reacts. For example, the training simulation includes a cooperative and an aggressive non-player character. The outcomes of the advanced game scenario were published in [51].

Figure 26: Virtual environment for the police control training simulation (left) and the game menu with feedback (right).

[1] `https://unity.com/de`, last accessed on March 3, 2021

Figure 27: Full-body avatars from the third-person perspective and the corresponding player's poses [51]. Note that players can view the virtual body from the first-person perspective.

6.1.2 *Multiplayer Virtual Reality Training Simulation*

Training scenarios must be as realistic as possible to enable users to learn and develop skills and tactics, which they can subsequently transfer to the real world. Thereby, full-body motion reconstruction is highly valuable in multiplayer VR scenarios. Without full-body motion reconstruction, players cannot see their real bodies, nor those of their teammates or opponents. Furthermore, training simulations need to ensure a high degree of presence so that users are able to behave in the virtual world similarly to how they behave in the real world. In multiplayer scenarios, this also includes non-verbal communication among teammates. Therefore, with full-body motion reconstruction, we enable users of a shared virtual environment to communicate non-verbally using body language and gestures.

The importance of full-body avatars to build more realistic VR applications has also been identified by Eller in the context of training environments for firefighters [74, 75]. Eller determines the quality and quantity of an application through immersion, presence, and interaction. Among other aspects, full-body avatars that naturally interact with a virtual environment and move naturally contribute to a higher sense of presence. Additionally, full-body avatars with natural facial expressions and gestures are also crucial for more realistic interaction. However, Eller [74] also points out that reconstruction errors disrupt the sense of presence or destroy the illusion of being in the virtual environment.

Training Scenario

We intend to investigate the impact of full-body motion reconstruction in multiplayer VR training simulation. To this end, we elaborated, in cooperation with police officers,

a concept for a stressful training environment simulating a shootout situation, which we submitted for publication in [49]. The developed simulation is not a pure shooting training but tactical behavior training with the primary goal of quickly finding bullet-proof cover behind vehicles. Similar to role-playing games in traditional training, one user plays the role of a police officer while the second user plays the role of an assailant. In different conditions, the users either see a full-body avatar of an assailant or only a partial-visible avatar with head and hands (see Figure 28). The condition with abstract avatar representation with only head and hands (i. e., floating controllers) is common in many popular VR games.

In the beginning, both users stand on their side of the car. The assailant is hiding her or his hands so that the police officer cannot see if they are armed. After a short conversation, the police officer asks the assailant to show their hands. In this case, the assailant draws the weapon and starts to shoot. The police officer needs to react quickly and should seek protection behind a car (see Figure 29).

The current training scenario implements a competitive multiplayer mode, allowing police officers to compete against each other. To further enable collaborative multiplayer mode, we could replace users with non-player characters so that they then play together against the assailants. With additional HTC Vive base stations that can track multiple HTC Vive devices, we could also offer additional police officers an opportunity to train in a shared virtual environment. However, the development of multiplayer games is challenging. Because multiple players act concurrently, it is often difficult to run through all possible variants in the game. Due to the challenges in the development of multiplayer games, authoring environments have already shown that they are beneficial in developing such games [183].

The developed VR scenario could further be expanded by additional buildings. Compared with traditional training, computer-based simulations enable the easy configuration of virtual training environments. For example, solutions for procedural generation of buildings [86] with multiple rooms connected by doors or stairways and the extension of existing building models [21] have been proposed. Such virtual environments are particularly suitable for training scenarios in which the police officers need to enter and search through unknown territory.

Figure 28: The assailant with a full-body avatar (left) or an abstract representation with only head and hands (right) [49].

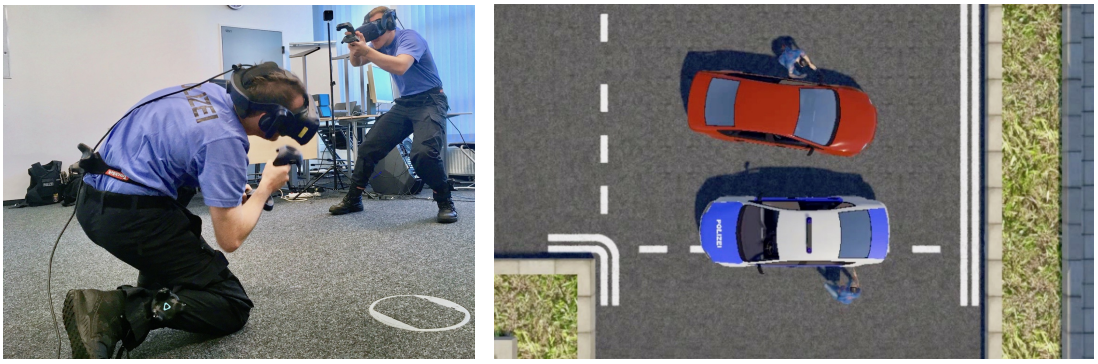


Figure 29: Two police officers acting in the immersive VR training environment (left) and the corresponding top-down view of the game scene (right) [49].

6.1.3 *Evaluation*

We conducted a within-subject experiment under two conditions: the participants could see a full-body avatar and an abstract representation of the assailant with head and hands. Hence, all participants tested both conditions in a counterbalanced order. A video[2] demonstrates the virtual representation of both avatars.

Measures

The aim of this study was to investigate which avatar representation is more threatening by investigating the stress level between both conditions. Previous studies on *rubber hand illusion* in VR (see Section 2.2.2) have already shown that the sense of body ownership plays an important role in the physiological response to a threat. Thereby, the stress level is often detected by analyzing the skin conductance response [9, 234], brain activity [73], heart rate deceleration [199], or cortisol level [32]. Another possibility to detect stress is to examine the heart rate variability [32, 120].

Consequently, to detect mental stress, we analyze the heart rate and heart rate variability collected from a heart rate sensor. We determine the heart rate variability by analyzing the time intervals between consecutive heartbeats, i. e., RR intervals (see Figure 30). The RR intervals indicate the amount of time between two R peaks. We calculate the mean heart rate and root mean square of successive RR interval differences by analyzing the time-domain measures. Additionally, by evaluating the frequency-domain measures, we obtain low- and high-frequency powers. For the analysis, we use the heart rate variability analysis software package for MATLAB [174, 175].

In addition to the heart rate variability, we validate our results by assessing the sense of presence using a questionnaire proposed by Witmer and Singer [227]. All questions could be responded to on a five-point Likert scale ranging from zero (not at all) to four (extremely). Moreover, we employ Kennedy's *Simulator Sickness Questionnaire* [113]

[2] `https://youtu.be/-B_gBn7b034`, last accessed on March 3, 2021

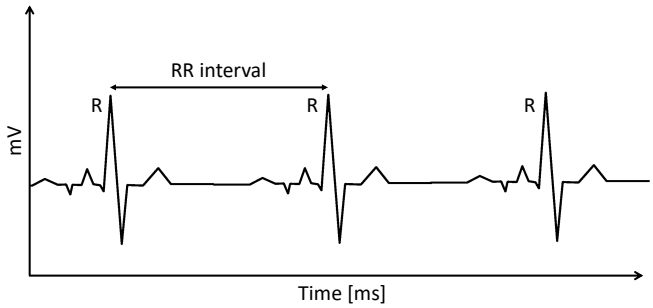


Figure 30: RR intervals.

to evaluate if the current VR simulation causes cybersickness. Each symptom on the *Simulator Sickness Questionnaire* can be assessed on a scale from zero (no symptoms) to three (severe symptoms), as described in Appendix A.1.

Participants

We initially recruited 38 police officers. Due to technical problems, we discarded the tracking data of six participants. Consequently, we use the data of 32 participants (23.1 ± 3.0 years, between 20 and 32 years, five females) for the analysis. All participants were police students at the Hessian University of Applied Sciences for Police and Public Administration. Unfortunately, at the time of the user study, only one heart rate sensor was available. For this reason, only the participants playing the role of a police officer were wearing a sensor. Thus, experiments involved the RR intervals of 16 participants (22.67 ± 2.23 years, between 20 and 27 years, two females).

Procedure

Before the participants put on the HTC Vive Head-Mounted Display (HMD), they attached three HTC Vive trackers to their hips and both feet. To enable full-body motion reconstruction, they furthermore held two HTC Vive controllers in the hands. The participant playing the role of the police officer additionally put on the Zephyr heart rate sensor.[3]

To access the resting values, the participants could explore the virtual environment without seeing each other for approximately 6 min. Afterward, both participants played the simulation two times, for approximately 3 min in each condition. Between both conditions, the participants had a short break (approximately 3 min) to ensure resting heart rate before the participants started with the second condition. Hence, all participants were immersed for approximately 15 min.

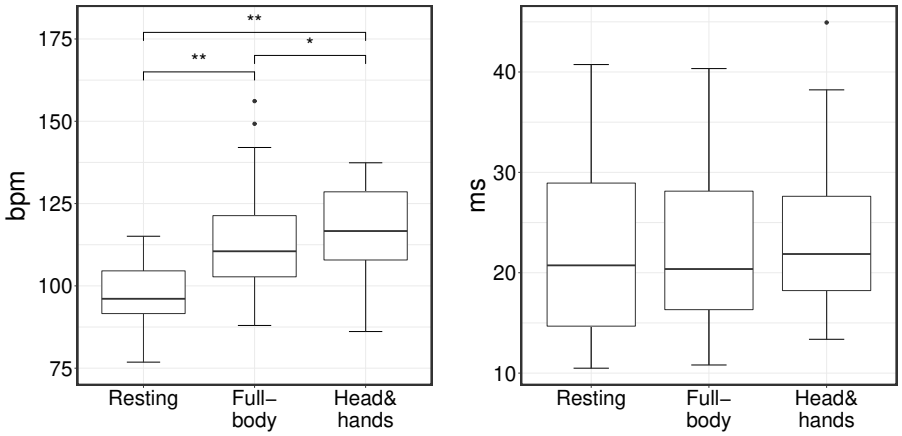
Results on Mental Stress

Because we found some errors and artifacts in the recorded data of the heart rate sensor, we first removed outliers and smoothed noisy data with a moving average filter.

[3] `https://www.zephyranywhere.com/system/hxm`, last accessed on March 3, 2021

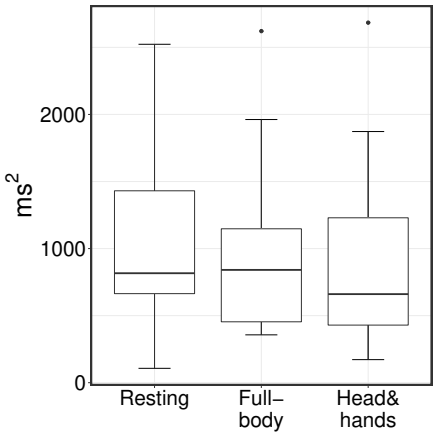
Afterward, we applied time-domain and frequency-domain analysis. The results of the analysis are shown in Figure 31. Here we focus on analyzing the low- and high-frequency power. Generally, it is assumed that the power in the high-frequency band is low for mentally or physically stressful events, while the power in the low-frequency band is high for mentally and low for physically stressful events, while the power in the low-frequency band is high for mentally and low for physically stressful events [185, 192].

Because the heart rate data do not follow a normal distribution, we explore the statistical significance by conducting Friedman's test. We apply the Conover post-hoc test with a Bonferroni correlation for pairwise comparisons to investigate if there are statistically significant differences between the two avatar conditions. Additionally, to explore the effect size, we calculate Hedge's g [186], with a value of 0.2 indicating a small, 0.5 a medium, and 0.8 a large effect size [55]. The results on effect size are detailed in Table 16.

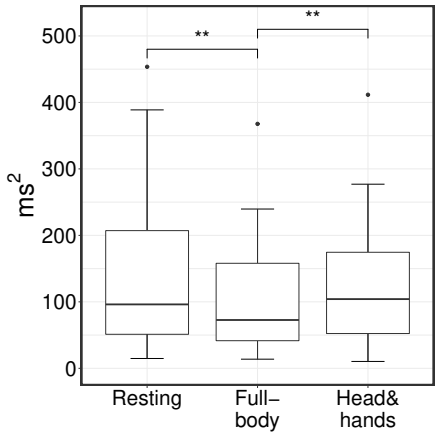


(a) Mean heart rate

(b) Root mean square of successive RR interval differences



(c) Low-frequency power



(d) High-frequency power

Figure 31: Heart-rate variability during the resting and stress phase with either a full-body avatar or an abstract representation with head and hands. * denotes statistic significance for $p < 0.05$ and ** statistical significance for $p < 0.01$.

Table 16: Effect sizes (Hedge's g) during the resting phase and stress condition with either a full-body avatar or an abstract representation with head and hands.

HRV	**Resting vs. full-body**	**Resting vs. head&hands**	**Full-body vs. head&hands**
mHR	1.08	1.04	−0.08
RMSSD	−0.02	0.14	0.15
LF	−0.18	−0.27	−0.11
HF	−0.28	−0.08	0.22

Interpretation: a value of 0.2 indicating a small, 0.5 a medium, and 0.8 a large effect size [55].
Abbreviations: *Mean Heart Rate* (mHR), *Root Mean Square of successive RR interval differences* (RMSSD), *Low-Frequency power* (LF), and *High-Frequency power* (HF).

The results of Friedman's test revealed a significant difference in heart rate ($\chi^2(2) = 14.63$, $p < 0.001$, Kendall's $W = 0.63$). The mean heart rate (measured in beats per minute, bpm) increases from $\mathrm{mHR}_R = 96.08\,\mathrm{bpm}$ in the resting condition to $\mathrm{mHR}_{S-FB} = 110.52\,\mathrm{bpm}$ and $\mathrm{mHR}_{S-HH} = 115.41\,\mathrm{bpm}$ in the stress condition with either the full-body avatar or head-and-hands, respectively (see Figure 31a). A pairwise comparison reveals a significant difference between the resting phase and full-body avatar ($p < 0.001$, Hedge's $g = 1.08$), the resting phase and head-and-hands condition ($p < 0.001$, Hedge's $g = 1.04$), as well as the full-body avatar and head-and-hands condition ($p = 0.02$, Hedge's $g = -0.08$).

Friedman's test shows no significant difference in root mean square of successive RR interval differences ($\chi^2(2) = 0.50$, $p = 0.78$, Kendall's $W = 0.53$) among the three conditions (see Figure 31b). Compared with the resting phase ($\mathrm{RMSSD}_R = 20.74\,\mathrm{ms}$), the value decreases for the full-body avatar ($\mathrm{RMSSD}_{S-FB} = 20.37\,\mathrm{ms}$) and increases for the head-and-hands condition ($\mathrm{RMSSD}_{S-HH} = 21.86\,\mathrm{ms}$). Similarly, Friedman's test shows no significant difference in low-frequency power ($\chi^2(2) = 0.38$, $p = 0.83$, Kendall's $W = 0.32$) among the three conditions (see Figure 31c). Compared with the resting phase ($\mathrm{LF}_R = 816.18\,\mathrm{ms}^2$), the low-frequency power increases for the full-body avatar ($\mathrm{LF}_{S-FB} = 840.98\,\mathrm{ms}^2$) and decreases for the head-and-hands condition ($\mathrm{LF}_{S-HH} = 660.39\,\mathrm{ms}^2$).

However, as depicted in Figure 31d, Friedman's test shows a significant difference in high-frequency power ($\chi^2(2) = 6.50$, $p = 0.04$, Kendall's $W = 0.43$). The post-hoc test shows a significant difference between the resting phase and full-body avatar ($p < 0.001$, Hedge's $g = -0.28$) as well as the full-body avatar and head-and-hands condition ($p < 0.001$, Hedge's $g = 0.22$), however not between the resting phase and full-body avatar ($p = 0.29$). Higher low-frequency and lower high-frequency values indicate increased mental stress level ($\mathrm{HF}_R = 96.17\,\mathrm{ms}^2$, $\mathrm{HF}_{S-FB} = 72.71\,\mathrm{ms}^2$, and $\mathrm{HF}_{S-HH} = 104.14\,\mathrm{ms}^2$).

Results on User Experience

The results on the sense of presence are shown in Figure 32 and further detailed in Table 37 in Appendix A.9. The majority of the participants (68%) stated that the environment was very responsive to their actions ["Responsiveness of actions"]. Participants did not experience delays between the actions and expected outcomes ["Delay between actions and outcomes"]. Furthermore, the participants found the feeling of moving in the virtual environment compelling ["Sense of moving"].

The results also show that participants were, on average, not aware of the events occurring in the real world ["Distraction due to events in the real world"]. However, some participants tend to disagree. We believe that the main reason why some participants were more aware of the real-world events is that because the tracking area was small (3 m × 3 m), they sometimes collided with real-world objects, e. g., a wall or a table. In this case, the supervisor tried to draw attention to the hazards verbally to prevent participants from injuring themselves.

Furthermore, participants were partially aware of the HMD and control devices ["Distraction due to the display and control devices"]; however, additional trackers used for full-body motion tracking did not disturb ["Distraction due to the trackers"]. Some participants stated that the visual display quality sometimes impeded them ["Display quality"]. Although the HTC Vive Pro HMD used in this study has a high-resolution display, it seems that even higher resolution is required for training simulations so that the police officers can better aim at the enemy. Furthermore, the control mechanism used in the simulation was not distracting ["Distraction due to control mechanism"]. The majority of the participants (82%) could concentrate on the assigned tasks rather than on the mechanism ["Mechanism quality"].

The results further indicate that the current VR training simulation also needs to be improved. For example, the virtual experience does not seem consistent with

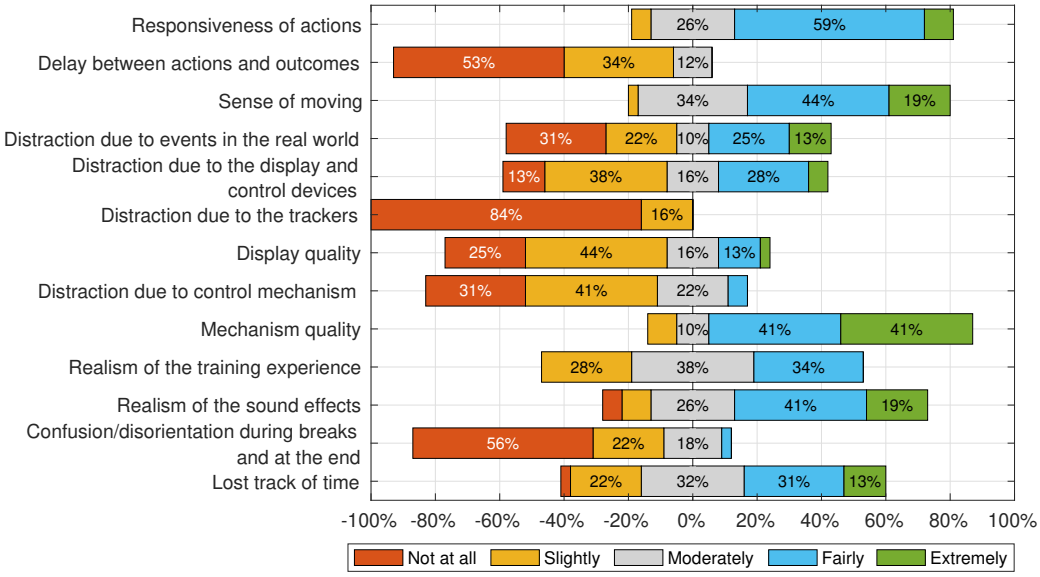


Figure 32: Sense of presence questionnaire results.

the real-world experience ["Realism of the training experience"]. Nevertheless, the sound effects contributed to the more realistic experience ["Realism of the sound effects"]. Furthermore, participants stated that they miss the recoil feedback of a weapon (backward movement of a weapon when discharged).

In the breaks or at the end of the experiment, the participant did not feel confused or disoriented ["Confusion/disorientation during breaks and at the end"]. Additionally, participants were, on average, involved in the experimental tasks and partially lost track of time ["Lost track of time"].

Moreover, by answering the question about the assailant ["Did the full-body avatar seem more threatening than the avatar with head and hands?"], around 78% of the participants responded that the full-body avatar was more threatening than the assailant with head and hands. Additionally, by answering the question about the training experience ["Which avatar contributed to a better training experience?"], the majority of the participants (94%) replied that the full-body avatar contributed to a better training experience.

Results on Cybersickness

We derived scores for nausea, oculomotor, disorientation, and the total score before and after the experiment from the *Simulator Sickness Questionnaire*. The data in Table 17 show high total scores before ($M = 10.17$, $SD = 13.94$) and after ($M = 18.12$, $SD = 15.84$) the training simulation. According to Stanney et al. [205], cybersickness symptoms are significant at scores above 10 and concern at scores above 15. We furthermore investigate if there is a significant increase in post total scores compared to pre total scores. Because according to the Anderson-Darling test, the data follow a normal distribution, we consequently conducted a one-tailed t-test. The results confirm that the total score is significantly higher after the simulation than before ($t(32) = -3.73$, $p < 0.001$).

However, the difference between the total scores before and after the simulation ($M = 7.95$) indicates that, according to Stanney et al. [205], the cybersickness symptoms are only minimal. During the evaluation, we could also observe that participants were sweating excessively. Because sweating is one of the nausea symptoms [113], we believe that the high total scores primarily arise due to physical activity and not from cybersickness.

Table 17: The mean scores on the *Simulator Sickness Questionnaire*, measured before and after the VR training simulation for police forces.

	Nausea	**Oculomotor**	**Disorientation**	**Total score**
	M (SD)	M (SD)	M (SD)	M (SD)
Pre	8.05 (11.39)	9.24 (13.98)	9.14 (18.91)	10.17 (13.94)
Post	20.57 (15.72)	12.08 (13.59)	15.23 (24.07)	18.12 (15.84)

Abbreviations: *Mean* (M) and *Standard Deviation* (SD).

Conclusions and Limitations

Heart rate variability has been previously shown to be an important indicator to detect mental stress. However, as Berntson et al. [17] point out, posture, age, activity, and aerobic fitness can influence heart rate variability. A study by van Rosenberg [185] explicitly reduces the ambiguities in stress categorizations. The researchers reveal that high-frequency power decreases for both physical and mental stress, while low-frequency power increases for mental stress and decreases for physical stress. Additional studies on mental stress also report a decrease in the high-frequency and an increase in low-frequency power [120, 192].

According to our results, the low-frequency power increases for full-body avatar compared to head-and-hands conditions; however, not significantly. In fact, due to combined stress (mental and physical), it is not surprising that the low-frequency power does not change significantly. Small changes in physical activity occlude the effects of mental stress in heart rate variability. Hence, low-frequency power decreases due to physical stress and increases due to mental stress. This could explain why low-frequency power remains constant.

However, our results show a statistically significant difference in high-frequency power for the full-body avatar compared to the head-and-hands condition. As the low-frequency power increases and high-frequency power significantly decreases in the stress condition with the full-body avatar compared to the stress condition with the head and hands, the results suggest that the full-body avatars induce higher mental stress. Hence, our results indicate that participants were rather exposed to mental stress than physical stress. However, as the results show only a small effect size (see Table 16), we cannot draw definite conclusions.

Another limitation of the study is the small sample size. According to the power analysis with G*Power [82], we need at least 22 participants to detect a medium effect with a power of 0.9 and an alpha of 0.05. However, in our study, we analyzed the heart rate variability of only 16 participants.

Although the current study is based on a small sample of participants, the questionnaire ratings and the analysis of heart rate variability show a tendency that the assailant with the full-body avatar appears more threatening and contributes to a better training experience than the assailant with head and hands. These results show an added value of full-body avatars over an abstract representation with head and hands only. In particular, as the full-body avatar assailant contributed to a stronger response to the threat, these results indicate that full-body motion reconstruction is beneficial for multiplayer VR-based training simulations for police forces. Eller [74] further supports the importance of full-body avatars. Especially in simulations, in which the training scenarios have to be as realistic as possible, the full-body movement reconstruction offers the user the opportunity to interact through body language and gestures. Such realistic virtual environments allow the users to develop skills and tactics, which they can then transfer to the real world.

Nevertheless, the current VR simulation has potential for improvements. The user study with 32 police officers showed that we need to provide a more realistic training environment. In particular, the police officers expressed that they missed feedback

when they got shot. To further enhance the realism and user experience, we need to include haptic feedback. For example, full-body haptic suits, such as Teslasuit[4], offer great potential for haptic feedback.

Furthermore, because the police officers missed the recoil feedback of the weapon, we could incorporate additional weapons. Recently, Krompiec and Park [125] have provided a template for weapons with haptic feedback using the HTC Vive controllers. Additionally, commercially available weapons with recoil feedback have been lately released [19]. Especially in stressful situations, realistic weapons are necessary so that police officers can train to change the magazine quickly, reload, and eventually clear the weapon (if a bullet gets stuck). The importance of real-world equipment to enhance the immersion was already shown in a collaborative VR training scenario for firefighters, e. g., the usage of an actual fire extinguisher allows users to learn new interactions [76]. Further examples demonstrate the usage of a vest to stimulate heat from a fire and a fog machine to simulate the smell and taste of smoke.[5]

6.2 VIRTUAL REALITY-BASED EXERGAME

Studies have already shown that exergames have a beneficial effect on the players' physical activity, e. g., they increase players' activity level [4], energy expenditure [128], and muscle strength [202]. However, many exergames focus on the game design and neglect proper training concepts or performance aspects such as accuracy and intensity. Only a minority of exergames involve an interdisciplinary team of sports scientists, game designers, and human-computer interaction researchers [150].

In addition to appropriate training concepts, we identified that high-quality exergames require suitable tracking technology to avoid adverse effects, such as accidents and injuries [45]. As already identified in Section 2.3, especially in exergames, we need to track movements accurately so that players perform the exercises correctly. To this end, we developed an exergame that simultaneously reconstructs and recognizes full-body movements. Due to full-body motion recognition, we detect activity execution errors and provide feedback so that players can then improve their movements.

(a) *Warrior I* (b) *Warrior II* (c) *Extended side angle*

Figure 33: Reconstructed full-body yoga poses used in the developed game scenario [47].

[4] `https://teslasuit.io`, last accessed on March 3, 2021

[5] `https://youtu.be/ftJcg9ligtc`, last accessed on March 3, 2021

6.2.1 *Game Design*

We developed and implemented a VR-based exergame that embeds yoga in a narrative and motivates players to train and repeat yoga poses. The elaborated game design and the results of the user study were published in [47]. Figure 33 depicts reconstructed full-body avatars for the yoga poses. As previous studies have already shown the benefits of yoga in reducing pain [37, 171], such an exergame could be used in a home environment to supervise players' improvements. Another possible application example would be a training or rehabilitation scenario where therapists or doctors can monitor the patient's rehabilitation.

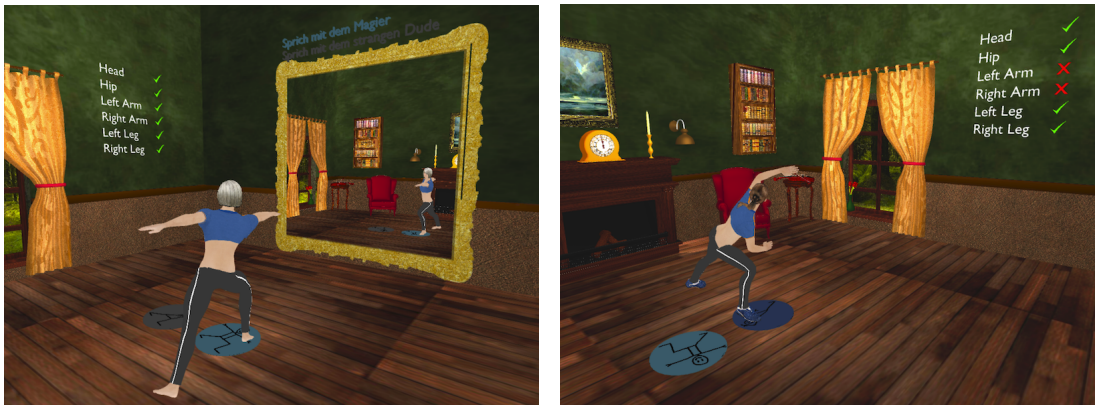
The game scenario consists of an adventure story that follows an accident of a circus acrobat (the player) who fell from a trapeze because somebody deliberately cut the rope. To solve the mystery and to find the culprit who cut the rope and intentionally caused the accident, the player needs to speak with different non-player characters by practicing yoga poses. Thus, the story proceeds as the player repeats yoga poses.

In total, the game scenario consists of 31 paths and 10 characters. In each new game state, the player can choose a character to speak with to receive more details about the culprit. To enable the player to learn new movements, each game state ensures that the player performs a different yoga movement as in the previous state. For example, as shown in Figure 34, the pose from the game state x is disabled in a subsequent game state $x + 1$.

Because several studies agree that feedback on a player's performance is important to enhance player enjoyment in games [65, 211] and to ensure that players can work towards achieving the characterizing goal [45], we display feedback immediately after a player performs a pose. We use *hidden Markov models* trained in Section 5.2.3 to identify activity execution errors. As demonstrated in Figure 35a, the feedback shows players which body parts were moved correctly. In case the recognition system detects an execution error (see Figure 35b), the player needs to repeat the pose. If the player

(a) Game state x (b) Game state $x + 1$

Figure 34: The game ensures that the pose from the current game state is not available in the subsequent game state [47].



(a) All tracked body parts were moved as intended by the game.

(b) Recognized movement execution errors for hands.

Figure 35: Feedback example for individual body parts [47].

cannot perform a movement correctly after ten trials, the new game state is loaded to avoid frustration.

The video[6] demonstrates how the full-body motion reconstruction and recognition are integrated into the VR-based exergame. Players can see their full-body avatar in a virtual mirror and by looking down towards their body. It should be noted that we used two additional trackers attached to the elbows for more precise upper-body movement reconstruction. The video further intentionally demonstrates a wrong execution of a yoga pose. For example, when the player takes a step backward (instead of taking a step forward), the system identifies this activity execution error and displays corresponding feedback.

6.2.2 *Evaluation*

We conducted a user study to evaluate the preliminary game design and to find out if the current exergame provides appropriate feedback and is suitable to motivate players to learn and practice yoga.

Measures

To gain further insights into players' immersion and flow experience, we use the core module of the *Game Experience Questionnaire* [108], which is widely used in game research to assess player experience, although recent research has raised issues with this questionnaire [132]. The *Game Experience Questionnaire* consists of 33 items to evaluate competence, sensory and imaginative immersion, flow, tension/annoyance, challenge, as well as negative and positive affect. Each item can be responded to on a five-point scale ranging from zero (not at all) to four (extremely). We calculate the mean (M) value and standard deviation (SD) to access components' scores.

[6] `https://youtu.be/J2bgYozfsDw`, last accessed on March 3, 2021

Furthermore, because cybersickness is still one of the main concerns in immersive virtual environments, we employ the *Simulator Sickness Questionnaire* proposed by Kennedy et al. [113]. Thus, to evaluate if the current game causes any symptoms, participants need to rate each symptom on the *Simulator Sickness Questionnaire* from zero (no symptoms) to three (severe symptoms), as described in Appendix A.1.

Participants

We recruited 32 participants (31.09 ± 9.2 years, between 22 and 60 years, 14 females, one other). The participants either had never done yoga before or were beginners. At the beginning of the evaluation, we obtained informed consent from all participants. We informed all participants about the purpose of the user study and explained that all collected data are confidential and will only be used in anonymized form. The participants were informed about the risks and that they can always end the experiment without specifying a reason. Afterward, all participants filled out a demographics questionnaire.

Procedure

Before the game started, each participant could choose between different male and female avatars. Prior to participants putting on the HTC Vive HMD, they attached additional sensors to enable full-body motion reconstruction and recognition. Subsequently, participants calibrated the avatar so that the full-body motion reconstruction and recognition work independently of the body height and position as well as rotation of the attached sensors. The participants then played the exergame for an average of 11.9 min.

Results on Game Experience

The *Game Experience Questionnaire* results are presented in Table 18 and further detailed in Table 38 in Appendix A.9. With high scores for flow ($M = 2.74$, $SD = 0.71$) and

Table 18: *Game Experience Questionnaire* results.

Factor	**M (SD)**
Competence	1.98 (0.77)
Sensory and imaginative immersion	2.20 (0.60)
Flow	2.74 (0.71)
Tension/Annoyance	0.65 (0.79)
Challenge	1.45 (0.56)
Negative affect	0.46 (0.56)
Positive affect	2.94 (0.73)

Abbreviations: *Mean* (M) and *Standard Deviation* (SD).

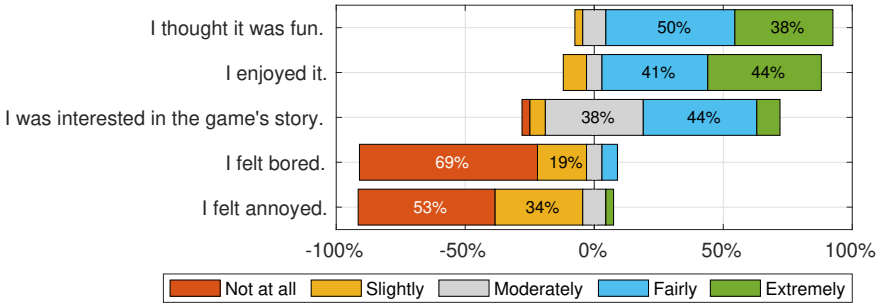


Figure 36: Selected items of the *Game Experience Questionnaire*.

positive affects (M = 2.94, SD = 0.73), we can say that, on average, the players were occupied with the game, lost track of time, had fun, and enjoyed the game. In particular, more than 85% of the participants stated that they found the exergame fun and enjoyed it (see Figure 36). We observed that some participants were also very interested in the game's story (e. g., they giggled).

The scores for negative affect (M = 0.46, SD = 0.56) and tension/annoyance (M = 0.65, SD = 0.79) show only low values. In particular, the participants did not feel bored or annoyed (see Figure 36). However, the player reported that they felt only moderately competent (M = 1.98, SD = 0.77) and challenged (M = 1.45, SD = 0.56). The low scores on competence and challenge can be attributed to the fact that most players had never done yoga before.

The score for the sensory and imaginative immersion is rather moderate (M = 1.45, SD = 0.6). We believe that the score is relatively low because players had no opportunity to explore the virtual world. Furthermore, the current game design is limited by a small number of exercises (only three yoga poses). In order to offer more variety, we would need to extend the existing approach by training new models, e. g., for additional yoga poses or different sports exercises.

The user study also pointed out some design issues. For example, although the feedback on the wall did not appear until the pose was completed, the players began to look to the side before completing the pose. Consequently, the movement of the head was often not as intended by the game. We subsequently redesigned the virtual environment and moved the feedback on the back wall so that players could see the feedback in the mirror. Another possibility would be to place the feedback in the player's field of view.

Results on Cybersickness

We calculated the scores for nausea, oculomotor, disorientation, and the total score before and after the participants played the exergame from the *Simulator Sickness Questionnaire*. The data in Table 19 highlight that the total score after the experiment (M = 10.40, SD = 11.35) is lower than before the experiment (M = 12.39, SD = 11.18). To determine if there is a statistically significant difference in post total scores, we first conducted the Anderson-Darling test to check if the data fit a normal distribution. Because the data do not follow a normal distribution, we performed a two-tailed

Table 19: The mean scores on the *Simulator Sickness Questionnaire*, measured before and after the VR-based exergame.

	Nausea	**Oculomotor**	**Disorientation**	**Total score**
	M (SD)	M (SD)	M (SD)	M (SD)
Pre	8.05 (11.39)	14.69 (12.76)	7.40 (11.17)	12.39 (11.18)
Post	9.84 (12.47)	9.00 (11.30)	7.83 (12.22)	10.40 (11.35)

Abbreviations: *Mean* (M) and *Standard Deviation* (SD).

Wilcoxon signed rank test. The results show no significant difference in post total score compared with the pre total score ($p = 0.28$, $z = 1.07$).

Conclusions and Limitations

The developed VR-based exergame has the potential to motivate players to exercise or train and practice new yoga poses. In a similar virtual environment, other full-body movements, such as Tai Chi, could be integrated into the gameplay. Nevertheless, there is potential for improvements with respect to the recognition performance. In particular, we need to improve the recognition precision as it affects the player experience, e. g., the level of immersion [161]. As already discussed in Section 5.2.3, to improve the recognition performance, we would need bigger training and testing data sets and would also need to optimize hyperparameters for individual *hidden Markov models*.

Another possibility to improve the recognition performance is to employ personal or hybrid models, as they have previously shown that they perform better than generic models [138]. However, personal and hybrid models require from each player to collect sensor data and re-train the recognition models. As collecting sensor data and re-training might be laborious for the players and would eventually discourage them from playing the game, we therefore used only generic models.

Furthermore, current trained *hidden Markov models* are also very susceptible to small mistakes. Additionally, the exergame is limited by the recognition of only three yoga poses. To expand the game, we would need to train *hidden Markov models* that are able to recognize new movements. Including other movements would also help eliminate repetitive tasks, which in turn increases motivation and variation in the game.

During the user study, we also observed that some participants needed additional instructions. In order to challenge novice and expert players, we need to adapt the difficulty level to the player's performance. In a bachelor thesis, Neitzel [158] examined different adaptation possibilities. Depending on the player's performance, Neitzel introduced an additional virtual character, i. e., a trainer who shows the correct execution of a pose and draws attention to the misplaced body parts.

Overall, the results of a user study demonstrate the capability of recognizing full-body movements in a game scenario and providing appropriate feedback. Simultaneous full-body motion reconstruction and recognition in VR-based applications are especially suitable at the present time, during the COVID-19 pandemic, e. g., to mo-

tivate patients to repeat specific exercises regularly and independently in a home environment. On the one hand, the exercises' repetition and execution can be monitored through full-body motion recognition. The system detects activity execution errors, provides appropriate feedback, and ensures that the movements are being performed correctly and as instructed. On the other hand, through full-body motion reconstruction, a therapist can remotely observe and assist the patient's movement execution. Hence, in contrast to desktop applications, VR enables therapists to observe a full-body avatar of the patient from all perspectives. In this case, regular check-ups could be done virtually.

7 SUMMARY, CONCLUSIONS, AND OUTLOOK

Concluding our work, we summarize the finding of the state-of-the-art analysis, elaborated methods for full-body motion reconstruction and recognition, and results on the validation studies. We highlight our main contributions and draw conclusions based on our obtained results. Finally, we discuss open issues and potential future work.

7.1 SUMMARY OF THE THESIS

In Chapter 1, we underlined the need for full-body motion reconstruction and recognition to improve the illusion of being in the virtual environment and provide an immersive Virtual Reality (VR) experience. In particular, we motivated the advantages of full-body motion reconstruction in multiplayer VR training simulations to improve interactions among teammates or opponents and, therefore, the training outcomes. Additionally, we motivated the advantages of full-body motion recognition to ensure that users perform activities as intended by the game and get the desired physical activity.

In Chapter 2, we conducted a systematic literature review on existing full-body tracking approaches. The analysis revealed the advantages and disadvantages of current motion capture systems and their potential for full-body motion tracking in VR. We identified that many studies focus merely on reconstruction and recognition of upper-body movements and neglect lower-body movements. Based on the requirement analysis, we proposed our overall approach for simultaneous reconstruction and recognition of full-body movements using only off-the-shelf VR devices in Chapter 3. In the following, we present the contributions of this thesis.

7.1.1 *Contributions*

Chapter 4 contains our first two contributions focusing on full-body motion reconstruction. As our *first contribution*, we explored relevant Jacobian-based inverse kinematics methods to estimate a full-body pose based upon six sensors attached to the head, hands, hips, and feet. We observed that the reconstructed full-body pose's quality depends on the inverse kinematics parameters. Therefore, we presented an optimized method that maximizes accuracy and minimizes latency. To evaluate the quality of full-body motion reconstruction, we compared the estimated full-body pose with the ground truth. We calculated the error in position and rotation of individual joints and analyzed the performance time needed to reconstruct the full-body pose as accurately as possible. Additionally, we evaluated the subjective quality of the reconstructed full-body avatars by assessing the sense of embodiment.

The analysis revealed that the *damped least squares* method outperforms other inverse kinematics methods employed in this thesis. The results showed that we reconstruct full-body movements with an average position error of 37.23 mm and a rotation error of 13.69°. We achieved a smaller error in position compared with related work, however a partially bigger error in rotation. The results of the user study further revealed that the majority of participants perceived the avatar's body as their body and could also control its movement through their movements. Additionally, they perceived only a minimal delay between the avatar's movements and their own. Hence, the proposed optimized method is capable of reconstructing full-body motions with reasonable accuracy and low latency.

Furthermore, because a high computational effort of an inverse kinematics method causes high latency and therefore provokes cybersickness, we investigated the effects of increased end-to-end latency on user experience and performance as our *second contribution*. The results of the user study showed that an end-to-end latency of 56.8 ms or higher induces significant cybersickness symptoms. The results also showed that increased end-to-end latency above 69 ms causes users to need statistically significantly more time to complete a task.

As our *third contribution*, we varied the sensor positions on the body to identify a suitable sensor subset for accurate full-body motion recognition in Chapter 5. Thereby, we employed machine learning algorithms to recognize activity execution errors to provide appropriate feedback. We analyzed different feature types and selected features that are not redundant and contribute to better overall recognition performance. A comparison of feature types showed that particularly position- and rotation-based features contributed to better overall performance, whereas velocity- and acceleration-based features did not have a substantial impact. The extensive evaluation of full-body motion recognition's quality under varying sensor positions revealed that within our database (three yoga poses), we achieve the best results with an F1 score equal to 0.91 with two sensors attached to the upper and lower body (right hand and left foot).

Finally, we implemented and validated our model on full-body motion reconstruction and recognition in two VR-based serious games in Chapter 6. We explored the added value of full-body avatars in the context of multiplayer VR training simulation for police forces. For this purpose, we elaborated, in cooperation with police forces, a training simulation that offers an opportunity to train dangerous and stressful situations in a shared immersive virtual environment. The analysis of the statistical significance and the effect size of the heart rate variability showed that a full-body avatar is significantly more threatening than an abstract representation of an assailant with head and hands. Questionnaire responses further support these results and show that around 78% of the police officers found the assailant with the full-body avatar more threatening, whereas 94% of the police officers confirmed that the full-body avatar contributed to a better training experience.

Moreover, we investigated how full-body motion reconstruction and recognition can be integrated into a VR-based exergame. To this end, we designed and implemented an exergame that aims to motivate players to train and practice new yoga poses. Due to full-body motion reconstruction, players see their avatar while looking down toward

their body and in a virtual mirror. Furthermore, due to full-body motion recognition, the system identifies the activity execution errors and provides feedback so that players can improve their movements. The evaluation results on game experience show that more than 85% of the participants found the exergame fun and enjoyed it.

7.2 CONCLUSIONS

In this thesis, we provided both objective and subjective data on the feasibility, performance, and positive impact of full-body motion reconstruction and recognition using off-the-shelf VR devices. We implemented the model on full-body motion reconstruction and recognition in two scenarios: (1) a multiplayer VR training simulation for police forces and (2) a VR-based exergame to practice yoga poses. The evaluation results of the two serious games showed the added value of full-body motion reconstruction and recognition.

In the first scenario, the results of a user study with police officers indicated that full-body avatars evoke a higher stress level and contribute to a more threatening situation. Due to full-body avatars, police officers can communicate through body language and gestures, which further increases the sense of presence and enables intuitive interaction. Such a realistic training simulation allows users to learn and develop skills and tactics more effectively, which they can then transfer to the real world.

Moreover, in the second scenario, we demonstrated how full-body motion reconstruction and recognition could be integrated into a VR-based exergame by embedding yoga poses in a narrative. We showed that two sensors attached to the upper and lower body are feasible for full-body motion recognition. Nevertheless, for appropriate feedback on individual body parts, we need to track individual limbs. We demonstrated that our approach provides appropriate feedback so that players can improve their movements and the training effects. Such a VR-based exergame is especially suitable at the present time, during the COVID-19 pandemic. For example, due to full-body motion reconstruction and recognition, a system can automatically monitor the exercises' repetitions and executions, and thus supervise players' improvements.

7.3 OUTLOOK

Our contributions to the simultaneous reconstruction and recognition of full-body movements fuel new and further research questions in the direction of highly immersive VR experiences. Our presented results prompt the inclusion of additional senses to improve immersion and enhance the training experience. The current VR training simulation for police officers includes a Head-Mounted Display (HMD) to enable first-person perspective and a headset to facilitate 360° sound. However, for a highly immersive VR experience, we must, above all, recreate the sensation of touch. Full-body haptic suits, such as Teslasuit, offer great potential to simulate physical feeling with electrical stimulation. Furthermore, we need to include real-world equipment, such as realistic weapons with recoil feedback for more realistic training.

Additionally, the VR-based exergame designed to motivate players to repeat specific exercises proposes new research challenges. For example, to enable home-based rehabilitation, we could again utilize a full-body haptic suit to assist the patient's movement execution. Furthermore, to challenge novice and expert players, we could adapt the difficulty level to the player's performance. In particular, it is necessary to employ our system in long-term interventions, e. g., to evaluate the capability of full-body motion reconstruction and recognition and explore potential improvements while playing the exergame regularly.

ACKNOWLEDGMENTS

This thesis was funded by the European Union (IWB-EFRE-Program of the State of Hessen) and the Hessian Ministry of Economics, Energy, Transport and Housing (Digitales Hessen) within the *WTT Serious Games* project and by the German Federal Ministry of Education and Research (BMBF) within the Software Campus project *TargetVR*.

BIBLIOGRAPHY

[1] Shailen Agrawal and Michiel van de Panne. "Task-Based Locomotion." In: *ACM Transactions on Graphics* 35.4 (2016). DOI: 10.1145/2897824.2925893.

[2] Rachel Albert, Anjul Patney, David Luebke, and Joohwan Kim. "Latency Requirements for Foveated Rendering in Virtual Reality." In: *ACM Transactions on Applied Perception* 14.4 (2017). DOI: 10.1145/3127589.

[3] Syed Faizan Ali, Shaheena Noor, Syed Aaqib Azmat, Ahmed Ullah Noor, and Hammad Siddiqui. "Virtual Reality as a Physical Training Assistant." In: *International Conference on Information and Communication Technologies*. ICICT 2017. Karachi, Pakistan: IEEE, 2017, pp. 191–196. DOI: 10.1109/ICICT.2017.8320189.

[4] Tim Althoff, Ryen W. White, and Eric Horvitz. "Influence of Pokémon Go on Physical Activity: Study and Implications." In: *Journal of Medical Internet Research* 18.12 (2016), e315. DOI: 10.2196/jmir.6759.

[5] Leonard A. Annetta. "The "I's" Have It: A Framework for Serious Educational Game Design." In: *Review of General Psychology* 14.2 (2010), pp. 105–113. DOI: 10.1037/a0018985.

[6] Christoph Anthes, Rubén Jesús García-Hernández, Markus Wiedemann, and Dieter Kranzlmüller. "State of the Art of Virtual Reality Technology." In: *2016 IEEE Aerospace Conference*. Big Sky, MT, USA, 2016, pp. 1–19. DOI: 10.1109/AERO.2016.7500674.

[7] Andreas Aristidou and Joan Lasenby. "FABRIK: A Fast, Iterative Solver For the Inverse Kinematics Problem." In: *Graphical Models* 73.5 (2011), pp. 243–260. DOI: 10.1016/j.gmod.2011.05.003.

[8] Andreas Aristidou, Joan Lasenby, Yiorgos Chrysanthou, and Ariel Shamir. "Inverse Kinematics Techniques in Computer Graphics: A Survey." In: *Computer Graphics Forum* 37.6 (2018), pp. 35–58. DOI: https://doi.org/10.1111/cgf.13310.

[9] Kathleen Carrie Armel and Vilayanur S. Ramachandran. "Projecting Sensations to External Objects: Evidence from Skin Conductance Response." In: *Proceedings of the Royal Society of London. Series B: Biological Sciences* 270.1523 (2003), pp. 1499–1506. DOI: 10.1098/rspb.2003.2364.

[10] Parks Associates. *Expected Virtual Reality Use Cases*. 2019. URL: https://www.parksassociates.com/blog/article/pr-04152019 (Last accessed on March 3, 2021).

[11] Audiosurf. *Audioshield*. 2016. URL: http://audio-shield.com (Last accessed on March 3, 2021).

[12] Tamas Bates, Karinne Ramirez-Amaro, Tetsunari Inamura, and Gordon Cheng. "On-Line Simultaneous Learning and Recognition of Everyday Activities from Virtual Reality Performances." In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IROS 2017. Vancouver, Canada, 2017, pp. 3510–3515. DOI: 10.1109/IROS.2017.8206193.

[13] Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains." In: *The annals of mathematical statistics* 41.1 (1970), pp. 164–171.

[14] Armin Becher, Jens Angerer, and Thomas Grauschopf. "Negative Effects of Network Latencies in Immersive Collaborative Virtual Environments." In: *Virtual Reality* 24.3 (2020), pp. 369–383. DOI: 10.1007/s10055-019-00395-9.

[15] Claudia Beleites, Ute Neugebauer, Thomas Bocklitz, Christoph Krafft, and Jürgen Popp. "Sample Size Planning for Classification Models." In: *Analytica Chimica Acta* 760 (2013), pp. 25–33. DOI: 10.1016/j.aca.2012.11.007.

[16] James Bergstra and Yoshua Bengio. "Random Search for Hyper-Parameter Optimization." In: *The Journal of Machine Learning Research* 13 (2012), pp. 281–305.

[17] Gary G. Berntson, Karen S. Quigley, Greg J. Norman, and David L. Lozano. "Cardiovascular Psychophysiology." In: *Handbook of psychophysiology*. Cambridge University Press, 2017, pp. 183–216.

[18] Johanna Bertram, Johannes Moskaliuk, and Ulrike Cress. "Virtual training: Making reality work?" In: *Computers in Human Behavior* 43 (2015), pp. 284–292. DOI: https://doi.org/10.1016/j.chb.2014.10.032.

[19] BeswinVR. *VR Gun Controller for HTC Vive*. 2021. URL: https://www.beswinvr.com/product-review/ (Last accessed on March 3, 2021).

[20] Inc. BigBox VR. *Population: ONE*. 2020. URL: http://www.populationonevr.com (Last accessed on March 3, 2021).

[21] Timo Bittner, Christian Eller, Marcus Dombois, and Uwe Rüppel. "BIM-unterstützte Erstellung interaktiver, immersiver VR-Szenarien am Beispiel der Brandbekämpfung." In: *Bauphysik* 40.6 (2018), pp. 410–415. DOI: 10.1002/bapi.201800028.

[22] Willem Bles, Jelte E. Bos, Bernd de Graaf, Eric Groen, and Alexander H. Wertheim. "Motion Sickness: Only One Provocative Conflict?" In: *Brain Research Bulletin* 47.5 (1998), pp. 481–487. DOI: 10.1016/S0361-9230(98)00115-4.

[23] Bobby Bodenheimer and Qiang Fu. "The Effect of Avatar Model in Stepping off a Ledge in an Immersive Virtual Environment." In: *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception*. SAP 2015. Tübingen, Germany, 2015, pp. 115–118. DOI: 10.1145/2804408.2804426.

[24] Felix Born, Sophie Abramowski, and Maic Masuch. "Exergaming in VR: The Impact of Immersive Embodiment on Motivation, Performance, and Perceived Exertion." In: *11th International Conference on Virtual Worlds and Games for Serious Applications*. VS-Games 2019. Vienna, Austria: IEEE, 2019, pp. 1–8. DOI: 10.1109/VS-Games.2019.8864579.

[25] Adrián Borrego, Jorge Latorre, Mariano Alcañiz, and Roberto Llorens. "Comparison of Oculus Rift and HTC Vive: Feasibility for Virtual Reality-Based Exploration, Navigation, Exergaming, and Rehabilitation." In: *Games for Health Journal* 7.3 (2018), pp. 151–156. DOI: 10.1089/g4h.2017.0114.

[26] Jelte E. Bos. "Nuancing the Relationship Between Motion Sickness and Postural Stability." In: *Displays* 32.4 (2011). Visual Image Safety, pp. 189–193. DOI: 10.1016/j.displa.2010.09.005.

[27] Jelte E. Bos, Willem Bles, and Eric L. Groen. "A Theory on Visually Induced Motion Sickness." In: *Displays* 29.2 (2008). Health and Safety Aspects of Visual Displays, pp. 47–57. DOI: 10.1016/j.displa.2007.09.002.

[28] Henrik Boström. "Predicate Invention and Learning From Positive Examples Only." In: *Machine Learning: ECML-98*. Vol. 1398. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence). Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 226–237. DOI: 10.1007/BFb0026693.

[29] Jean Botev and Steffen Rothkugel. "High-Precision Gestural Input for Immersive Large-Scale Distributed Virtual Environments." In: *Proceedings of the 9th Workshop on Massively Multiuser Virtual Environments*. MMVE 2017. Taipei, Taiwan, 2017, pp. 7–11. DOI: 10.1145/3083207.3083209.

[30] Matthew Botvinick and Jonathan Cohen. "Rubber Hands 'Feel' Touch that Eyes See." In: *Nature* 391 (1998), p. 756. DOI: 10.1038/35784.

[31] Matthew Brand, Nuria Oliver, and Alex Pentland. "Coupled Hidden Markov Models for Complex Action Recognition." In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. San Juan, PR, USA: IEEE, 1997, pp. 994–999. DOI: 10.1109/CVPR.1997.609450.

[32] Anne-Marie Brouwer, Mark A. Neerincx, Victor Kallen, Leslie van der Leer, and Michiel ten Brinke. "EEG Alpha Asymmetry, Heart Rate Variability and Cortisol in Response to Virtual Reality Induced Stress." In: *Journal of Cybertherapy & Rehabilitation* 4.1 (2011), pp. 21–34.

[33] Susan Bruck and Paul A. Watters. "The Factor Structure of Cybersickness." In: *Displays* 32.4 (2011). Visual Image Safety, pp. 153–158. DOI: 10.1016/j.displa.2011.07.002.

[34] Samuel R. Buss. *Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Samped Least Squares Methods*. Tech. rep. University of California, 2009.

[35] Samuel R. Buss. *Source Code for Inverse Kinematics Solvers*. 2014. URL: https://www.math.ucsd.edu/~sbuss/ResearchWeb/ikmethods/ (Last accessed on March 3, 2021).

[36] Samuel R. Buss and Jin-Su Kim. "Selectively Damped Least Squares for Inverse Kinematics." In: *Journal of Graphics Tools* 10.3 (2005), pp. 37–49. DOI: 10.1080/2151237X.2005.10129202.

[37] Arndt Büssing, Andreas Michalsen, Sat Bir S. Khalsa, Shirley Telles, and Karen J. Sherman. "Effects of Yoga on Mental and Physical Health: A Short Summary of Reviews." In: *Evidence-Based Complementary and Alternative Medicine* 2012 (2012). DOI: 10.1155/2012/165410.

[38] Polona Caserman. *Real-Time Body Reconstruction and Recognition in Virtual Reality using HTC Vive Trackers and Controllers*. 2017. URL: https://github.com/CatCuddler/BodyTracking (Last accessed on March 3, 2021).

[39] Polona Caserman, Philipp Achenbach, and Stefan Göbel. "Analysis of Inverse Kinematics Solutions for Full-Body Reconstruction in Virtual Reality." In: *IEEE 7th International Conference on Serious Games and Applications for Health*. SeGah 2019. Kyoto, Japan: IEEE, 2019, pp. 1–8. DOI: 10.1109/SeGAH.2019.8882429.

[40] Polona Caserman, Miriam Cornel, Michelle Dieter, and Stefan Göbel. "A Concept of a Training Environment for Police Using VR Game Technology." In: *Serious Games*. Vol. 11243. JCSG 2018. Lecture Notes in Computer Science. Darmstadt, Germany: Springer International Publishing, 2018, pp. 175–181. DOI: 10.1007/978-3-030-02762-9_18.

[41] Polona Caserman, Augusto Garcia-Agundez, Alvar Gámes Zerban, and Stefan Göbel. "Cybersickness in Current-Generation Virtual Reality Head-Mounted Displays: Systematic Review and Outlook." In: *Virtual Reality* (2021), pp. 1–19. DOI: 10.1007/s10055-021-00513-6.

[42] Polona Caserman, Augusto Garcia-Agundez, and Stefan Göbel. "A Survey of Full-Body Motion Reconstruction in Immersive Virtual Reality Applications." In: *IEEE Transactions on Visualization and Computer Graphics* 26.10 (2020), pp. 3089–3108. DOI: 10.1109/TVCG.2019.2912607.

[43] Polona Caserman, Augusto Garcia-Agundez, Robert Konrad, Stefan Göbel, and Ralf Steinmetz. "Real-time Body Tracking in Virtual Reality Using a Vive Tracker." In: *Virtual Reality* 23.2 (2018), pp. 155–168. DOI: 10.1007/s10055-018-0374-z.

[44] Polona Caserman and Stefan Göbel. "Become a Scrum Master: Immersive Virtual Reality Training to Learn Scrum Framework." In: *Serious Games*. Vol. 12434. JCSG 2020. Lecture Notes in Computer Science. Stoke-on-Trent, UK: Springer International Publishing, 2020, pp. 34–48. DOI: 10.1007/978-3-030-61814-8_3.

[45] Polona Caserman, Katrin Hoffmann, Philipp Müller, Marcel Schaub, Katharina Straßburg, Josef Wiemeyer, Regina Bruder, and Stefan Göbel. "Quality Criteria for Serious Games: Serious Part, Game Part, and Balance." In: *JMIR Serious Games* 8.3 (2020), e19037. DOI: 10.2196/19037.

[46] Polona Caserman, Patrick Krabbe, Janis Wojtusch, and Oskar von Stryk. "Real-Time Step Detection Using the Integrated Sensors of a Head-Mounted Display." In: *IEEE International Conference on Systems, Man, and Cybernetics*. SMC 2016. Budapest, Hungary, 2016, pp. 3510–3515. DOI: 10.1109/SMC.2016.7844777.

[47] Polona Caserman, Shule Liu, and Stefan Göbel. "Full-Body Motion Recognition in Immersive Virtual Reality-based Exergame." In: *IEEE Transactions on Games* (2021), pp. 1–10. DOI: 10.1109/TG.2021.3064749.

[48] Polona Caserman, Michelle Martinussen, and Stefan Göbel. "Effects of End-to-end Latency on User Experience and Performance in Immersive Virtual Reality Applications." In: *Entertainment Computing and Serious Games*. Vol. 11863. ICEC-JCSG 2019. Lecture Notes in Computer Science. Arequipa, Peru: Springer International Publishing, 2019, pp. 57–69. DOI: 10.1007/978-3-030-34644-7_5.

[49] Polona Caserman, Philip Schmidt, Thorsten Göbel, Jonas Zinnäcker, André Kecke, and Stefan Göbel. *Impact of Full-Body Avatars in Immersive Multiplayer Virtual Reality Training for Police Forces*. Submitted for publication in IEEE Transactions on Games.

[50] Polona Caserman, Thomas Tregel, Marco Fendrich, Moritz Kolvenbach, Markus Stabel, and Stefan Göbel. "Recognition of Full-Body Movements in VR-Based Exergames Using Hidden Markov Models." In: *Serious Games*. Vol. 11243. JCSG 2018. Lecture Notes in Computer Science. Darmstadt, Germany: Springer International Publishing, 2018, pp. 191–203. DOI: 10.1007/978-3-030-02762-9_20.

[51] Polona Caserman, Hongtao Zhang, Jonnas Zinnäcker, and Stefan Göbel. "Development of a Directed Teleport Function for Immersive Training in Virtual Reality." In: *11th International Conference on Virtual Worlds and Games for Serious Applications*. VS-Games 2019. Vienna, Austria: IEEE, 2019, pp. 1–8. DOI: 10.1109/VS-Games.2019.8864599.

[52] Sylvain Chagué and Caecilia Charbonnier. "Real Virtuality: A Multi-User Immersive Platform Connecting Real and Virtual Worlds." In: *Proceedings of the 2016 Virtual Reality International Conference*. VRIC 2016. Laval, France, 2016, pp. 1–3. DOI: 10.1145/2927929.2927945.

[53] Jacky C. P. Chan, Howard Leung, Jeff K. T. Tang, and Taku Komura. "A Virtual Reality Dance Training System using Motion Capture Technology." In: *IEEE Transactions on Learning Technologies* 4.2 (2011), pp. 187–195. DOI: 10.1109/TLT.2010.27.

[54] Donglin Chen, Hao Liu, and Zhigang Ren. "Application of Wearable Device HTC VIVE in Upper Limb Rehabilitation Training." In: *2nd IEEE Advanced Information Management,Communicates,Electronic and Automation Control Conference (IMCEC)*. IMCEC 2018. Xi'an, China, 2018, pp. 1460–1464. DOI: 10.1109/IMCEC.2018.8469540.

[55] Jacob Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Academic press, 1977.

[56] Maria Cornacchia, Koray Ozcan, Yu Zheng, and Senem Velipasalar. "A Survey on Activity Detection and Classification using Wearable Sensors." In: *IEEE Sensors Journal* 17.2 (2017), pp. 386–403. DOI: 10.1109/JSEN.2016.2628346.

[57] HTC Corporation. *Vive Cosmos*. 2021. URL: https://www.vive.com/eu/product/vive-cosmos/overview/ (Last accessed on March 3, 2021).

[58] HTC Corporation. *Vive Pro Full Kit*. 2021. URL: https://www.vive.com/eu/product/vive-pro-full-kit/ (Last accessed on March 3, 2021).

[59] HTC Corporation. *Vive Tracker*. 2021. URL: https://www.vive.com/eu/accessory/vive-tracker/ (Last accessed on March 3, 2021).

[60] Valve Corporation. *Half-Life: Alyx*. 2020. URL: https://www.half-life.com/de/alyx/ (Last accessed on March 3, 2021).

[61] Valve Corporation. *The Year's Top VR-Exclusive Experiences as Measured by Gross Revenue*. 2021. URL: https://store.steampowered.com/sale/BestOf2020 (Last accessed on March 3, 2021).

[62] John J. Craig. *Introduction to Robotics: Mechanics and Control*. 3rd ed. Pearson Education International, 2005.

[63] Simon Davis, Keith Nesbitt, and Eugene Nalivaiko. "A Systematic Review of Cybersickness." In: *Proceedings of the 2014 Conference on Interactive Entertainment*. IE2014. Newcastle, NSW, Australia, 2014, pp. 1–9. DOI: 10.1145/2677758.2677780.

[64] Mark S. Dennison, A. Zachary Wisti, and Michael D'Zmura. "Use of Physiological Signals to Predict Cybersickness." In: *Displays* 44 (2016). Contains Special Issue Articles – Proceedings of the 4th Symposium on Liquid Crystal Photonics (SLCP 2015), pp. 42–52. DOI: 10.1016/j.displa.2016.07.002.

[65] Heather Desurvire and Charlotte Wiberg. "Game Usability Heuristics (PLAY) for Evaluating and Designing Better Games: The Next Iteration." In: *Online Communities and Social Computing*. Vol. 5621. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 557–566. DOI: 10.1007/978-3-642-02774-1_60.

[66] Darpan Dhawan, Michael Barlow, and Erandi Lakshika. "Prosthetic Rehabilitation Training in Virtual Reality." In: *IEEE 7th International Conference on Serious Games and Applications for Health*. SeGAH 2019. Kyoto, Japan, 2019, pp. 1–8. DOI: 10.1109/SeGAH.2019.8882455.

[67] DHL. *Augmented & Virtual Reality*. 2020. URL: https://www.dhl.com/global-en/home/insights-and-innovation/thought-leadership/trend-reports/augmented-reality.html (Last accessed on March 3, 2021).

[68] Tim Dittmar, Claudia Krull, and Graham Horton. "A New Approach for Touch Gesture Recognition: Conversive Hidden non-Markovian Models." In: *Journal of Computational Science* 10 (2015), pp. 66–76. DOI: 10.1016/j.jocs.2015.03.002.

[69] Paul DiZio and James R. Lackner. *Motion Sickness Side Effects and Aftereffects of Immersive Virtual Environments Created with Helmet-Mounted Visual Displays*. Tech. rep. Brandeis University, Waltham MA, Ashton Graybiel Spatial Orientation Laboratory, 2000.

[70] Ralf Dörner, Stefan Göbel, Wolfgang Effelsberg, and Josef Wiemeyer. *Serious Games: Foundations, Concepts and Practice*. Springer, 2016. DOI: 10.1007/978-3-319-40612-1.

[71] Sheldon M. Ebenholtz, Malcolm M. Cohen, and Barry J. Linder. "The Possible Role of Nystagmus in Motion Sickness: A Hypothesis." In: *Aviation, space, and environmental medicine* 65.11 (1994), pp. 1032–1035. URL: https://europepmc.org/article/med/7840743.

[72] Sean R. Eddy, Graeme Mitchison, and Richard Durbin. "Maximum Discrimination Hidden Markov Models of Sequence Consensus." In: *Journal of Computational Biology* 2.1 (1995), pp. 9–23. DOI: 10.1089/cmb.1995.2.9.

[73] H. Henrik Ehrsson, Katja Wiech, Nikolaus Weiskopf, Raymond J. Dolan, and Richard E. Passingham. "Threatening a Rubber Hand That You Feel Is Yours Elicits a Cortical Anxiety Response." In: *Proceedings of the National Academy of Sciences* 104.23 (2007), pp. 9828–9833. DOI: 10.1073/pnas.0610011104.

[74] Christian Eller. "Bewertung und Nutzung immersiver Methoden der Digitalen Realität für Bau- & Umweltingenieuranwendungen innerhalb der Lehre und Praxis." PhD thesis. Darmstadt, DE: Technical University of Darmstadt, 2020. DOI: 10.25534/tuprints-00012788.

[75] Christian Eller. *Pan-Faktor*. 2020. URL: https://www.pan-factor.info (Last accessed on March 3, 2021).

[76] Christian Eller, Timo Bittner, Marcus Dombois, and Uwe Rüppel. "Collaborative Immersive Planning and Training Scenarios in VR." In: *Advanced Computing Strategies for Engineering*. Vol. 10863. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018, pp. 164–185. DOI: 10.1007/978-3-319-91635-4_9.

[77] Red Storm Entertainment. *Star Trek: Bridge Crew*. 2017. URL: https://www.ubisoft.com/en-us/game/star-trek/bridge-crew (Last accessed on March 3, 2021).

[78] Sony Interactive Entertainment Europe. *PlayStation Store: Top downloads of 2020*. 2021. URL: https://blog.playstation.com/2021/01/13/playstation-store-top-downloads-of-2020/ (Last accessed on March 3, 2021).

[79] Marc Fabri, David J. Moore, and Dave J. Hobbs. "The Emotional Avatar: Non-verbal Communication Between Inhabitants of Collaborative Virtual Environments." In: *Gesture-Based Communication in Human-Computer Interaction*. Vol. 1739. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 269–273. DOI: 10.1007/3-540-46616-9_24.

[80] Shih-Hau Fang, Yu-Xaing Fei, Zhezhuang Xu, and Yu Tsao. "Learning Transportation Modes From Smartphone Sensors Based on Deep Neural Network." In: *IEEE Sensors Journal* 17.18 (2017), pp. 6111–6118. DOI: 10.1109/JSEN.2017.2737825.

[81] Nua Faric, Henry W. W. Potts, Adrian Hon, Lee Smith, Katie Newby, Andrew Steptoe, and Abi Fisher. "What Players of Virtual Reality Exercise Games Want: Thematic Analysis of Web-Based Reviews." In: *Journal of Medical Internet Research* 21.9 (2019), e13833. DOI: 10.2196/13833.

[82] Franz Faul, Edgar Erdfelder, Axel Buchner, and Albert-Georg Lang. "Statistical Power Analyses using G* Power 3.1: Tests for Correlation and Regression Analyses." In: *Behavior research methods* 41.4 (2009), pp. 1149–1160. DOI: 10.3758/BRM.41.4.1149.

[83] Andrea Ferracani, Daniele Pezzatini, Jacopo Bianchini, Gianmarco Biscini, and Alberto Del Bimbo. "Locomotion by Natural Gestures for Immersive Virtual Environments." In: *Proceedings of the 1st International Workshop on Multimedia Alternate Realities*. AltMM 2016. Amsterdam, The Netherlands, 2016, pp. 21–24. DOI: 10.1145/2983298.2983307.

[84] Samantha Finkelstein, Andrea Nickel, Zachary Lipps, Tiffany Barnes, Zachary Wartell, and Evan A. Suma. "Astrojumper: Motivating Exercise with an Immersive Virtual Reality Exergame." In: *Presence: Teleoperators and Virtual Environments* 20.1 (2011), pp. 78–92. DOI: 10.1162/pres_a_00036.

[85] Roger Fletcher. *Practical Methods of Optimization*. 2nd ed. John Wiley & Sons, 1987.

[86] Jonas Freiknecht and Wolfgang Effelsberg. "Procedural Generation of Multistory Buildings With Interior." In: *IEEE Transactions on Games* 12.3 (2020), pp. 323–336. DOI: 10.1109/TG.2019.2957733.

[87] Rebecca Fribourg, Ferran Argelaguet, Anatole Lécuyer, and Ludovic Hoyet. "Avatar and Sense of Embodiment: Studying the Relative Preference Between Appearance, Control and Point of View." In: *IEEE Transactions on Visualization and Computer Graphics* 26.5 (2020), pp. 2062–2072. DOI: 10.1109/TVCG.2020.2973077.

[88] Sebastian Friston and Anthony Steed. "Measuring Latency in Virtual Environments." In: *IEEE Transactions on Visualization and Computer Graphics* 20.4 (2014), pp. 616–625. DOI: 10.1109/TVCG.2014.30.

[89] Philippe Fuchs. *Virtual Reality Headsets - A Theoretical and Pragmatic Approach*. CRC Press, 2017.

[90] Henrique G. Debarba, Sidney Bovet, Roy Salomon, Olaf Blanke, Bruno Herbelin, and Ronan Boulic. "Characterizing First and Third Person Viewpoints and Their Alternation for Embodied Interaction in Virtual Reality." In: *PLOS ONE* 12.12 (Dec. 2017), pp. 1–19. DOI: 10.1371/journal.pone.0190109.

[91] Brook Galna, Gillian Barry, Dan Jackson, Dadirayi Mhiripiri, Patrick Olivier, and Lynn Rochester. "Accuracy of the Microsoft Kinect Sensor for Measuring Movement in People with Parkinson's Disease." In: *Gait & Posture* 39.4 (2014), pp. 1062–1068. DOI: 10.1016/j.gaitpost.2014.01.008.

[92] Beat Games. *Beat Saber*. 2018. URL: https://www.beatsaber.com (Last accessed on March 3, 2021).

[93] Augusto Garcia-Agundez. "Clinical Decision Support Systems with Game-Based Environments, Monitoring Symptoms of Parkinson's Disease with Exergames." PhD thesis. Darmstadt, DE: Technical University of Darmstadt, 2020. DOI: 10.25534/tuprints-00014137.

[94] Augusto Garcia-Agundez, Christian Reuter, Hagen Becker, Robert Konrad, Polona Caserman, André Miede, and Stefan Göbel. "Development of a Classifier to Determine Factors Causing Cybersickness in Virtual Reality Environments." In: *Games for Health Journal* 8.6 (2019), pp. 439–444. DOI: 10.1089/g4h.2019.0045.

[95] Augusto Garcia-Agundez, Christian Reuter, Polona Caserman, Robert Konrad, and Stefan Göbel. "Identifying Cybersickness Through Heart Rate Variability Alterations." In: *International Journal of Virtual Reality* 19.1 (2019), pp. 1–10. DOI: 10.20870/IJVR.2019.19.1.2907.

[96] Augusto Garcia-Agundez, Aiko Westmeier, Polona Caserman, Robert Konrad, and Stefan Göbel. "An Evaluation of Extrapolation and Filtering Techniques in Head Tracking for Virtual Environments to Reduce Cybersickness." In: *Serious Games*. Vol. 10622. JCSG 2017. Lecture Notes in Computer Science. Valencia, Spain: Springer International Publishing, 2017, pp. 203–211. DOI: 10.1007/978-3-319-70111-0_19.

[97] Stefan Göbel. "Autorenumgebung für Serious Games - StoryTec: Eine Autorenumgebung und narrative Objekte für personalisierte Serious Games." Habilitation. Darmstadt, DE: Technical University of Darmstadt, 2017. URL: http://tuprints.ulb.tu-darmstadt.de/6941/.

[98] John F. Golding. "Motion Sickness Susceptibility Questionnaire Revised and Its Relationship to Other Forms of Sickness." In: *Brain Research Bulletin* 47.5 (1998), pp. 507–516. DOI: 10.1016/S0361-9230(98)00091-4.

[99] Geoffrey Gorisse, Olivier Christmann, Etienne Armand Amato, and Simon Richir. "First- and Third-Person Perspectives in Immersive Virtual Environments: Presence and Performance Analysis of Embodied Users." In: *Frontiers in Robotics and AI* 4 (2017), p. 33. DOI: 10.3389/frobt.2017.00033.

[100] Cyril Goutte and Eric Gaussier. "A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation." In: *Advances in Information Retrieval*. Vol. 3408. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 345–359. DOI: 10.1007/978-3-540-31865-1_25.

[101] Stefan Greuter and David J. Roberts. "SpaceWalk: Movement and Interaction in Virtual Space with Commodity Hardware." In: *Proceedings of the 2014 Conference on Interactive Entertainment*. IE20 2014. Newcastle, NSW, Australia, 2014, pp. 1–7. DOI: 10.1145/2677758.2677781.

[102] Jess Grey. *Review: Oculus Rift S*. 2020. URL: https://www.wired.com/review/oculus-rift-s-review/ (Last accessed on March 3, 2021).

[103] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton. "Enhanced Computer Vision with Microsoft Kinect Sensor: A Review." In: *IEEE Transactions on Cybernetics* 43.5 (2013), pp. 1318–1334. DOI: 10.1109/TCYB.2013.2265378.

[104] Ping-Hsuan Han, Da-Yuan Huang, Hsin-Ruey Tsai, Po-Chang Chen, Chen-Hsin Hsieh, Kuan-Ying Lu, De-Nian Yang, and Yi-Ping Hung. "Moving Around in Virtual Space with Spider Silk." In: *ACM SIGGRAPH 2015 Emerging Technologies*. SIGGRAPH 2015. Los Angeles, California, 2015. DOI: 10.1145/2782782.2792483.

[105] Thomas Helten, Meinard Müller, 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 (ICCV)*. Sydney, NSW, Australia: IEEE, 2013, pp. 1105–1112. DOI: 10.1109/ICCV.2013.141.

[106] Thomas Hilfert and Markus König. "Low-Cost Virtual Reality Environment for Engineering and Construction." In: *Visualization in Engineering* 4.1 (2016). DOI: 10.1186/s40327-015-0031-5.

[107] Thuong N. Hoang, Martin Reinoso, Frank Vetere, and Egemen Tanin. "Onebody: Remote Posture Guidance System using First Person View in Virtual Environment." In: *Proceedings of the 9th Nordic Conference on Human-Computer Interaction*. NordiCHI 2016. Gothenburg, Sweden, 2016, pp. 1–10. DOI: 10.1145/2971485.2971521.

[108] Wijnand A. IJsselsteijn, Yvonne A. W. De Kort, and Karolien Poels. "The Game Experience Questionnaire." In: *Eindhoven: Technische Universiteit Eindhoven* (2013).

[109] Jan-Keno Janssen. *Oculus Quest: High-End-VR ohne PC oder Konsole*. 2019. URL: https://www.heise.de/tests/Oculus-Quest-High-End-VR-ohne-PC-oder-Konsole-4430044.html (Last accessed on March 3, 2021).

[110] Fan Jiang, Xubo Yang, and Lele Feng. "Real-Time Full-Body Motion Reconstruction and Recognition for Off-the-Shelf VR Devices." In: *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry - Volume 1*. VRCAI 2016. Zhuhai, China, 2016, pp. 309–318. DOI: 10.1145/3013971.3013987.

[111] Marcelo Kallmann. "Analytical Inverse Kinematics with Body Posture Control." In: *Computer Animation and Virtual Worlds* 19.2 (2008), pp. 79–91. DOI: 10.1002/cav.176.

[112] Svebor Karaman, Jenny Benois-Pineau, Vladislavs Dovgalecs, Rémi Mégret, Julien Pinquier, Régine André-Obrecht, Yann Gaëstel, and Jean-François Dartigues. "Hierarchical Hidden Markov Model in Detecting Activities of Daily Living in Wearable Videos for Studies of Dementia." In: *Multimedia Tools and Applications* 69 (2014), pp. 743–771. DOI: 10.1007/s11042-012-1117-x.

[113] Robert S. Kennedy, Norman E. Lane, Kevin S. Berbaum, and Michael G. Lilienthal. "Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness." In: *The International Journal of Aviation Psychology* 3.3 (1993), pp. 203–220. DOI: 10.1207/s15327108ijap0303_3.

[114] Ben Kenwright. "Inverse Kinematics – Cyclic Coordinate Descent (CCD)." In: *Journal of Graphics Tools* 16.4 (2012), pp. 177–217. DOI: 10.1080/2165347X.2013.823362.

[115] Ben Kenwright. "Real-Time Character Inverse Kinematics using the Gauss-Seidel Iterative Approximation Method." In: *Proceedings of the 4th International Conference on Creative Content Technologies*. Vol. 4. Nice, France, 2012, pp. 63–68.

[116] Eamonn J. Keogh and Michael J. Pazzani. "Derivative Dynamic Time Warping." In: *Proceedings of the 2001 SIAM international conference on data mining*. 2001, pp. 1–11. DOI: 10.1137/1.9781611972719.1.

[117] Behrang Keshavarz and Heiko Hecht. "Validating an Efficient Method to Quantify Motion Sickness." In: *Human Factors* 53.4 (2011), pp. 415–426. DOI: 10.1177/0018720811403736.

[118] Konstantina Kilteni, Raphaela Groten, and Mel Slater. "The Sense of Embodiment in Virtual Reality." In: *Presence: Teleoperators and Virtual Environments* 21.4 (2012), pp. 373–387. DOI: 10.1162/PRES_a_00124.

[119] Eunju Kim, Sumi Helal, and Diane Cook. "Human Activity Recognition and Pattern Discovery." In: *IEEE Pervasive Computing* 9.1 (2010), pp. 48–53. DOI: 10.1109/MPRV.2010.7.

[120] Hye-Geum Kim, Eun-Jin Cheon, Dai-Seg Bai, Young Hwan Lee, and Bon-Hoon Koo. "Stress and Heart Rate Variability: A Meta-Analysis and Review of the Literature." In: *Psychiatry investigation* 15.3 (2018), pp. 235–345. DOI: 10.30773/pi.2017.08.17.

[121] Manon Kok, Jeroen D. Hol, and Thomas B. Schön. "Using Inertial Sensors for Position and Orientation Estimation." In: *Foundations and Trends in Signal Processing* 11.1-2 (2017), pp. 1–153. DOI: 10.1561/2000000094.

[122] Ryota Kondo, Maki Sugimoto, Kouta Minamizawa, Takayuki Hoshi, Masahiko Inami, and Michiteru Kitazaki. "Illusory Body Ownership of an Invisible Body Interpolated Between Virtual Hands and Feet via Visual-Motor Synchronicity." In: *Scientific reports* 8.7541 (2018), pp. 1–8. DOI: 10.1038/s41598-018-25951-2.

[123] Robert Konrad. *Kinc*. 2013. URL: https://github.com/Kode/Kinc (Last accessed on March 3, 2021).

[124] Oliver Kreylos. *Lighthouse Tracking Examined*. 2016. URL: http://doc-ok.org/?p=1478 (Last accessed on March 3, 2021).

[125] Przemyslaw Krompiec and Kyoungju Park. "Enhanced Player Interaction using Motion Controllers for First-Person Shooting Games in Virtual Reality." In: *IEEE Access* 7 (2019), pp. 124548–124557. DOI: 10.1109/ACCESS.2019.2937937.

[126] Fedwa Laamarti, Mohamad Eid, and Abdulmotaleb El Saddik. "An Overview of Serious Games." In: *International Journal of Computer Games Technology* 2014 (2014). DOI: 10.1155/2014/358152.

[127] Belinda Lange, Evan A. Suma, Brad Newman, Thai Phan, Chien-Yen Chang, Albert Rizzo, and Mark Bolas. "Leveraging Unencumbered Full Body Control of Animated Virtual Characters for Game-Based Rehabilitation." In: *Virtual and Mixed Reality - Systems and Applications*. Vol. 6774. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 243–252. DOI: 10.1007/978-3-642-22024-1_27.

[128] Lorraine Lanningham-Foster, Teresa B. Jensen, Randal C. Foster, Aoife B. Redmond, Brian A. Walker, Dieter Heinz, and James A. Levine. "Energy Expenditure of Sedentary Screen Time Compared With Active Screen Time for Children." In: *Pediatrics* 118.6 (2006), e1831–e1835. DOI: 10.1542/peds.2006-1087.

[129] Óscar D. Lara and Miguel A. Labrador. "A Survey on Human Activity Recognition using Wearable Sensors." In: *IEEE Communications Surveys (Tutorials)* 15.3 (2013), pp. 1192–1209. DOI: 10.1109/SURV.2012.110112.00192.

[130] 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: *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. VRST 2017. Gothenburg, Sweden, 2017, pp. 1–10. DOI: 10.1145/3139131.3139156.

[131] Joseph J. LaViola. "A Discussion of Cybersickness in Virtual Environments." In: *SIGCHI Bulletin* 32.1 (2000), pp. 47–56. DOI: 10.1145/333329.333344.

[132] Effie L.-C. Law, Florian Brühlmann, and Elisa D. Mekler. "Systematic Review and Validation of the Game Experience Questionnaire (GEQ) - Implications for Citation and Reporting Practice." In: *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*. CHI PLAY '18. Melbourne, VIC, Australia: Association for Computing Machinery, 2018, pp. 257–270. DOI: 10.1145/3242671.3242683.

[133] Lee, Juyoung and Ahn, Sang Chul and Hwang, Jae-In. "A Walking-in-Place Method for Virtual Reality using Position and Orientation Tracking." In: *Sensors* 18.9 (2018), pp. 1–19. DOI: https://doi.org/10.3390/s18092832.

[134] Paolo Leoncini, Bogdan Sikorski, Vincenzo Baraniello, Francesco Martone, Carlo Luongo, and Mariano Guida. "Multiple NUI Device Approach to Full Body Tracking for Collaborative Virtual Environments." In: *Augmented Reality, Virtual Reality, and Computer Graphics*. Vol. 10324. Lecture Notes in Computer

Science. Cham: Springer International Publishing, 2017, pp. 131–147. DOI: 10.1007/978-3-319-60922-5_10.

[135] Patrick Lieser, Alaa Alhamoud, Hosam Nima, Björn Richerzhagen, Sanja Huhle, Doreen Böhnstedt, and Ralf Steinmetz. "Situation Detection Based on Activity Recognition in Disaster Scenarios." In: *Proceedings of the International Conference on Information Systems for Crisis Response and Management (ISCRAM)*. 2018.

[136] Li Liu, Yuxin Peng, Shu Wang, Ming Liu, and Zigang Huang. "Complex Activity Recognition using Time Series Pattern Dictionary Learned From Ubiquitous Sensors." In: *Information Sciences* 340-341 (2016), pp. 41–57. DOI: 10.1016/j.ins.2016.01.020.

[137] Jeffrey W. Lockhart and Gary M. Weiss. "Limitations with Activity Recognition Methodology & Data Sets." In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. UbiComp '14 Adjunct. Seattle, Washington: Association for Computing Machinery, 2014, pp. 747–756. DOI: 10.1145/2638728.2641306.

[138] Jeffrey W. Lockhart and Gary M. Weiss. "The Benefits of Personalized Smartphone-Based Activity Recognition Models." In: *Proceedings of the 2014 SIAM international conference on data mining*. SIAM. 2014, pp. 614–622. DOI: 10.1137/1.9781611973440.71.

[139] Sébastien Lozé. *Efficient Police Virtual Training Environment in VR by V-Armed*. 2019. URL: https://www.unrealengine.com/en-US/spotlights/efficient-police-virtual-training-environment-in-vr-by-v-armed (Last accessed on March 3, 2021).

[140] Noitom Ltd. *Hi5 VR GLOVE*. 2021. URL: https://hi5vrglove.com (Last accessed on March 3, 2021).

[141] RUST LTD. *Hot Dogs, Horseshoes & Hand Grenades*. 2016. URL: http://h3vr.com (Last accessed on March 3, 2021).

[142] Astrid J. A. Lubeck, Jelte E. Bos, and John F. Stins. "Motion in Images Is Essential to Cause Motion Sickness Symptoms, but Not to Increase Postural Sway." In: *Displays* 38 (2015), pp. 55–61. DOI: 10.1016/j.displa.2015.03.001.

[143] Kathryn Luttgens, Helga Deutsch, and Nancy Hamilton. *Kinesiology: Scientific Basis of Human Motion*. Brown & Benchmark, 1992.

[144] Charles Malleson, Andrew Gilbert, Matthew Trumble, John Collomosse, Adrian Hilton, and Marco Volino. "Real-Time Full-Body Motion Capture from Video and IMUs." In: *International Conference on 3D Vision*. 3DV 2017. Qingdao, China, 2017, pp. 449–457. DOI: 10.1109/3DV.2017.00058.

[145] Hiroshi Mamitsuka. "A Learning Method of Hidden Markov Models for Sequence Discrimination." In: *Journal of Computational Biology* 3.3 (1996), pp. 361–373. DOI: 10.1089/cmb.1996.3.361.

[146] Jenny Margarito, Rim Helaoui, Anna M. Bianchi, Francesco Sartor, and Alberto G. Bonomi. "User-Independent Recognition of Sports Activities From a Single Wrist-Worn Accelerometer: A Template-Matching-Based Approach." In: *IEEE Transactions on Biomedical Engineering* 63.4 (2016), pp. 788–796. DOI: 10.1109/TBME.2015.2471094.

[147] Derek W. Marks, Lauren Rispen, and Gabriel Calara. "Greater Physiological Responses While Playing XBox Kinect Compared to Nintendo Wii." In: *International Journal of Exercise Science* 8.2 (2015), pp. 164–173.

[148] R. Timothy Marler and Jasbir S. Arora. "Survey of Multi-Objective Optimization Methods for Engineering." In: *Structural and multidisciplinary optimization* 26.6 (2004), pp. 369–395. DOI: 10.1007/s00158-003-0368-6.

[149] Anna Lisa Martin-Niedecken, Elena Márquez Segura, Katja Rogers, Stephan Niedecken, and Laia Turmo Vidal. "Towards Socially Immersive Fitness Games: An Exploratory Evaluation Through Embodied Sketching." In: CHI PLAY '19 Extended Abstracts. Barcelona, Spain: Association for Computing Machinery, 2019, pp. 525–534. DOI: 10.1145/3341215.3356293.

[150] Anna Lisa Martin-Niedecken, Katja Rogers, Laia Turmo Vidal, Elisa D. Mekler, and Elena Márquez Segura. "ExerCube vs. Personal Trainer: Evaluating a Holistic, Immersive, and Adaptive Fitness Game Setup." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI 2019. Glasgow, Scotland Uk, 2019, pp. 1–15. DOI: 10.1145/3290605.3300318.

[151] Uwe Maurer, Asim Smailagic, Daniel P. Siewiorek, and Michael Deisher. "Activity Recognition and Monitoring using Multiple Sensors on Different Body Positions." In: *International Workshop on Wearable and Implantable Body Sensor Networks*. BSN 2006. Cambridge, MA, USA, 2006. DOI: 10.1109/BSN.2006.6.

[152] Michael Meredith and Steve Maddock. "Using a Half-Jacobian for Real-Time Inverse Kinematics." In: *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*. 2004, pp. 81–88.

[153] Johannes Moskaliuk, Johanna Bertram, and Ulrike Cress. "Impact of Virtual Training Environments on the Acquisition and Transfer of Knowledge." In: *Cyberpsychology, Behavior, and Social Networking* 16.3 (2013), pp. 210–214. DOI: 10.1089/cyber.2012.0416.

[154] Florian Mueller and Katherine Isbister. "Movement-Based Game Guidelines." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: Association for Computing Machinery, 2014, pp. 2191–2200. DOI: 10.1145/2556288.2557163.

[155] Subhas Chandra Mukhopadhyay. "Wearable Sensors for Human Activity Monitoring: A Review." In: *IEEE Sensors Journal* 15.3 (2015), pp. 1321–1330. DOI: 10.1109/JSEN.2014.2370945.

[156] Gergely Nagymáté and Rita M. Kiss. "Application of OptiTrack Motion Capture Systems in Human Movement Analysis." In: *Recent Innovations in Mechatronics* 5.1 (2018), pp. 1–9. DOI: 10.17667/riim.2018.1/13.

[157] Yoshihiko Nakamura and Hideo Hanafusa. "Inverse Kinematic Solutions with Singularity Robustness for Robot Manipulator Control." In: *Journal of Dynamic Systems, Measurement, and Control* 108.3 (1986), pp. 163–171. DOI: 10.1115/1.3143764.

[158] Jan Thorsten Neitzel. "Entwicklung von adaptiven Ansätzen zur Steigerung der Spielerleistung in einem Exergame." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2020.

[159] Pedro Neto, Dário Pereira, J. Norberto Pires, and A. Paulo Moreira. "Real-Time and Continuous Hand Gesture Spotting: An Approach Based on Artificial Neural Networks." In: *International Conference on Robotics and Automation*. IEEE, 2013, pp. 178–183. DOI: 10.1109/ICRA.2013.6630573.

[160] Diederick C. Niehorster, Li Li, and Markus Lappe. "The Accuracy and Precision of Position and Orientation Tracking in the HTC Vive Virtual Reality System for Scientific Research." In: *i-Perception* 8.3 (2017), pp. 1–23. DOI: 10.1177/2041669517708205.

[161] Jasmir Nijhar, Nadia Bianchi-Berthouze, and Gemma Boguslawski. "Does Movement Recognition Precision Affect the Player Experience in Exertion Games?" In: *Intelligent Technologies for Interactive Entertainment*. Vol. 78. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 73–82. DOI: 10.1007/978-3-642-30214-5_9.

[162] Niels C. Nilsson, Stefania Serafin, Morten H. Laursen, Kasper S. Pedersen, Erik Sikström, and Rolf Nordahl. "Tapping-In-Place: Increasing the Naturalness of Immersive Walking-In-Place Locomotion Through Novel Gestural Input." In: *IEEE Symposium on 3D User Interfaces (3DUI)*. 3DUI 2013. Orlando, FL, USA, 2013, pp. 31–38. DOI: 10.1109/3DUI.2013.6550193.

[163] Nintendo. *Ring Fit Adventure*. 2019. URL: https://www.nintendo.com/games/detail/ring-fit-adventure-switch/ (Last accessed on March 3, 2021).

[164] David E. Orin and William W. Schrader. "Efficient Computation of the Jacobian for Robot Manipulators." In: *The International Journal of Robotics Research* 3.4 (1984), pp. 66–75. DOI: 10.1177/027836498400300404.

[165] Xueni Pan and Antonia F. de C. Hamilton. "Why and How to Use Virtual Reality to Study Human Social Interaction: The Challenges of Exploring a New Research Landscape." In: *British Journal of Psychology* 109.3 (2018), pp. 395–417. DOI: 10.1111/bjop.12290.

[166] Mathias Parger, Joerg H. Mueller, Dieter Schmalstieg, and Markus Steinberger. "Human Upper-Body Inverse Kinematics for Increased Embodiment in Consumer-Grade Virtual Reality." In: *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*. VRST 2018. Tokyo, Japan, 2018. DOI: 10.1145/3281505.3281529.

[167] Alexandra Pfister, Alexandre M. West, Shaw Bronner, and Jack Adam Noah. "Comparative Abilities of Microsoft Kinect and Vicon 3D Motion Capture for Gait Analysis." In: *Journal of Medical Engineering & Technology* 38.5 (2014), pp. 274–280. DOI: 10.3109/03091902.2014.909540.

[168] Ivan Phelan, Madelynne Arden, Carol Garcia, and Chris Roast. "Exploring Virtual Reality and Prosthetic Training." In: *IEEE Virtual Reality*. VR 2015. Arles, France, 2015, pp. 353–354. DOI: 10.1109/VR.2015.7223441.

[169] Ignazio Pillai, Giorgio Fumera, and Fabio Roli. "Designing Multi-Label Classifiers that Maximize F Measures: State of The Art." In: *Pattern Recognition* 61 (2017), pp. 394–404. DOI: /10.1016/j.patcog.2016.08.008.

[170] John C. Platt. *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*. Tech. rep. 1998.

[171] Paul Posadzki and Edzard Ernst. "Yoga for Low Back Pain: A Systematic Review of Randomized Clinical Trials." In: *Clinical rheumatology* 30.9 (2011), pp. 1257–1262. DOI: 10.1007/s10067-011-1764-8.

[172] Kjetil Raaen and Ivar Kjellmo. "Measuring Latency in Virtual Reality Systems." In: *Entertainment Computing - ICEC 2015*. Vol. 9353. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 457–462. DOI: 10.1007/978-3-319-24589-8_40.

[173] Lawrence R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition." In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286. DOI: 10.1109/5.18626.

[174] John T. Ramshur. "Design, Evaluation, and Application of Heart Rate Variability Analysis Software (HRVAS)." Master Thesis. Memphis, TN: University of Memphis, 2010.

[175] John T. Ramshur. *Heart Rate Variability Analysis Software (HRVAS) Package*. 2010. URL: https://github.com/jramshur/HRVAS (Last accessed on March 3, 2021).

[176] Werner Siegfried Ravyse, A. Seugnet Blignaut, Verona Leendertz, and Alex Woolner. "Success Factors for Serious Games to Enhance Learning: A Systematic Review." In: *Virtual Reality* 21.1 (2017), pp. 31–58. DOI: 10.1007/s10055-016-0298-4.

[177] James T. Reason. "Relations Between Motion Sickness Susceptibility, the Spiral After-Effect and Loudness Estimation." In: *British Journal of Psychology* 59.4 (1968), pp. 385–393. DOI: 10.1111/j.2044-8295.1968.tb01153.x.

[178] James T. Reason and Joseph John Brand. *Motion Sickness*. Academic press, 1975.

[179] Lisa Rebenitsch and Charles Owen. "Review on Cybersickness in Applications and Visual Displays." In: *Virtual Reality* 20.2 (2016), pp. 101–125. DOI: 10.1007/s10055-016-0285-9.

[180] Paula Rego, Pedro Miguel Moreira, and Luís Paulo Reis. "Serious Games for Rehabilitation: A Survey and a Classification Towards a Taxonomy." In: *5th Iberian Conference on Information Systems and Technologies*. Santiago de Compostela, Spain: IEEE, 2010, pp. 1–6.

[181] Research and Markets. *Serious Game - Global Market Outlook (2017-2026)*. Tech. rep. 2018.

[182] Research and Markets. *Virtual Reality Gaming Market: Global Industry Trends, Share, Size, Growth, Opportunity and Forecast 2019-2024*. Tech. rep. 2019.

[183] Christian Reuter. "Authoring Collaborative Multiplayer Games - Game Design Patterns, Structural Verification, Collaborative Balancing and Rapid Prototyping." PhD thesis. Darmstadt, DE: Technical University of Darmstadt, 2016. URL: 10.26083/tuprints-00017499.

[184] Gary E. Riccio and Thomas A. Stoffregen. "An Ecological Theory of Motion Sickness and Postural Instability." In: *Ecological Psychology* 3.3 (1991), pp. 195–240. DOI: 10.1207/s15326969eco0303_2.

[185] Wilhelm von Rosenberg, Theerasak Chanwimalueang, Tricia Adjei, Usman Jaffer, Valentin Goverdovsky, and Danilo P. Mandic. "Resolving Ambiguities in the LF/HF Ratio: LF-HF Scatter Plots for the Categorization of Mental and Physical Stress from HRV." In: *Frontiers in Physiology* 8 (2017), pp. 1–12. DOI: 10.3389/fphys.2017.00360.

[186] Ralph L. Rosnow and Robert Rosenthal. "Effect Sizes for Experimenting Psychologists." In: *Canadian Journal of Experimental Psychology* 57.3 (2003), pp. 221–237. DOI: 10.1037/h0087427.

[187] Franca Rupprecht, Bo Heck, Bernd Hamann, and Achim Ebert. "Signal-Processing Transformation from Smartwatch to Arm Movement Gestures." In: *Advances in Human Factors and Systems Interaction*. Vol. 781. Cham: Springer International Publishing, 2019, pp. 109–121. DOI: 10.1007/978-3-319-94334-3_13.

[188] Neil J. Salkind. *Exploring Research*. 8th ed. Pearson Education International, 2012.

[189] Gayani Samaraweera, Alex Perdomo, and John Quarles. "Applying Latency to Half of a Self-Avatar's Body to Change Real Walking Patterns." In: *IEEE Virtual Reality*. VR 2015. Arles, France, 2015, pp. 89–96. DOI: 10.1109/VR.2015.7223329.

[190] Atsushi Sato and Asako Yasuda. "Illusion of Sense of Self-Agency: Discrepancy Between the Predicted and Actual Sensory Consequences of Actions Modulates the Sense of Self-Agency, but Not the Sense of Self-Ownership." In: *Cognition* 94.3 (2005), pp. 241–255. DOI: 10.1016/j.cognition.2004.04.003.

[191] Philip Schäfer, Marius Koller, Julia Diemer, and Gerrit Meixner. "Development and Evaluation of a Virtual Reality-System with Integrated Tracking of Extremities Under the Aspect of Acrophobia." In: *2015 SAI Intelligent Systems Conference*. IntelliSys 2015. 2015, pp. 408–417. DOI: 10.1109/IntelliSys.2015.7361173.

[192] Fred Shaffer and James P. Ginsberg. "An Overview of Heart Rate Variability Metrics and Norms." In: *Frontiers in public health* 5 (2017), pp. 1–17. DOI: 10.3389/fpubh.2017.00258.

[193] Robert Shewaga, Alvaro Uribe-Quevedo, Bill Kapralos, and Fahad Alam. "A Comparison of Seated and Room-Scale Virtual Reality in a Serious Game for Epidural Preparation." In: *IEEE Transactions on Emerging Topics in Computing* 8.1 (2020), pp. 218–232. DOI: 10.1109/TETC.2017.2746085.

[194] Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul J. M. Havinga. "Complex Human Activity Recognition using Smartphone and Wrist-Worn Motion Sensors." In: *Sensors* 16.4 (2016), p. 426. DOI: 10.3390/s16040426.

[195] Ken Shoemake. "Animating Rotation with Quaternion Curves." In: *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH. New York, NY, USA, 1985, pp. 245–254. DOI: 10.1145/325334.325242.

[196] Hubert P. H. Shum, Edmond S. L. Ho, Yang Jiang, and Shu Takagi. "Real-Time Posture Reconstruction for Microsoft Kinect." In: *IEEE Transactions on Cybernetics* 43.5 (2013), pp. 1357–1369. DOI: 10.1109/TCYB.2013.2275945.

[197] Mel Slater. "Immersion and the Illusion of Presence in Virtual Reality." In: *British Journal of Psychology* 109.3 (2018), pp. 431–433. DOI: 10.1111/bjop.12305.

[198] Mel Slater and Maria V. Sanchez-Vives. "Enhancing Our Lives with Immersive Virtual Reality." In: *Frontiers in Robotics and AI* 3 (2016), p. 74. DOI: 10.3389/frobt.2016.00074.

[199] Mel Slater, Bernhard Spanlang, Maria V. Sanchez-Vives, and Olaf Blanke. "First Person Experience of Body Transfer in Virtual Reality." In: *PloS one* 5.5 (2010), e10564. DOI: 10.1371/journal.pone.0010564.

[200] Mel Slater and Sylvia Wilbur. "A Framework for Immersive Virtual Environments (FIVE): Speculations on the Role of Presence in Virtual Environments." In: *Presence: Teleoperators and Virtual Environments* 6.6 (1997), pp. 603–616. DOI: 10.1162/pres.1997.6.6.603.

[201] Breannan Smith, Chenglei Wu, He Wen, Patrick Peluse, Yaser Sheikh, Jessica K. Hodgins, and Takaaki Shiratori. "Constraining Dense Hand Surface Tracking with Elasticity." In: *ACM Transactions on Graphics* 39.6 (2020). DOI: 10.1145/3414685.3417768.

[202] Jan Sohnsmeyer, Hajo Gilbrich, and Burkhard Weisser. "Effect of a Six-Week-Intervention With an Activity-Promoting Video Game on Isometric Muscle Strength in Elderly Subjects." In: *International Journal of Computer Science in Sport (International Association of Computer Science in Sport)* 9.2 (2010), pp. 75–79.

[203] Kate A. Spitzley and Andrew R. Karduna. "Feasibility of Using a Fully Immersive Virtual Reality System for Kinematic Data Collection." In: *Journal of Biomechanics* 87 (2019), pp. 172–176. DOI: 10.1016/j.jbiomech.2019.02.015.

[204] Misha Sra and Chris Schmandt. "MetaSpace: Full-Body Tracking for Immersive Multiperson Virtual Reality." In: *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15 Adjunct. Daegu, Kyungpook, Republic of Korea: Association for Computing Machinery, 2015, pp. 47–48. DOI: 10.1145/2815585.2817802.

[205] Kay M. Stanney, Robert S. Kennedy, and Julie M. Drexler. "Cybersickness is Not Simulator Sickness." In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 41.2 (1997), pp. 1138–1142. DOI: 10.1177/107118139704100292.

[206] Statista. *Fields of Application for Virtual Reality Headsets in the U.S. 2017*. 2017. URL: https://www.statista.com/forecasts/790469/fields-of-application-for-virtual-reality-headsets-in-the-us (Last accessed on March 3, 2021).

[207] Statista. *Forecast for the Number of Active Virtual Reality Users Worldwide From 2014 to 2018*. 2017. URL: https://www.statista.com/statistics/426469/active-virtual-reality-users-worldwide/ (Last accessed on March 3, 2021).

[208] Statista. *Share of Global Virtual Reality Software Revenue in 2018 and 2022, By Category*. 2019. URL: https://www.statista.com/statistics/528739/virtual-reality-software-revenue-share-worldwide-by-category/ (Last accessed on March 3, 2021).

[209] Statista. *Virtual Reality (VR) Gaming Revenue Worldwide From 2017 to 2024*. 2020. URL: https://www.statista.com/statistics/499714/global-virtual-reality-gaming-sales-revenue/ (Last accessed on March 3, 2021).

[210] Jan-Philipp Stauffert, Florian Niebling, and Marc Erich Latoschik. "Effects of Latency Jitter on Simulator Sickness in a Search Task." In: *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. Reutlingen, Germany, 2018, pp. 121–127. DOI: 10.1109/VR.2018.8446195.

[211] Penelope Sweetser and Peta Wyeth. "GameFlow: A Model for Evaluating Player Enjoyment in Games." In: *Computers in Entertainment* 3.3 (2005). DOI: 10.1145/1077246.1077253.

[212] James S. Thomas, Christopher R. France, Samuel T. Leitkam, Megan E. Applegate, Peter E. Pidcoe, and Stevan Walkowski. "Effects of Real-World Versus Virtual Environments on Joint Excursions in Full-Body Reaching Tasks." In: *IEEE Journal of Translational Engineering in Health and Medicine* 4 (2016), pp. 1–8. DOI: 10.1109/JTEHM.2016.2623787.

[213] Deepak Tolani, Ambarish Goswami, and Norman I. Badler. "Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs." In: *Graphical Models* 62.5 (2000), pp. 353–388. DOI: 10.1006/gmod.2000.0528.

[214] Dorra Trabelsi, Samer Mohammed, Faicel Chamroukhi, Latifa Oukhellou, and Yacine Amirat. "An Unsupervised Approach for Automatic Activity Recognition Based on Hidden Markov Model Regression." In: *IEEE Transactions on Automation Science and Engineering* 10.3 (2013), pp. 829–835. DOI: 10.1109/TASE.2013.2256349.

[215] Thomas Tregel. "Online Content Generation in Mobile Game Applications – Adaptation and Personalization for Location-based Game Systems." PhD thesis. Darmstadt, DE: Technical University of Darmstadt, 2021. DOI: 10.26083/tuprints-00017499.

[216] Thomas Tregel, Andreas Gilbert, Robert Konrad, Petra Schäfer, and Stefan Göbel. "Examining Approaches for Mobility Detection Through Smartphone Sensors." In: *Joint International Conference on Serious Games*. Vol. 11243. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018, pp. 217–228. DOI: 10.1007/978-3-030-02762-9_22.

[217] Michel Treisman. "Motion Sickness: An Evolutionary Hypothesis." In: *Science* 197.4302 (1977), pp. 493–495. DOI: 10.1126/science.301659.

[218] TriCAT. *Simulation Training for Medical Personnel in Highly Immersive Multi-User VR Environment*. 2020. URL: https://tricat.net/enterprise-solutions/i-medtasim/ (Last accessed on March 3, 2021).

[219] Roman Uhlig. "Analysis of Machine Learning Approaches for Motion Recognition in Virtual Reality." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[220] Luis Unzueta, Manuel Peinado, Ronan Boulic, and Ángel Suescun. "Full-Body Performance Animation with Sequential Inverse Kinematics." In: *Graphical Models* 70.5 (2008), pp. 87–104. DOI: 10.1016/j.gmod.2008.03.002.

[221] Thomas Waltemate, Dominik Gall, Daniel Roth, Mario Botsch, and Marc E. Latoschik. "The Impact of Avatar Personalization and Immersion on Virtual Body Ownership, Presence, and Emotional Response." In: *IEEE Transactions on Visualization and Computer Graphics* 24.4 (2018), pp. 1643–1652. DOI: 10.1109/TVCG.2018.2794629.

[222] Thomas Waltemate, Irene Senna, Felix Hülsmann, Marieke Rohde, Stefan Kopp, Marc Ernst, and Mario Botsch. "The Impact of Latency on Perceptual Judgments and Motor Performance in Closed-Loop Interaction in Virtual Reality." In: *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. VRST 2016. Munich, Germany, 2016, pp. 27–35. DOI: 10.1145/2993369.2993381.

[223] Qifei Wang, Gregorij Kurillo, Ferda Ofli, and Ruzena Bajcsy. "Evaluation of Pose Tracking Accuracy in the First and Second Generations of Microsoft Kinect." In: *International Conference on Healthcare Informatics*. Dallas, TX, USA: IEEE, 2015, pp. 380–389. DOI: 10.1109/ICHI.2015.54.

[224] Colin Ware and Ravin Balakrishnan. "Reaching for Objects in VR Displays: Lag and Frame Rate." In: *ACM Transactions on Computer-Human Interaction* 1.4 (1994), pp. 331–356. DOI: 10.1145/198425.198426.

[225] Jeremy D. Wendt, Mary C. Whitton, and Frederick P. Brooks. "GUD WIP: Gait-Understanding-Driven Walking-In-Place." In: *IEEE Virtual Reality Conference*. VR 2010. Boston, MA, USA, 2010, pp. 51–58. DOI: 10.1109/VR.2010.5444812.

[226] Markus Wistrand and Erik L. L. Sonnhammer. "Improving Profile HMM Discrimination by Adapting Transition Probabilities." In: *Journal of Molecular Biology* 338.4 (2004), pp. 847–854. DOI: 10.1016/j.jmb.2004.03.023.

[227] Bob G. Witmer and Michael J. Singer. "Measuring Presence in Virtual Environments: A Presence Questionnaire." In: *Presence: Teleoperators and Virtual Environments* 7.3 (1998), pp. 225–240. DOI: 10.1162/105474698565686.

[228] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd ed. Morgan Kaufmann Publishers, 2011.

[229] Xiaomao Wu, Lizhuang Ma, Zhihua Chen, and Yan Gao. "A 12-DOF Analytic Inverse Kinematics Solver for Human Motion Control." In: *Journal of Information & Computational Science* 1.1 (2004), pp. 137–141.

[230] Shuo Yan, Gangyi Ding, Zheng Guan, Ningxiao Sun, Hongsong Li, and Longfei Zhang. "OutsideMe: Augmenting Dancer's External Self-Image by using A Mixed Reality System." In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA 2015. Seoul, Republic of Korea, 2015, pp. 965–970. DOI: 10.1145/2702613.2732759.

[231] Jie Yang and Yangsheng Xu. *Hidden Markov Model for Gesture Recognition*. Tech. rep. Carnegie Mellon University, Pittsburgh, Pennsylvania, The Robotics Institute, 1994.

[232] Soojeong Yoo and Judy Kay. "VRun: Running-in-Place Virtual Reality Exergame." In: *Proceedings of the 28th Australian Conference on Computer-Human Interaction*. OzCHI 2016. Launceston, Tasmania, Australia, 2016, pp. 562–566. DOI: 10.1145/3010915.3010987.

[233] Mary K. Young, John J. Rieser, and Bobby Bodenheimer. "Dyadic Interactions with Avatars in Immersive Virtual Environments: High Fiving." In: *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception*. SAP 2015. Tübingen, Germany, 2015, pp. 119–126. DOI: 10.1145/2804408.2804410.

[234] Ye Yuan and Anthony Steed. "Is the Rubber Hand Illusion Induced by Immersive Virtual Reality?" In: *IEEE Virtual Reality Conference*. VR 2010. 2010, pp. 95–102. DOI: 10.1109/VR.2010.5444807.

[235] Chun Zhu and Weihua Sheng. "Motion- and Location-Based Online Human Daily Activity Recognition." In: *Pervasive and Mobile Computing* 7.2 (2011), pp. 256–269. DOI: 10.1016/j.pmcj.2010.11.004.

A

APPENDIX

A.1 CYBERSICKNESS

Cybersickness Factors

Prior research has thoroughly investigated the primary factors that contribute to the cause of cybersickness and categorizes them into personal, environmental, and hardware factors [63, 131]. The main factors are summarized in Table 20.

Table 20: Factors affecting cybersickness. Adapted from [63, 131].

Personal	**Effects**
Illnesses	Illnesses and poor health condition generally induce cybersickness
Age	Cybersickness symptoms decrease with age
Gender	Women seem to be more susceptible to cybersickness
Personal adaptability and experience	Repeatedly exposures to VR could reduce cybersickness
Environmental	**Effect**
Controllability	Lower controllability induces cybersickness
Duration	Longer exposure time elicits cybersickness
Posture	Sitting seems to be a better position to reduce cybersickness
Hardware	**Effect**
Lag	A high delay between users' actions and visual feedback causes cybersickness
Refresh rate	Low refresh rates elicit cybersickness
Flicker	Flicker is distracting and induces cybersickness
Positional tracking error	Inaccurate tracking causes cybersickness
Field of view	Wider field of view induces cybersickness
Calibration (interpupillary distance)	Poor calibration induces cybersickness

Cybersickness Symptoms

The total score on the *Simulator Sickness Questionnaire* is calculated as follows [113]: For each of 16 possible symptoms (see Table 21), a single score of zero (no symptoms), one (slight symptoms), two (moderate symptoms), or three (severe symptoms) is assigned. Then, the scores for each symptom group, i. e., *nausea*, *oculomotor*, and *disorientation*, are summed and multiplied by 3.74. Weighted scores for each individual group are obtained by multiplying *nausea* by 9.54, *oculomotor* by 7.58, and *disorientation* by 13.92. Depending on the score, the symptoms are categorized as negligible (total score lower than 5), minimal (5 to 10), significant (10 to 15), or concerning (15 to 20) [205].

Table 21: *Simulator Sickness Questionnaire* symptoms [113].

Symptoms	**Nausea**	**Oculomotor**	**Disorientation**
General discomfort	×	×	
Fatigue		×	
Headache		×	
Eyestrain		×	
Difficulty focusing		×	×
Increased salivation	×		
Sweating	×		
Nausea	×		×
Difficulty concentrating	×	×	
Fullness of head			×
Blurred vision		×	×
Dizzy (eyes open)			×
Dizzy (eyes closed)			×
Vertigo			×
Stomach awareness	×		
Burping	×		

A.2 FOUNDATIONS OF INVERSE KINEMATICS

The inverse kinematics problem describes the problem of defining a possible set of the joint configuration so that the end-effector of the manipulator attains the target position and orientation [62]. Similar to manipulators, also virtual characters consist of multiple joints and links. Therefore, we can apply the inverse kinematics problem to an articulated model to move an end-effector (a hand and a foot) to the target position and orientation as rapidly and accurately as possible. In other words, by solving the inverse kinematics problems, we determine the rotations of each bone in a kinematic chain so that the end-effectors reach the desired targets.

Several inverse kinematics approaches have already been presented in the '80s, e. g., in robotics to control (industrial) manipulators [157, 164]. Other researchers proposed various solutions for character animation in computer graphics [7, 115, 152, 213, 220]. However, they often do not generate visual smooth animations, result in unnatural poses, or require many iterations to converge, and thus cause high computational costs.

Formulation of Inverse Kinematics

In the following, we provide a short overview of the inverse kinematics problem. The end-effector's position and rotation are represented as $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k]^T$, where k is the number of end-effectors. The column vector $\mathbf{s}$ contains $m = 6k$ scalar entries, where the first three parameters denote the position and the following three parameters the rotation of the end-effector. The target positions and rotations of k end-effectors are represented as $\mathbf{t} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k]^T$. The desired change in position and rotation of the i^{th} end-effector is then given by $\mathbf{e}_i = \mathbf{t}_i - \mathbf{s}_i$. Furthermore, the joint angles are represented as $\theta = [\theta_1, \theta_2, \ldots, \theta_n]^T$. The column vector θ consists of n scalar entries, where n is the number of degrees of freedom in the kinematic chain.

The goal of the inverse kinematics problem is to find values for the joint angles θ so that the resulting configuration moves the end-effector as close as possible to the desired target [34]:

$$\mathbf{t}_i = \mathbf{s}_i(\theta), \text{ for all i.} \tag{10}$$

Solving Equation 10 is challenging because, depending on the degrees of freedom and the end-effector target, there may not always be a unique solution. Instead, multiple possible solutions or no solution (e. g., when the desired position is out of the reachable workspace) for θ may also exist. For simple, the inverse kinematics problem can be solved analytically for trivial articulated models with a low number of degrees of freedom [54, 110, 111, 229]. However, analytical methods are insufficient for non-trivial models with long kinematic chains; therefore, alternative methods are required. Previous research proposed several methods to provide a solution to the inverse kinematics problem, e. g., numerical methods including Jacobian [36, 115, 164], Newton [85], and heuristic methods [7, 114], hybrid methods [7, 220], or data-driven methods [1]. A survey by Aristidou et al. [8] provides a detailed overview, including the advantages and disadvantages of several inverse kinematics solvers.

Jacobian Inverse Methods

The Jacobian methods offer a linear approximation of the inverse kinematics problem to gradually move the end-effector to the desired target. The Jacobian $\mathbf{J} \in \mathbb{R}^{m \times n}$ is a matrix of partial derivatives and is described as a function of θ values [34]:

$$\mathbf{J}(\theta)_{ij} = \frac{\partial \mathbf{s}_i}{\partial \theta_j}, \text{ for } i = 1, \ldots, k \text{ and } j = 1, \ldots, n, \tag{11}$$

where k is the number of end-effectors and n the number of degrees of freedom. Using the Jacobian matrix, we can solve Equation 10 and approximate the solution to the inverse kinematics problem. We aim to find the joint angles θ to minimize the error $\mathbf{e}$ between the current and target position and rotation of the end-effector. For this purpose, we make a small change in the joint angles θ and approximate the consequent change in end-positions as:

$$\Delta \mathbf{s} \approx \mathbf{J} \Delta \theta. \tag{12}$$

As we are searching for $\Delta \mathbf{s}$, which is approximately equal to $\mathbf{e}$, we can estimate the change in $\Delta \theta$ to be:

$$\Delta \theta \approx \mathbf{J}^{-1} \mathbf{e}. \tag{13}$$

However, because the Jacobian matrix is possibly not a square matrix or invertible and can also suffer from singularity problems (loss of a degree of freedom), several researchers presented methods for approximating the Jacobian inverse. An excellent explanation of the Jacobian inverse methods to solve Equation 13 is given by Aristidou et al. [8], including *Jacobian transpose*, *pseudoinverse*, *damped least squares*, *damped least squares with singular value decomposition*, and *selectively damped least squares*.

One of the mathematically easiest methods to solve the inverse kinematics problem is to use a transpose matrix: $\Delta \theta = \alpha \mathbf{J}^T \mathbf{e}$, for $\alpha = \langle \mathbf{e}, \mathbf{J}\mathbf{J}^T \mathbf{e} \rangle / \langle \mathbf{J}\mathbf{J}^T \mathbf{e}, \mathbf{J}\mathbf{J}^T \mathbf{e} \rangle$, where $\langle a, b \rangle$ indicates the dot product between vectors a and b. The *Jacobian transpose* is very easily computed; however, it converges very slowly for multiple end-effectors [36]. *Pseudoinverse* is another very popular method to solve the inverse kinematics problem and is calculated as: $\Delta \theta = \mathbf{J}^T (\mathbf{J}\mathbf{J}^T)^{-1} \mathbf{e}$. However, in contrast to *Jacobian transpose*, the disadvantage of this method is the increased computing effort, especially if the character model has many degrees of freedom.

Because *pseudoinverse* suffers from singularity problems, other methods such as *damped least squares* and *damped least squares with singular value decomposition* were introduced [34]. The *damped least squares* method aims to find a solution for the inverse kinematics problem using $\Delta \theta = \mathbf{J}^T (\mathbf{J}\mathbf{J}^T + \lambda^2 \mathbf{I})^{-1} \mathbf{e}$, with an appropriate damping constant $\lambda > 0$. Moreover, the *selectively damped least squares* method, which was presented by Buss and Kim in [36], is another extension of the *damped least squares* and does not require a damping constant but instead needs an appropriate value for maximum change in any joint angle.

Newton methods

An alternative approach to solve the inverse kinematics problem is based on Newton methods, whereby the problem is posed as a minimization problem [8]. Newton methods generate smooth motion and do not suffer from singularity problems; however, they are difficult to implement and suffer from high computational cost per iteration [8].

Heuristic methods

Heuristic methods solve the inverse kinematics gradually, as they iteratively apply simple operations to reduce the distance between the desired and actual target [8]. *Cyclic coordinate descent* [114] and *forward and backward reaching inverse kinematics* [7] methods are the two of the most popular heuristic methods to solve the inverse kinematics problem. *Cyclic coordinate descent* requires only a dot and cross product and has low computational costs, is algorithmically simple, and straight-forward [114]. However, it can generate unnatural pose, even when joint constraints are satisfied [8]. *Forward and backward reaching inverse kinematics* avoids the usage of rotational angles or matrices and instead updates the joints' positions along a line to the next joint [7].

A.3 JOINT CONSTRAINTS

This section summarizes the joint constraints of the skeleton model. Table 22 defines the joints' range of motions, whereas joints' local frames are depicted in Figure 37.

Table 22: Angular constraints for the joints on the left (L) and right (R) side.

Kinematic chain	**Joint**	**x-axis**		**y-axis**		**z-axis**	
		Min	Max	Min	Max	Min	Max
Head	*Neck*	−55°	55°	−105°	105°	−75°	75°
	Spine	−35°	15°	/	/	−50°	50°
Arm	*Wrist* (L)	−45°	45°	−95°	−95°	−75°	75°
	Wrist (R)	−45°	45°	−95°	−95°	−75°	75°
	Elbow (L)	0°	155°	/	/	/	/
	Elbow (R)	0°	155°	/	/	/	/
	Shoulder (L)	−65°	145°	−105°	105°	−105°	105°
	Shoulder (R)	−65°	145°	−105°	105°	−105°	105°
Leg	*Knee* (L)	0°	165°	/	/	/	/
	Knee (R)	0°	165°	/	/	/	/
	Hip (L)	−125°	45°	−65°	55°	−55°	15°
	Hip (R)	−125°	45°	−55°	65°	−15°	55°

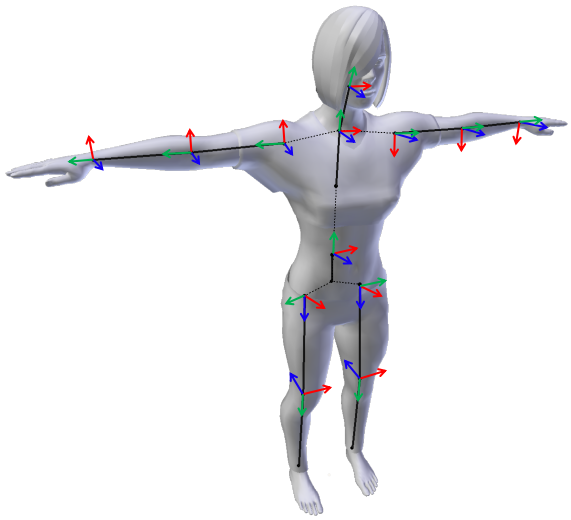


Figure 37: Local frames: The x, y, and z-axis are represented as red, green, and blue, respectively.

A.4 MOVEMENT DESCRIPTION

Tables 23-24 describe movements for training and testing data set used for full-body motion reconstruction. The training data set is used to optimize parameters for full-body motion reconstruction, whereas the testing data set is used to determine the accuracy of the estimated pose. Table 25 describes the three yoga movements used to train and test machine learning models for full-body motion recognition.

Table 23: Description of movements used for training data set for full-body motion reconstruction.

Movement	**Description**	**Number of repetitions/ duration**
Shoulder forward flexion	Raise the arms to eye level and lower them back to the side	10 each side
Shoulder abduction	Raise the arms to the side until they are pointing to the ceiling and lower them back to the side	10 each side
Shoulder horizontal abduction	Raise the arms to eye level and move them horizontally	10 each side
Rotation with arms at side	With the elbow flexed, rotate the arms inward and outward	10 each side
Rotation with arms in abduction	With the elbow flexed, raise arms to shoulder level and rotate them inward and outward	10 each side
Elbow flexion	Flex the elbow while the palm is facing up and straighten the elbow again	10 each side
Hand pronation	With the elbow flexed, move the palm from facing up to facing down	10 each side
Knee flexion	Flex the knee while standing on one foot	10 each side
Hip flexion	Raise the knee as high as possible while standing on one foot	10 each side

Table 24: Description of movements used for testing data set for full-body motion reconstruction.

Movement	**Description**	**Number of repetitions/ duration**
Standing	Stand as still as possible	30 s
Walking (in a circle)	Walk in the tracking area	30 s
Punching	Box invisible punching bag	10 each side
Kicking	Kick invisible punching bag	10 each side
Squats	Sit down as if there is an invisible chair and stand up	10 each side
Lunges	Take a large step forward, move the knee to the ground and return to comfortable standing	10 each side

Table 25: Description of yoga movements used for training and testing data set for full-body motion recognition.

Movement	**Description**	**Number of repetitions**
Warrior I	Sink the body into a lunge until the right knee is bent to a right angle. Turn the left foot halfway inwards and keep the back leg straight. Move the arms upwards.	100
Warrior II	Sink the body into a lunge until the right knee is bent to a right angle. Turn the left foot halfway inwards and keep the back leg straight. Raise the arms to the side at shoulder height so that they are parallel to the floor.	100
Extended side angle	Sink the body into a lunge until the right knee is bent to a right angle. Turn the left foot halfway inwards and keep the back leg straight. Bring the right arm down with the elbow on the right thigh. Reach the left arm over the head, next to the left ear.	100

A.5 ACCURACY OF FULL-BODY MOTION RECONSTRUCTION

Tables 26-29 summarize overall accuracy for *Jacobian transpose*, *pseudoinverse*, *damped least squares with singular value decomposition*, and *selectively damped least squares*, respectively. The results for the *damped least squares* method are presented in Section 4.3.1.

Table 26: Overall error in joint position and rotation for the *Jacobian transpose* method.

Joint	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]
Head	14.77 ± 8.39	0.01 ± 0.01
Hips	3.13 ± 2.20	0.44 ± 0.34
Left hand	25.52 ± 19.25	0.03 ± 0.34
Right hand	21.20 ± 15.63	0.02 ± 0.03
Left elbow	67.40 ± 23.50	44.27 ± 10.23
Right elbow	66.52 ± 29.78	41.67 ± 9.03
Left foot	31.90 ± 18.95	0.93 ± 0.64
Right foot	28.93 ± 16.93	0.81 ± 0.51
Left knee	61.52 ± 34.45	23.16 ± 15.98
Right knee	62.42 ± 72.33	24.83 ± 15.99
Average	38.33 ± 24.14	13.62 ± 5.31

Table 27: Overall error in joint position and rotation for the *Jacobian psedoinverse* method.

Joint	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]
Head	199.25 ± 149.59	53.77 ± 54.05
Hips	3.13 ± 2.120	0.44 ± 0.34
Left hand	562.48 ± 301.82	126.67 ± 61.26
Right hand	566.18 ± 306.36	126.23 ± 62.13
Left elbow	381.62 ± 170.87	128.37 ± 54.85
Right elbow	384.47 ± 176.60	128.44 ± 55.37
Left foot	786.94 ± 352.70	86.00 ± 46.99
Right foot	781.79 ± 351.31	87.17 ± 46.72
Left knee	433.03 ± 205.68	91.34 ± 46.41
Right knee	442.21 ± 225.73	92.21 ± 46.23
Average	454.11 ± 224.29	92.16 ± 47.43

Table 28: Overall error in joint position and rotation for the *damped least squares with singular value decomposition* method.

Joint	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]
Head	14.74 ± 8.39	0.001 ± 0.01
Hips	3.13 ± 2.120	0.44 ± 0.34
Left hand	21.24 ± 19.60	0.34 ± 0.33
Right hand	17.49 ± 15.93	0.28 ± 0.27
Left elbow	68.57 ± 24.26	43.08 ± 10.25
Right elbow	67.32 ± 29.87	41.45 ± 8.83
Left foot	31.06 ± 18.68	1.38 ± 0.87
Right foot	28.15 ± 16.69	1.22 ± 0.67
Left knee	59.74 ± 33.60	23.56 ± 16.12
Right knee	60.86 ± 71.76	25.20 ± 16.09
Average	37.23 ± 24.10	13.69 ± 5.38

Table 29: Overall error in joint position and rotation for the *selectively damped least squares* method.

Joint	**Error in position** e_{pos} [mm]	**Error in rotation** e_{rot} [°]
Head	14.75 ± 8.39	0.001 ± 0.01
Hips	3.13 ± 2.20	0.44 ± 0.34
Left hand	21.30 ± 19.67	0.06 ± 0.06
Right hand	17.45 ± 16.06	0.06 ± 0.10
Left elbow	69.44 ± 24.19	42.90 ± 10.23
Right elbow	67.64 ± 29.78	41.31 ± 8.67
Left foot	32.40 ± 19.00	1.58 ± 0.92
Right foot	30.07 ± 17.20	1.45 ± 0.75
Left knee	59.45 ± 33.55	23.74 ± 16.13
Right knee	60.52 ± 71.67	25.41 ± 16.13
Average	37.62 ± 24.17	13.70 ± 5.33

This section summarizes statistical results on the impact of end-to-end latency on user experience, in particular, cybersickness (see Table 30) and user performance (see Table 31).

Table 30: Statistical results for the effect of end-to-end latency on cybersickness. The results represent pairwise comparisons using Conover post-hoc test. The first value specifies the adjusted p-value (using Bonferroni correction) and the second value the effect size (Hedge's g).

	0 ms	50 ms	54 ms	58 ms	63 ms	69 ms	75 ms	83 ms	94 ms	104 ms	121 ms	150 ms
0 ms												
50 ms	1; 0.03											
54 ms	1; 0.23	1; 0.22										
58 ms	1; 0.06	1; 0.04	1; -0.16									
63 ms	1; 0.39	1; 0.40	1; 0.17	1; 0.33								
69 ms	1; 0.28	1; 0.28	1; 0.09	1; 0.23	1; -0.06							
75 ms	<0.001; 0.65	<0.001; 0.67	0.58; 0.45	<0.001; 0.60	0.99; 0.29	0.64; 0.33						
83 ms	<0.001; 1.06	<0.001; 1.10	<0.001; 0.88	<0.001; 1.01	<0.001; 0.72	<0.001; 0.72	0.72; 0.41					
94 ms	<0.001; 0.85	<0.001; 0.87	<0.001; 0.67	<0.001; 0.80	<0.001; 0.52	<0.001; 0.54	1; 0.24	1; -0.15				
104 ms	<0.001; 0.96	<0.001; 0.99	<0.001; 0.78	<0.001; 0.91	<0.001; 0.64	<0.001; 0.65	0.89; 0.35	1; -0.03	1; 0.11			
121 ms	<0.001; 1.25	<0.001; 1.29	<0.001; 1.08	<0.001; 1.20	<0.001; 0.93	<0.001; 0.92	0.01259; 0.63	1; 0.23	1; 0.37	1; 0.25		
150 ms	<0.001; 1.50	<0.001; 1.55	<0.001; 1.35	<0.001; 1.46	<0.001; 1.20	<0.001; 1.20	<0.001; 0.92	0.06; 0.54	<0.001; 0.67	0.0443; 0.55	1; 0.32	

Table 31: Statistical results for the effect of end-to-end latency on user performance. The results represent pairwise comparisons using Conover post-hoc test. The first value specifies the adjusted p-value (using Bonferroni correction) and the second value the effect size (Hedge's g).

	50 ms	54 ms	58 ms	63 ms	69 ms	75 ms	83 ms	94 ms	104 ms	121 ms	150 ms
50 ms											
54 ms	1; 0.12										
58 ms	1; 0.36	1; 0.29									
63 ms	0.48; 0.36	1; 0.30	1; 0.06								
69 ms	0.01; 0.55	0.81; 0.49	1; 0.21	1; 0.14							
75 ms	0.01; 0.49	0.66; 0.39	1; -0.03	1; -0.09	1; -0.27						
83 ms	0.02; 0.48	0.99; 0.38	1; -0.02	1; -0.08	1; -0.26	1; 0.01					
94 ms	<0.001; 0.54	0.03; 0.46	0.17; 0.09	1; 0.01	1; -0.15	1; 0.15	1; 0.13				
104 ms	<0.001; 0.93	<0.001; 0.83	<0.001; 0.26	0.01; 0.15	0.43; -0.02	0.53; 0.41	0.35; 0.38	1; 0.19			
121 ms	<0.001; 0.92	<0.001; 0.85	<0.001; 0.39	<0.001; 0.28	<0.001; 0.12	<0.001; 0.53	<0.001; 0.51	0.04392; 0.34	1; 0.20		
150 ms	<0.001; 1.26	<0.001; 1.18	<0.001; 0.58	<0.001; 0.58	<0.001; 0.26	<0.001; 0.80	<0.001; 0.76	<0.001; 0.55	0.02; 0.43	1; 0.17	

A.7 FEATURE SELECTION

Figure 38 shows the effect of feature types. We calculate the F_1 scores for a power set of all feature types (excluding the empty set). By evaluating the individual models' performance, we then find the optimal feature subset and detect when redundant features are added. The results show that we achieve the best performance ($F_1 = 0.86$) if we use position- and rotation-based features combined with dual-sensor features.

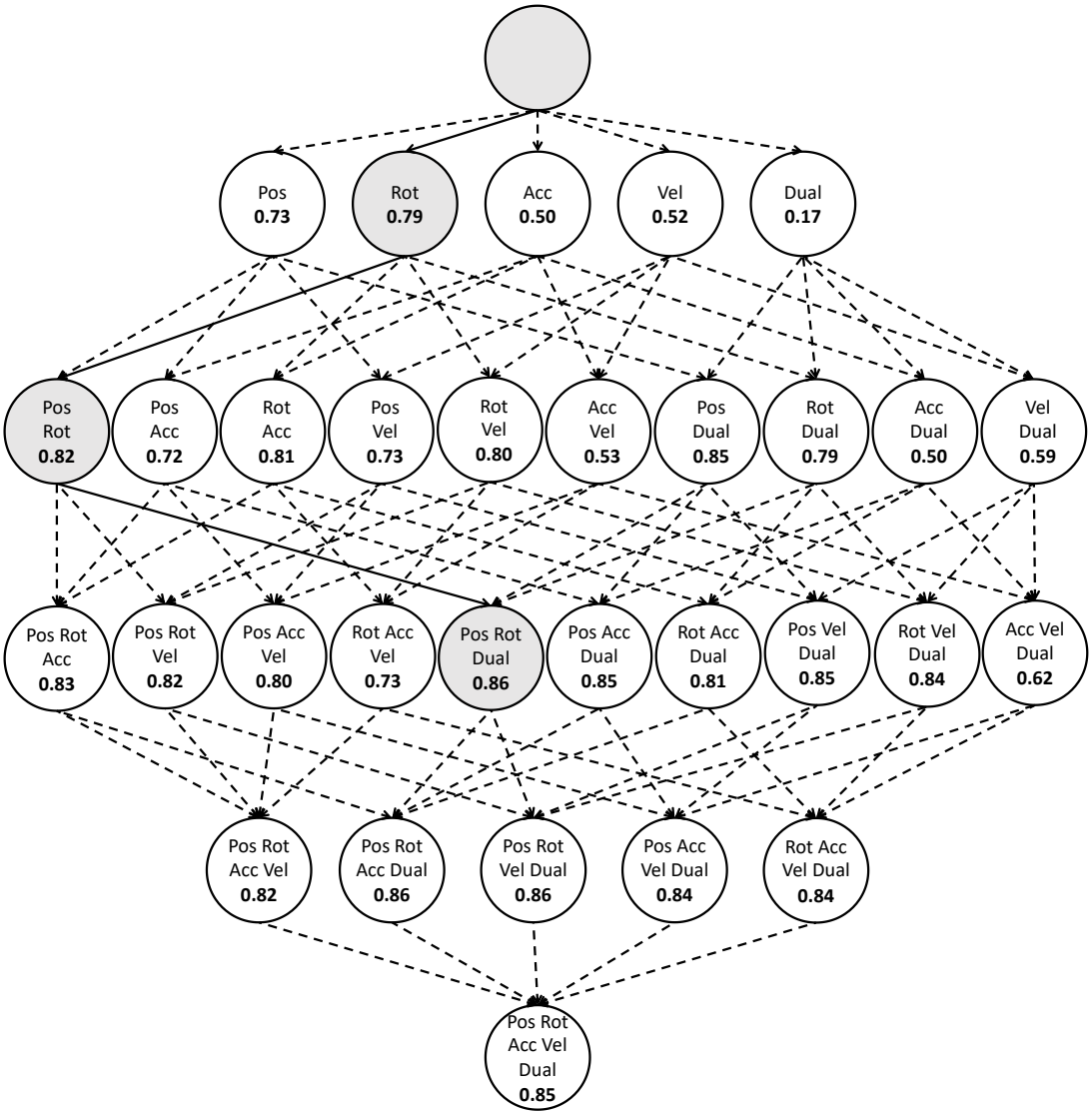


Figure 38: Macro-averaged F_1 scores for *Naïve Bayes* depending on the feature subset. Abbreviations: *Acceleration* (Acc), *Velocity* (Vel), *Position* (Pos), *Rotation* (Rot), *Dual-Sensor Features* (Dual).

A.8 PERFORMANCE OF FULL-BODY MOTION RECOGNITION

This section summarizes the performance of meta-algorithms, using the hyperparameters detailed in Table 32. The results of the base classifiers are presented in Section 5.2. *AdaBoostM1* and *Bagging* both use ensembles of the base classifiers to improve the classification rule. We use *AdaBoostM1* and *Bagging* with *J48*, *JRip*, *REP-Tree*, *Random Forest*, *Naïve Bayes*, *Bayesian Net*, and *SMO*. The results are detailed in Tables 33-34.

For all classifiers, we report the F_1 score as [100]:

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP}, \\ \text{Recall} &= \frac{TP}{TP + FN}, \\ F_1 &= 2 \cdot \frac{\cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \end{aligned} \tag{14}$$

where *TP* represents *true positives* (number of correctly classified positive samples), *FN* represents *false negatives* (number of positives incorrectly classified as negatives), *TN* represents *true negatives* (number of correctly classified negative samples), and *FP* represents *false positive* (number of negatives incorrectly classified as positives).

Table 32: Selected values for hyperparameters.

Classifier	**Hyperparameters**
Random Forest	100 iterations; at least one instance per leaf; maximum depth of 20
REP-Tree	At least two instances per leaf; 3-folds for reduced pruning, maximum depth of 20
JRip	3-folds for reduced pruning
J48	At least two instances per leaf; pruning confidence $c = 0.25$
AdaBoostM1	10 iterations
Bagging	10 iterations

Table 33: F_1 scores for relevant sensor subsets using *AdaBoostM1* as meta-algorithm.

Classifier	6 sensors	3 sensors		2 sensors						1 sensor					
	Full-body	Upper-body	Lower-body	lHand-rHand	lFoot-rFoot	lHand-lFoot	rHand-lFoot	rHand-rFoot	lHand-rFoot	Head	rHand	lHand	Hips	rFoot	lFoot
Random Forest	1.00	1.00	0.75	1.00	0.52	0.92	0.96	0.99	0.93	0.98	0.98	0.92	0.78	0.43	0.54
SMO	1.00	1.00	0.81	0.99	0.50	0.96	0.92	0.92	0.92	1.00	0.93	0.93	0.87	0.46	0.50
REP-Tree	0.97	0.97	0.79	0.92	0.52	0.89	0.87	0.91	0.89	1.00	0.95	0.91	0.73	0.40	0.41
Bayes Net	1.00	1.00	0.54	1.00	0.48	0.94	0.91	0.89	0.94	0.91	0.95	0.89	0.65	0.40	0.58
Naïve Bayes	0.95	0.95	0.57	0.99	0.48	0.93	0.90	0.90	0.86	0.88	0.93	0.87	0.54	0.50	0.58
JRip	0.95	0.93	0.75	0.93	0.56	0.83	0.86	0.72	0.80	0.98	0.87	0.86	0.73	0.47	0.44
J48	0.90	0.94	0.80	0.98	0.48	0.83	0.89	0.90	0.79	0.95	0.90	0.83	0.75	0.40	0.45

Abbreviations: *Left Hand* (lHand), *Right Hand* (rHand), *Left Foot* (lFoot), and *Right Foot* (rFoot). The *full-body* configuration includes all six sensors. The *upper-body* configuration includes the HMD and two controllers held in the hands, whereas the *lower-body* configuration includes sensors attached to the hips and both feet.

Table 34: F_1 scores for relevant sensor subsets using *Bagging* as meta-algorithm.

Classifier	6 sensors	3 sensors		2 sensors						1 sensor					
	Full-body	Upper-body	Lower-body	lHand-rHand	lFoot-rFoot	lHand-lFoot	rHand-lFoot	rHand-rFoot	lHand-rFoot	Head	rHand	lHand	Hips	rFoot	lFoot
Random Forest	1.00	1.00	0.77	0.96	0.50	0.92	0.98	0.98	0.93	0.98	0.98	0.90	0.78	0.44	0.53
SMO	1.00	1.00	0.81	0.99	0.50	0.96	0.92	0.92	0.92	1.00	0.93	0.93	0.87	0.46	0.50
REP-Tree	0.99	0.99	0.70	0.93	0.54	0.82	0.90	0.91	0.85	0.96	0.91	0.85	0.72	0.45	0.36
Bayes Net	1.00	1.00	0.58	1.00	0.49	0.96	0.91	0.87	0.94	0.91	0.96	0.90	0.64	0.43	0.58
Naïve Bayes	0.96	0.96	0.55	0.99	0.55	0.93	0.89	0.90	0.90	0.90	0.92	0.92	0.56	0.50	0.57
JRip	0.95	0.93	0.55	0.94	0.49	0.83	0.86	0.71	0.80	0.98	0.78	0.82	0.58	0.51	0.38
J48	0.99	0.99	0.63	0.95	0.47	0.82	0.90	0.90	0.86	0.97	0.90	0.87	0.68	0.42	0.44

Abbreviations: *Left Hand* (lHand), *Right Hand* (rHand), *Left Foot* (lFoot), and *Right Foot* (rFoot). The *full-body* configuration includes all six sensors. The *upper-body* configuration includes the HMD and two controllers held in the hands, whereas the *lower-body* configuration includes sensors attached to the hips and both feet.

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	100		
Warrior II		97	3
Side angle	1		99

(a) Head

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	76	23	1
Warrior II	17	82	1
Side angle	7	8	85

(b) Hips

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	82		18
Warrior II		97	3
Side angle			100

(c) Left hand

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	100		
Warrior II	1	96	3
Side angle	1	1	98

(d) Right hand

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	49	24	27
Warrior II	23	57	20
Side angle	12	16	72

(e) Left foot

Predicted class

Actual class	*Warrior I*	*Warrior II*	*Side angle*
Warrior I	36	54	10
Warrior II	53	33	14
Side angle	14	12	74

(f) Right foot

Table 35: Confusion matrices of *Random Forest*, using only one sensor.

A.9 QUESTIONNAIRES

This section includes questionnaires that have been used in user studies. We used the questionnaire in Table 36 for the user study introduced in Section 4.3.2 to evaluate the subjective quality of the reconstructed full-body avatars. Additionally, we used the questionnaire in Table 37 to assess the sense of presence during the VR training simulation for police forces, introduced in Section 6.1. Furthermore, we used the questionnaire in Table 38 to rate the game experience during the VR-based exergame, which we introduced in Section 6.2.

Table 36: Sense of embodiment questionnaire used to evaluate the subjective quality of the reconstructed full-body avatars.

Item	Question	Category
1.	I felt as if the virtual body was my body.	Sense of body ownership
2.	It seemed like the virtual arms belonged to me.	Arms' body ownership
3.	It seemed like the virtual legs belonged to me.	Legs' ownership
4.	It felt as if I was controlling the movements of the virtual avatar.	Sense of agency
5.	The movement of the hands and feet seemed to be another person's movement.*	Full-body disconnection
6.	It seemed like the size of the avatar matched the real one.*	Estimation of body parts
7.	It seemed like the movements of the virtual avatar were delayed.*	Latency

All questions were responded to on a five-point Likert scale from zero (strongly disagree) to four (strongly agree). The questionnaire items are adapted from Kilteni et al. [118]. Additional elements that are specific to our simulation are marked with *.

Table 37: Sense of presence questionnaire used to evaluate the VR training simulation for police forces.

Item	Question	Factor
1.	How responsive was the environment to actions that you initiated (or performed)?	Responsiveness of actions
2.	How much delay did you experience between your actions and expected outcomes?	Delay between actions and outcomes
3.	How compelling was your sense of moving around inside the virtual environment?	Sense of moving
4.	How aware were you of events occurring in the real world around you?	Distraction due to events in real world
5.	How aware were you of your display and control devices?	Distraction due to the display and control devices
6.	How aware were you of the additional trackers?*	Distraction due to the trackers
7.	How much did the visual display quality interfere or distract you from performing assigned tasks or required activities?	Display quality
8.	How distracting was the control mechanism?	Distraction due to control mechanism
9.	How well could you concentrate on the assigned tasks or required activities rather than on the mechanisms used to perform those tasks or activities?	Mechanism quality
10.	How much did your experiences in the virtual environment seem consistent with your real-world experiences?	Realism of the training experience
11.	The sound effects contributed to a more realistic experience.*	Realism of the sound effects
12.	To what degree did you feel confused or disoriented at the beginning of breaks or at the end of the experimental session?	Confusion/disorientation during breaks and at the end
13.	Were you involved in the experimental task to the extent that you lost track of time?	Lost track of time

All questions were responded to on a five-point Likert scale from zero (not at all) to four (extremely). The questionnaire items are adapted from Witmer and Singer [227]. Additional elements that are specific to our simulation are marked with *.

Table 38: *Game experience questionnaire* used for the evaluation of the VR-based exergame.

Item	Question	Component
2.	I felt skillful.	Competence
10.	I felt competent.	
15.	I was good at it.	
17.	I felt successful.	
21.	I was fast at reaching the game's targets.	
3.	I was interested in the game's story.	Sensory and imaginative immersion
12.	It was aesthetically pleasing.	
18.	I felt imaginative.	
19.	I felt that I could explore things.	
27.	I found it impressive.	
30.	It felt like a rich experience.	
5.	I was fully occupied with the game.	Flow
13.	I forgot everything around me.	
25.	I lost track of time.	
28.	I was deeply concentrated in the game.	
31.	I lost connection with the outside world.	
22.	I felt annoyed.	Tension/ Annoyance
24.	I felt irritable.	
29.	I felt frustrated.	
11.	I thought it was hard.	Challenge
23.	I felt pressured.	
26.	I felt challenged.	
32.	I felt time pressure.	
33.	I had to put a lot of effort into it.	
7.	It gave me a bad mood.	Negative affect
8.	I thought about other things.	
9.	I found it tiresome.	
16.	I felt bored.	

Continued on next page

Table 38: *Game experience questionnaire* used for the evaluation of the VR-based exergame (continued).

Item	Question	Component
1.	I felt content.	Positive affect
4.	I thought it was fun.	
6.	I felt happy.	
14.	I felt good.	
20.	I enjoyed it.	

All questions were responded to on a five-point Likert scale from zero (not at all) to four (extremely). The questionnaire items are taken from IJsselsteijn et al. [108]. The item numbers indicate the order of the questions.

A.10 LIST OF ACRONYMS

HMD	Head-Mounted Display
VR	Virtual Reality

A.11 SUPERVISED STUDENT THESES

[1] Philipp Achenbach. "Implementierung eines IK-Ansatzes zum Tracken von Körperbewegungen unter Berücksichtigung verschiedener Aspekte der Wahrnehmung." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[2] Md Saiful Ahmad. "Disaster Training Simulator based on Mixed Reality." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[3] Sergej Alexeev. "Erkennung von Körperbewegungen in einem VR-basierten Exergame." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[4] Sven Appel. "Analysing Position Error in Full-Body Tracking." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2020.

[5] Benedikt Böhning. "Analyse von Ganzkörperbewegungen in VR-basierten Serious Games." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[6] Oliver Eschrich. "Analyse von Datenhandschuhen für immersive Handsteuerungen in Virtual Reality." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[7] Lars Hieronymi. "Entwicklung eines VR-basierten Exergames mit mehreren Eingabemöglichkeiten zur Steuerung." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[8] Daniel Jente. "Analysis of Methods for Locomotion in a Virtual Reality based Serious Game." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[9] Shule Liu. "Recognition of Full-body Movements in Virtual Reality using Hidden Markov Models." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[10] Philipp Malkmus. "Automatische Generierung urbaner 3-D Modelle aus geotopographischen Basisinformationen." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[11] Michelle Meinhardt Martinussen. "Analysis of the Latency Perception in Virtual Reality Applications." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[12] Martin Möller. "Concepts and Methods for Locomotion in Virtual Reality." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[13] Alexander Josef Müller. "Entwicklung eines Infrarot-Trackers zur Erkennung von Körperbewegungen in Virtual Realities." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

[14] Jan Thorsten Neitzel. "Entwicklung von adaptiven Ansätzen zur Steigerung der Spielerleistung in einem Exergame." Bachelor Thesis. Darmstadt, DE: Technical University of Darmstadt, 2020.

[15] Philip Schmidt. "Entwicklung einer immersiven VR-Mehrspieleranwendung für Polizeieinsatztraining." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[16] Roman Uhlig. "Analysis of Machine Learning Approaches for Motion Recognition in Virtual Reality." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2019.

[17] Hongtao Zhang. "Analysis of 3D Storytelling in Virtual Reality." Master Thesis. Darmstadt, DE: Technical University of Darmstadt, 2018.

B

AUTHOR'S PUBLICATIONS

MAIN PUBLICATIONS

[1] Polona Caserman, Philipp Achenbach, and Stefan Göbel. "Analysis of Inverse Kinematics Solutions for Full-Body Reconstruction in Virtual Reality." In: *IEEE 7th International Conference on Serious Games and Applications for Health*. SeGah 2019. Kyoto, Japan: IEEE, 2019, pp. 1–8. DOI: 10.1109/SeGAH.2019.8882429.

[2] Polona Caserman, Miriam Cornel, Michelle Dieter, and Stefan Göbel. "A Concept of a Training Environment for Police Using VR Game Technology." In: *Serious Games*. Vol. 11243. JCSG 2018. Lecture Notes in Computer Science. Darmstadt, Germany: Springer International Publishing, 2018, pp. 175–181. DOI: 10.1007/978-3-030-02762-9_18.

[3] Polona Caserman, Augusto Garcia-Agundez, Alvar Gámes Zerban, and Stefan Göbel. "Cybersickness in Current-Generation Virtual Reality Head-Mounted Displays: Systematic Review and Outlook." In: *Virtual Reality* (2021), pp. 1–19. DOI: 10.1007/s10055-021-00513-6.

[4] Polona Caserman, Augusto Garcia-Agundez, and Stefan Göbel. "A Survey of Full-Body Motion Reconstruction in Immersive Virtual Reality Applications." In: *IEEE Transactions on Visualization and Computer Graphics* 26.10 (2020), pp. 3089–3108. DOI: 10.1109/TVCG.2019.2912607.

[5] Polona Caserman, Augusto Garcia-Agundez, Robert Konrad, Stefan Göbel, and Ralf Steinmetz. "Real-time Body Tracking in Virtual Reality Using a Vive Tracker." In: *Virtual Reality* 23.2 (2018), pp. 155–168. DOI: 10.1007/s10055-018-0374-z.

[6] Polona Caserman and Stefan Göbel. "Become a Scrum Master: Immersive Virtual Reality Training to Learn Scrum Framework." In: *Serious Games*. Vol. 12434. JCSG 2020. Lecture Notes in Computer Science. Stoke-on-Trent, UK: Springer International Publishing, 2020, pp. 34–48. DOI: 10.1007/978-3-030-61814-8_3.

[7] Polona Caserman, Katrin Hoffmann, Philipp Müller, Marcel Schaub, Katharina Straßburg, Josef Wiemeyer, Regina Bruder, and Stefan Göbel. "Quality Criteria for Serious Games: Serious Part, Game Part, and Balance." In: *JMIR Serious Games* 8.3 (2020), e19037. DOI: 10.2196/19037.

[8] Polona Caserman, Patrick Krabbe, Janis Wojtusch, and Oskar von Stryk. "Real-Time Step Detection Using the Integrated Sensors of a Head-Mounted Display." In: *IEEE International Conference on Systems, Man, and Cybernetics*. SMC 2016. Budapest, Hungary, 2016, pp. 3510–3515. DOI: 10.1109/SMC.2016.7844777.

[9] Polona Caserman, Shule Liu, and Stefan Göbel. "Full-Body Motion Recognition in Immersive Virtual Reality-based Exergame." In: *IEEE Transactions on Games* (2021), pp. 1–10. DOI: 10.1109/TG.2021.3064749.

[10] Polona Caserman, Michelle Martinussen, and Stefan Göbel. "Effects of End-to-end Latency on User Experience and Performance in Immersive Virtual Reality Applications." In: *Entertainment Computing and Serious Games*. Vol. 11863. ICEC-JCSG 2019. Lecture Notes in Computer Science. Arequipa, Peru: Springer International Publishing, 2019, pp. 57–69. DOI: 10.1007/978-3-030-34644-7_5.

[11] Polona Caserman, Philip Schmidt, Thorsten Göbel, Jonas Zinnäcker, André Kecke, and Stefan Göbel. *Impact of Full-Body Avatars in Immersive Multiplayer Virtual Reality Training for Police Forces*. Submitted for publication in IEEE Transactions on Games.

[12] Polona Caserman, Thomas Tregel, Marco Fendrich, Moritz Kolvenbach, Markus Stabel, and Stefan Göbel. "Recognition of Full-Body Movements in VR-Based Exergames Using Hidden Markov Models." In: *Serious Games*. Vol. 11243. JCSG 2018. Lecture Notes in Computer Science. Darmstadt, Germany: Springer International Publishing, 2018, pp. 191–203. DOI: 10.1007/978-3-030-02762-9_20.

[13] Polona Caserman, Hongtao Zhang, Jonnas Zinnäcker, and Stefan Göbel. "Development of a Directed Teleport Function for Immersive Training in Virtual Reality." In: *11th International Conference on Virtual Worlds and Games for Serious Applications*. VS-Games 2019. Vienna, Austria: IEEE, 2019, pp. 1–8. DOI: 10.1109/VS-Games.2019.8864599.

CO-AUTHORED PUBLICATIONS

[14] Augusto Garcia-Agundez, Ann-Kristin Folkerts, Robert Konrad, Polona Caserman, Stefan Göbel, and Elke Kalbe. "PDDanceCity: An Exergame for Patients with Idiopathic Parkinsons Disease and Cognitive Impairment." In: (2017). URL: https://dl.gi.de/handle/20.500.12116/3298.

[15] Augusto Garcia-Agundez, Ann-Kristin Folkerts, Robert Konrad, Polona Caserman, Thomas Tregel, Mareike Goosses, Stefan Göbel, and Elke Kalbe. "Recent Advances in Rehabilitation for Parkinsons Disease with Exergames: A Systematic Review." In: *Journal of Neuroengineering and Rehabilitation* 16.1 (2019), pp. 1–17. DOI: 10.1186/s12984-019-0492-1.

[16] Augusto Garcia-Agundez, Christian Reuter, Hagen Becker, Robert Konrad, Polona Caserman, André Miede, and Stefan Göbel. "Development of a Classifier to Determine Factors Causing Cybersickness in Virtual Reality Environments." In: *Games for Health Journal* 8.6 (2019), pp. 439–444. DOI: 10.1089/g4h.2019.0045.

[17] Augusto Garcia-Agundez, Christian Reuter, Polona Caserman, Robert Konrad, and Stefan Göbel. "Identifying Cybersickness Through Heart Rate Variability Alterations." In: *International Journal of Virtual Reality* 19.1 (2019), pp. 1–10. DOI: 10.20870/IJVR.2019.19.1.2907.

[18] Augusto Garcia-Agundez, Aiko Westmeier, Polona Caserman, Robert Konrad, and Stefan Göbel. "An Evaluation of Extrapolation and Filtering Techniques in Head Tracking for Virtual Environments to Reduce Cybersickness." In: *Serious Games*. Vol. 10622. JCSG 2017. Lecture Notes in Computer Science. Valencia, Spain: Springer International Publishing, 2017, pp. 203–211. DOI: 10.1007/978-3-319-70111-0_19.

[19] Robert Konrad, Polona Caserman, Stefan Göbel, and Ralf Steinmetz. "Conceptual Approach Towards Recursive Hardware Abstraction Layers." In: *Serious Games*. Vol. 10622. JCSG 2017. Lecture Notes in Computer Science. Valencia, Spain: Springer International Publishing, 2017, pp. 284–295. DOI: 10.1007/978-3-319-70111-0_26.

EDITED BOOKS

[20] Stefan Göbel, Augusto Garcia-Agundez, Thomas Tregel, Minhua Ma, Jannicke Baalsrud Hauge, Manuel Oliveira, Tim Marsh, and Polona Caserman, eds. *Proceedings of the 4th Joint International Conference on Serious Games*. JCSG 2018. Darmstadt, Germany: Springer, 2018. ISBN: 978-3-030-02761-2. DOI: 10.1007/978-3-030-02762-9.

[21] Stefan Göbel, Robert Konrad, Florian Mehm, Polona Caserman, Thomas Tregel, and Augusto Garcia-Agundez. "Serious-Games-Forschung und -Lehre in der Informatik." In: *Games studieren – was, wie, wo?* 2019, pp. 613–656. DOI: 10.14361/9783839440322-042.

C

ERKLÄRUNG LAUT PROMOTIONSORDNUNG

§8 Abs. 1 lit. c PromO

Ich versichere hiermit, dass die elektronische Version meiner Dissertation mit der schriftlichen Version übereinstimmt.

§8 Abs. 1 lit. d PromO

Ich versichere hiermit, dass zu einem vorherigen Zeitpunkt noch keine Promotion versucht wurde. In diesem Fall sind nähere Angaben über Zeitpunkt, Hochschule, Dissertationsthema und Ergebnis dieses Versuchs mitzuteilen.

§9 Abs. 1 PromO

Ich versichere hiermit, dass die vorliegende Dissertation selbstständig und nur unter Verwendung der angegebenen Quellen verfasst wurde.

§9 Abs. 2 PromO

Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Darmstadt, 16. März 2021

Polona Caserman