DESIGN LENSES FOR 3D USER INTERFACES IN EXTENDED REALITY

by

Po-Wen Yao

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(COMPUTER SCIENCE)

May 2024

Dedication

To my family, my Ph.D. Thesis Advisor Michael Zyda, and my Ph.D. Committee. My friend and editor for this thesis, Sean Zimmerman. Also, to my girlfriend Shitong Shen, who is one of the reasons why the Ph.D. took as long as it did, but I also wouldn’t have made it without her.

Acknowledgements

Thank you to everyone who has helped me on my journey. Including, but not limited to the following:

PhD Thesis Advisor Michael Zyda

My PhD Defense Committee members Jernej Barbic, Andrew Nealen, and Heather Culbertson.

My PhD Qualification Committee members Jernej Barbic, Azad Madni, Heather Culbertson, and Paul Debevec

Professors who helped me grow Vangelis Lympouridis, Souti (Rini) Chattopadhyay, Chris Swain, Laird Malamed, Scott Easley, Joseph Olin, Jeremy Gibson, Peter Brinson, Richard Lemarchand, St. John Colon, and Cynthia Woll.

My fellow PhD Labmates at the GamePipe Lab Marc Spraragen, Balki (Balakrishnan) Ranganathan, Jerry Lin, Fotos Frangoudes, Tian Zhu, David Young, and Brandon Booth.

Sean Zimmerman and Jonathan Ko for editing assistance. Shitong Shen and Inkarat Mittongtare for art assistance. Tungying Liu for being my sounding board.

USC Academia support Julie Michele Slayton for assistance with the Institutional Review Board at USC. Computer Science Ph.D. Advisors at the Viterbi Engineering School Lizsl A. De Leon, Asiroh Cham, and Andy Shangson Chen.

Toward the Stars team Design: Christopher Barakian, Alex De Bont, Dominic Ricci Engineering: Tian Zhu, Kevin Chiang, Richard Chou, Byron Choy, Neetu George, Tim Huang, Tianyu Lang, George Li, Brandon Nahigian, Vidhi Sampat, Satyajeet Sasmal, Yongxiang Tang, Derek Truong, Elsie Tu, Austin Von Spiegel, Jeff Xiao, Huayu Yang Art: Kyna Sherman, Eric Andrade, Murilo Ottoni Costa, Dillon Gu, Kai Jiang, Inkarat Mittongtare, Fernando Olmedo, Juliana Ouyang, Brittany Rotstad, Kyle Swinney, Steven Walther Sound: Bob Lydecker, Eric Pratt. Production: Amber Baluch, Jonathan Chu, Ailsa Chiu, Rui Zeng Special Thanks: Mike Zyda, St. John Colon, Fotos Frangoudes, Waqas Hussain, Sandy Appleoff, Brooke Hubert, Bryce Kho, David Landau, Travis McLain, Sean Saleh, Woodhouse Team, TJ Darcy, Brianna Lei, Josie Noronha, Evan Stern, Matthew Torres, Yu-chi Wang

Arkology team Core Members: Yen Kuo Kao, Shou Yi Hou, Siqi Guo, Qing Mao, Huanyuan Huang Supporting Members: Shitong Shen, Ting Gong, Subhayu Chakravorty, Jian Zhang, Inkarat Mittongtare, Jeff Xiao Contributors: Guangxi Jin, Divyanshu Bhardwaj, Yue Wang, Nicolas Lai Music: Camden Barkley

XROS UI team XrInVr pod: Huijun Jiang, Hanyuan Xiao, Hyuck Ju Kim, Hong Liu, Feiyu Miao, Ruoxi Jia, Xinyi Lin

Machine Learning pod: Sloan Swieso, Andrew Zhao, Mark Miller, Adityan Jothi Carnival pod: Michael Du, Raegan Brown

SmartHome NLP pod: Yu (Hope) Hou, Huanpu Hu, Yuan (Tiger) He, Cheng Da TUI pod: Chi Xu, Chao Cao, Wenhuang Huang, Tian Yang

CV pod: Satya Naraparaju, Pranavi Jalapati.

General pod: Yangzhen Zhang, Michael Yuen, Vinay Gupta. Especially, Tian Yang with the collaborative work on Flick Typing and taxonomy and Jackie Ye with the collaborative work on Extradimensional Storage.

Table of Contents

List of Tables

List of Figures

Abstract

Arthur C. Clarke's Third Law states, "Any sufficiently advanced technology is indistinguishable from magic." With Extended Reality (XR), we move ever closer to magic-like interactions. However, the current state of the art continues to be dominated by prior computing paradigms, which do not take full advantage of XR's unique properties. Techniques such as three-dimensional user interfaces and Natural User Interfaces allow for intuitive XR interactions. Still, they can inconvenience users with their strict adherence to real-world rules for naturalness.

Hyperphysical User Interfaces, which do not follow the rules of real-world physics, have been used to address these limitations. This allows the creation of novel and efficient interaction techniques across many more contexts. However, there is no systematic approach to creating hyperphysical interactions. Furthermore, users lack a clear way to access, manage, or customize their repository of techniques.

This thesis presents a systematic and practical approach to the design and development of user interfaces and user interactions in XR that feels like Clarke's magic. The theory portion presents a model for spatial interaction, three categories of design lenses (Hyperphysical User Interface, Whole-body Interaction, and Extradimensional Space), and a gesture taxonomy that serves as a road map for gesture design. A series of exploratory prototypes (School of Spatial Sorcery, Virtual Equipment System, and Extradimensional Space Storage) demonstrate the theory's practical applications. Together, the theory and prototypes guide the design of novel hyperphysical interfaces for any scenario and techniques to access the best tools for the situation.

Chapter 1

Introduction

ch1:Introduction

For the latest work, please visit `http://powenyao.com/`

1.1 Personal Background and Inspiration

ch1sec:PersonalBackground

In 2016, Professor Michael Zyda, my Ph.D. advisor, excitedly shared with us Ph.D. students at the GamePipe Lab on the advancement in Virtual Reality (VR). He exclaimed that the HTC Vive headsets were revolutionary and nothing like the VR devices he had seen before. Not having seen the new generation of Virtual Reality headsets at work, I was initially skeptical. I couldn't help but wonder: how much better could it be compared to what our Advanced Game Projects class students were doing with their VR setup?

After actually getting hands-on with the HTC Vive headsets, I was convinced. The new generation of VR allows you to see and interact in three dimensions while being commercially available and relatively affordable. Seeing the potential of VR, I decided to switch my research area from game artificial intelligence to learning the ins and outs of virtual reality.

I started to work on converting the real-time strategy sandbox environment I was building for my Game AI research to a Virtual Reality Real-Time Strategy game. With some friends' help, we built the Virtual Reality Real-Time Strategy game Arkology. It was showcased as an indie game at the Taipei Game Show (Taipei), Indiebox (Osaka), and the University Student Game Showcase at the Game Developers

Conference (San Francisco). However, before submitting any demo or paper to an academic conference, I had to take a leave of absence and return to my home country, the Republic of China, to serve in the mandatory military service.

After finishing my military service, I returned to USC to continue my Ph.D. work. It just so happened that Professor Zyda was teaching a class on this very subject - CSCI 538 AR/VR/MR class along with Professor Vangelis Lympouridis. I became the TA for the class and worked to understand the field better. I had to catch up on the rapidly advancing area of Extended Reality (XR), which includes Augmented Reality (AR) and Mixed Reality (MR) in addition to VR.

I have discovered that in the initial excitement of the return and start of a new VR era, many jumped on the VR bandwagon, and ideas and techniques flourished. As time passed, people realized that the adoption of VR would be slow and many challenges lie ahead before major commercial successes and returns on investment. Community interests and investments slowed along with the advancement.

Although XR technology has matured over the past years, the user interface (UI) and user experience (UX) remain under-explored. Many games and experiences tackle the low-hanging fruit of emulating real-life experiences and are thus restricted in their designs. Much less effort has been spent exploring how VR can go beyond mimicking reality. Innovative ideas may appear in one VR title but not be utilized again. For example, major VR titles have not utilized the space around the head as storage space since Fantastic Contraption in 2017. While using the head as a storage space may create fatigue and injury to the neck in real life, the same does not have to be true for virtual objects stored around the head.

My work aims to explore and push XR UI and UX beyond the familiar. This work will help developers create better experiences that are only possible in XR as we explore what's beyond the limits of reality.

1.2 Motivation

ch1sec:Motivation

1.2.1 Extended Reality and its Qualities

Extended Reality (XR) is an umbrella term that includes Augmented Reality (AR), Mixed Reality (MR), Virtual Reality (VR), and other potential classifications within the reality–virtuality continuum[92]. Regardless of whether the user interacts with the subset of AR, MR, or VR technology, a few qualities are emerging and becoming common to XR. The motivation behind this work is to realize the potential that Extended Reality brings.

In XR, the user exists in and is part of the computing environment. Instead of looking at a display device affixed somewhere in the world, visuals are delivered directly to the users' eyes. The user can also directly interact with the environment using their own body, extensions of their body (e.g. motion controllers), and other tools driven by their body (e.g. a virtual ray interactor). In addition to this active participation from the user in interactions, tracking devices can also measure passive participation from the user. Pulse, mood, tone, and other often signals generated unconsciously from the user can be factors in human-computer interaction (HCI). While these user-tracking technologies are not exclusive to XR, they are gaining popularity and have a more prominent role in XR compared to other computing paradigms.

The user's interactions are also becoming spatial in nature. Instead of interacting with an input device (such as a mouse or keyboard) located in a specific fixed location, users are part of the interaction environment and have access to a wide variety of interactors. These interactors can include the head, hands, motion controllers in place of hands, or other body parts. Users can rotate their heads to see the world in three-dimensional (3D) space. Users' hands (or motion controllers) can also interact with the world directly in 3D space. That is to say, users are not limited to a specific location fixed in space but are free to move around in 3D space as needed.

Similar to how touchscreen and GPS technology are not exclusive to mobile computing but greatly shape its paradigm, the user being part of the computing environment and the 3d spatial interaction greatly shape the XR computing paradigm.

1.2.2 XR Future

With users being part of the interaction environment and having access to the third dimension, we can envision a future powered by XR that enables the users to:

- Interact Anywhere - For Convenience
- Interact Any Way - For Health
- Interact Spatially - For Efficiency
- Interact Hyper-physically - For Adaptability

Interact Anywhere Users will not be restricted to interactions in front of a computer, such as a cubicle at work. They will be able to interact while walking or commuting on public transport. As long as users have their XR devices, they can pick up and resume their computing activities anywhere.

Interact Any Way As users are not bound to a physical location, they can move and adjust to a more ergonomic or comfortable pose. Users can use the computing environment to fit their needs in various situations, such as standing, sitting, walking, or lying down. The standard recommendations for an office worker with tables and monitors at specific heights would differ in XR and must be updated to reflect the new computing paradigm.

Interact Spatially Users can interact with spatial content directly without navigating through 2D representations of 3D spatial content. For example, in contemporary 3D computer graphics applications, seeing the side view of a 3D model (e.g. a teapot) would involve using a series of keystrokes or mouse interactions. In XR, users can instead change their perspective using heads and bodies, or use their hands to move and rotate the model. These actions might involve more effort but are intuitive and simple. Working spatially allows for intuitive interaction as the actions are directly applied to the content.

Interact Hyperphysically Users can perform actions that involve hyperphysicality, which allows them to perform actions they cannot otherwise do in real life. A classic example of hyperphysical interactions in the traditional PC environment would be Ctrl+Z to undo previous actions. Like Ctrl+z, hyperphysical interactions can offer more efficient interactions that users wish they could do in real life.

1.3 State of XR Interaction

ch1sec:stateOfXr

The current state of the XR Interactions could be described as the Wild West. Whereas WIMP (Window, Icon, Menu, Pointer) Graphical User Interface dominates Personal Computers, there is still a long way to go before a universal user interface for XR emerges.

Compared to a traditional Desktop environment where the users need to adapt to the system, users in XR are part of the XR system. Newer VR Devices (E.G., HP Reverb G2 Omnicept[*], DecaGear[†], Apple Vision Pro[‡]) are incorporating familiar technology into the headset to track your heart rate, eye movement, or facial expressions.

At the same time, new approaches and integrative methods enable Wearables to track users in novel ways. Some examples include miniature cameras on headphones to track facial contours and determine the users' facial expressions[24]. Companies, such as Meta, and scholars[68, 72] are working on technology that can track the users' hands through a wristband utilizing surface electromyography and machine learning. Another example is the Amazon Halo Band, which performed tone of voice analysis in addition to the typical step, pulse, and sleep tracking.

These emerging immersive technologies provide users with additional opportunities and space for interaction. However, this additional space remains under-explored and under-utilized. In this thesis, I provide a framework and tools for future designers to take advantage of this interaction space.

---

[*] https://www.hp.com/us-en/vr/reverb-g2-vr-headset-omnicept-edition.html

[†] https://www.deca.net/

[‡] https://www.apple.com/apple-vision-pro/

1.4 XR and Its Many Contexts

ch1sec:Contexts

Users in Extended Reality can find themselves in a variety of situations. These situations can be based on the physical environment, the user's body posture, social contexts, as well as the interactive content within the computing environment.

1.4.1 Virtual Reality: Physical Space and Body Postures

Commercial VR devices typically require users to perform an initial room setup before using the device. During setup, the user must choose between a stationary or a room-scale experience, which impacts the physical space available to the VR experiences. This is to ensure a safe and enjoyable experience for the user. This is reflected in Virtual Reality content stores. Stores categorize their content based on physical space and posture[§]. Some stores may further break down the stationary category into sitting posture and standing posture. [¶]. It is worth noting that these store categories reflect the general trend and exist for customer convenience. They should not be treated as restrictions on design nor that they cover all use cases.

Sitting, Standing, and Room-Scale are the three main ways of interacting in commercial virtual experiences for home users. They have significantly shaped the design of user interfaces and user experiences of commercial products. First, we'll discuss sitting and standing, two different postures where the user is assumed to be stationary.

Stationary - Sitting For the sitting posture, VR headset vendors recommend that users sit on an office swivel chair. In practice, the user may sit anywhere, such as on a couch, bed, or the floor. This affects their eye level and ease of access to peripersonal space (the space within arm's reach), which in turn can impact interaction and user interface design.

[§] In Room setup, the player can set up a stationary or a room-scale experience, `https://www.vive.com/us/support/vive/category_howto/what-is-the-play-area.html`

[¶] Meta Store explicitly lists "Sitting, Standing, and Roomscale" as supported player modes `https://www.meta.com/help/quest/articles/accounts/purchasing-apps/support-player-modes-rift/`

Stationary - Standing The standing posture is similar to sitting in that the user is expected to be stationary, with the potential of occasionally taking a step from the starting location.

While the two stationary postures sound similar, there are many differences to consider when designing the user interface. The designers should adjust the virtual environment and interfaces to accommodate the differences in the user's head height, the reach of the user's hands in accessing content, potential issues with the user colliding with real-world objects, assumed level of physical activity, and many other traits[170].

Room-scale Room-scale refers to having an area (such as a 3-meter by 3-meter or larger space) that can be used for interaction and is free from obstructions or obstacles. Within the room-scale space, the user can freely move and interact within that space as if they were in the real world. Room-scale provides a balance between having realistic movement and the restriction of having limited physical space to dedicate to XR interactions. Research such as Redirected Walking[102, 137, 94] convinces the user that they are experiencing their movement around the space in one way while moving around in the space in another way. For an extreme example, users may be convinced that they are walking in straight lines but are actually walking in circles. This allows the user to get more out of the use of limited physical space as if the space were larger or even unrestricted.

Arena-scale While the prior three are the typical setups available to most users, there are also commercial experiences for "arena-scale"[‖] VR, where the user interacts in a much bigger area than Room-scale. These arena-scale experiences typically involve multiple users in the same physical space, in contrast with the VR experiences at home where the user is typically alone. This impacts interface design and safety concerns due to the potential for in-person physical interactions between users in Arena-Scale. This is in contrast with VR multiplayer experiences at home where every user is remote and there is no risk of collision with other players.

[‖] `https://www.xrvrcade.com/arena-scale`

Stationary - Lying Down Although not listed as a category for content, recumbent or lying down on a bed or couch is another posture for users to interact with VR[55]. Common activities for users lying down include meditation or watching video content. While lying down is a stationary posture like sitting and standing, it has very different implications for the design of the user interface and user interaction.

For starters, users who are lying down would prefer what they see to be displayed in front of their eyes, just like sitting and standing, but now the content would be above the users' heads (on the ceiling). Users need tools to define a preferred orientation for the content or a new 'up' direction. More so than sitting or standing, users would not want the content to be located behind their heads as it would be difficult to view or interact with those content. If using motion controllers, the users' arms are more restricted when compared to upright postures. The users' arms would also get tired more easily if they had to hold it up to interact with the content in front of their eyes above their body.

More recent updates to VR frameworks (SteamVR and OculusVR) allow users to redefine and reorient where they are looking within the VR itself. However, it only supports easily changing the view without support to better interact with the content that is now on the ceiling.

1.4.2 Mobile Augmented Reality - Walking, Driving, Riding

Due to Virtual Reality's inability to easily see or hear the user's physical surroundings, most Virtual Reality experiences are confined to a relatively safe and private space. In contrast, Augmented Reality and Mixed Reality may involve travel, uncertain environments, and public spaces. In terms of traveling, the user may be walking, operating a vehicle, or riding along as a passenger. There is user interface research into Mobile Augmented Reality for walking[82]. Research is also being done to produce a more efficient user interface for the driver to operate a vehicle[56]. As a passenger, the user may seek to be entertained [64] or try to catch up on work while commuting [85].

1.4.3 Unusual Environments - Space and Underwater

Scholars have proposed using Augmented Reality in space, such as Holo-SEXTANT[7] (an AR overlay to help astronauts navigate paths in low visibility environment) or Space Walker used for mission training[48]. Some researchers utilize the neutral buoyancy of an underwater environment and Virtual Reality to simulate zero gravity in space[130]. There is also a series of research focused on building a Virtual Reality system to be used underwater[32, 33].

1.4.4 XR for Animals

There are even Virtual Reality displays developed for animals such as mice[111], zebrafish[70], honey bees[123], nonhuman primates[43], Augmented Reality for cows to improve milk production[129], fruit flies[23], and many other types of animals[100]. These works suggest the potential for the design principles in this work to also apply to animals; however, this thesis will focus on human-centric interactions.

1.4.5 Different content

Virtual Reality allows the simulation of our reality or an alternate reality. Augmented Reality tries to augment or enhance our existing reality. As such, anything conceivable could involve extended reality. Some examples are listed below:

- Entertainment [106]
- Virtual Events (Concerts, sports, experiences, etc)
- Sports
- Games (e.g., Beat Saber, Walking Dead: Saints & Sinners, Half-Life: Alyx, etc.)
- Health [120]
- Fitness (e.g Y.U.R, supernatural, etc)
- Rehabilitation
- Meditation
- Training

- Medical Training and Simulation (e.g. Osso VR)
- Retail (e.g., IKEA [104])
- Work
- Group Collaboration (vSpatial, MeetinVR, etc)
- Social via Social platform (e.g. VRChat, Rec Room, Meta Horizon World)

The wide variety of content means it's difficult to have a one-shoe-fits-all approach. For simulation of real-life activities, fidelity to real life is likely the highest priority, whereas, for a Virtual Reality experience involving magic and fantastic settings, it only needs to be familiar enough for the player to learn to play and enjoy the experience.

1.5 Existing XR Interaction Techniques

With the need to adapt to the ever-expanding landscape of XR, it is easy to see why developers and researchers have created a plethora of techniques to improve user interfaces, user interaction, and user experience.

In the comprehensive textbook, "3D User Interfaces: Theory and Practice" [83], 3D interaction techniques are divided into three categories:

- Selection and Manipulation
- Travel
- System Control

In Travel (or Locomotion) techniques, we see that researchers have proposed many techniques. Locomotion Vault [36] is a database that focuses on recording and preserving different locomotion techniques, tracking attributes such as what kind of hardware is involved, whether the technique fits their definition of magical or not, and so on. As of the writing of this thesis, it tracks 109 different locomotion techniques. These 100+ listed locomotion techniques may just be the tip of the iceberg. Prinz et al. [114] identified 29

Locomotion Techniques	**Immersion**	**Effort**	**Difficulty**
Limited Room-scale	High	High	Easy
Walk-in-place	High	High	Easy
Joystick-driven	Medium	Low	Medium
Controller-based Teleportation	Low	Low	Medium

Table 1.1: Comparison of different locomotion techniques by immersi tab:LocomotionComparison

publications with a taxonomy for locomotion techniques, with seven of these taxonomies having a major impact in the field. The field of VR locomotion is far from being fully developed, mapped, and classified.

The same applies to the entertainment side of VR. Many commercial VR games have adopted either limited Room-scale movement, walk-in-place using controllers, joystick-driven locomotion, or controller-based teleportation. These locomotion techniques range on the spectrum of realistic to hyperphysical interaction and are often offered together to allow for user preference.

Limited Room-scale movement is the most realistic and requires no effort to learn, but it restricts the user to a limited physical space in Virtual Reality. Walk-in-place locomotion simulates human walking, but with the lack of full-body tracking, it's generally based on arm motion. Though intuitive, the motion becomes tiring with extended use. Joystick-driven locomotion mimics human walking and is familiar to video game players, but users are more prone to motion sickness. Teleportation can alleviate Virtual Reality sickness but is unrealistic. It can be deeply disruptive to experiences that lean toward fidelity to reality, such as breaking player immersion in horror games or in training and simulation. All in all, there is unlikely to be a universal standard for locomotion, just techniques that best suit the given scenario.

1.6 Brief Summary

ch1sec:briefSummary

XR interactions involve many different factors for consideration, such as contexts or situations. In terms of content, XR is similar to the desktop computing paradigm; the content exists for almost every aspect of everyday life, from watching a movie to writing a PhD thesis. In terms of environments, XR is similar to mobile computing, where the user could be in various postures, situations, social contexts, and locations. What truly sets XR apart from other computing paradigms is the immersive technology that allows users to interact with their full body in 3D space. This brings XR ever closer to interactions in real life. At the same time, XR can allow for interactions that are impossible in real life. Thus, there is a need for new user interfaces and novel paradigms for this new interaction space. These three qualities of the hyperphysical user interface, whole-body interaction, and extradimensional space are what we will examine in Chapter 2

1.7 Thesis Road Map

ch1sec:Roadmap

This thesis covers a large body of work that is shown in Figure 1.1 and is structured as follows:

Chapter 1 introduces Extended Reality, its current state, and unique challenges relevant to this work.

Chapter 2 covers user research in Extended Reality as well as prior work on the concept and use of design lenses itself. It also covers related user interface research in the design lenses of Hyperphysical User Interface, Whole-body Interaction, and Extradimensional Space.

Chapter 3 primarily covers the designs and concepts. Chapter 3a defines the Spatial Interaction Model, this work's three classes of design lenses, and how they work together.

Chapter 3b examines a gesture taxonomy and associated system architecture, which were created based on prior works through our design lenses. Chapter 3c discusses concepts and prototypes in the game-like sandbox environment known as the School of Spatial Sorcery. Chapter 3d and Chapter 3e discuss specific prototypes designed and implemented based on the design lenses. Chapter 3d covers the Virtual Equipment System, an egocentric equipment framework. Chapter 3e covers Extradimensional Space Storage, an inventory system that heavily utilizes the concepts of Extradimensional Space.

Chapter 4b covers methodologies and evaluation of the prototype implementations of Virtual Equipment System.

Section 5.1 covers the data, results, and analysis of the Virtual Equipment System.

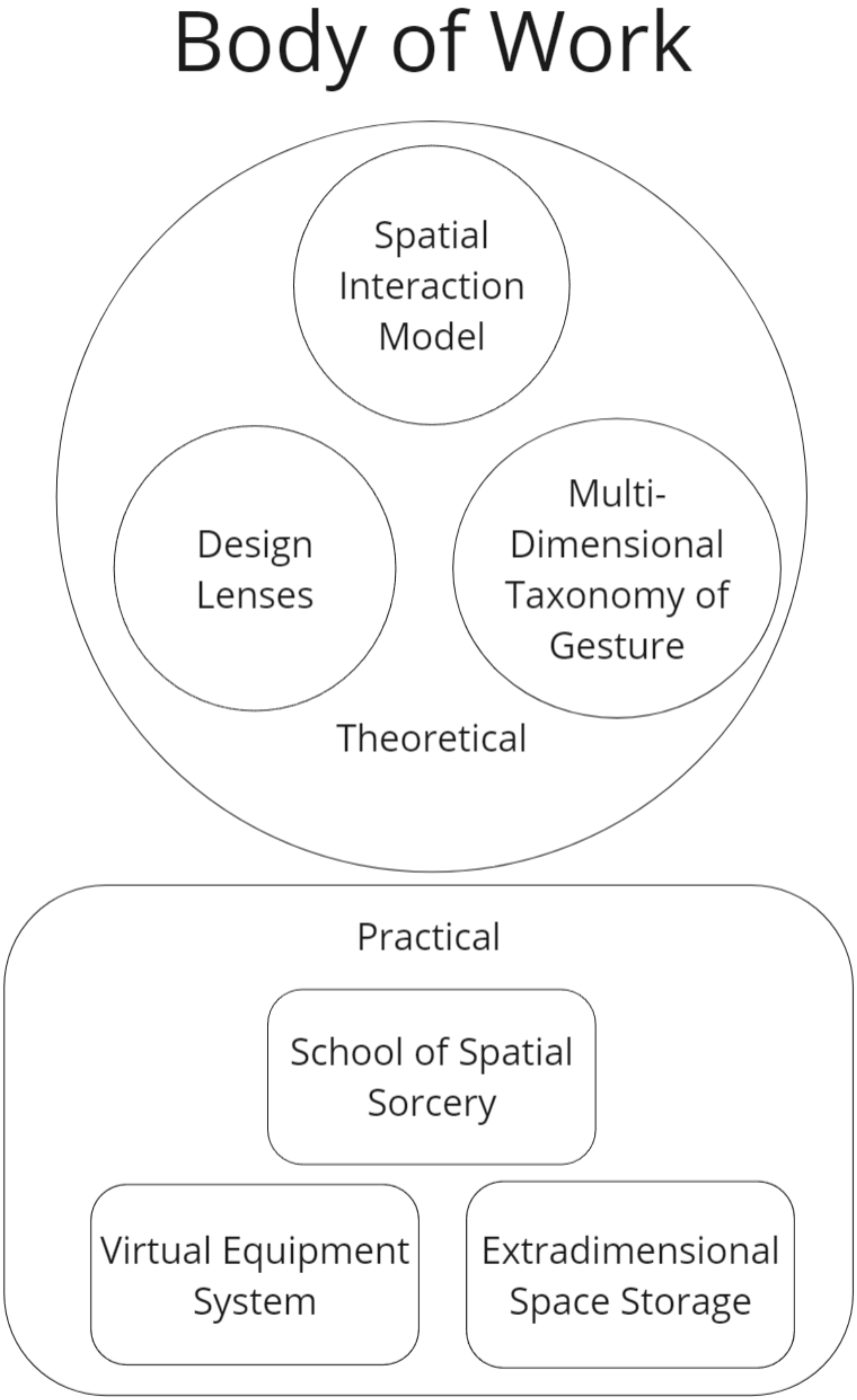


Figure 1.1: This figure depicts the body of work discussed in this thesis. Circle shape indicating theoretical work and round square indicating implementations fig:bodyofwork

Chapter 6 discusses future work.

Chapter 7 summarizes this thesis and discusses future direction.

1.8 List of Publications

ch1sec:Publications

The following publications contributed to this thesis and the respective areas they belong to. Publications in bold are works where I am the primary author. This typically means that I was responsible for the majority of the implementation, design, experiment, and conduction.

- Taxonomy
  - **Powen Yao, Tian Yang, and Michael Zyda. "Toward a Gesture System Architecture in Extended Reality Based on a Multi-dimensional Taxonomy of Gestures". In: *International Conference on Human-Computer Interaction.* Springer. 2023, pp. 340–347**
- Virtual Equipment System
  - **Powen Yao, Tian Zhu, and Michael Zyda. "Adjustable Pointer in Virtual Reality for Ergonomic Interaction". In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2020, pp. 828–829**
  - **Powen Yao, Tian Zhu, and Michael Zyda. "Designing Virtual Equipment Systems for VR". in: *International Conference on Human-Computer Interaction.* Springer. 2020, pp. 137–144**
  - **Powen Yao et al. "Interfacing with Sensory Options Using a Virtual Equipment System". In: *Symposium on Spatial User Interaction.* 2020, pp. 1–2**
  - **Powen Yao, Vangelis Lympouridis, and Michael Zyda. "Virtual Equipment System: Expansion to Address Alternate Contexts". In: *International Conference on Human-Computer Interaction.* Springer. 2021, pp. 353–360**

  - **Powen Yao, Vangelis Lympouridis, and Michael Zyda. “Virtual Equipment System: Face Mask and Voodoo Doll for User Privacy and Self-Expression Options in Virtual Reality”. In: *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2021, pp. 747–748**
  - **Powen Yao, Shitong Shen, and Michael Zyda. “Virtual Equipment System: First Evaluation of Egocentric Virtual Equipment for Sensory Settings”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I.* Springer International Publishing Cham. 2022, pp. 131–149**
- Extradimensional Space Storage
  - **Powen Yao, Zhankai Ye, and Michael Zyda. “Virtual Equipment System: Toward Bag of Holding and Other Extradimensional Storage in Extended Reality”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I.* Springer International Publishing Cham. 2022, pp. 113–130**
- Text Entry
  - **Powen Yao et al. “Punch Typing: Alternative Method for Text Entry in Virtual Reality”. In: *Symposium on Spatial User Interaction.* 2020, pp. 1–2**
  - Tian Yang, Powen Yao, and Mike Zyda. “Flick Typing: Toward A New XR Text Input System Based on 3D Gestures and Machine Learning”. In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2022, pp. 888–889

  - Tian Yang, Powen Yao, and Michael Zyda. “Flick Typing: A New VR Text Input System Based on Space Gestures”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I*. Springer International Publishing Cham. 2022, pp. 379–392

- Smart Home / Simulated Environment (Precursor to School of Spatial Sorcery)
  - Zezhen Xu, Powen Yao, and Vangelis Lympouridis. “Virtual Control Interface: A System for Exploring AR and IoT Multimodal Interactions Within a Simulated Virtual Environment”. In: *International Conference on Human-Computer Interaction*. Springer. 2021, pp. 345–352
  - **Powen Yao et al. “Toward Using Multi-Modal Machine Learning for User Behavior Prediction in Simulated Smart Home for Extended Reality”. In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE. 2022, pp. 688–689**
  - **Powen Yao et al. “Using Multi-modal Machine Learning for User Behavior Prediction in Simulated Smart Home for Extended Reality”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I*. Springer International Publishing Cham. 2022, pp. 94–112**
- Machine Learning Explorations
  - Sloan Swieso et al. “Toward Using Machine Learning-Based Motion Gesture for 3D Text Input”. In: *Proceedings of the 2021 ACM Symposium on Spatial User Interaction*. 2021, pp. 1–2

- Adityan Jothi et al. “Toward Predicting User Waist Location From VR Headset and Controllers Through Machine Learning”. In: *Proceedings of the 2021 ACM Symposium on Spatial User Interaction.* 2021, pp. 1–2
- Mark Miller et al. “Virtual Equipment System: Toward Peripersonal Equipment Slots with Machine Learning”. In: *Symposium on Spatial User Interaction.* 2021, pp. 1–2
- Pranavi Jalapati et al. “Integrating Sensor Fusion with Pose Estimation for Simulating Human Interactions in Virtual Reality”. In: *HCI International 2022–Late Breaking Papers: Interacting with eXtended Reality and Artificial Intelligence: 24th International Conference on Human-Computer Interaction, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings.* Springer Nature Switzerland Cham. 2022, pp. 74–87

Chapter 2

Fundamentals and Previous Work

ch2:Fundamentals

2.1 Introduction to Fundamentals and Related Work

ch2sec:IntrofundamentalsAndRelatedWork

In this chapter, we will first establish the terminology Section 2.2 used in this work. Next, we'll examine related work in Hyperphysicality, Whole-body Interaction, and Extradimensional Space. Finally, we'll examine relevant XR research and the use of design lenses in the literature.

2.2 Terminology

ch2sec:Terminology

This section lists the definitions and abbreviations for terminologies used in this work.

Interaction in Extended Reality (XR) is deeply linked with Human-Computer Interaction (HCI); however, there are no standards in the terminologies used in HCI to describe some novel concepts that exist in XR. By providing an established terminology for this work, it will allow better communication of the ideas and concepts in this thesis.

The section is divided into Common Terminology and Unique Terminology. Common Terminology is defined using prior works when possible. This ensures the terminology is consistent with the literature and has a clear definition. Unique Terminology is a collection of new terms introduced to describe the concepts and ideas in this work.

2.2.1 Common Terminology

ch2sec:commonTerminology

Human-Computer Interaction (HCI)

> "A field of study that examines all aspects of the interplay between people and interactive technologies. One way to think about HCI is as the process of communication between human users and computers (or interactive technologies in general). Users communicate actions, intents, goals, queries, and other such needs to computers. Computers, in turn, communicate to the user information about the world, about their internal state, about the responses to user queries, and so on. This communication may involve explicit dialog, or turn-taking, in which a user issues a command or query, the system responds, and so on, but in most modern computer systems, the communication is more implicit, free form, or even imperceptible." [83]

User Interface (UI)

> "The medium through which the communication between users and computers takes place. The UI translates a user's actions and state (inputs) into a representation the computer can understand and act upon, and it translates the computer's actions and state (outputs) into a representation the human user can understand and act upon." [83]

User Experience (UX)

> "A broader concept encompassing a user's entire relationship with an artifact, including not only usability but also usefulness and emotional factors such as fun, joy, pride of ownership, and perceived elegance of design." [83]

(Six) Degree of Freedom (DOF/6DOF)

> "The number of independent dimensions of the motion of a body. DOF can be used to describe the input possibilities provided by input devices, the motion of a complex articulated object such as a human arm and hand, or the possible movements of a virtual object." [83]

This thesis deals with 6DOF XR, where the user is tracked in terms of positional and rotational motion. In 6DOF XR, three degrees of freedom come from positional motion along the x, y, and z axes. Similarly, three degrees of freedom come from rotational motion around the x, y, and z rotational axes. The three degrees of freedom from positional motion combined with the three degrees of freedom from rotational motion are combined for six degrees of freedom total. In contrast, some XR systems only have 3DOF, as they only track the rotational position of pitch, yaw, and roll.

3D Interaction

> "Human–computer interaction in which the user's tasks are performed directly in a real or virtual 3D spatial context."[83]

3D User Interface (3DUI)

> "A UI that involves 3D interaction." [83]

Natural User Interface (NUI)

> "A NUI is not a natural *user interface*, but rather an interface that makes your user act and feel like a natural. An easy way of remembering this is to change the way you say "natural user interface"—it's not a *natural* user interface, but rather a *natural user* interface" [146]

Supernatural User Interface (SNUI)

> "Supernatural user interfaces (SNUIs) are interfaces which are still inspired by the ways humans interact with one another or with their environment, but not limited by it. SNUIs permit actions which are not possible in the physical world. Examples would include teleportation or floating interface elements. Since virtual realities allow developers or users to set the rules within a world, the way users interact with virtual environments (VEs) can also be supernatural. Natural interaction can still inspire these interactions, however, they are less limited by natural constraints." [87]

Virtual Environment

> "A synthetic, spatial (usually 3D) world seen from a first-person point of view. The view in a virtual environment is under the real-time control of the user." [83]

Reality-Virtuality Continuum A spectrum where the Reality side consists of solely real objects and the Virtuality side consists of solely virtual objects. Augmented Reality leans toward the reality side, Mixed Reality (where real world and virtual world objects are presented in a single display) in the middle, and Virtual Reality leans toward the Virtuality side. [92]

Augmented Reality (AR)

> An approach that uses displays, tracking, and other technologies to enhance (augment) the user's view of a real-world environment with synthetic objects or information.[83]

Virtual Reality (VR)

> An approach that uses displays, tracking, and other technologies to immerse the user in a VE. Note that in practice VE and VR are often used almost interchangeably. [83]

Mixed Reality (MR)

> A set of approaches, including both VR and AR, in which real and virtual information is mixed in different combinations. A system's position on the mixed reality continuum indicates the mixture of virtuality and reality in the system (with the extremes being purely virtual and purely real). Mixed reality systems may move along this continuum as the user interacts with them [83]

The above definition is in turn based on Milgram et al. [92]

Extended Reality (XR) Extended Reality or XR is a term that describes any reality on the reality-virtuality continuum[92]. It is used as a substitute to describe the combination of Augmented Reality, Virtual Reality, Mixed Reality, and other realities. The X is intended to be a substitute for any of the letters or combination of letters that describes different realities (e.g. AR/VR/MR).

Head-Locked Content Head-locked Content has a fixed spatial relationship with the display. * The content's position and rotation do not change in the user's head's local coordinate system.

Body-Locked Content

> An attached frame of reference moves with the user as they walk around, with a fixed heading defined when the app first creates the frame. This lets the user comfortably look around at content placed within that frame of reference. Content rendered in this user-relative way is called body-locked content.

"Body-Locked" may appear as "Body-Stabilized," "Body-Fixed," or "Body-anchoring" in other works. "Content" may appear as "HUD," "Content," "text", or "User Interface (UI)" in other works.

World-Locked Content World-locked content is an attached frame of reference that does not move with the user as they walk around. It is often contrasted with Body-Locked Content.

The same concept may also be described as World-Anchoring or exocentric.

Personal Space The area immediately bordering the body.

Peripersonal Space or Near Space The area that is within the reach of the user's hands.

Extrapersonal Space or Far Space The area that is beyond the reach of the user's hands, but can be reached with the assistance of tools.

Proprioception Proprioception is the human sense that provides information about the position and angles of the body joints without visual information.

Interactor An entity that is used to produce an effect.

Some examples include the user's body parts (such as fingers, hands, head), physical hardware (motion controllers), and tools (Grabber,

---

* `https://docs.microsoft.com/en-us/windows/mixed-reality/design/coordinate-systems`

Interactable An optional entity that the interactor may perform its action on. It may also be referred to as interactee in some work. An interactor can also serve as an interactable to a different interactor.

Some examples include traditional UI elements (buttons, sliders, etc), objects, or even empty spaces (Zones).

Voodoo Doll Technique Originally defined by Pierce et al. as

> Use of a humanoid shape object to represent the user interaction.[110]

In this work, the definition of the Voodoo Doll Technique is expanded. The Voodoo Doll Technique makes use of a humanoid-shaped object as a surrogate interactable, typically representing the user. Interactions performed on the Voodoo Doll will apply their effects to the represented object.

Surface Gesture A gesture involving interactions with the surface of an interactable using an interactor, such as moving a stylus on a tablet's surface or fingers on a touchscreen.

Motion Gesture A gesture involving interactions by moving an interactable in 3D space, such as rotating a phone screen 90 degrees to switch between landscape and portrait mode.

Crossing-based User Interface Crossing-based User Interface uses an interactor moving across a boundary of a representation[2]. It has its roots in 2D interaction and is contrasted with point-and-click. Whereas point-and-click involves entering a boundary and performing a separate click action to achieve an effect, crossing-based involves passing through and leaving a boundary in a specific way to cause an effect.

Windows, Icons, Menus, Pointer (WIMP) The most common computing paradigm that makes use of Windows, Icons, Menus, and Pointers for its user interface and interaction.

2.2.2 Unique Terminology

ch2sec:UniqueTerminology

This section contains terminology that has specific usage in this work. This includes terminology that others have not defined before, as well as existing technology that is used differently.

Hyperphysical/Hyperphysicality Qualities of physical laws and phenomena that are not found in our own physical world.

“Hyperphysical” may also appear as “supernatural,” “hyper-natural,” or “magical” in the literature.

Hyperphysical User Interface Hyperphysical User Interface is a user interface that has qualities of hyperphysicality. In other words, user interface with qualities of physical laws and phenomena that are not found in our own physical world. An example would be a zone (empty space) that will provide different effects based on the direction one enters from, exits to, or a combination of both.

Hyperphysical Interface Properties Hyperphysical Interface Properties are the properties of a user interface with a virtual manifestation that follows physical laws and phenomena that are not necessarily found in our physical world.

Whole-body Interaction Ergonomics Whole-body Interaction Ergonomics are the measurable properties when using the body as an interface. As its name suggests, it differs from traditional human factors or ergonomics in its focus on the user’s whole body.

Virtual Equipment A virtual object that provides functionality to the user. In this work, it is implied to be Egocentric Equipment unless explicitly stated.

Egocentric Equipment Objects that belong to and follow the user, often because they are physically attached to the user, though this is not required.

Real-life examples include a kitchen apron, a watch, a holstered gun, etc.

Exocentric Equipment Objects that do not follow the user when the user moves. As the user moves, they remain stationary and and fixed to the world. Real-life examples include a kitchen oven, a bookcase, a television, etc.

Personal Equipment Egocentric Equipment that exists in the user’s personal space. In other words, equipment that is immediately next to the user’s body.

With the proximity, personal equipment often appears to be physically attached to the body. It's a specific type of Egocentric Equipment alongside Peripersonal and Extrapersonal Equipment.

Peripersonal Equipment Egocentric Equipment that exists in the user's peripersonal space. In other words, equipment that is within the user's hand reach.

Extrapersonal Equipment Egocentric Equipment exists in the user's extrapersonal space. In other words, equipment that is within the user's tool reach.

Object Gesture Gestures that can be performed on or performed with an object, whether the object is real or virtual.

Equipment Gesture Gestures that can be performed on or performed with a piece of equipment. This is a subset of Object Gestures.

Interaction Zones Interaction Zones (or Zones) refer to an empty area in which **Dimension** Dimension is a loaded term that are used in many ways. While this term has common connotations in mathematics referring to spatial axes (such as the x, y, and z axes for position or rotation), this work makes use of another common usage as defined below. With regards to sections involving Extradimensional Space in Chapter 3e, a dimension is the sum of all space and time that are connected physically.

With regards to sections involving the multi-dimension taxonomy of gestures in Section 3b.5, a dimension is a collection of categories, similar to enumerated types.

Extradimensional Space (EDS) Extradimensional space describes a physical space that is inaccessible by normal physical traversal. They must be accessed using specific extradimensional interaction techniques that are meant to interact with the extradimensional space.

Extradimensional Space Storage (EDS Storage) Extradimensional Space Storage (or simply as Extradimensional Storage) is storage that can store more items than the physical space the container occupies by linking it to another dimension.

Extradimensional Interaction Techniques Extradimensional Interaction Techniques allow the user to interact with extradimensional space. An example would be using portals that link two dimensions together.

Dimensional Shift Dimensional Shift describes the phenomenon where the user experiences multiple parts of the same dimension or multiple dimensions overlapping spatially.

2.3 Related Work - Hyperphysical User Interface

ch2sec:RelatedHyperphysical

Many works make use of hyperphysicality without explicitly discussing hyperphysicality as a concept or how they came to their hyperphysical design. The interaction techniques are often presented as they are. Occasionally, the authors may mention inspiration from popular media such as the go-go interaction technique[113] based on the animated TV series Inspector Gadget and his phrase "go-go gadget." This thesis addresses how hyperphysicality can be used on the Spatial Interaction Model in Section 3a.1 as a lens for design in Section 3a.3.

A small sample of how hyperphysicality is used in literature follows.

2.3.1 Enabling Novel Interaction Space with Hardware

Creating hardware that enables interactions that were not possible previously. For example, having helmets that respond to touch [78], belts[41], pockets[42], wristbands[40], or headsets [58][59].

2.3.2 Manipulating Space and Time

Other works deal with manipulating space and time, such as Spacetime: Enabling Fluid Individual and Collaborative Editing in Virtual Reality[151]. More specifically, many researchers[86, 94, 116, 125] study the impact of teleportation and portal techniques on the users in terms of metrics such as immersion, comfort, or ease of use.

2.3.3 Surrogates and Metaphors

Designers often draw upon the power of metaphors to help users understand and utilize interaction techniques. These metaphors may involve surrogate interactors, interactables, or actions. Understanding the different ways we can use metaphors and surrogates can help with interaction design and is discussed in detail in Section 3b.8.

2.3.3.1 Using Surrogate Interactors

Some works deal with using a surrogate interactor to perform actions on behalf of another interactor. For example, using arms in place of legs[8] or using fingers in place of legs[124][†].

2.3.3.2 Using Surrogate Objects / Interactables

There are works that deal with using a surrogate object for interaction, such as interacting with a voodoo doll that is a proxy for a distant object[110, 109]. Another type is based on the World in Miniature metaphor[132]. A miniature representation of the world is used as a proxy for the world to allow for interactions such as object selection, navigation, and visualization.

2.3.3.3 Using Surrogate Actions

Instead of performing the intended action, some research seeks to use surrogate actions such as cycling[51] or walking in place[21] to simulate the user walking around the virtual world.

2.3.4 Redirected Touching & Walking

Redirected Touching[80] and Redirected Walking [102] [137] create a discrepancy between the real world and the virtual world to achieve certain effects. One effect of Redirected Touching is for the user to think that there are more physical objects for the user to grab than there are in real life. For example, the user

---

[†] https://twitter.com/pushmatrix/status/1227302127862734849

is able to grab three objects located at different positions in the virtual environment when there is really just one object in the physical environment. As the user tries to grab each virtual object, different visuals are presented to guide the user to grab the same physical object. For Redirected Walking, the user may be convinced that they are walking in a straight line when they are walking in circles, which allows a large walkable virtual space to be simulated in a small walkable physical space.

2.3.5 Research that advocates for hyperphysicality

There are a few works that specifically argue for the incorporation of hyperphysicality in research, such as "To Mimic Reality or to Go Beyond?“Superpowers” in Virtual Reality, the Experience of Augmentation and Its Consequences" [101], and "Developing Embodied Familiarity with Hyperphysical Phenomena"[34]. These works, however, do not provide guidelines on how to utilize hyperphysicality.

2.4 Related Work - Whole-body Interaction

ch2sec:RelatedWholeBody

2.4.1 Whole Body Tracking

Much research has been done on tracking the entirety of the user’s body, such as using hardware trackers, computer vision, radio frequency, or Wi-Fi.

A common but more expensive solution to the end user is to utilize hardware trackers with embedded sensors to track the user’s body position and movement in 3D space. This typically requires tracking the head, left hand, right hand, left feet, right feet, and waist. The rest of the body would be extrapolated using inverse kinematic.

Another solution is to use egocentric[62] or exocentric cameras and computer vision to track the user’s body.

Other user tracking techniques include radio frequency sensing and acoustic data to track gestures or fine-grained human activities such as breathing. For example, tracking head gestures with a smartwatch using millimeter-wave radar[157],tracking gestures using a pair of Wi-Fi devices[47], and facial recognition with radio-frequency identification[88].

Some examples of acoustic tracking the user include tracking breathing using acoustic data from a smartphone [66] , smart glasses that estimate upper body poses [89], tracking hands [142] , tracking fine-grained human activities (breathing, blinking, and finger tapping)[84], , and tracking vital signs [141].

These works push the boundary of which parts of the body and bodily functions can be tracked and available for interaction. This thesis builds on top of their contribution, addressing what developers can provide and what the users can do with improved and more extensive tracking technology as it becomes available.

2.4.2 Body-Centric Interaction

In this section, we discuss prior works that use body-centric interaction and how this thesis builds upon or differs from these prior works.

Pierce et al.'s "Toolspaces and glances: storing, accessing, and retrieving objects in 3D desktop applications"[108] is an early example of utilizing storage spaces attached to the user. Toolspaces are storage spaces for virtual objects and the user can access them using glance metaphors with a mouse. This thesis utilizes the same concept of egocentric storage space but makes use of 3D spatial interactors (6DOF motion controllers) in place of 2D interactors (mouse). This allows for a more natural and straightforward interaction.

Ängeslevä et al. have previously come up with the design of an interface design called Body Mnemonics as described in [10] and [9]. In Body Mnemonics, a portable device can retrieve and store information, such as apps, from the user's body space using proprioception. Surveys explored what users would store in different body locations, resulting in a classification of the four basic mapping strategies: emotional, associative, functional, and logical. After asking participants to store apps in different locations, a follow-up survey also found that participants could recall, on average, 10.2/12 of the stored apps, suggesting that such a system was feasible. This is similar to the Virtual Equipment System described in Chapter 3d in that they both use a tracked device to perform interactions around the user's body. However, their system remains in the exploratory design stage. It also does not make use of peripersonal space or extrapersonal space.

Chen et al. built a series of works based on Mobile Devices and made use of Motion Gestures in work such as "Body-centric interaction with a screen-based handheld device" [25], "Body-Centric Interaction with Mobile Devices"[26], " "Extending a Mobile Device's Interaction Space through Body-Centric Interaction"[27], and "Around-body interaction: sensing & interaction techniques for proprioception-enhanced input with mobile devices"[28]. The work of Chen et al. uses mobile phones and their motion gestures(a class of interaction techniques involving an object's position and rotation changes). This allows the user to position and orient their mobile devices to interact with user interfaces stored in the space around the body. This earlier work is limited to mobile devices and motion gestures, whereas this thesis provides a framework to use any interactables (physical objects, virtual objects, or empty space) and different classes of gestures.

Dobbelstein et al. have published a series of works that involve the use of physical input devices that can work in conjunction with wearable displays for unobtrusive interaction. This includes "Belt: An unobtrusive touch input device for head-worn displays"[41], "FaceTouch: Touch Interaction for Mobile Virtual Reality"[58][59], "PocketThumb: A wearable dual-sided touch interface for cursor-based control of smart-eyewear"[42], "SnapBand: a Flexible Multi-Location Touch Input Band"[40], "Unobtrusive Interaction for Wearable Computing"[39], and ultimately Dobbelstein's PhD Thesis "Near-body interaction for wearable interfaces"[38]. Dobblestein et al.'s work involves building novel physical input devices to support near-body interaction and demonstrating its benefits from such near-body interactions. While Dobblestein's thesis also makes use of near-body space, my thesis makes use of space beyond near-body (personal space), such as peripersonal space and extrapersonal space. Furthermore, my thesis also emphasizes novel interactions through the use of hyperphysicality on existing hardware in place of building new custom physical hardware.

In a similar vein, Khadka et al. have published a series of works[77, 74, 75, 76] that deal with creating special hardware worn on the body to explore tangible user interfaces using the body as storage. The work compares egocentric and exocentric virtual object storage techniques and finds that the egocentric technique could improve task performance. Khadka makes use of specially constructed physical hardware and focuses on storage techniques, whereas the work presented in this thesis involves the use of virtual equipment and supports a variety of interaction techniques.

2.4.3 Reach Envelope

Some research, such as "Reach envelope of a 9-degree-of freedom model of the upper extremity"[154] and "Human reach envelope and zone differentiation for ergonomic design"[153], maps out the physical space that is accessible by the user's arms, referred to as Peripersonal Space and Personal Space in this work. Whereas previous works discuss them in terms of where interactors can be for interaction, these spaces are also used as areas for user interface storage in this work. Thus, these spaces are utilized in this work as usable spaces for interaction as well as locations for interfaces.

2.5 Related Work - Extradimensional Space

ch2sec:RelatedExtradimensional

Extradimensional space (EDS) is a term often used in popular culture, fantasy works, or science fiction. When the term extradimensional space is used in academia, it describes a very different concept.

In Astrophysics[169], the term "universe" describes the sum of all space and time that we live in. Universe, dimension, plane, world, space, or reality are often used in works of fiction interchangeably without a clear distinction or definition for each. In this work, we will primarily use dimension and offer the following definition. A dimension is the sum of all space and time that are connected physically.

Fiction work posits that there are other distinct and separate sums of space and time that are not connected physically with our own. This may be described using terms such as alternate, another, extradimensional, or parallel to refer to a dimension other than our own. Some examples are a parallel universe, alternate universe, parallel dimension, pocket dimension (a parallel dimension that is much smaller in size), another plane of existence, parallel world, or alternate reality. These separate sums of space and time cannot be accessed physically. That is, no matter how much one travels in their own dimension, they will never reach the other dimension. Instead, there are alternative ways (such as magic in fantasy works or technology in science fiction) that must be used to allow one to view, access, or enter another dimension. Thus, Extradimensional space is a term used to describe a physical space that is inaccessible by normal physical traversal.

There is little research around extradimensional space as it remains a work of fiction. However, it is a useful hyperphysical concept that is often employed to build better user experiences without being specifically called out or acknowledged by the designer.

For example, A 2D menu storage system in XR could be considered to be using the concept of extradimensional space. A virtual object like a large hammer can be stored and represented in a much smaller space. The physical space that the hammer would normally take up is instead taking up space in extradimensional space storage.

Games[‡] and XR experiences also often use the concept of portals to connect physical space, which is not possible in real life. Portals allow the user to physically step through it and end up in a new location. This

---

[‡] https://store.steampowered.com/app/400/Portal/

new location could be within the same or a separate dimension. If it leads the user to another dimension, it can be considered as an extradimensional interaction technique[§].

Teleportation is often employed in Virtual Reality[16] [35] [14] . While it does not necessarily involve the use of another dimension, it allows the user to take a shortcut through space that's not possible in the real world.

Other related techniques involve the concept that will be referred to as Dimensional Shift in this work. Dimensional Shift describes the phenomenon where the user experiences multiple parts of the same dimension or multiple dimensions overlapping spatially. Schjerlund et al.'s "Ovrlap: Perceiving multiple locations simultaneously to improve interaction in vr"[127] is an example of a dimensional shift of multiple parts of the same dimension, allowing the user to simultaneously see and interact with multiple locations at once.

2.6 Related Work - General User Interface Research in XR

ch2sec:RelatedXrui

Many of the general user interfaces research in XR make use of hyperphysicality, though few specifically call out hyperphysicality as a core concept. None provides a systematical approach to creating new hyperphysical interactions. Below, we discuss a few research work and their differences to this thesis.

Mike Alger's Master's thesis "Visual design methods for virtual reality"[4], "VR interface design manifesto" [5], and "VR interface design pre-visualisation methods"[6] tackles XR UI research focuses on how to translate existing 2D paradigm in operating systems to an ergonomic user interface designed for head mounted displays. In contrast, my thesis emphasizes the use of hyperphysicality and aims to provide different techniques for different situations.

In the Master's thesis "User Experience Guidelines for Design of Virtual Reality Graphical User Interfaces"[52] by Sofia Fröjdman, Sofia focuses on a specific type of VR experience with GUI that is controlled

---

[§] https://tvtropes.org/pmwiki/pmwiki.php/Main/ExtraDimensionalShortcut by head orientation. In contrast, my thesis deals with 6DOF spatial interaction involving multiple body parts.

The Master's thesis "Supernatural and comfortable user interfaces for basic 3d interaction tasks"[87] by Paul Lubos shares similarity to this thesis in that it also argues for hyperphysical user interfaces (supernatural user interfaces). Lubos' thesis differs in that it focuses on investigating how hyperphysicality could be beneficial as applied to selection and locomotion tasks. In contrast, my thesis focuses on how one can make use of hyperphysicality to create new interaction techniques.

In the Master's thesis "Control systems in virtual reality video games"[103], Sergi Olive tackles the XR UI research by focusing on control systems in VR games. Specifically, a user study on locomotion was conducted. The thesis notes that the control systems used are heavily impacted by game genre and design. It also found that some control systems, such as picking up items and firing weapons, are standardized, but not others, such as locomotion or inventory. My work differs in that it seeks to provide a framework for the design of new interaction techniques and the usage of the best interaction techniques for the situation at hand.

2.7 Related Work - Spatial Interaction Model & Design Lenses

ch2sec:RelatedLenses

2.7.1 Spatial Interaction Model

XR interactions must deal with the many different contexts mentioned in Section 1.4. This emphasizes components that are considered unimportant or implicit in previous interaction models used in WIMP and other computing paradigms.

In "A New Model for Handling Input", Myers emphasizes the concept of interactors [99] in the model used for the Garnet System. In this model, interactors handle input, graphics, and programs separately and independently, allowing multiple input devices to operate at the same time. This is similar to the

Model-View-Controller paradigm used in the program Smalltalk[81]. Within Model-View-Controller, the concept of an interactor falls under the Controller.

The Spatial Interaction Model presented in this thesis is partly inspired by the Unity Game Engine[¶]'s XR Interaction Toolkit[‖]. The XR Interaction Toolkit provides developers with a hardware-agnostic solution to build interactions for VR and AR experiences. The toolkit describes itself as "a high-level, component-based, interaction system for creating VR and AR experiences." It provides the user with a variety of interactors to use on an even wider variety of interactables. The interactables in the toolkit are assumed to be virtual objects such as an apple or a button. This work expands the definition of interactables more loosely to cover anything that can be interacted with, including a seemingly empty space.

2.7.2 Design Lenses

This work makes use of the concept of lenses for design from the game design book by Jesse Schell [126]. In the book, Schell states

> Good game design happens when you view your game from as many perspectives as possible. I refer to these perspectives as lenses, because each one is a way of viewing your design. The lenses are small sets of questions you should ask yourself about your design. They are not blueprints or recipes, but tools for examining your design.
>
> Many other HCI works also make use of lenses for designing. For example, a series of works discusses lenses for exercise games, including but not limited to "Five lenses for designing exertion experiences."[49], "10 Lenses to Design Sports-HCI"[96]. and the subsequent "13 game lenses for designing diverse interactive jogging systems"[97].
>
> However, these works use the concept without explicitly defining the concept of design lenses or how to utilize them. Thus, we use and follow the definition offered by Schell.

[¶] https://unity.com/

[‖] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.5/manual/index.html

Some works may refer to lenses as themes, such as in "How bodies matter: five themes for interaction design"[79], or as analytical lenses, such as "Games for Change: Looking at Models of Persuasion Through the Lens of Design" [11]. In Schell's usage, however, lenses for design are more than just a short phrase or word for consideration. Lenses for design also include a short list of questions and considerations, so that a more formal approach may be applied to the design challenge at hand.

Chapter 3

Body of Work

ch3:Body

This chapter discusses the body of work that makes up this thesis. First, in Chapter 3a, we introduce a Spatial Interaction Model and three classes of Design Lenses that tie the different bodies of work together.

Then, we discuss how the model and lenses can be applied to classify gestures. This is used to create a gesture taxonomy and associated architecture for a gesture system in Chapter 3b.

In Chapter 3c, we introduce the School of Spatial Sorcery, an environment for exploring spatial interaction with the backdrop of a fantasy setting. This is where we implement and test new interaction techniques based on the design lenses.

In Chapter 3d and Chapter 3e, we discuss the specific implementation of the Virtual Equipment System and Extradimensional Space Storage. These two works were designed and built as guided by the three design lenses.

Finally, we discuss other related work that contributed to the creation of this thesis in Chapter 3f.

Chapter 3a

Spatial Interaction Model & Design Lenses

ch3:ModelandLenses

This thesis presents a Spatial Interaction Model (SIM) and offers design lenses that can be used together to guide the design of XR user interfaces.

3a.1 Spatial Interaction Model

ch3sec:SpatialInteractionModel

In Table 3a.1, we provide a definition of each of the entities in the SIM. In SIM, there are two agents. An actor agent that wants an effect to happen, and an observer agent that is capable of producing that effect. These agents can be human or AI systems.

The actor agent tries to convey its intention to the observer agent to get the observer agent to produce some effect. This intent is conveyed through an interaction. Interaction is the user's interactor's action with an optional interactable within contexts that an observer agent interprets to produce a desired effect by another agent. Put it plainly, Interactor + Action + Interactable + Context = Effect.

The observer agent observes the physical characteristics of the interaction. Now, the observer agent has to make sense of the physical characteristics to infer the intent and produce the desired effect. This is done by utilizing the entities in the Context Mapping category (further

Entity	**Description**
Physical Characteristics	Observable qualities of the entities directly involved in the interaction (action, interactor, interactable)
Interaction	The process in which interactor performs an action with an interactable within some contexts
Interactor	An entity that is used in an interaction to ultimately produce an effect
(Interactor) Action	Observable changes in the interactor's physical characteristics. More plainly, what the interactor does
Interactable	An optional entity that the interactor may perform its action on
Context Mapping	Entities that can be used to determine whether the observed physical characteristics should result in which effects
Source of Meaning	The source of knowledge that can be used to understand and derive meaning from the physical characteristics to produce the desired effect
Action Mapping	Whether the action maps performed interaction to the effect based on action
Effect Mapping	Whether the action maps performed interaction to the effect based on the effect
Target	Whether the interaction has an intended target
Context	Any considerations not included within the interactor, its action, or interactable that may affect the interpretation of the intent to produce an effect
Intent	What the Actor Agent is aiming to achieve
Actor Agent (User)	An entity that is performing an interaction to convey its intent to the observer agent to create a desired effect
Observer Agent (Audience)	The entity that interprets the interactor, its action, interactable, and context to deduce the intent based on a shared source of meaning to produce the desired effect
Interpretation	What the Observer Agent believes the Actor Agent is doing to achieve some effect
Effect	Any outcome based on the observer agent's interpretation of the intent behind the interaction

Table 3a.1: Entities in the Spatial Interaction Model

tab:SimEntities

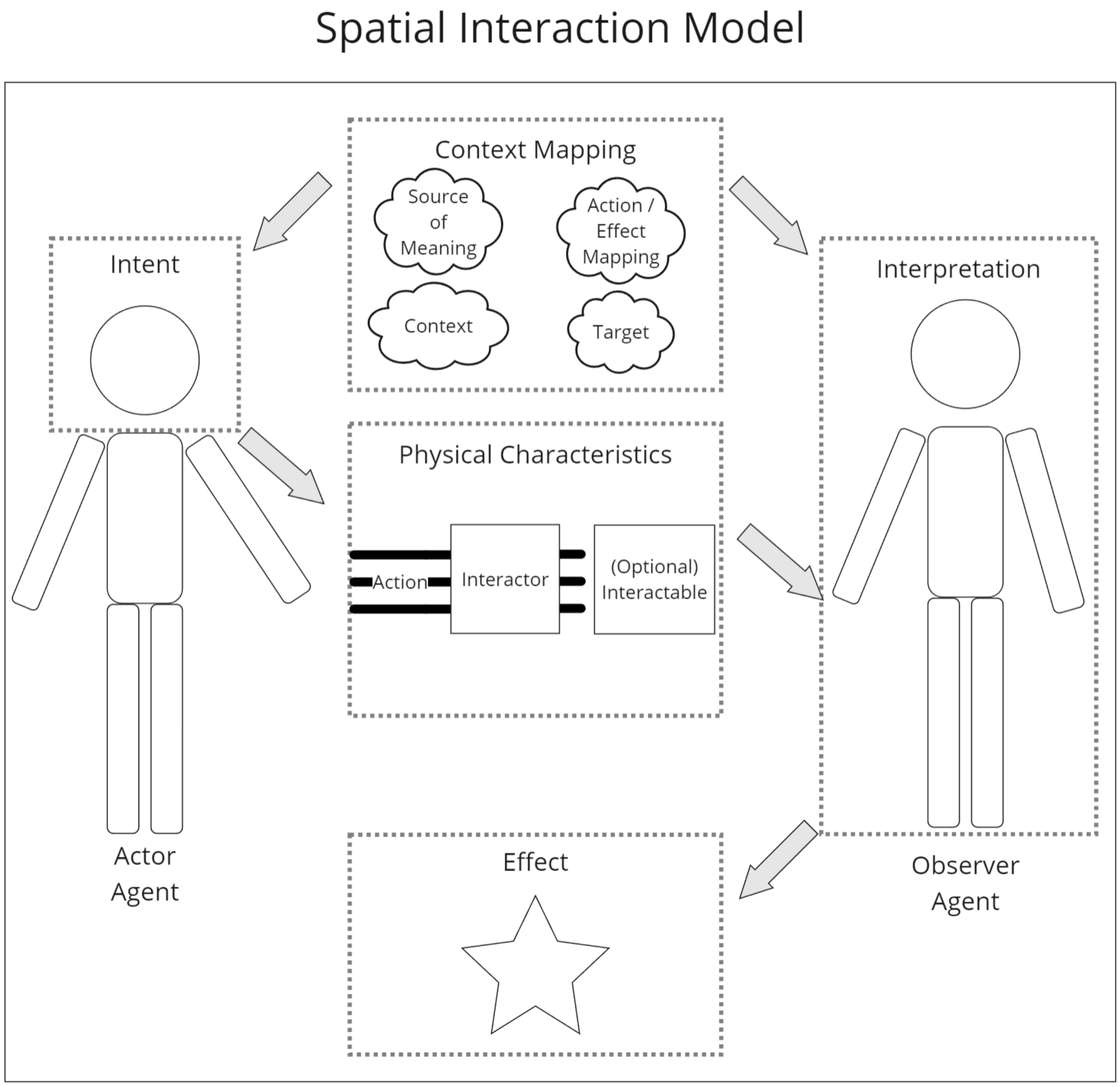


Figure 3a.1: This figure depicts the Spatial Interaction Model. It shows an interaction agent performing an action with an interactor with an optional interactable to an observer agent in order to produce some effect. The observer agent interprets the intent of the interaction agent through the shared source of meaning and contexts to determine what the actions map to

fig:Spatial Interaction Model

discussed in Chapter 3b. The observer agent will draw upon knowledge of how a gesture can map to different effects based on the context mapping.

The physical characteristics and contexts are the considerations. Source of Meaning, Action Mapping, Effect Mapping, and Target are how to interpret those considerations. The sources of meaning refer to the knowledge base that the agents can use to make sense of the interaction. Whether the source of meaning is based on physical laws, hyperphysical laws, cultural conventions, or previously agreed-upon meaning, it establishes a baseline for understanding. Other characteristics of context mapping, such as target, action mapping, or effect mapping, will allow the observer agent to further disambiguate between the potential effects and determine exactly the desired effect to produce.

This interaction model is utilized with the design lenses in creating a multidimensional gesture taxonomy as discussed in Chapter 3b.

3a.2 Lenses for Design

ch3sec:Lenses

This thesis offers three classes of Design Lenses when designing for XR User Interfaces. The three classes are Hyperphysical User Interface, Whole-body Interaction, and Extradimensional Space. Each will be discussed in the following sections to illustrate what they are and how they have been used.

3a.3 Hyperphysicality in XR

ch3sec:lensHyperphysicality

In Ivan Sutherland's classic article from 1965, "The Ultimate Display"[134], he stated, "There is no reason why the objects displayed by a computer have to follow the ordinary rules of physical reality with which we are familiar."

Given that we don't have to follow the ordinary rules of physical reality, the possible space for interaction is infinitely large. However, there are aspects of hyperphysicality that we can look at to guide user interface design in XR to be more efficient, intuitive, or fun.

3a.3.1 Aspects of Hyperphysicality

ch3sec:aspectsOfHyperphysicality

We start our journey into hyperphysicality by looking at various XR research and applications for examples of hyperphysicality as applied to different aspects of spatial interaction and in different degrees. We focus on VR games as they offer plentiful, easily accessible examples of spatial interaction applied in many different domains. We follow the example of the paper "Directions for 3D User Interface Research From Consumer VR Games"[131] by Steed et al., who have similar insights into referring to consumer VR games for reference.

Hyperphysicality can apply to different aspects of interaction:

1. **Manifestation** of the user and the physical extension of the user (e.g. motion controller)
2. **Action** of the representation to the virtual world
3. **Effects** or the outcome of an interaction in the virtual world

Hyperphysicality can also be applied to objects, the world itself, the user, and other users or agents in the virtual world.

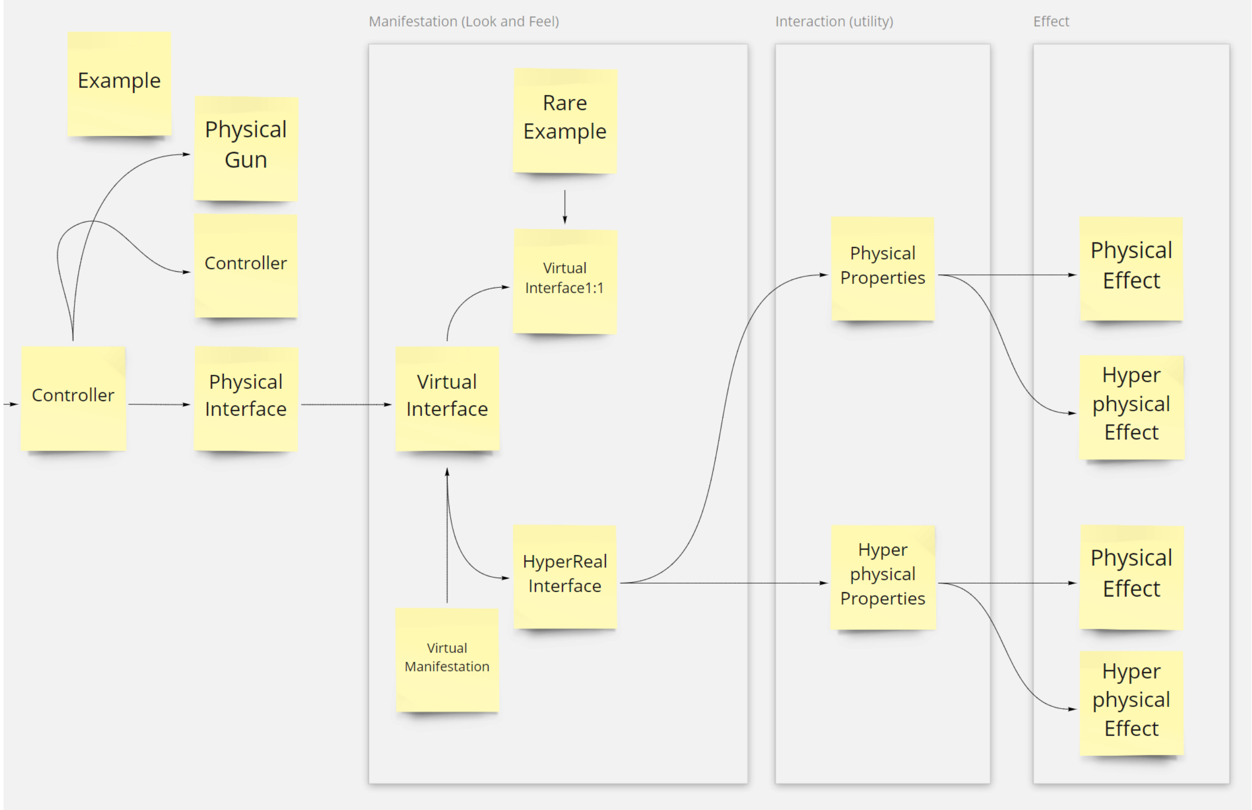


Figure 3a.2: This figure shows that hyperphysicality can be applied to either fig:hyperphysicality-diagram

3a.3.2 Hyperphysicality Aspects - Manifestation

ch3sec:examplesOfHyperphysicalityAspects

Hyperphysicality can be applied to an entity in terms of how that entity manifests in the virtual world. In other words, a virtual entity can take on a different representation than what it actually physically is, allowing the use of visual metaphors to better match its virtual representation to its function. For example, the user's controller could appear as a fork and a knife while eating.

3a.3.2.1 Using Hardware Accessories for Real Physicality

For most VR games, the 6DOF motion controllers serve as the users' main means of interaction with the virtual world. While these controllers can take on many shapes, they often differ from their virtual counterparts. That is to say, while a controller may look and function in one way in real life, its virtual counterpart can look and function very differently in the virtual world.

Having the form match the function can further add to realism and immersion. Thus, some equipment makers provide accessories for motion controllers, such as a gun attachment. These hardware accessories help the form match the function not just in visual appearance but in tactile feel. In the case of Sony's PlayStation Move Controller, as shown in Figure 3a.3, the accessory provides the user with a physical item that looks and feels like a gun, albeit a toy gun.

In the case of the gun controller attachment for HTC Vive's motion controller, as shown in Figure 3a.4, the motion controller retains all its previous functionality, but the controller attachment turns it into a gun shape. There are no additional hardware changes to the input, just a modification to the controller that mimics the ergonomics of a real-life firearm.

Figure 3a.3: This figure depicts a Sony PlayStation Move Shooting Attachment. The attachment helps the PlayStation Move Controller to feel and handle more like a pi fig:hyperphysicality-hardware-accessories

Figure 3a.4: A Rifle Gun Attachment for HTC Vive Motion Controller. The attachment helps the HTC Vive Motion Controller to feel and handle more like a rifle fig:hyperphysicality-hardware-accessories2

3a.3.2.2 Hyperphysical Virtual Representations

Although physical representations can be modified with accessories to better match the virtual representations, it is not common due to cost and the variety of virtual representations desired.

Instead, most games use the most common form of hyperphysicality, which is having a virtual representation that looks and functions differently than the real-life motion controller.

The most popular VR game, Beat Saber, showcases an example of hyperphysicality in its controller representation. As shown in Figure 3a.5, the motion controllers in real life have been transformed into blue and red sabers in VR. These virtual sabers allow the user to reach and slice through objects without having a physical blade on the controller.

Figure 3a.5: A mixed-reality image of the VR game Beat Saber. It shows a woman holding motion controllers in real life that have been transformed into blue and red sabers in virtual reality fig:beat-saber

3a.3.2.3 Transformative Nature of Representation

In many VR games and applications, we also see the ability for motion controllers to transform into different virtual tools in seconds based on the user's actions and needs.

In the VR Game Budget Cut, the user can, at the push of a button, summon different tool attachments out of thin air. The user can then insert the base tool (as represented by the controller) into the desired tool attachment to gain new functionality as shown in Figure 3a.6. This allows the user to customize the tool depending on the situation at hand.

In the VR drawing application Tilt Brush by Google, the two motion controllers have separate roles. The controller wielded by the dominant hand is the user's interactor with different user interfaces. The controller wielded by the non-dominant hand is where multiple user interfaces are attached and serve as a recipient to the dominant controller's interactions. Figure 3a.7 shows the user holding a controller on the left hand where the color palette and other UI reside and a controller on the right hand that is drawing the colored lines.

Figure 3a.6: Screenshot capture from the game Budget Cut. The player is presented with multiple tool attachments at the touch of a button. The player chooses the attachment to use by moving the controller to one of the attachments.

fig:budget-cut

We mention dominant and non-dominant hands as the hyperphysical nature of representation allows the roles of the two controllers to be transformed and swapped quickly. Users can choose the hand that is most comfortable to control the primary interactor.

3a.3.3 Hyperphysicality Aspects - Action

In the Nintendo Switch game Arms shown in Figure 3a.8, the motion controllers represent the fist/boxing glove. The hyperphysicality comes from the action. As the user punches with the controller, the boxing glove flies forward to hit the enemy, something boxing gloves in the real world cannot do. Similar behavior can be found in the literature regarding non-linear amplification [144] where the user's movement is not mapped one-to-one from the physical movement to the resulting movement.

Figure 3a.7: Mixed reality image of Google's VR Drawing Application Tilt Brush. It depicts virtual interfaces attached to the physical VR controller fig:tilt-brush

SUPERHOT VR Figure 3a.9 showcases a different example of hyperphysicality in the action. In SUPERHOT VR, the user can punch, slice with a knife, and fire a bullet from a gun. Most of these actions will have the expected result of punching, cutting, and piercing through things. The game is unique, however, in what happens when you do not perform any action. Advertised as a game where "time moves when you move," it is hyperphysical in its flow of time.

3a.3.4 Hyperphysicality Aspect - Effect

3a.3.4.1 Menu Selection in Cloudlands : VR Minigolf

In the game Cloudlands : VR Minigolf, the motion controller is represented as a minigolf putter in VR. The main menu of the game is structured like a minigolf course. The player must putt the golf ball into the different holes to choose different menu options as shown in Figure 3a.10.

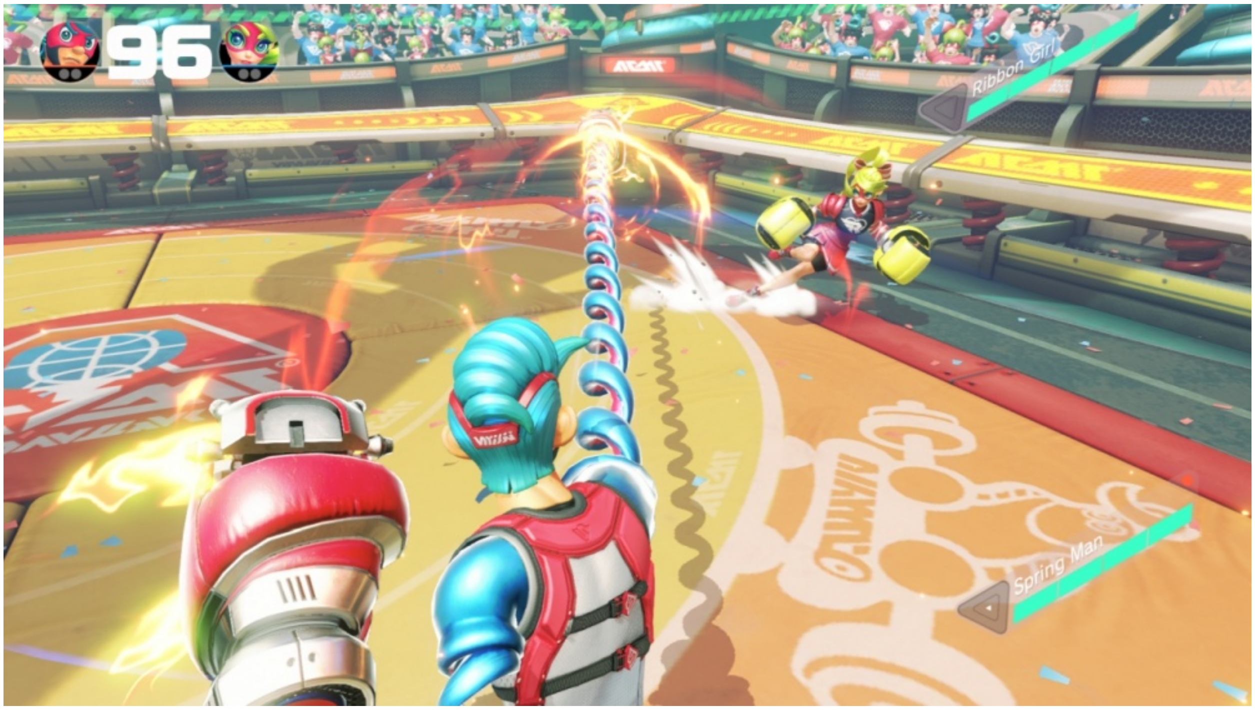


Figure 3a.8: Screenshot of the game Arms showing a player character’s fist flying t fig:nintendo-arms

The minigolf club is also an example of hyperphysicality in representation, but it differs from the previous example of Beat Saber. In addition to functioning like a golf club, it also serves the purpose of menu selection when combined with a golf ball, something a typical minigolf club cannot do. That is to say, it has hyperphysicality in both its functionality and effects.

Requiring the user to putt a golf ball to select a menu option serves a few interesting purposes. First, it reinforces the core mechanics of putting by requiring the player to putt the golf ball before actually starting the game.

The game developers have also chosen to avoid making the menu selection a low-calorie interaction. It would require much less physical exertion for the user to select a menu option using the common laser pointer interaction.

Figure 3a.9: Mixed Reality image of a person playing SUPERHOT VR. Time passes extremely slowly when the player is still. Time passes normally as the player moves. fig:superhot

3a.3.4.2 Another Method for Menu Selection

In the prior example of Beat Saber, we can imagine a scenario where the saber does not slice objects in the world following the laws of physics of our world. Instead, it may be used to slice a virtual representation of connections between objects, such as the connection between a button used for alarms and an audio player to create a ringing sound.

The saber could also slice cubes representing different menu options, much like what Cloudlands : VR Minigolf does. This is done in the VR Game Fruit Ninja VR as shown in Figure 3a.11. This implementation would also be a crossing-based UI[1] in three dimensions.

Figure 3a.10: Mixed Reality image of a person playing Cloudlands : VR Minigolf. By putting the golf ball into the appropriate openings, the player can access different menus fig:minigolf

3a.3.5 Hyperphysicality of Virtual Worlds

Hyperphysicality in the user's world is a common occurrence. It is not limited to the user or the physical extensions of the user. It can also exist within the virtual world in its objects or agents.

In the game Fantastic Contraption, where the player uses different construction materials to build a contraption, the creators deliberately moved away from providing a 2D UI for object selection. Instead, the construction materials are made available in the form of a colorful cartoon cat that serves as a storage interface, as shown in Figure 3a.12.

Figure 3a.11: Screenshot of Fruit Ninja VR showing different menu options as fruits to be sliced by the swords

fig:fruit-ninja

Games	Physical World	Virtual World Representation	Virtual World Actions	Virtual World Effects
VR Shooter games	Gun Shape	Gun	Gun-based actions such as fire and reload	Gun-related effects
Beat Saber	VR Controller	Saber	Saber-based actions such as slice or thrust	Saber-related effects
Nintendo's Arms	Motion Controller	Boxing Gloves	Boxing Gloves that extend and retract	Boxing Glove-related effects
SUPERHOT VR	VR Controller	Hand	Time moves when user moves	Hand-related effects
Cloudland VR Minigolf	VR Controller	Golf Club	Golf-based actions such as putt	Golf-ball related effect menu selection

Table 3a.2: Hyperphysicality in select games shov tab:hyperphysicality-summary

3a.3.6 Hyperphysicality Summary

3a.3.7 Quantify Hyperphysicality's Usefulness

Hyperphysicality can be applied to the different entities (interactor, interactable) in the Spatial Interaction Model in terms of the different hyperphysicality aspects (representations, actions, and effects). These can be represented as Lenses for Design, where there's a central concept with questions for the designer to ask themselves. This leads to further grouping of questions for ease of use. Each of the Hyperphysical Lenses provides questions to consider when designing with hyperphysicality

Figure 3a.12: The cat has a blue stick and a brown stick as tails. On the cat's back, there's a rotating cylinder shape. Players can make use of the above objects to construct a fantastic contraption. On the cat's head, there are multiple pins that can be used to remove any placed objects. fig:fantastic-contraption

The Lens of Hyperphysical References. Consider common sources of seemingly impossible and fantastical phenomena with which the user may be familiar. Draw inspiration and establish common grounds using cartoon physics[19], the principles of stage magic[138], magic systems from real-life rituals or fantasy novels, technologies in science fiction, etc.

- What source of hyperphysicality can be referenced to aid user understanding?
- How easily can this concept of hyperphysicality be taught to the user?

The Lens of Surrogate Representation

- Can this interactor represent a different object?
- Can this interactable represent a different object?

The Lens of Surrogate Action

Demos	**Physical World**	**Virtual World Representation**	**Virtual World Actions**	**Virtual World Effects**
Adjustable Pointer [167]	VR Controller	Adjustable laser pointer	Laser-pointer	Laser-pointer selector
Virtual Equipment as Sensory Options [160]	None (Space next to the user's ear)	Headphones	Headphone-like controls (Sony's headphones)	Adjust Volume, Song Selection
Virtual Equipment as Privacy Mask [163]	None (Space above the user's face)	Mask & Voodoo Doll	Manipulating a doll	Adjust privacy settings
Punch Typing [161] & Flick Typing [156, 155]	None (Space the user has chosen)	Virtual 3D Keyboard	Touching / Punching Buttons	Text Entry
Virtual Equipment as Avatar Doll	None (Space above the back)	Small Doll Figure	Placing the Avatar Doll where the ray ends	Enabling / Disabling sensory inputs from the avatar
Peripersonal Equipment	None (space around the user's body, reachable by hand)	Empty space but with a translucent sphere when visualization is needed	grab and place	storing or accessing Virtual Equipment

Table 3a.3: Hyperphysicality in select author's work tab:hyperphysicality_own_work

- Can this interaction be performed using a different action?
- Can this interaction be performed by a different interactor?
- Can this interaction be performed on a different interactable?
- Can this interaction achieve an indirect effect?

The Lens of Surrogate Effect

- Can this interaction achieve an indirect effect?

Factors	Sub-Factor	Examples
XR Content	-	Precision Work
User's Physical Traits	Age	Children, Adult, Elderly
	Fitness	
	Height	Short, Average, Tall
User's Mental Traits	Preference	Preferred Effort Level
User's Environmental Setting	-	Moving, Standing, Sitting, Supine, Floating
	Room Setup	Stationary, RoomeScale, Arena Scale, Open Space

Table 3a.4: Factors to consider for Whole Body Interacti tab:factors_whole_body

3a.4 Whole-body Interaction

ch3sec:lensWholeBodyInteraction

A unique quality of XR is that it involves our bodies more than traditional user interaction. Beyond the traditional mouse and keyboard and the more recent voice and video inputs, XR brings interactors that exist in three-dimensional space, such as the headset and 6DOF controllers. (There is also overlap with wearable computing.) The XR industry also continues to push towards increased participation of our body by tracking hands, heart pulse, facial expressions, etc.

The human body can assume various postures, such as sitting, standing, walking, or lying down. The environment may also restrict the user in various ways, such as being in the rear seat of the car[85]. It could also be used in more exotic conditions such as underwater[32] or floating in zero-gravity[*].

This differs from mobile devices because the interaction is not fixed to the hardware device. With a mobile device, the user must interact with the device at its location, whether ergonomic or not. We've all had times when our arms got sore from holding up our phones while lying in our beds. In XR, it is possible for users to switch to more ergonomic interactions based on different contexts. The context could come from the content the users are experiencing, such as entertainment or simulation. The context could also come from how the user chooses to

---

[*] `https://redmondmag.com/articles/2019/11/08/vr-headset-in-space.aspx`

engage with the content, such as the user's posture (e.g. standing, sitting, moving, lying down, etc).

3a.4.1 User Tracking

3a.4.1.1 Height Tracking

This subsection provides some examples of VR platforms and games that do height-tracking

Many VR systems (Vive, Oculus) track the user's height and use that information to place the floor of the virtual world accordingly for a more immersive experience. However, few games and experiences truly utilize the user's height as the foundation to build their experiences.

Returning to the example of Beat Saber, the game involves slicing objects and dodging obstacles coming toward you. To create obstacles that are dodged by crouching, the game must know the user's height so the obstacle can be placed at an appropriate height. This height calibration is done within the game and is separate from the height calibration from the VR Operating System. Beat Saber's need to support multi-platform suggests that they can not rely on information provided by the VR Operating System.

Moon Rider is an open-source game that is very similar to Beat Saber. In addition to a sword mode, Moon Rider has a fist mode in which the user punches the incoming colored block instead. An issue that becomes obvious in the fist mode that was not apparent in the sword mode is that the user's body was not considered for ergonomic design. A common pattern for the incoming blocks is a red block on the far left and a blue block on the far right, immediately followed by a blue block on the far left and a red block on the far right. With a long-reaching sword, it is easy to reach both blocks even when your arms are crossed. However, the task becomes much more difficult when it has to be achieved with just the user's arm span. The left arm can utilize its full reach for the initial block on the left, but the right arm has to first cover the distance of the body's width, likely reaching no more than the left arm's elbow. To mitigate this, the designers must manually adjust the block placement. Even then, it is difficult to make proper adjustments without information about the user's height and arm length.

Another example that uses the user's height is the game Job Simulator. Designed for a standing experience, the user stands in the middle of a workspace with everything within easy reach.

Developers can also make use of surrogate measures for height. For example, the Ape Index refers to the ratio of arm span to height, which is typically 1 in human[91]. The game Job Simulator uses the Ape Index to infer player height from armspan. During the calibration mode[†], the game asks players to stretch their arms while holding the tracked motion controllers to form a T-pose and calibrates the height based on this pose.

3a.4.1.2 Calorie Tracking

There are a few solutions for tracking calories in VR, such as YUR [‡], FitXR[§], Oculus Move[¶], and Supernatural [‖].

These solutions track calories based on estimates given by your headset and two motion controllers; they also track the time spent moving. Some solutions, such as YUR fit, may be linked to an external Bluetooth heart-rate monitor. YUR Fit also uses motion tracking to measure the number of squats a user makes.

[†] `https://uploadvr.com/job-sim-calibrating-height-ps-vr/`

[‡] `https://yur.fit/`, `https://store.steampowered.com/app/1188920/YUR/`

[§] `https://www.youtube.com/watch?v=6dnq6CzahSs`

[¶] `https://www.youtube.com/watch?v=UV3Tx8Vp0Ds`

[‖] `https://www.getsupernatural.com/`

Tracked calorie estimates help users quantify how many calories they have burned. As it's geared toward consumers for tracking calories, it is not meant for UI/UX designers to measure the effects of their designed interactions.

3a.4.2 Whole-body Interaction and Hyperphysicality

With the different parts of the body being tracked in XR, it is useful to look at how they can be utilized hyperphysically.

3a.4.2.1 Hyperphysicality for Head Inputs

In a number of VR games, the user's head becomes a participant and receptacle to interactions. In many games, 'eating' is featured as a game mechanic. The user can place food or drink items near the mouth to take a bite of the food. These interactions can be seen in games for cosmetic purposes, such as in Job Simulator. They can also be a gameplay mechanic and make use of hyperphysicality, such as eating food or drinking potions to recover health and mana.

In Valve's virtual reality demo and sandbox experience, The Lab, the user has a few different interactions with the head. Aside from the eating and drinking metaphors, it also uses the head as part of a transportation metaphor. As the human head has many sensory organs (visual, audio, etc.) and is one of the few things tracked by VR hardware, it's natural that it is used in metaphors for transporting to another world. The user can grab spheres representing different locations, move them close to the head, and be transported to another location as depicted in Figure 3a.13. The act of bringing the sphere near the head becomes a metaphor for entering another location/world.

In the previously mentioned game, Fantastic Contraption, the game developers also utilize the space around the user's head. Instead of grabbing materials from the colorful cat, expert users

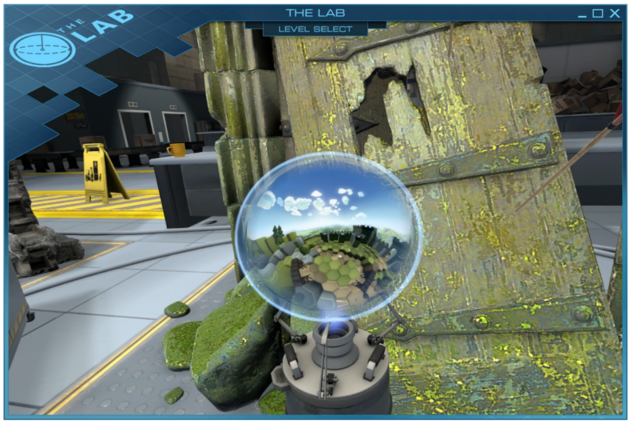


Figure 3a.13: A Screenshot of the VR experience The Lab by Valve. The glass sphere in front of the user can be moved near the face for the user to enter the game world contained within. fig:the-lab-sphere

can choose to grab different materials stored near the user's head. The user can grab the blue stick from the right ear, the brown stick from the left ear, and the pin directly above the head.

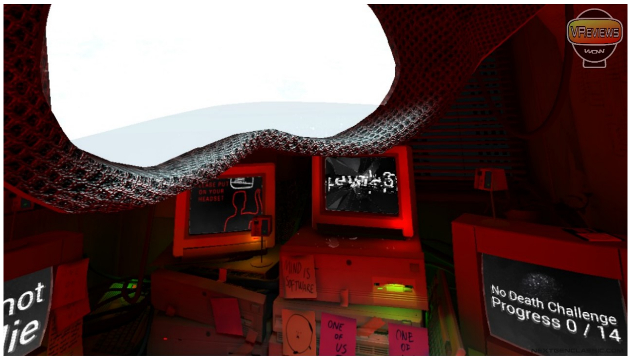


Figure 3a.14: A screenshot of the game SUPERHOT VR showing the user taking off a VR headset within the game to see the 'real life' in the game, which is the environment in red. fig:superhot-vr-in-vr

In many VR games, the player can also put on a virtual VR headset to enter Virtual Reality within the game. This occurs in the game "SUPERHOT VR" as shown in Figure 3a.14, the game "Virtual Reality Reality" as shown in Figure 3a.15, and the game "Accounting" as shown in Figure 3a.16.

The game Falcon Age provides an example of interacting with the user's head outside of the above examples of entering another world. In Falcon Age, the player can move the controller

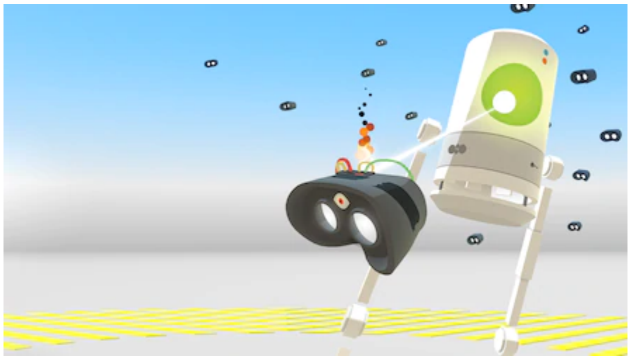


Figure 3a.15: A screenshot of the game Virtual Virtual Reality. The white robot forcibly removes the user's VR headset within the game.

fig:virtual-reality-reality

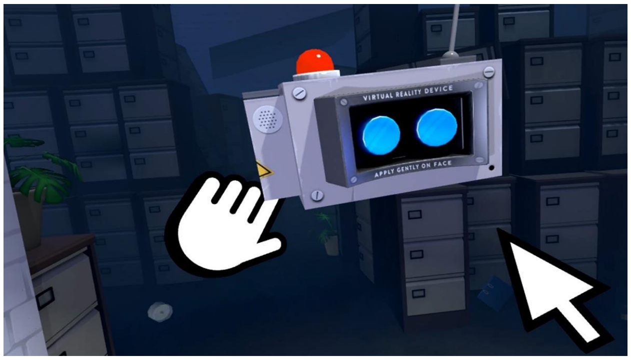


Figure 3a.16: A screenshot of the game Accounting in which the user has a primitive-look

fig:accounting

to head height to summon a falcon. While the falcon is flying and carrying an item, the player can raise the hand up high to receive the item.

3a.4.2.2 Hyperphysicality for Hand Input

In Facebook Connect Keynote 2020[**], Chief Scientist Michael Abrash revealed their work in using electromyography for hand tracking as shown in Figure 3a.17.

Although still in the research phase, an electromyography device can read motor neurons of the hand to track hand movement. Furthermore, they showed a person born with less than five fingers controlling five fingers in their demo, as shown in Figure 3a.18 and Figure 3a.19.

[**] https://youtu.be/woXmJMw2lTM?t=4722

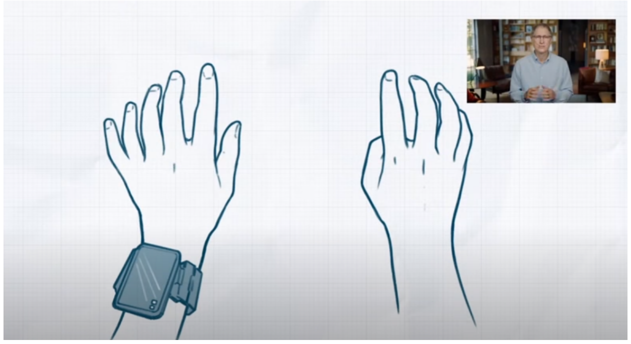


Figure 3a.17: Concept image of a person wearing an electromyography device to fig:electromyograph-3

Figure 3a.18: Image of a person born without five fingers fig:electromyograph-1

Abrash also suggested that with the human brain's neural plasticity, we could control body parts that we do not have in real life, such as a sixth or a seventh finger. This suggests the potential for hyperphysicality in interaction with hands and other body parts.

XR developer Daniel Beauchamp showcases another hyperphysical interaction hand example[††]. Daniel showed how a virtual hand can be thrown past the iron bars of a jail cell to access items beyond the user's physical reach. The hand can then be controlled remotely to crawl up the table and push the red button to free the user as shown in Figure 3a.20.

Hyperphysicality can also be used to trick the user into thinking their interaction is more natural and physical than it really is. One example would be redirected touching, such as the work[61] by Han et al. In this work, there is only one physical block, but three blocks in the

---

[††] `https://www.youtube.com/watch?v=CSqUgiEGs5s`

Figure 3a.19: Image of the virtual representation of the person born without five fingers. In the virtual representation, the person can make use of five fingers fig:electromyograph-2

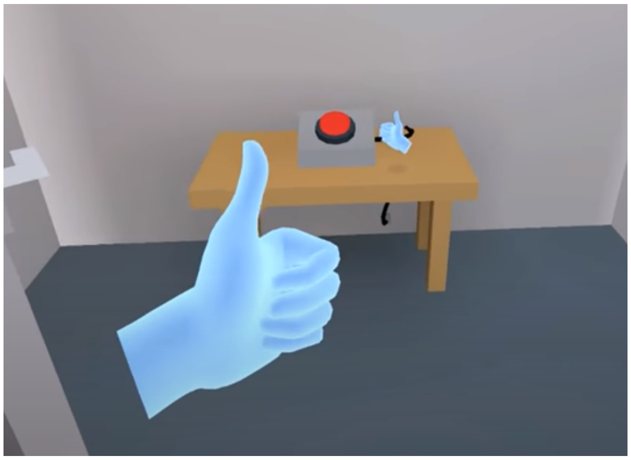


Figure 3a.20: Work by Daniel Beauchamp, showcasing how the user's hand can be disembodied and detached from the user. fig:throw-your-hand

virtual world. The user would reach their hand out to grab one of the three blocks, however, the visual provided by the VR system would always guide them into grabbing the one physical block. The user is convinced that there are multiple blocks when there is really one. This type of hyperphysicality can allow designers to reuse the same physical prop, which can provide new design opportunities and reduce the cost of providing a tangible user interface to the users.

3a.4.2.3 Hyperphysicality of Whole Body

Similar to redirected touching, the whole body can be engaged in hyperphysicality. One example would be redirected walking, which is remapping the user's position as the user moves. In extreme cases, the user could be walking in circles while convinced that they are walking in straight lines.

Another example of hyperphysicality of the whole body comes from the game Unseen Diplomacy as shown in Figure 3a.21. In this game, the user plays using Room-Scale VR. The user has to constantly navigate through small closet-like rooms, solving one challenge after another. The catch is that the small rooms are not constructed with real geometry. The user may enter a room and exit through the right-hand side door four times in a row only to find themselves in a completely new room each time without realizing that this layout does not make sense. This design trick allows the game world to feel much bigger than its physical space provides. While it may not make sense when the player stops and thinks about it, it makes enough sense for the player at the time to navigate and traverse through the world to get what they want.

3a.4.2.4 Hyperphysicality Derived Body Locations

Given that most commercial VR systems only track the user's head and hand position via motion controllers, few commercial experiences involve the use of other body parts. With the head position, however, we can derive other body locations. While these locations may not be accurate, they serve as a starting point for many experiences.

A common derived body location is the waist. This is typically calculated based on a simple spatial offset from the user's headset, though some games allow users to manually set their waist height.

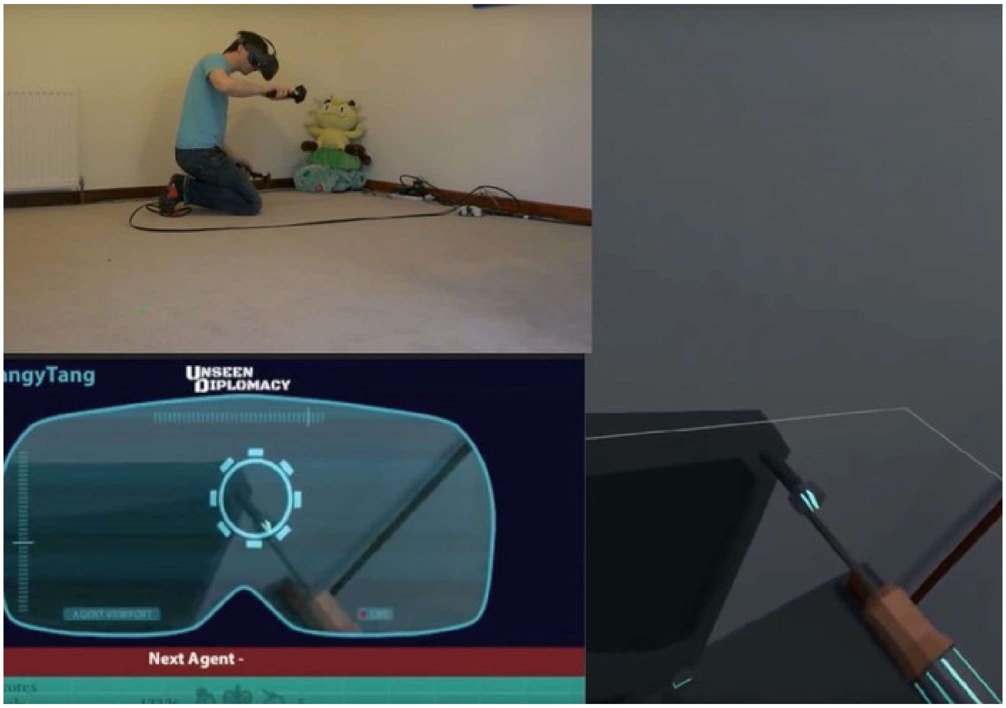


Figure 3a.21: The figure shows a player playing the VR game Unseen Diplomacy. The top left is the player in real life. The bottom left depicts what is shown on the PC screen for spectators. The bottom right depicts what the player sees in the VR game. Due to the limited physical space, the player moves and navigates through a game world composed of non-euclidean geometry. fig:unseen-diplomacy

Near-waist storage locations are often utilized from the waist to access equipment, such as holsters, belts, and pockets. We can find examples in games of weapon holsters (VR shooter games, e.g., RoboRecall, Payday 2 in VR), Utility Belts (Payday 2 and Batman: Arkham VR), and cellphones (stored in pockets from Dance Central VR).

Another commonly derived body location is the near-hand location of the wrist. This 'wrist' is derived from an offset from the motion controllers or based on a kinematic model. It is often used as a user interface in the form of a virtual wristwatch (Rec Room, Rick and Morty: Virtual Rick-ality) or as a storage interface (popularized by Half-Life: Alyx).

3a.4.3 Quantifying Whole Body Interaction's Usefulness

Whole-body interaction sets the limit to what hyperphysicality can do. Hyperphysicality suggests that anything can happen, but whole-body interaction grounds the design to what's ergonomic. Whole-body interaction can also provide useful creative constraints to the design. Each of the Whole Body Interaction lenses provides questions to consider when designing.

The Lens of Interaction Roles

- Can this body part serve as an interactor?
- If a body part serves as an interactor, can it also serve as an interface to affect the properties of the interactor?

The Lens of Soma-Semantic

- Can this interaction be stored and accessed using the body to bridge the interaction and the outcome?
- Can the interaction be stored or accessed at a location that is semantically linked to a body part?
- Can the effect of this interaction technique be semantically linked to a body part?

The Lens of Body Capability

- Can this interaction technique be performed using a different body part?
- Can Hyperphysicality be used to provide accessibility for those with restricted body movement?

3a.5 Extradimensional Space

ch3sec:lensExtradimensionalStorage

3a.5.1 Quantifying Usefulness of Extradimensional Space

Lens of Dimensional Sight

1. What dimensions should be currently visible?
2. What objects should be visible but not interactable?

Lens of Relevancy

1. Is this entity relevant to the current task, goal, scenario, or context?
2. If not, should it be stored in another dimension?

Lens of Permission

1. Can this interaction technique access entities in extradimensional space?
2. If not, should this interaction technique be an extradimensional interaction technique?
3. What are the criteria to access entities in this extradimensional space?

Chapter 3b

Gesture Taxonomy

ch3:Taxonomy

3b.1 Taxonomy Introduction

sec:Introduction

This section presents a multi-dimensional taxonomy of gesture for Extended Reality. This taxonomy is a general taxonomy that builds on top of the Spatial Interaction Model and the three design lenses in Chapter 3a.

XR brings users into an environment very similar to the real world. In the case of VR, the environment is almost entirely virtual. In the case of AR, the environment includes virtual elements to augment our real world. In each case, many previously difficult, expensive, or impossible actions become possible thanks to the virtual aspects of XR. In the same way that touchscreens, miniaturization, and wireless technology brought touch interaction and portability qualities that became the foundation of the mobile computing paradigm, XR technologies bring certain qualities to the spotlight, such as the user's body and the environment, which greatly influence Human-Computer Interaction (HCI) in XR. These qualities are discussed further in Section 3b.4.

3b.1.1 Existing Taxonomies

In this thesis's exploration of spatial gesture design in XR, design lenses and other XR qualities guide the designs. When my collaborators and I delved into research papers in the XR field to determine if our designs were truly novel, we found that some of our gestures did not fit into existing taxonomies. The existing taxonomies we have examined have not reflected these XR qualities. Many qualities we will discuss are lumped together under the catch-all dimension of "Context" in previous taxonomies. By explicitly identifying and considering these qualities, gesture designers gain additional reference points when designing or choosing the right gesture for the intended task.

At the same time, existing taxonomies can be very specific. Some taxonomies deal with different classes of gestures, such as taxonomies for Surface Gestures [147], taxonomies for Motion Gestures [122], and the increasingly common taxonomies of Mid-Air Gestures [3, 140, 20]. Given that all of these different classes of gestures can be utilized in XR, an XR taxonomy should unify and include dimensions in these specific taxonomies when possible. The presented taxonomy provides designers with a taxonomy that includes important considerations for XR as a starting point. At the same time, designers can delve into existing literature based on these specific gesture classes and taxonomies when they need to consider gestures at a different level of granularity or dimensions that are specific to a class of gestures.

We have also found many taxonomies to be a side discussion of their main work, to be terse and cryptic in their explanations, or to assume prior knowledge of prior taxonomies, making them difficult to understand or utilize.

3b.1.2 Motivation and Contribution

All in all, the many qualities, techniques, and contexts specific to XR call for a taxonomy specific to the XR field. Such XR taxonomy must account for the unique characteristics of XR, build on top of and unify existing taxonomies, and serve as a starting point for readers to leverage the more specific gesture taxonomies that were developed for different classes of gestures. Creating the basis for such a taxonomy is a continual goal and contribution of our work.

Novelty-wise, the contribution comes in three folds. First, this taxonomy considers many emergent qualities of XR that are made possible by advancements in technology. Second, we break down the Nature dimension found in many taxonomies work into these additional dimensions: Source of Meaning, Target, Action Mapping, and Effect Mapping. These new dimensions provide a different perspective on gestures and help consolidate proposed multi-dimension taxonomies with taxonomies based on McNeill's work [90], such as many of the mid-air hand gestures taxonomies, which is discussed in detail in Section 3b.7 and Section 3b.8. Thirdly, we provide a general architecture for a gesture system based on this taxonomy in Section 3b.10.

3b.1.3 Taxonomy Reading Guide

We have structured the sections to fulfill different needs. Readers seeking only to use this as a gesture taxonomy should go directly to Section 3b.5, Section 3b.6, and Section 3b.7. Readers trying to understand the differences between this taxonomy and prior work should refer to Section 3b.2 but crucially also Section 3b.8 and Section 3b.9. Readers interested in the implementation of a general gesture system can refer to Section 3b.10. Finally, readers interested in examples of gestures in this taxonomy and how the taxonomy can be used to guide development can refer to Section 3b.11.

3b.2 Background and Related Work

ch3:TaxBackground

3b.2.1 Early work on linguistic-based gesture categorization

Early efforts to classify gestures into a taxonomy started in the field of linguistics[45, 73]. These taxonomies examined gestures in a conversational setting (between human agents), and the gestures discussed during this stage were mid-air bare-hand gestures.

One of the earliest works is by Efron[45], who classified gestures into two categories and five sub-categories: Logical-discursive gestures (Batons and Ideographic gestures) and Objective gestures (Deictic gestures, Physiographic gestures, and Symbolic gestures). This classification provided the basic terminology for human gesture categorization and analysis, which greatly influenced later research on gesture taxonomy. After Efron, Kendon's analysis[73] of the relationship between gestures and linguistics played an important role. Kendon established a spectrum of human gestures based on speech dependency and social regularity, including Gesticulation, Language-like Gestures, Pantomimes, Emblems, and Sign Languages. Kendon's spectrum provided a comprehensive structure of the relationship between human gestures and communication. Later, McNeill[90] conducted a series of analyses on human gestures and human communication, which greatly influenced future works. McNeill proposed a taxonomy of gestures with four categories (Iconic, Metaphoric, Deictic, and Beats), which summarized the ideas from Efron, Kendon, etc., and provided abundant examples and experiments. Many later studies[115, 107, 22, 147, 122] followed McNeill's terminologies and taxonomy. The linguistic origin and scope of these early-stage studies, however, provide limited usefulness to HCI.

3b.2.2 Gesture taxonomy with Human-Computer Interaction involved

As computer science developed, the concept of gestures was adopted to communicate user intents to computer systems through user actions. There are many studies [115, 107, 22] on gestures in HCI after McNeill's work, which focused on mid-air bare-hand gestures. Quek proposed a hierarchy-style gesture taxonomy, which incorporated previous works, including McNeill's [90]. There are two general categories: Symbols and Acts. The former are subdivided into Mimetic gestures and Deictic gestures, and the latter contains Referential gestures and Modalizing gestures. Later, Pavlovic et al.[107] followed and expanded on Quek's taxonomy by introducing Manipulative gestures. Additionally, Cassell[22] made an extension to McNeill's taxonomy by adding two new categories (Emblematic gestures and Propositional gestures). Recently, Vuletic et al.[140] made a comprehensive literature review of the works from this period.

Gesture taxonomies in this period applied early analysis of linguistic mid-air bare-hand gestures to the HCI field. These early studies focused on hand gesture recognition and interpretation in computer systems. However, as human-computer interaction technologies and theories were not as developed, efforts at this stage were limited to linguistic analysis. They did not take the complexities of HCI into consideration.

3b.2.3 Multi-Dimensional Gesture Taxonomy

Later, HCI researchers found that it was not enough to categorize gestures by only one dimension, so they utilized gesture taxonomies with multiple dimensions, where each dimension contains multiple categories. Poggi's work[112] is one of the early important multi-dimensional gesture taxonomies. Poggi's taxonomy contains four dimensions from a linguistic perspective, including Relationship to Other Signals, Cognitive Construction, Gesture-Meaning Relationship, and Semantic Content.

As computer science technology grew rapidly, the concepts and content of gesture-based interaction were considerably enriched. For example, the input interactor was no longer limited to hands, but was expanded to mice, mobile phones, motion controllers, etc. Therefore, HCI researchers continued using multi-dimensional gesture taxonomies for their works[71, 147, 122, 139] to account for the rapid growth of computer science. These works allow gestures to be examined in a different way, which helps designers to better explore the design space. The categories from previous taxonomies were now mostly limited to one dimension known as "Gesture Style"[71] or "Nature"[147, 122, 139]. Karam's taxonomy[71] was one of the important multi-dimensional works at the early period of this stage, as it is one of the first efforts to introduce multi-dimensional taxonomy into gesture-based interaction in HCI. However, Karam's multi-dimensional taxonomy is overly broad and of limited utility to designing the complex gestures used in HCI applications of the XR domain.

Many important works on multi-dimensional gesture taxonomy in the HCI field are based on user-elicitation studies. Researchers proposed and verified gesture taxonomies according to the user-defined HCI gesture set as elicited by user studies. One of the most foundational works in this field was by Wobbrock et al.[147]. Based on their research on user-defined surface gesture sets, they proposed a taxonomy of surface gestures including four dimensions: Form (considering hand pose and movement path), Nature (the mapping from gestures to their meaning), Binding (spatial dependence), and Flow (whether the responses to gestures occur while or after the performer acts).

Later, Ruiz et al.[122] applied Wobbrock et al.'s user elicitation study method to motion gesture research. They proposed a motion gesture taxonomy based on the user-defined gesture set they collected. The taxonomy is based on Wobbrock et al. but with additional dimensions. Most importantly, Ruiz et al.[122] divided the dimensions of gesture taxonomy into two classes: Physical Characteristics and Gesture Mapping. Physical Characteristics consider the physical attributes of the gestures, including three dimensions: Kinematic Impulse (the range of jerk of the gestures), Dimension (the number of spatial coordinate axes involved in the gestures' motion), and Complexity (whether the gesture is a single or a composition of multiple discrete gestures). Gesture Mapping "involves how users map motion gestures to device commands" and contains three dimensions: Nature, Context (dependence on specific context), and Temporal (Flow in Wobbrock et al.'s taxonomy).

Following Ruiz et al., Vafaei[139] combined, modified, and expanded Wobbrock et al.'s and Ruiz et al.'s work, leading to a comprehensive taxonomy on HCI gestures. The modifications were made to adapt the taxonomy to one primarily focused on mid-air gestures. Utilizing Ruiz et al.'s categorization of taxonomy dimensions, Vafaei's Gesture Mapping contains five dimensions: Nature, Form, Binding, Temporal, and Context. Physical Characteristics include six dimensions: Dimensionality (same as Ruiz et al.), Complexity, Body Part (which part of the user's body is involved), Handedness (whether it involves the dominant or non-dominant hand), Hand Shape (literally), and Range of Motion (the extent of rotation of the user's joints).

There are also other works considering gesture taxonomy in the HCI field [46, 119, 50, 3, 20], which made combinations, modifications or expansions to the above taxonomies.

Most gesture taxonomy in the HCI field proposed so far has been limited to specific research areas. For example, Wobbrock's work[147] focused on surface gestures on 2D screens, Ruiz's work[122] focused on motion gestures, Freeman's work[50] focused on multi-touch whole-hand gestures, and Ens' work focused on AR [46]. There has been some effort toward the taxonomy of gestures used in VR domains in recent years, such as Bhowmick et al.[3] on VR object selection, Wu et al.[150] on VR shopping and work of Arora et al. on VR animating [13], but these works are focused on some specific application domains in VR. To the best of our knowledge, there have been no efforts working on the general applications to the XR domain or considering the multiple qualities we mentioned before, such as User, Environment, Interactors, Interactables, etc. We built our taxonomy based on these previous multi-dimensional taxonomies to create an overarching taxonomy that can include more specific taxonomies.

3b.3 Methodology

ch3:TaxMethodology

In our initial attempt to classify gestures and interactions, we started with a specific search for "XR gesture taxonomy" on Google Scholar. Then, we broadened the scope of our search terms as we failed to find a taxonomy to classify our gestures. We used terms such as "gesture virtual reality", "3D gesture virtual reality", "user defined gestures", "motion gesture", "gesture based interaction", "controller interaction virtual reality", "gesture interaction virtual reality", and "controller gesture virtual reality". We recognized that there is a need for an XR taxonomy at this point and worked toward a qualitative study on gestures and gestures taxonomy by examining prior works. The searches served as the starting point. Our goal was not to perform an exhaustive literature review but to create a taxonomy of classifications for use with XR. Thus, we chose to evaluate the following works and taxonomies[45, 73, 117, 90, 115, 107, 22, 112, 71, 147, 122, 105, 139, 46, 119]. We found five taxonomies to be particularly representative and focused our efforts on unifying them. McNeill[90] provided comprehensive research on linguistic mid-air bare-hand gestures and proposed a taxonomy, which was the basis for future works on gesture taxonomy. Wobbrock et al.[147] and Ruiz et al.[122] established multi-dimensional taxonomies for surface gestures and motion gestures, respectively. Vafaei[139] built on top of their works but with an emphasis on hand gestures. These three works provided a paradigm for future studies on gesture taxonomy in HCI, so our taxonomy followed their frameworks. Additionally, Ens et al.[46] proposed using a taxonomy to inform design space and suggested new design space dimensions, offering a different perspective from designers. The taxonomies we examined and the reasons why we considered them to be representative are detailed further in Section 3b.2.

For the qualitative study and actual creation of the taxonomy, it was done by using an adapted grounded theory approach. Grounded theory analysis[57] uses collected data to explore novel domains and to form a theory. The approach mainly consists of three steps: open coding, axial coding, and selective coding.

1. In open coding, data is collected. Preliminary labels are applied to create codes. Codes are grouped together as concepts based on shared qualities.
2. In axial coding, codes and concepts are analyzed to find the relationships between them by creating categories.
3. In selective coding, the categories created are used to form a general theory.

Adapting this method to our use, we started with a closed-book coding approach where we applied open coding, axial coding, and selective coding to the select few taxonomies mentioned above. Two researchers performed inductive coding by negotiating the terms of dimensions and classes until they came to an agreement. After the initial three-step process, we moved to a hybrid approach where we used the results of the previous process in combination with new taxonomies and gestures to verify our taxonomy's classification power. If the taxonomy is insufficiently powerful in its classification power, we add or modify encoding and concepts as needed. We repeated the process multiple times until we found our taxonomy to be sufficiently strong in its classification power. By sufficiently strong, we mean that our taxonomy can be used instead of a previous taxonomy for all of that taxonomy's dimensions. But for certain taxonomies, we found this impractical. Instead, we were able to determine that the problematic dimensions of the taxonomies were not relevant to the scope of our taxonomy.

For example, the Motion Gesture-based taxonomy[122] has kinematic pulses with categories of low, medium, and high mapped to gestures where the range of jerk is below 3m/s^3, between 3m/s^3 and 6m/s^3, and above 6m/s^3. As this taxonomy specifically deals with modern smartphones, the values are specific to objects of similar size. In our taxonomy, that dimension is covered by Motion Threshold with the categories of None, Position, Velocity, Acceleration, and Jerk. The kinematic pulses would thus be covered by Motion Threshold/Jerk, and the specific categories under it would be irrelevant to the scope of our taxonomy.

3b.4 XR Qualities

ch3:TaxQualities

XR has several qualities that distinguish its computing paradigms from other paradigms. We'll briefly examine some unique qualities, how they impact XR, and why they need to be considered in a gesture taxonomy for XR.

3b.4.1 User

The most important quality of XR is perhaps the relationship between the user and the computing environment. Users are increasingly included as part of the computing environment.

In the Windows, Icon, Mouse, and Pointer (WIMP) environment, the user is represented by a disembodied mouse cursor or a text caret. In XR, the user can interact with the environment directly. Interaction is not through an indirect intermediary such as a mouse, a keyboard, or a mobile phone. It is directly translated to their virtual counterparts in the form of heads, hands, other body parts, or motion controllers as extensions of body parts.

With the presence of the user's body, users can leverage their kinaesthetic senses. They can also take advantage of their personal, peripersonal, and extrapersonal space for spatial interactions[164]. Other user characteristics (e.g., posture, pulse rate, or facial expression) are increasingly being tracked and utilized [149]. While XR is not the only computing environment that utilizes these characteristics, there is a greater prevalence and emphasis in this medium.

3b.4.2 Environment

The users' physical presence allows 3D spatial interactions with the environment, changing the users' relationship with it. The environment can be structured to follow the same laws of physics as the real world. This allows users to adapt quickly by utilizing existing real-life skills to navigate a familiar environment. For example, users can use their visuomotor skills from real life in the XR environment for object selection and manipulation. The skills gained from the XR environment can also transfer back to the real world, allowing XR to be used for training or rehabilitation.

3b.4.3 Hyperphysical Environment

XR environment can also adapt different laws of physics as mentioned in Section 3a.3. This can be useful for entertainment, necessary to overcome real-world restrictions (such as accessible physical space or restricting cables), or used to make the XR experience more convenient. Teleportation as a locomotion method is an example of hyperphysical interaction. It does not follow the laws of physics in the real world; however, its usefulness in fulfilling the above needs makes it one of the most common locomotion options in VR.

3b.4.4 Environment/Region

6DOF XR is inherently spatial and emphasizes the spatial context. For example, as the user traverses the environment, such as coming home from a grocery trip and moving from the car into the kitchen, the environmental context changes rapidly. The user has gone from a public space into a private space while traveling through several regions, each intended for different purposes. User interactions in XR need to consider these spatial contexts.

3b.4.5 Object as an Interactor

Technology advancements have allowed for hardware setups that track the user and the environment, which is particularly prevalent in XR. Computer vision or hardware trackers may track real objects. XR system can also generate and track virtual objects. This allows any object the user picks up to subsequently be turned into interactors, which can be used to perform gestures. The system can then give additional hyperphysical properties to these objects. For example, the user can shake an apple (virtual or real). This will activate the associated effect of the gesture, such as a popup with information regarding the apple as shown in Figure 3b.1.

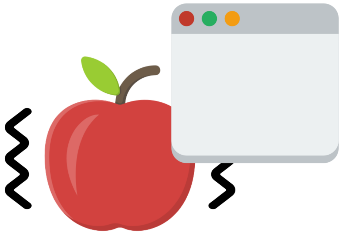


Figure 3b.1: User shaking an apple to get additional information in the form of a popup window. The apple can be viewed either as an interactor performing the action or as additional context (e.g., interactable) to the hand interactor.

Fig:apple

Of note, the designer could treat objects picked up by the user as an individual and separate interactor or as a contextual cue (e.g. interactable) of the hand/motion controller interactor that will subsequently modify its mapping. It is up to the designer to determine which classification works better for them.

3b.4.6 Interactor Qualities

In our classification, we include objects that are being tracked in their motion to determine gestures in the interactor dimension as it allows us to better encapsulate gestures for similar types of interactors. We can consider and create gestures based on shared qualities and traits of the same class of interactors. For example, some examples of categories for interactors could be handheld objects and wearable objects based on their spatial relationship to the user. Alternatively, a gesture can apply to a variety of weapons and tools, based on their function.

The properties of the individual interactor, such as the sharp edge of a blade compared to the blunt end of a club, can also be used to determine the effect of gestures.

3b.4.7 Multiple Interactors

Given that any object can be an interactor and the user's body parts can also serve as interactors, for there to be multiple interactors in a single gesture or interaction is a common occurrence.

XR tracking technology provides additional information about interactors that can lead to new possibilities. For example, traditional surface gestures track points of contact. When trying to track a pinch gesture, it would be unable to differentiate a pinching motion with one hand, or two fingers from two separate hands. However, immersive technologies (e.g. cameras and computer vision) could track which fingers they are, their angle, and which hand each finger is from. The same surface gesture can now have different effects depending on whether it is performed by the dominant hand, off-hand, or both hands. The additional context can reframe how we think about surface gestures and other interactions in XR.

3b.4.8 Interactables

We also examine the concept of interactables, which are the targets of a gesture. Having an interactable as the target of a gesture is not exclusive to XR. In a WIMP environment, right-clicking on the desktop and right-clicking on a file will bring up different options. However, we have found it useful to think about additional qualities of interactables in XR and how they fit into the gesture taxonomy.

Interactor Modifying Interactor With multiple spatial interactors, an emergent property of interactors is that they can modify other interactors directly. This allows one interactor to perform an action and the other interactor to change the end results of that action, even while the action occurs.

This can be seen in the WIMP environment, where some keyboard keys (e.g., Ctrl, Shift, or Alt keys) may modify the behavior of the mouse. For example, while selecting GUI files, a single click highlights a file. Following that with another click with the Shift key pressed down allows the user to select all files within that range.

This can be viewed as a compound gesture involving two simple gestures, with one gesture providing an additional context for the other gesture. This combination is worth noting as it can dramatically increase the interaction space.

3b.4.9 Additional Contexts

An emerging theme from these qualities mentioned above is how expansive XR interactions can be. In addition to the obvious spatial context, XR being based in reality means it can also include any of the situations and contexts that one may face in real life. From work to entertainment, personal to public, mobile computing to desktop computing, anything a user can interact with, and any context a user can face in real life can be part of XR.

Based on the many contexts that an XR user may need to deal with, many techniques have been developed. In the text 3D User Interfaces [83], LaViola et al. classify 3D interaction techniques into the broad categories of Selection and Manipulation, Travel, and System Control. In the Travel category, the locomotion vault[36] tracks 109 travel techniques for using VR alone. Depending on the context, one technique may be preferred over another. While not all 109 techniques are based on gestures, this illustrates the importance of understanding and classifying gestures based on the contexts to identify the most appropriate ones available.

3b.5 Multi-Dimensional Taxonomy of Gesture

ch3:TaxTaxonomy

The taxonomy presented here is based primarily on the work "Taxonomy of Gestures in Human-Computer Interaction" by Vafaei[139], which is in turn based on the prior taxonomy work of Wobbrock et al.[147] and Ruiz et al.[122]. Those authors, in turn, based their work on prior work as outlined in the Section 3b.2. We also incorporate elements of design space from works by Ens et al.2018.

This multi-dimension taxonomy contains the following terms, defined below for clarity.

- Class: A logical grouping of dimensions, which aids understanding of how the dimensions are related and how they impact gestures.
- Dimension: A collection of categories. Similar to the enumerated type. [*]
- Category: Unique qualities within a dimension. Each gesture can be of one category within a dimension. Similar to enumerator.

Classes of Taxonomy Dimensions Ruiz et al., followed by Vafaei, divided the dimensions of a gesture into two classes - Physical Characteristics and Gesture Mapping. In our taxonomy, we continue to use these same classes but with the following definitions.

Physical Characteristics Ruiz et al. defined Physical Characteristics as “characteristics of the gestures themselves.” As they deal primarily with surface gestures, motion gestures, and

---

[*] Dimension used here is different from the dimension relating to extradimensional space. Refer to the terminologies in Section 2.2.2

mobile computing paradigms, the dimensions within this category are mainly restricted to the gestures' inputs themselves, as the properties of interactors for Surface Gestures (fingers or stylus) and Motion Gestures (phones) are not fully captured.

In contrast, Vafaei primarily deals with mid-air gestures with the user's hands as the interactor. Thus, Vafaei defined it as "involve physical attributes of the gestures themselves" and includes dimensions unique to the hands such as Handedness, Hand Shape, and Range of Motion.

We broaden the scope of Ruiz et al.'s Physical Characteristics class to include any interactor and convert hand-based dimensions in Vafaei's work to more general dimensions based on the broader concept of interactors. Physical Characteristics can be viewed as **"physical characteristics of the interactor and the action performed by the interactor."** Our taxonomy of Physical Characteristics is listed in Table 3b.1. Put simply, Physical Characteristics are about the properties of the interactor and its actions.

Gesture Mapping Ruiz et al. define Gesture Mapping as "how users map motion gestures to device commands" whereas Vafaei defined it as "how users map gestures to tasks."

The Physical Characteristics class deals with dimensions that describe an action. The same action, however, can be interpreted in many ways. This is where Gesture Mapping comes in. The Gesture Mapping class contains dimensions that provide the necessary contexts for the audience to understand the intent of the gesture performer and respond accordingly.

Both Ruiz et al. and Vafaei include Nature, Temporal, and Context as dimensions in Gesture Mapping, with Vafaei further including the Form and Binding dimensions from Wobbrock's work. Regardless of which dimensions are used, they all serve the purpose of disambiguating physical actions into different results.

The Context dimension is used as a catchall for all other possibilities that are not mentioned in the Taxonomies. In XR, however, we find that many dimensions normally categorized under the dimension Context are worth listing out as separate dimensions for consideration.

Thus, we view any dimensions within the Gesture Mapping class as providing context from different perspectives and define the Gesture Mapping class as **"any quality that can be used as context to map an action to a different effect"**. Our taxonomy of Physical Characteristics is listed in 3b.2. Put simply, Gesture Mapping is the context to decipher the action within a gesture to determine the effect.

Utilizing/Thinking about these Two Classes If a dimension involves altering the action that the user will perform or changes the property of the interactor, it belongs in the class of Physical Characteristics. On the other hand, if the same interactor performs the same action but has different results, it must be due to the difference in a dimension belonging to the Gesture Mapping class. The Gesture Mapping class covers dimensions that can be used as context to disambiguate one result from another.

3b.6 Physical Characteristics

ch3:TaxPhysicalCharacteristics

Physical Characteristics describe the motion and other easily seen characteristics of a gesture. Thus, our Physical Characteristics class, for the most part, is very similar to the ones proposed by Vafaei and others. We generalize the dimensions in this class to better fit XR and for the taxonomy to be used by the Gesture System Architecture in Section 3b.10.

Table 3b.1: The dimension class of Physical Characteristics in the gesture taxonomy

Table:physical

Dimensions	Sub-dimensions	Categories	Sub-categories	Description
Position	***x***	*Relevant*		Gesture involves the x-axis of position
		Irrelevant		Gesture ignores the x-axis of position
	y	*Relevant*		Gesture involves the y-axis of position
		Irrelevant		Gesture ignores the y-axis of position
	z	*Relevant*		Gesture involves the z-axis of position
		Irrelevant		Gesture ignores the z-axis of position
Rotation	***x***	*Relevant*		Gesture involves the roll-axis of rotation
		Irrelevant		Gesture ignores the roll-axis of rotation
	y	*Relevant*		Gesture involves the pitch-axis of rotation
		Irrelevant		Gesture ignores the pitch-axis of rotation
	z	*Relevant*		Gesture involves the yaw-axis of rotation
		Irrelevant		Gesture ignores the yaw-axis of rotation
Form		*Static*		No motion/change is in gesture
		Dynamic		Motion/change occurs in gesture
Interactor		*User Body*	*Hand*	Arm is fixed, but palm or fingers move
			Arm	Arm moves (hand moves as well)
			Head	Gesture is performed by head movement
			Shoulder	Gesture is performed by shoulder movement
			Foot	Gesture is performed by foot movement
			Eyes	Gesture is performed by pupil movement
			Mouth	Gesture is performed by lip movement
			Other body parts	Gesture is performed by other body part
		Input Device		Gesture utilizes Input Device as an interactor
		In-Environment Object		Gesture utilizes In-Environment Object as an interactor
Multiplicity		*Simple*		Gesture utilizes only one interactor
		Multiple		Gesture utilizes multiple interactors
Action Symmetry		*Symmetric*		Gesture utilizes multiple interactors performing similar motion
		Asymmetric		Gesture utilizes multiple interactors doing performing different motion
Interactor Homogeniety		*High*		Interactors are very similar in its form and function
		Medium		Interactors are only somewhat similar in its form and function
		Low		Interactors are not similar in its form and function
Complexity		*Simple*		Gesture involves an atomic gesture
		Compound		Gesture involves multiple atomic gestures
Motion Threshold		*None*		Gesture does not involve changes in position or rotation
		Position / Angle		Gesture involves changes in position/angle
		Velocity		Gesture involves changes in velocity/angular velocity
		Acceleration		Gesture involves changes in acceleration/angular acceleration
		Jerk		Gesture involves changes in jerk/angular jerk

Position and Rotation We have replaced the dimensionality dimension with the categories of single-axis, double-axis, tri-axis, and six-axis from Vafaei into separate dimensions of Position and Rotation. While it may be easy to categorize all surface gestures as having the dimension of Double-Axis (gesture occurs on a surface, a 2D plane), a multi-touch surface gesture can involve rotation, as in the case of one stationary contact point with another contact point rotating and tracing a circle. In XR, where the interactor is tracked in 3D space, a gesture can involve interaction on a 2D surface while considering the interactor's rotation in 3D space. The traditional surface gesture would fit within this taxonomy as a special case of a broader surface gesture class that deals with actions on a 2D plane by a 3D interactor.

Object movement of objects in 3D space can be defined as translation (the change of position) and rotation (the change of self-orientation). Therefore, these two aspects are essential for defining a spatial gesture. In our taxonomy, we define two dimensions of Position and Rotation, each containing three sub-dimensions: x, y, and z, denoting the three axes in Cartesian coordinates. Each of these sub-dimensions can be enumerated as either Relevant or Irrelevant. Relevant means that this axis is used in the gesture, and Irrelevant means that this gesture does not care about this axis, i.e., that the movement of this gesture involved in this axis is not considered.

If one gesture requires the user to move the interactor, it involves the change of position or/and rotation of the interactor. Thus, the movement is relevant to some of the six axes while possibly irrelevant to the other axes. For example, if a gesture requires moving the interactor to draw a circle, with no limitation on the rotation of the interactor itself, then this gesture is relevant to the x, y, and z axes of Position and irrelevant to the x, y, and z axes of Rotation. However, if the gesture requires drawing a circle with the interactor's front direction remaining aligned with the tangential direction of the circle, then this gesture is relevant to all six axes.

It is worth mentioning that if a gesture does not allow movement on specific axes, it does not mean that it is irrelevant to these axes. For example, consider an XR fighting game where the character can make a special attack when the user thrusts a virtual sword ahead in front of them with the point of the blade remaining forward. It might seem that such a gesture only involves the position change along one axis, but it also involves two axes of rotation.

Form The Form dimension is modified from Vafaei's definition to only include Static and Dynamic. The Stroke category is removed[†].

Motion Threshold Ruiz et al. categorized a motion gesture's jerk under Kinematic Impulse with a gesture being either low, moderate, or high (which corresponds to 3 m/s^3, 3-6 m/s^3, and greater than 6 m/s^3). In this taxonomy, we instead focus on the different ways position or rotation changes over time. This ranges from distance, velocity, acceleration, to jerk as well as the angular equivalents. Some gestures may involve a simple change in distance to trigger the effects, while another may require a sudden, forceful jerk. We do not assign a specific numeric value as the value would differ depending on the interactor.

Complexity The complexity dimension remains unchanged from Vafaei with two categories: Simple and Compound. A gesture is simple (also referred to as atomic) if it cannot be broken down into a simpler gesture. A compound gesture is a gesture that can be decomposed into two or more simple gestures.

Interactor The interactor dimension specifies whether the interactor is an input device, an object in the environment, or a part of the user's body. Input Device includes headsets, motion

---

[†] It may be brought back as a third form of a repeating pattern

controllers, 6DOF trackers, and so on. In-environment objects are any tracked objects, be they virtual or physical, that the user can interact with.

Interactor/User Body The Body Part dimension introduced by Vafaei's work is considered a sub-dimension of the Interactor. Bringing in the Design Lens of Whole-Body Interaction, we incorporate commonly available body parts for interaction. In addition to the Hand, Arm, Head, Shoulder, and Foot, we also include Eyes, Mouth, Face, and Legs as they are increasingly tracked. Lastly, we include Other Body Part as a category to account for future advances in body tracking technologies. For simplicity, it does not include biosignals generated by the user that cannot be seen or consciously controlled, such as the user's pulse rate. Those are instead categorized under Gesture Mapping as additional modalities that can be considered in conjunction with gestures to map the effect.

Multiplicity The Multiplicity dimension considers whether the gesture involves one or multiple interactors. As previously mentioned, the existence of multiple interactors makes this an important dimension to consider when designing gestures.

In the work of Ens et al. dealing with hand gestures, this was represented under the dimension of Hand Coordination with the categories Unimanual and Bimanual. Unimanual and Bimanual are still useful categories to consider when designing any gestures in XR that require hands. In many cases, having multiple interactors will require the user to use both hands to manipulate the interactors (e.g., motion controllers). In this case, the same Bimanual considerations would be made, such as the user's kinespheres based on the shoulder joint and arm length.

Multiplicity/Action Symmetry With multiple interactors involved, it can also be useful to examine what the multiple interactors are doing. If they are performing the same action, such as a pinch gesture, the actions are symmetrical. If one hand is turning an imaginary steering wheel and the other hand is grabbing an imaginary shift stick, the actions are asymmetrical.

Multiplicity/Interactor Homogeneity With multiple interactors, the interactors can have very different properties from each other. For example, the user's head and hands may both be interactors in a gesture. If the interactors are very similar, such as two hands clapping, they would have high homogeneity. If they have very different properties, such as a head and hand, they would have low homogeneity. Interactor Homogeneity also considers how the properties of the interactors are involved in the gesture. The user may be holding and using a motion controller with the left hand while using the right hand without a motion controller. In this case, the two interactors are the motion controller and the right hand. There may still be Action Symmetry in a gesture despite using different interactors.

3b.7 Gesture Mapping

ch3:TaxGestureMapping

The Gesture Mapping class differs the most from prior taxonomies. It consists of 1) dimensions that users utilize to understand how their actions are linked to the end result and 2) dimensions that are considered outside of the immediate interactor, its movement, and its interactable. In other words, it's the 'how' and 'what else' to consider.

3b.7.1 Dimensions About Understanding the Gesture/Effect connection

These dimensions are unpacked from the Nature dimension in previous taxonomies. These new dimensions deal with the relationship of the performed gesture to its effect.

Table 3b.2: The dimension class of Gesture Mapping in the gesture taxonomy

Table:mapping

Dimensions	Sub-dimensions	Categories	Description
Source of Meaning		*Physical*	Gesture derives its meaning from laws of physics
		Hyperphysical	Gesture derives its meaning from an alternate laws of physics
		Universal	Gesture derives its meaning from a generally universal set of values
		Cultural	Gesture derives its meaning from shared cultural values
		None	Gesture does not derives its meaning from any particular source; its arbitrary
Target		*Has Target*	Gesture has at least a target to interact with
		No Target	Gesture has no target to interact with
Action Mapping		*Direct*	Direct Relationship between Intent and Action
		Indirect	Indirect Relationship to Gesture between Intent and Action
Effect Mapping		*Direct*	Direct Relationship between Intent and Effect
		Indirect	Indirect Relationship to Gesture between Intent and Effect
Coordinate System		*Object / Interactable*	Coordinate system to process the gesture is based on an object
		Region	Coordinate system to process the gesture is based on a region
		World	Coordinate system to process the gesture is based on the world
		User	Coordinate system to process the gesture is based on the user
		Interactor	Coordinate system to process the gesture is based on the interactors
		Independent	No particular coordinate system is needed to process the gesture
		Mixed Dependencies	Use combination of coordinate system
Interaction Context		*Object*	Gestures require interaction with specific objects
		Region	Gestures require interaction with region-specific context
		World	Gesture require interaction with the world
		User	Gestures require interaction with the user
		Mixed Interaction Context	Gestures require interaction with multiple kinds of above components
		No Context	Gestures do not require interaction within any context
Temporal		*Continuous*	Action/task is performed during gesture
		Discrete	Action/task is performed after completion of gesture
Environment	***Region***	*Region-Associated*	Effect of gesture is relevant to which region it is performed in
		Region-Independent	Effect of gesture is independent from region
Social Context		*Private*	Only me
		Personal	My friends
		Social Groups	My teammates
		Public	Anyone else
User	***Proximity***	*Personal Space*	Gesture is performed in user's Personal Space
		Peripersonal Space	Gesture is performed in user's Peripersonal Space
		Extrapersonal Space	Gesture is performed in user's Extrapersonal Space
		Not Relevant	Gesture is performed in any space
	Posture	*Standing*	Gesture is based on User in the standing posture
		Sitting	Gesture is based on User in the sitting posture
		Lying Down	Gesture is based on User in the lying posture
		Other postures	Gesture is based on User in other postures not listed above
	Mobility	*Stationary*	Gesture is based on if the User is stationary
		Mobile	Gesture is based on if the User is travelling
		Mobile/Transportation Method	Gesture is based on the User's transportation method, e.g. bicycle, bus, car
Audience		*Human*	Gesture has human as the audience
		Rigid System	Gesture has a rule-based program as the audience
		Intelligent Agent	Gesture has an intelligent agent similar to human

Source of Meaning The Source of Meaning dimension is the pool of knowledge and experience that the user and the audience draw upon to make sense of the gesture. The dimension can take on the value of Physical, Hyperphysical, Universal, Cultural, or None.

The source of meaning between a gesture and its effect could be derived from the Physical category, which is the understanding of the laws of physics in the real world.

Alternatively, the Hyperphysical category refers to the user understanding the gesture based on some alternate set of laws of physics that explain how objects and the world would work. For example, telekinesis or using portals are impossible in real life, but are often portrayed in media and XR games.

Although very few things are truly universal, the Universal category refers to knowledge that is commonly known. In contrast, the Cultural category refers to knowledge that is limited to a specific group. Lastly, None (or Arbitrary) refers to gestures where there is no real-world source of meaning to draw upon.

Action Mapping A gesture has a direct relationship in Action Mapping if the performed action matches the utilized meaning, such as the user running with legs as the interactor to convey the meaning of running. If the user mimics running with other body parts, such as with fingers or arms, it would have an indirect relationship.

Effect Mapping A gesture has a direct relationship in Effect Mapping if the resulting effect matches the utilized meaning, such as the user's character running in response to the user's action conveying the meaning of running. If the same action and meaning lead to a different character running or to a machine operating, there is an indirect relationship.

3b.7.2 Contexts

Coordinate System The Coordinate System dimension is critical in providing the audience with a spatial reference point and frame to make sense of a gesture. For this dimension, we propose the following categories of reference points: Object, Region, World, User, Interactor, Independent, and Mixed Dependencies.

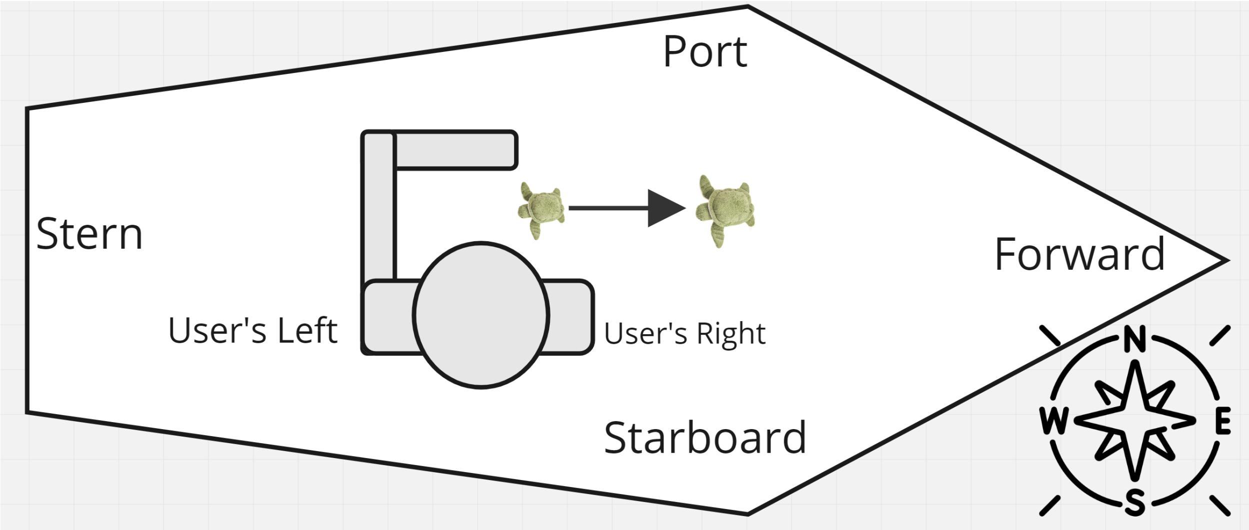


Figure 3b.2: The user moves a turtle on a boat. The same movement can be interpreted differently based on the perspective of the coordinate system used. The turtle moves backward according to its own perspective, forward to the user's left hand, right to the user's head, forward to the ship, and east in terms of Cardinal Directions.

fig:boat

In Figure 3b.2, we show an example of how the exact same motion can be mapped to different gestures and thus to different effects, depending on the Coordinate System used as the reference. Figure 3b.2 shows a top-down view of a user performing a motion gesture on a stuffed animal by moving the stuffed animal to the user's right-hand side. The direction of the gesture with respect to these different coordinate systems:

Thus, the same exact motion may map to at least five different effects. Although this is an extreme case, it is something for designers to consider. The designer may want to use other

Categories	Object	Direction
Interactable	Stuffed Animal	Backward
Interactor	User's Left Hand	Forward
User	User	Right
Region	Ship	Forward
World	World	East

Table 3b.3: This table shows the movement direction of the stuffed animal in Figure 3b.2 when considered by different categories under the coordinate system dimension

contexts to ensure the correct gesture and effect are considered. For example, if the user is grabbing the stuffed animal, the motion gesture associated with the stuffed animal would likely take precedence, and be interpreted as a backward motion gesture. Another option is for the gesture system to refer to a hierarchical order and conditions for the different effects. The order can be based on simple rules (e.g., most local coordinate systems take priority), designer decisions, or customized by users based on preferences.

Even gestures that are static and symbolic may need to utilize a coordinate system to decipher their meaning. Holding out the thumb with the rest of the fingers enclosed in a fist can be a thumbs-up or a thumbs-down gesture, depending on the direction of "up" with respect to the user.

Temporal The Temporal dimension remains the same as Vafaei with continuous and discrete categories. With a continuous gesture, users can see the effect of the gesture during the gesture. For example, a user pushes a remote object with the palm, and the object moves as the palm moves. With a discrete gesture, the effects occur at the completion of the gesture.

Context from Other Modalities Gestures can work in concert with other inputs or modalities to produce an effect. The most common would be using speech in conjunction with gestures. This would be referred to as gesticulation in early taxonomies [73]. As our taxonomy focuses on gesture, we view speech as an additional context to consider when determining an effect. For example, voice commands and gestures can complement each other and be used to fill in the desired effect in the form of verb-target [159].

Any number of other modalities can be utilized, such as biosignals from the user in the forms of pulse, electromyography, blood pressure, and so on. Once again, as the advancement of technology tracks new body parts and bodily functions, the lens of whole-body interaction can provide guidance on how to incorporate them into our design.

3b.7.3 Additional Contexts

In XR, the user is increasingly becoming part of the computing system.

Taking into account the qualities of XR, we include these additional items to consider and the associated dimensions with each item, such as User, Environment, Target, Audience, and Social Context.

User Whereas previous taxonomies only considered the resulting actions of users and parts of the user (e.g., hands by Vafaei), characteristics of the user can be utilized as context. For example, pulse rate, mood, user posture, body temperature, blood sugar, etc.

While we can view the user as a collection of multiple interactors, much like a hand is a collection of individual interactions (fingers), there are also benefits to considering multiple parts as a whole.

Take, for example, a user in different postures. The user could perform a hand gesture while standing, sitting, lying down, or moving about. We can consider this a compound gesture involving the simple gesture of the hand and multiple simple gestures involving the user's torso and other body parts in different positions. Alternatively, we can view it as the torso and other body parts working together as a compound gesture to modify the user's posture from one value to another. Then, the hand gesture, considering the user's posture along with the dimension of the user, results in a different effect.

Environment The Environment is not explicitly considered as a dimension in previous taxonomies. In XR, however, the user may travel between different environments, whether the travel takes place virtually or physically. The user performing the same finger-snapping gesture may open the garage door in the garage, play smooth jazz in the bedroom, or flush the toilet in the restroom.

This is somewhat covered in the Binding dimension in previous taxonomies. Binding in Vafaei's work refers to the coordinate system that the gesture may be utilizing. However, our definition of the Environment dimension is about the semantic meaning associated with a region of space. Thus, we explicitly separate and define each region. Furthermore, within a region, there could be other smaller or overlapping regions. Where the user stands near a garage workbench may be considered a region. The garage, house, city block, city, and country are all different regions. Once again, the same action may lead to different results.

Lastly, other qualities of the Environment may be considered as context. For example, the ambient volume, the brightness, or the temperature of a room may be used to drive a gesture. As there are too many potential qualities to list, we include them as a catch-all and reminder for designer consideration.

Target The Target dimension deals with whether the gesture is applied to specific targets or not. The target can be designated by prior actions such as a voice command or another gesture. For example, users may say “Light” to designate a target and then proceed to move their hands left or right to adjust the brightness. The target may also be designated in other ways and may not immediately change after the completion of the gesture.

The target can also be designated as part of a compound gesture using a simple gesture. For example, users can look at a light-bulb and move their hands left and right to adjust its brightness.

For additional explanations and examples, see Section 3b.8.

Audience To consolidate our taxonomy with earlier taxonomy work, we consider the Audience as a dimension. The Audience of the gesture in early linguistic-based taxonomy was humans, and humans are capable of dealing with pantomimic gestures that are invented on the spot. Most of the HCI gesture taxonomies deal with simple rule-based programs that we refer to as rigid systems. Rigid systems require the gesture to be defined ahead of time and cannot learn from the user to modify their understanding of the gesture. As technologies such as machine learning and deep learning advance, we may be moving toward intelligent audience agents that are capable of learning from the user and dealing with improvised gestures.

Social Context Ens et al. mention Social Context with three categories: personal, professional, and public. As no detail was provided, we modified these categories to add the "Private" category and offer the following definitions and examples.

The private category refers to a private setting with just the user. The personal category refers to when the user is with friends or other people whom the user trusts. Social groups refer to occasions where users share a space with strangers and/or friends who share a purpose. The public category is when the user shares the space with strangers and/or friends without a shared purpose.

3b.8 Breaking down the Nature dimension

ch3:TaxNatureDimension

Wobbrock first proposed Nature as a dimension and offered four categories: symbolic, physical, metaphorical, and abstract. These concepts are based on early research on gesture taxonomy by Efron[45], McNeill[90], and Karam et al.[71]

- Symbolic: Gesture visually depicts a symbol.
- Physical: Gesture acts physically on objects.
- Metaphorical: Gesture indicates a metaphor.
- Abstract: Gesture-referent mapping is arbitrary.

The nature dimension has been a source of great debate among the authors and led to the most important change within our Gesture Mapping Class. In our taxonomy, we break down the Nature dimension into these additional dimensions: Target, Source of Meaning, Action Mapping, and Effect Mapping. We'll go through how the Nature dimension came to be and how the new dimensions can better classify gestures.

3b.8.1 On the Target Dimension

By its original definition, Physical is the only category that specifies acting on objects, which makes it different from all the other categories. For Physical gestures, Wobbrock says that "Physical gestures should ostensibly have the same effect on a table with physical objects."

However, gestures in the other categories in this dimension could have a target. A thumbs-up gesture is often offered as an example of a symbolic gesture in previous taxonomies. However, imagine a thumbs-up gesture with a target. For example, the user could aim a thumbs-up gesture at an object to show approval as shown in Figure 3b.3. In Wobbrock's taxonomy, this

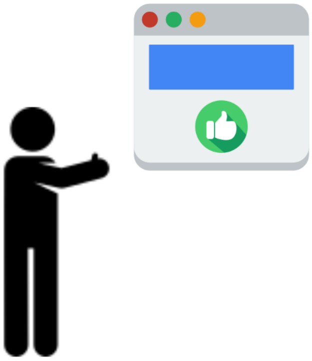


Figure 3b.3: User making a thumbs-up gesture at a window to express approval. A gesture that would be considered as having symbolic nature in previous taxonomies and not used with a target. fig:thumbsup

gesture would either belong to symbolic, meaning it cannot be mapped to a different context, or forced to change into an arbitrary gesture, suggesting it is not depicting a symbol.

Figure 3b.4: User mimicking running with fingers to control a car. A gesture that would be considered as having metaphorical or arbitrary nature in previous taxonomies and not used with a ta fig:fingerrunning

We can see the same issue when considering metaphorical gestures. The user performing a metaphorical gesture with two fingers running on the other hand's palm could cause a target to move as shown in Figure 3b.4. Lastly, by definition, an abstract gesture with arbitrary mapping can also have a target. Then, in the earlier taxonomy, within the Physical category, there is an explicit target, while the other categories, by definition, have an implicit lack of a target. Thus, we extract this implicit hidden assumption into its own Target dimension.

3b.8.2 On the Source of Meaning Dimension

Given that all the categories in the Nature dimension can have a target, what sets the Physical category apart from Symbolic and Arbitrary is how one can make sense of the gestures being performed. Physical, as its name suggests, derives its meaning from an understanding of the laws of physics. If someone nudges a ball, it will roll. In contrast, symbolic gestures, such as a thumbs-up gesture, derive their meaning from shared cultural knowledge. Without that shared cultural knowledge, it would simply be Arbitrary. Thus, another dimension unpacked is the Source of Meaning. The dimension Source of Meaning is the knowledge base for the Interaction Agents draw upon to perform a gesture to convey a desired effect and for the Observer Agent to interpret a performed gesture to produce the appropriate effect.

3b.8.3 On the Action Mapping and Effect Mapping Dimensions

Lastly, we discuss the Metaphorical category. Metaphorical suggests an indirect connection. To make sense of this, we examine the following cases (also illustrated in Figure 3b.5), looking at the action, intention, meaning, and effect.

1. User avatar moves when the user's legs move
2. User avatar moves when the user's fingers mimic legs and move
3. Another character (car) moves when the user's legs move
4. Another character (car) moves when the user's fingers mimic legs and move
5. User avatar moves when the user draws a symbol to represent run, such as a circle or an ideographic character
6. Another character moves when the user draws a symbol to represent run, such as a circle or an ideographic character

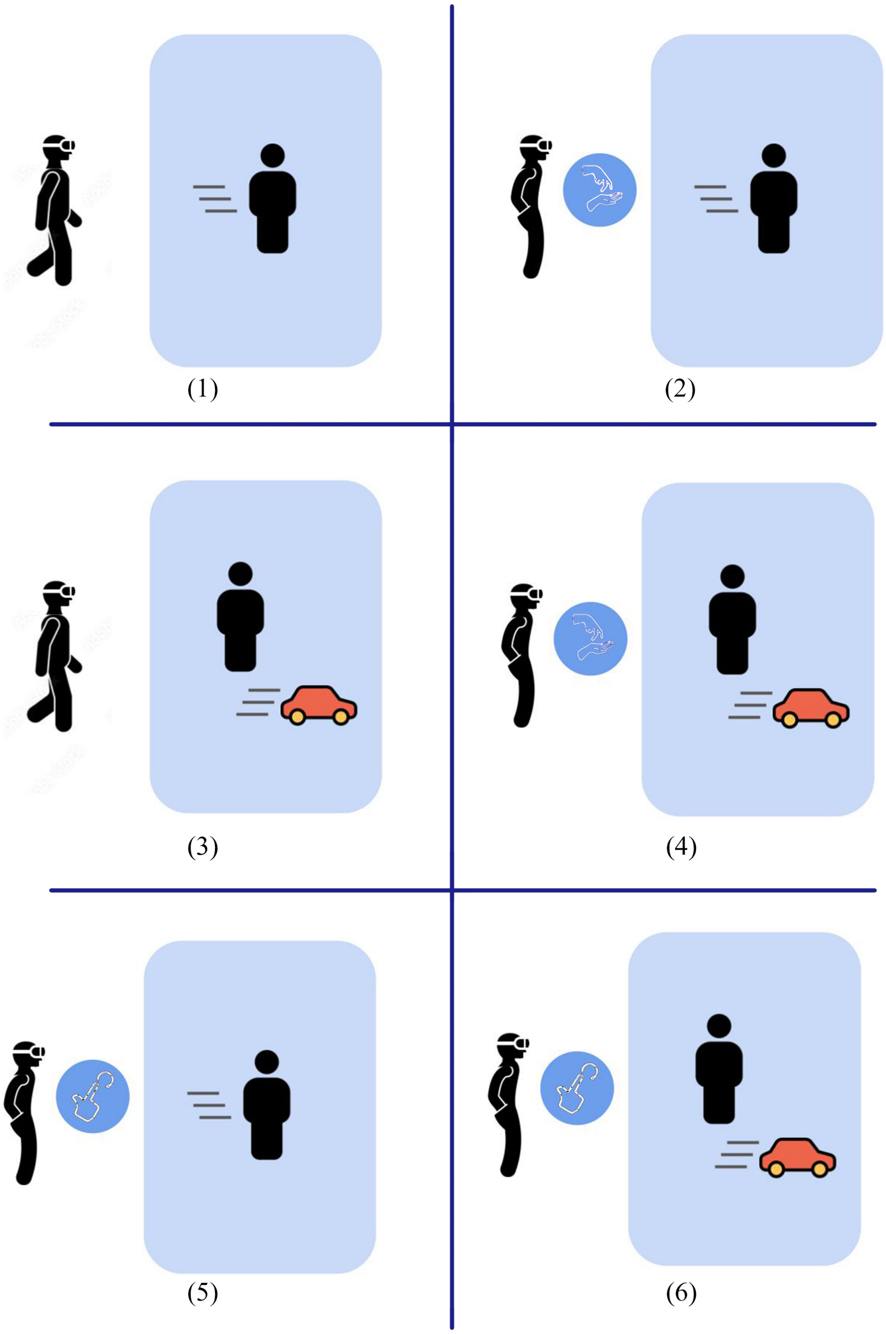


Figure 3b.5: Each image shows a user with a VR headset and shows a blue square representing Virtual Reality. Within Virtual Reality, there may be the player's avatar or a car. Case 1: User avatar moves when the user's legs move. Case 2: User avatar moves when the user's fingers mimic legs and move. Case 3: A car moves when the user's legs move. Case 4: A car moves when the user's fingers mimic legs and move. Case 5: User avatar moves when the user draws a symbol (wheel) to represent run. Case 6: A car moves when the user draws a symbol (wheel) to represent run.

fig:metaphoricalrun

In all cases, the user performs an action to achieve a certain effect. The user utilizes the meaning "run," but conveys that meaning through different actions and uses the same meaning to achieve different effects.

In the first case, it could be considered as a Physical Gesture since the user is acting on the floor object, and the floor object responds by exerting a force back and thus moving the user. In the second case, the action is changed to mimic leg movement using fingers. The difference here is that there is an indirect mapping between the action and the intent. In the third case, we are back to the user running with legs, but a car moves instead. The difference here is that there is an indirect mapping between the effect and the intent. In the fourth case (shown in Figure 3b.4), there's an indirect mapping between the action and the intent as well as the effect and the intent. With the exception of the first case, the rest would all be considered metaphorical gestures. Thus, we argue that metaphorical is an emergent property derived from any of the connections mentioned above being indirect.

Case 5 and case 6 are based on symbolic gestures. As argued earlier, symbolic gestures could also have targets. The difference between cases 1-4 and cases 5-6 is that the source of understanding for 1-4 is physical, and for 5-6 is cultural. One could argue that cases 5-6 are metaphorical and not symbolic. Thus, ultimately, we discard the use of the Nature dimension and replace it with Source of Meaning, Action Mapping, Effect Mapping, and Target with the understanding that the categories in Nature dimensions are simply shorthand for a specific configuration of these dimensions.

This is corroborated by the changes later taxonomies made to the Nature dimension. Ruiz et al.[122] followed Wobbrock's categorization of the Nature dimension, with just slight modifications. Vafaei[139] combined Wobbrock's and Ruiz's work and made an expansion. Vafaei's taxonomy on nature dimension modified Physical gestures to Manipulative gestures, which table:nature

Nature	Source of Meaning	Action Mapping	Effect Mapping	Target
Physical	Physical	Direct	Direct	Has Target
Symbolic	Cultural	Any	Any	Any
Metaphorical	Universal/Cultural	Any	Any	Any
Arbitrary	None	Any	Any	Any

Table 3b.4: Converting the Nature dimension in previous taxonomy into our taxonomy
Note: Symbolic Gestures are also often associated with being static in the Form dimension, which is another hidden assumption. For previous definitions of Metaphorical, either the Action Mapping or the Effect Mapping must be indirect.

followed Pavlovic's [107] and Karam's [71] terminology. Metaphorical was expanded to Pantomimic, which not only contained gestures indicating metaphors of some real objects, but also included all gestures that do pantomimes "to mimic an action to do a task." Additionally, Vafaei introduced Pointing gestures, which were contained in many early linguistic gesture researches [45, 90, 115, 107, 22, 71] as Deictic gestures. The Pointing gesture also has somewhat of a different role from other gestures in that it does not appear to have an effect directly. It simply identifies a target for an effect and is thus often used as part of a complex gesture involving multiple simple gestures. However, identifying a target can be considered as an effect and can be achieved in ways other than pointing. For example, the user may use their fingers to draw the number 2 to identify the object that is marked with the numeral 2. In either case, the environment must work in concert with the gesture to correctly identify the object. The user cannot use a Pointing gesture to target an object that is not spatially present in the environment, just as the user cannot use a number to target an object that is not included in an enumerated list.

3b.9 Comparison on other dimensions

ch3:TaxComparisonOthers

In this section, we describe in detail how the previous works were transformed to create our taxonomy. We have found that past work on gesture taxonomies focused too narrowly on specific areas, such as linguistic scenarios, 2D surface gestures, special applications in VR, etc. There seems to be no effort toward establishing a practicable and comprehensive taxonomy that applies to the XR domain. This work aims to fill this gape while maintaining continuity with previous works. To do so, we are incorporating and modifying concepts from previous taxonomies that can be applied to XR applications, as well as adding some new concepts with significant roles in XR.

3b.9.1 Position and Rotation

Ruiz et al. [122] introduced an important dimension named "Dimension," which was later expanded by Vafaei [139] using the new name "Dimensionality." This dimension "is used to describe the number of axes involved in the movement" [122]. For example, Single-Axis gestures are performed along a single axis. A Surface gesture performed on a screen is a perfect example of a Double-Axis gesture. Tri-Axis gestures involve either translation or rotation in 3D space, while Six-Axis gestures involve both translation and rotation.

For designing gesture interaction in XR scenarios, it is clearly not enough to just consider the above aspects. It is usually necessary to consider exactly which axes are involved in a gesture in XR applications. XR applications run in immersive 3D spaces, so translation and rotation along any axis is possible. Therefore, we split and expand the "Dimension" or "Dimensionality" to Position and Rotation, each containing the x, y, and z-axis, so that XR developers can decide which axes are Relevant and which are Irrelevant.

3b.9.2 From Body Part to Interactor

Vafaei[139] proposed the dimension "Body Part," which "describes which part(s) of the body is/are involved to do the gesture". This is also an important topic in XR, especially in VR domains, since users can perform gestures using different body parts in VR applications. However, a dominant feature of XR is that all objects involved in the interaction between the human user and the computer system are virtually represented in the virtual environment. Any gesture is performed by one or some virtual objects in the environment, which we identify as interactors in XR applications. This is also applicable for the mid-air bare-hand gestures that were widely discussed in previous works, since the hands, the fingers, and all body parts can be viewed as interactors in the virtual environment.

For example, when the user uses a fist to punch an object, the hand is an interactor. If the user closes the five fingers of one hand in order to grasp multiple objects, the five fingers can be viewed as five interactors that work together. In this context, the interactors can not only be body parts, but also be input devices, such as VR controllers, physical trackers, mobile phones, etc. Interactors can also be specific objects in the virtual environment. For example, using one hand as a knife to cut something and "grabbing" a virtual knife to cut the same thing can be viewed as two different gestures, since they can be identified as performed by different interactors, and can have different effects on the target object.

Since we are building a classification of the Interactor dimension from a relatively general level, we ignore some details and specific knowledge of specific interactors. For example, for the gesture of imitating a "V" using two fingers, we may treat the two fingers as two interactors, but others may choose to treat it as a Cultural gesture for "victory." If developers are focusing on the designing and interaction of mid-air bare-hand gestures, those developers might find other taxonomies on that subject helpful, such as [139]. Similarly, those focused on linguistic aspects might want to refer to these previous works[107, 22, 3].

3b.9.3 From Binding to Coordinate System

Wobbrock et al.[147] and Vafaei et al.'s taxonomies include a dimension called "binding", which consists of Object-centric, World-dependent, World-independent, and Mix-dependencies categories. We expand this dimension as the Coordinate System to accommodate XR scenarios. We add the categories Region, User, and Interactor, since they are necessary elements in XR environments related to gesture interactions. There might be different areas/regions in the virtual environment where the same gesture can have different effects, making the Region category an important focus for certain XR developers and users. Additionally, the performer of a gesture can be an interactable object and an interactor simultaneously, while there might be a specific user coordinate system that considers Whole-body Interaction rather than separate body parts. This led us to split the Object-centric dimension into Interactable, User, and Interactor.

3b.9.4 Interaction Context

Ruiz et al.[122] and Vafaei et al.[139] mentioned Context, which is a dimension that "describes if context is needed to determine the meaning of the gesture." These taxonomies only contain In-context and No-context, which is clearly not enough for XR applications, since the context in XR environment can be extraordinarily complex. Therefore, we expand this dimension into the Interaction Context that specifying which contexts are involved in the gesture interaction. Our category names are similar to those in the Coordinate System dimension, but the meanings are very different. The former dimension focuses on what context the gesture is interacting with, and the latter focuses on which coordinate system is used to identify the gesture.

3b.9.5 Other inherited dimensions

In addition to the expanded or modified dimensions mentioned above, we also inherited some dimensions directly from previous works which we find useful in XR applications, with slight or no modifications:

- Form [147, 139] → Form
- Hand Coordination [46] → Multiplicity and Action Symmetry
- Complexity [122, 139] → Complexity
- Flow [147] / Temporal [122, 139] → Temporal
- Social Context [46] → Social Context
- Proximity [46] → User-Proximity

3b.10 Gesture System Architecture

ch3:TaxArchitecture

We propose an XR gesture system architecture based on the Spatial Interaction Model in Section 3a.1 and the gesture taxonomy, shown in Figure 3b.6.

This architecture has two responsibilities that correspond to the two classes in the gesture taxonomy. The first is responsible for detecting gestures based on the class of Physical Characteristics. The other is responsible for the class of Gesture Mapping, which requires the system to track different contexts and how they map to an effect.

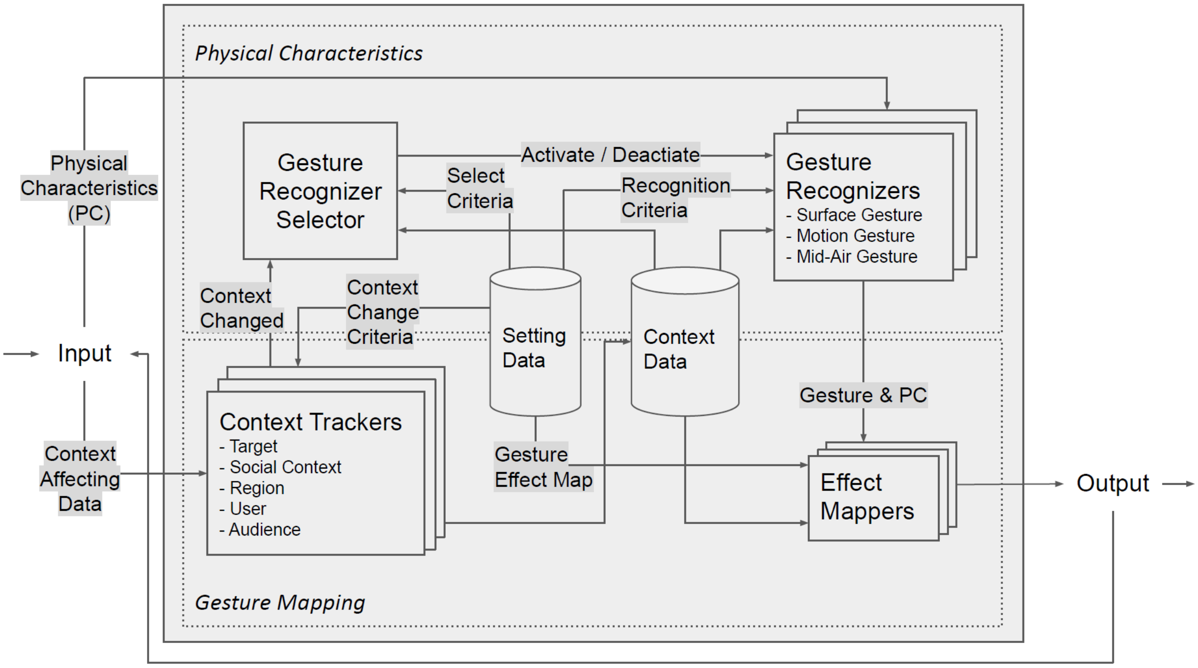


Figure 3b.6: A Gesture System Architecture based on the taxonomy. The modules are divided based on the two classes of Physical Characteristics and Gesture Mapping. Fig:architecture

A dotted region indicates whether a module contributes to Physical Characteristics or Gesture Mapping. Rectangular shapes indicate modules that are responsible for processing data. Stacked rectangular shapes indicate multiple modules are available to process different data. Lastly, the cylinder shape indicates data used by the Gesture System.

Information about the world comes into this Gesture System as Input to the Context Trackers. Context Trackers process these Context Affecting Data to determine the current contexts, which are then stored in Context Data. It then sends an event to inform the Gesture Recognizer Selector which contexts have changed.

Upon receiving the event, the Gesture Recognizer Selector processes the Context Data and Setting Data to determine and set the states of different Gesture Recognizers to be active or inactive.

If a Gesture Recognizer is active, it will process PC Data along with Setting Data and Context Data to determine whether a gesture has been performed. Once it recognizes a gesture, it will send an event containing Gesture Data and relevant PC Data to the Effect Mappers.

The Effect Mapper will look at Gesture, PC, Setting, and Context Data to determine the effects. Each Effect Mapper would be associated with other systems that are capable of producing the desired effects.

Setting Data are used by each of the modules in the architecture. Context Trackers use it to determine the condition when a context would switch, the Gesture Recognizer Selector for selection, the Gesture Recognizer to adjust its recognition criteria, and the Effect Mapper to determine which effect a gesture would cause. There is no explicit arrow going into the Setting Data, as a typical gesture recognition process would not change the Setting Data. However, users could have access to a gesture edit mode to modify the Setting Data at run-time.

The modular nature of the Gesture Recognizers and Context Trackers allows the architecture to support adding contexts and classes of gestures as needed. As not all XR experience will need to utilize all the contexts or gestures, Gesture Recognizers and Context Trackers can also be disabled to save on computing resources.

3b.11 Prototypes of Novel Gestures

ch3:TaxPrototypes

For our ongoing exploration of spatial gestures, we have developed a variety of novel gestures and discuss some here to illustrate how they fit in the proposed taxonomy.

3b.11.1 Virtual Equipment

In the Virtual Equipment System discussed in detail later in Chapter 3d, users are equipped with and can interact with different Virtual Equipment. For example, the user may be wearing earbuds for audio in the real world, but is also equipped with Virtual Headphones located around their ears that can be used to adjust audio settings.

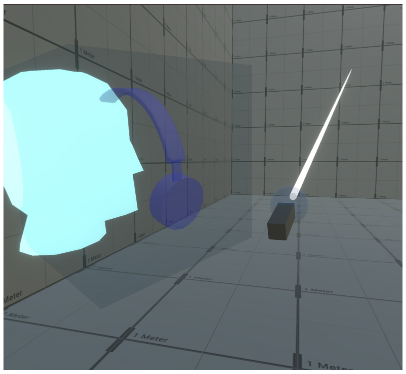


Figure 3b.7: Virtual Headphones are located by the user's ears and allow the user to adjust audio settings. fig:ves11

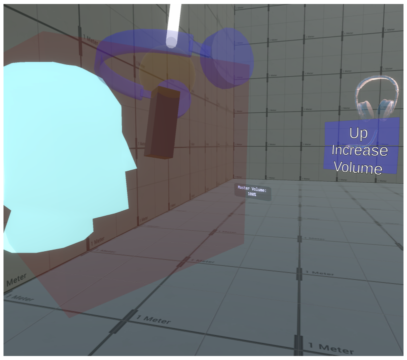


Figure 3b.8: The user performs a motion gesture to increase the audio volume by grabbing the headphone and moving it up. fig:ves2

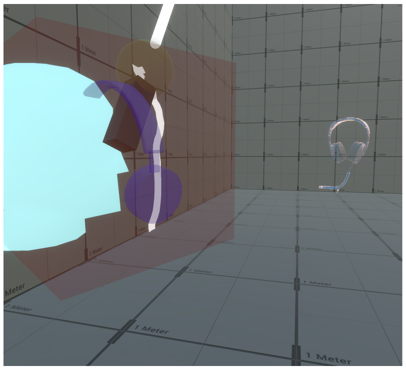


Figure 3b.9: The user performs "surface gesture" to increase the audio volume by holding down the trigger button and drawing a line. figure:ves3

With Virtual Headphones, users can interact with it through motion gestures. To do this, they would "grab" the Virtual Headphones with their hand or their controllers. Virtual Headphones can be moved up to increase the audio volume, down to lower the audio volume, forward to skip to the next song, or back to rewind.

Users can also interact with it through "surface gestures". In the absence of a surface that can provide tactile feedback, users would instead use their hand or the controller to "draw" (indicated by holding down the trigger button) in the space where the Virtual Equipment resides. The user can also perform simple draw gestures by drawing up, down, forward, and back with respect to the user to raise the audio volume, lower the audio volume, skip to the next song, or rewind, respectively.

In either case, the motion of the hand is largely the same. What differentiates the two gestures are the dimensions under gesture mapping, such as the properties of the interactor. To avoid the accidental trigger of the surface gesture, for example, users may need to stick their index fingers out or hold a button on the controller.

Further, consider other sets of Virtual Equipment, grouped together for different purposes. Virtual Headphones are part of a collection of Virtual Equipment that deals with sensory settings (Data Input). Users may also be simultaneously equipped with a set of Virtual Equipment dealing with Privacy Options (Data Output). Other than Virtual Headphones by the user's ear, there may also be a Virtual Ear (Cosmetics) that affects whether others see the users' ears as human ears or pointy elf ears. In this case, the user may perform the same motion in the same space, but utilize different context dimensions in the gesture-mapping class to interact with the different sets of equipment.

3b.11.2 Cube of Wonder

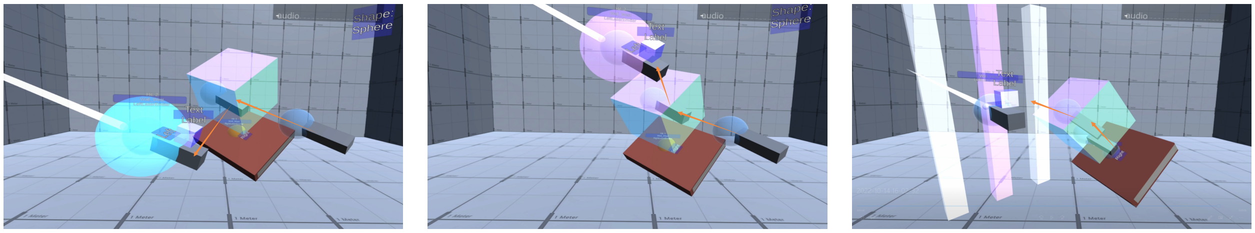


Figure 3b.10: Examples for Cube of Wonder. Entering and leaving from different faces yields different kinds of spells.

fig:cubes

The Cube of Wonder involves interaction between an interactor and the boundary of a space, in this case, different cube faces. The Cube of Wonder can be seen as a 3D implementation of a type of surface gesture involving the user interacting with the surface edge. In 2D, these gestures may be called edge gestures, bezel gestures, or swipe gestures[128, 95]. It can also be seen as a 3D implementation of the crossing-based interface by Accot et al.[2]

One implementation of the Cube of Wonder utilizes two cube faces: the faces the interactor hits when entering the cube and when leaving the cube. The six entry faces determine the element and the six exit faces determine the shape. It provides a total variation of 36 combinations in a small space. When the interactor leaves the cube, it will have selected a spell of a certain element and shape as shown in Figure 3b.10.

This gesture's action involves the interactor moving in 3D space, which in our case is the VR hand controller. This gesture involves the x, y, and z axes of the Position dimension, but the axes of Rotation are irrelevant. The Gesture Mapping involved is the Object Coordinate System and requires the interactor to not be holding an object.

In another implementation, the Cube of Wonder functions as a trigger to start recognizing a motion gesture with objects. Instead of utilizing the entry face and the exit face to determine the gesture, the entry face is only used to activate the gesture recognition system. The exit face is used to stop recognizing a motion gesture. In this case, the cube utilizes the Coordinate System's region dimension. It defines a region in which a specific motion will be mapped to a different result.

3b.11.3 Flick Casting

There is an example of utilizing motion gestures in VR system called Flick Typing. When using Flick Typing, users perform gestures by rotating the controllers in different directions to input different characters. Users can imagine that the characters are evenly distributed on a sphere according to the QWERTY keyboard layout, while the controller (interactor) is located in the sphere's center. Users first need to set up a starting controller status, which is set to point at the key located at the center, and the rotation of future gestures is calculated with respect to the starting status. Then, users rotate the controller to point to different characters to select it. We have applied and expanded the mechanism of Flick Typing to general target selection tasks. We have implemented Flick Casting functionality in VR games, which can select different spells to cast. Players can either select spells with one hand or use each hand to select different attributes of complex spells or skills.

Considering our taxonomy described before, Flick Typing is defined as an object coordinate-based gesture set, where the reference object is considered as a virtual object attached to the starting status. It uses the x and y axes of the Rotation dimension while ignoring the z-axis of the Rotation and all three axes of the Position, i.e. does not care about rolling movement and translation of the controller.

Figure 3b.11: Example for Flick Typing. The left controller is rotating to the right to select the key G, while the right controller to the upper left to select the key Y.

fig:flick

3b.12 Limitations and Future Work

ch3:TaxLimitations

This taxonomy serves as a primary step toward a unified taxonomy of gesture for XR. As XR continues to grow and mature, many additional contexts will likely emerge as significant, and previously unimportant contexts will likely need to be further expanded upon.

We kept the two classifications of Physical Characteristics and Gesture Mapping to avoid unnecessarily creating additional discrepancies from prior work. However, it may be more useful to sort the dimensions based on additional classifications, such as having classifications dealing with the movement, interactor, gesture mapping, and contexts separated.

In this particular iteration, the taxonomy is lacking in the area of multiple interactors. It can likely benefit from a more in-depth look at additional taxonomies of surface gestures. While other works don't deal with the entirety of the multiple interactors, they still often deal with the inputs to the system from those multiple interactors.

We have listed Social as one of the many contexts, but it is also an area that requires much further attention. We have not examined gestures involving multiple users in detail. Multiple users would also naturally involve the under-explored multiple interactors.

3b.13 Conclusion

ch3:TaxConclusions

This section presents a Multi-Dimensional Taxonomy of Gesture for Extended Reality. The taxonomy unifies prior HCI taxonomies of gestures and taxonomies focused on a specific class of gestures. This taxonomy also expands and elaborates on many qualities and contexts that are important to XR. Furthermore, we discuss separating the Nature dimension in previous work into Source of Meaning, Target, Action Mapping, and Effect Mapping to better expose the hidden assumptions. Lastly, we also present an architecture based on this taxonomy as well as some prototypes of novel gestures and demonstrate how they fit within the taxonomy.

With a multi-dimensional approach, XR gesture designers can explore new gestures by examining various different combinations of the dimensions. The taxonomy can serve as a guide to understand better how a given gesture fits in the design space as well as be used in combination with the design lenses to create new gestures or find appropriate existing gestures for the task at hand.

Chapter 3c

School of Spatial Sorcery

ch3:SSS

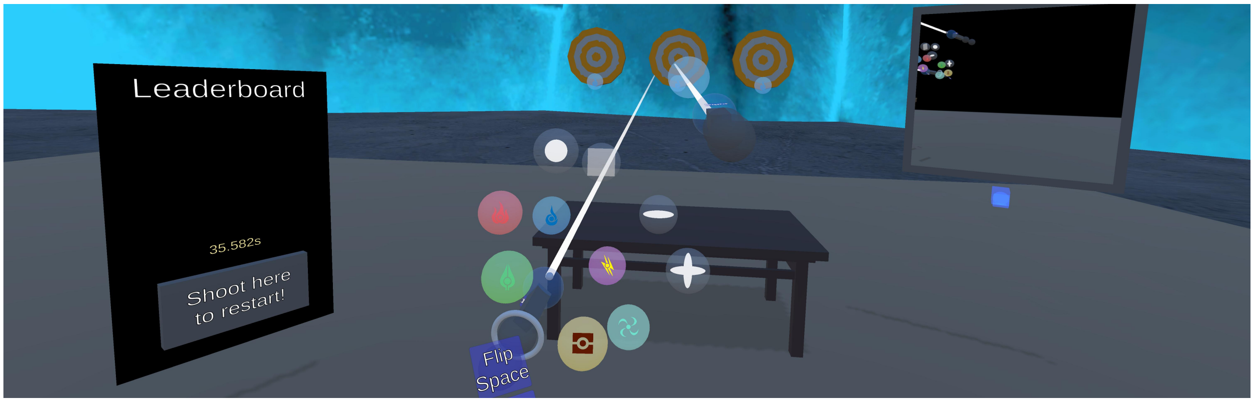


Figure 3c.1: User firing spells at targets using the Arcane Symbols located at the bottom of the left motion controller as provided by the Flip Space Bracelet

fig:sss-teaser

3c.1 Introduction

The School of Spatial Sorcery (SSS) explores spatial interaction within the backdrop of a fantasy setting. Using a fantasy setting for this testing ground instead of the more traditional laboratory environment sets the tone and prepares the users for more unusual hyperphysical interactions that draw inspiration from works of fiction. The user takes on the role of an apprentice spellcaster, learning about a novel spellcasting system with various equipment that encapsulates and provides unique spatial interactions for casting spells. This work's contributions are 1) the novel spellcasting system, 2) the use of virtual equipment to encapsulate user settings and to enable different interactions, and 3) the wide variety of spatial interactions to select or modify a spell for casting.

The project aims to create an environment for developers and users to build, experiment, and learn about the different methods to utilize three-dimensional (3D) space in virtual reality under the guise of a fantasy setting.

To this end, the user is placed in a sandbox gaming environment with instructions on the spellcasting system, the equipment system, and each individual piece of equipment. In addition, the game features a simple shooting range for the user to get familiarized with the different equipment and spatial interactions.

3c.2 Spellcasting System

A spellcasting system in fiction must function like any user interaction system. For a spellcaster to cast spells, the spells need to be translated or mapped from internal thoughts into external effects. This process may involve the spell caster simply thinking about the spell (i.e., Brain Control Interface), saying words to identify the spell (i.e., Natural Language Processing), moving parts around (i.e., Gestures and Spatial Interaction), through the use of an item (i.e., Controllers and other input devices), or other methods.

The spellcasting system in the School of Spatial Sorcery revolves around using arcane symbols to create spells. A spell is, at minimum, composed of an elemental effect and a shape. In other words, element + shape = spell. Minimally, users have access to seven element symbols and five different shape symbols, for a total of 35 spells that can be cast. By selecting different arcane symbols from different types of arcane symbols, the user can modify the spell to achieve different outcomes.

Although the spellcasting system is intended for a fantasy setting and for a game experience, the same system can be applied to any XR interaction tasks in different settings and experiences. For example, 3D painting applications use brushes. The brush is made of different qualities such as color, texture, width, etc. The same spatial interaction could be applied to 3D painting for the user to quickly modify brushes (used as interactables) to take on a new set of qualities. The brushes (used as interactors) can then be used to paint the world.

3c.2.1 Arcane Symbols: Element, Shape, and Spells

Arcane Symbols are representations of spell components within the School of Spatial Sorcery. These symbols help spellcasters channel magic power into spells. In the 3D user interface context, these representations are visualized and made tangible. They can be visualized in many ways. In the basic version of SSS, they appear as translucent spheres with icons embedded inside. The player can form spells using arcane symbols simply by selecting them (e.g., using the Grip Button on the Motion controller to select them).

The user has to grab the Arcane Symbol to indicate that the spell being prepared should have the quality associated with that symbol. As a novice spellcaster and as a part of our minimal viable product, the user has access to three types of Arcane Symbols. They are Element Symbols, Shape Symbols, and Spell Symbols.

Element Symbols: Element symbols bestow properties associated with an element. The six elements are water, fire, plant, earth, wind, and electricity. An element has an advantage over another in the order mentioned above. For example, the water element has an advantage over the fire element. When faced with a monster of earth element (e.g., earth golem), the user

Figure 3c.2: Element Symbols. From top to bottom, left to right: Earth, Plant, Wind, Magic, Fire, Electricity, Water.

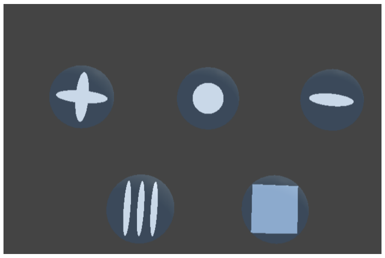


Figure 3c.3: Shape Symbols. From top to bottom, left to right: Cross, Sphere, Pole, Spikes, Wall

should attack it with spells imbued with the plant element. To make the target practice in the demo easier, however, the user simply has to match the element color to the target (e.g., hit the fire target with spells imbued with the fire element)

Shape Symbols: In the basic version of SSS, the novice can use Shape Symbols to modify the form of the spell with a few basic shapes. The shapes available are Spheres (the default projectile), Crosses, Walls, Poles, and Spikes. The different spell shapes change the area of effect that spell would have.

For example, a fire spell can come in the form of a sphere, commonly referred to in fantasy as a firebolt. It could also come in the shape of a rectangular block, commonly referred to in fantasy as a fire wall. These different shapes help the spell achieve different goals. For the purpose of the demo, the wall shape covers a much larger space and can hit multiple targets at

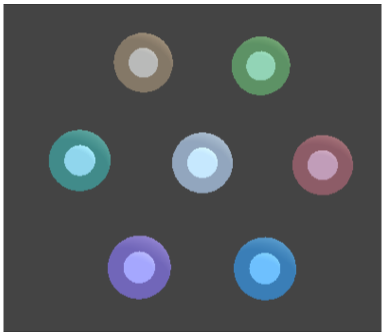


Figure 3c.4: A collection of Spell Symbols consists of the basic elements and the spherical shape.

once. In more advanced implementation, these differences commonly come at the expense of other resources (e.g., mana, a common resource required for casting spells in fiction settings).

Spell Symbols: Spell Symbols are representations of a minimally complete spell. In other words, it has the minimum qualities of shape and element. Once the user selects a Spell Symbol, the prepared spell will take on the quality of the shape and element specified and is ready to be cast. The user can aim and then cast the spell with an appropriate action (e.g., release the selection button). By default, the user can access all the spell symbols representing the standard shape in the various elemental forms.

In the basic version of SSS, the Spell Symbols are primarily found as a combination of the sphere shape and one of the elements. However, Spell Symbols can represent different combinations. For example, Earth + Wall creates a stone wall, Wood + Spikes creates wooden splinter projectiles, Water + Pole creates a water pillar, etc. To truly create a spellcasting system equivalent to those found in fantasy work, however, the players will have to utilize Advanced Arcane Symbols that are still in the conceptual stage.

3c.3 Equipment

In the basic version of SSS, the user will find two classes of equipment: Handheld Equipment and Wearable Equipment. Equipment can confer special abilities to the user, such as teleportation, seeing otherwise invisible objects, accessing storage for items, etc. More importantly, equipment can allow users to cast spells directly or indirectly by providing access to Arcane Symbols.

3c.3.1 Handheld Equipment

Handheld Equipment works as its name suggests. As the user grabs the equipment (using the grip button on the motion controller), it will enable access to different styles of spellcasting. When the user stops holding the Handheld Equipment, the user loses access to the abilities provided by the Handheld Equipment.

Some Handheld Equipment also has additional abilities that can be activated by using the trigger button on the motion controller.

3c.3.1.1 Handheld Equipment - Spellbooks

The most straightforward example of Handheld Equipment would be the implementation of various spellbooks.

Upon picking up the spellbook (i.e., using the Grip Button), the user will see a collection of Spell Symbols floating on top of the spellbook. The user can grab one of the Spell Symbols with the other controller using the grip button and then release the grip button to cast the spell.

There are many variations of the spellbooks. The Simple Spellbook contains seven Spell Symbols that represent the basic spherical shape combined with the different elements. The Combo Spellbook contains all seven of the elemental symbols and all five shape symbols.

3c.3.1.2 Handheld Equipment - Wand of Rotational Casting

Another example of Handheld Equipment is the Wand of Rotational Casting. With this Handheld Equipment, spell selection is modified not through selecting arcane symbols by moving the motion controller to where the symbols are located in 3D space, but by changing the orientation of the wand with respect to its local space. Instead of relying on the position, it relies simply on the rotation of the controller's roll axis. To change the selected spell, the user needs to rotate the wand as it is pointed at the target, then simply push the trigger button to cast the selected spell.

An alternative implementation of the Wand of Rotational Casting relies on a sudden change in rotation (i.e., angular acceleration or angular jerk) to switch to the previous or the next selection. This allows the user to cast spells more ergonomically. The downside, however, is that it requires more physical ability to complete. Another disadvantage is that it may take longer to cycle to the desired selection. For a Wand of Rotational Casting with seven spell options, the desired spell may be as far as three actions away, potentially more if the user does not know which direction to go. This further reduces this technique's usefulness when there are many options to cycle through.

3c.3.2 Wearable Equipment

To take advantage of a piece of Wearable Equipment, it must first be equipped. The equip action is done by grabbing the equipment using the grip button and releasing it in an Equipment Slot.

When equipped, Wearable Equipment will offer new ways to cast spells. This differs from Handheld Equipment in that the user does not need to hold it in their hand. The benefits are conferred so long as the Wearable Equipment is equipped.

If the user no longer wishes for the Wearable Equipment to confer its benefits, they can simply remove it. Alternatively, the user can interact with the equipped Wearable Equipment's User Interface to disable the abilities temporarily.

In the demo, the user will find Wearable Equipment in the form of bracelets that can be placed in the equipment slot by the wrist. Equipment slots are represented as translucent gray spheres located behind the VR controller near the user's forearm.

The demo includes a series of bracelets for locomotion such as Teleportation, Smooth Movement, Snap Rotation, and Smooth Rotation. Users can freely equip as well as enable or disable an equipped bracelet's ability to change the desired locomotion methods.

3c.4 Extradimensional Space

The concept of Extradimensional Space is common in fiction works. Below, we discuss different ways of viewing extradimensional space and interacting with extradimensional space.

3c.4.1 Extradimensional Sight

In the demo, a Monocle of Extradimensional Sight confers the ability to see the world from different perspectives.

Within SSS, there are:

1. Monocle of Spellcaster Sight
2. Monocle of Developer Sight
3. Monocle of Scholar Sight
4. Monocle of Multi Sight

For Spellcaster, Developer, and Scholar Sight, each of these monocles confers the ability to see the world as intended for that particular demographic. The Monocle of Spellcaster Sight allows users to see things relevant to the fantasy experience, such as ghosts, secret doors, or enemy elemental weaknesses. The Monocle of Developer Sight allows users to see and interact with user interfaces for debugging, such as a command console or level reset button. Finally, the monocle of Scholar Sight is useful for scholars from academia. It allows users to see information about an item or a technique, such as relevant works, citations, or a summary of significance to academia. It is easy to imagine monocles developed for other demographics, such as novices or gamers.

Finally, the monocle of Multi Sight is a monocle that includes the different sights mentioned above. The user can interact with the monocle to switch between the different types of visions without juggling between multiple monocles.

These different types of monocles help isolate information intended for different audiences to be accessed through different dimensions. This helps to avoid overwhelming the user and allows users to easily access the information they need as the context changes.

Type	Implementation	Description
Flip Space	Flip Space	Access by flipping an interactable upside-down
Reach Space	Sleeve Space, Cloak Space	Access by passing the interactor through an entry
Sliding Door Space	Lapel Space, Trunk Space	Access by first grabbing a handle to open the space, then by reaching in
Drawer Space	Chest Space	Access by first grabbing a handle to pull out the space, the spatial location of the content changes as the handle position changes

Table 3c.1: Different Types of Simple Extradimensional Space S tab:differentSimpleEds

3c.4.2 Simple Extradimensional Space Storage

Aside from bracelets that provide locomotion ability, a second series of bracelets utilize the Lens of Extradimensional Space to give users access to spells or items.

In popular culture, Hammer Space (often stylized as Hammerspace) refers to a space that fictional characters can access to retrieve or store objects to suit their needs. For example, a large comical-size hammer can be retrieved when there is no reasonable storage on the person that could hold the hammer, hence the name and the attempt to justify how it could be done.

Flip Space, Reach Space, Sliding Door Space, and Drawer Space can all be considered variants and sub-classes of Hammer Space. They differ in terms of the additional requirements that need to be fulfilled before the stored content can be accessed.

3c.4.3 Flip Space

Flip Space is a space made available when the associated object is flipped upside down. One implementation is the Flip Space Bracelet. When equipped, the Flip Space Bracelet provides the user the ability to access spell symbols when the controller is flipped upside down. The spell symbols do not exist in the same space as the user until the flip action is performed.

Figure 3c.5: Characters from the cartoon show Animaniacs about to pull something out of their trousers by reaching

fig:animaniacs-1

The concept of Flip Space can also be useful for providing additional information. Similar to how objects may be flipped in real life to see information placed at the bottom of the object, detailed information for an object could be made accessible when it is flipped upside down. Flipping then becomes a metaphor for seeing additional information that is normally hidden away to reduce clutter.

3c.4.4 Reach Space

Reach Space is a space made available when the user's interactor (e.g., hand or motion controllers) passes through the associated entryway from a specific direction. The space is unavailable if the entryway is passed through in the wrong direction to avoid accidental access. Examples of implementation include the Cloak Space Bracelet and the Sleeve Space Bracelet.

Figure 3c.6: Characters from the cartoon show Animaniacs pulled some paper from fig:animaniacs-2

3c.4.4.1 Cloak Space Bracelet

Upon equipping the Cloak Space Bracelet, the user would gain access to Cloak Space, a variant of Reach Space. Accessing Cloak Space is similar to the user grabbing a worn cloak (or a trench coat) with one hand and pulling it forward in order for the other hand to access the contents stored in the pockets inside the cloak.

In the actual implementation to access Cloak Space, users would move their right controller toward the area underneath the side of the left controller. By moving past a translucent entry area, the user can access the contents stored within Cloak Space that have just appeared.

3c.4.4.2 Sleeve Space Bracelet

Upon equipping the Sleeve Space Bracelet, the user would gain access to Sleeve Space, a variant of Reach Space. Sleeve Space refers to a storage space near the user's wrist. It is accessed by

Figure 3c.7: Characters from the cartoon show Animaniacs storing baloney into the pants by first pulling on the waistband. This is similar to the Trunk Space implementation of the more general Sliding Door Space

fig:animaniacs-3-trouser

reaching below the wrist as if the user is wearing clothes with long sleeves. With access to Sleeve Space, the user could produce objects from seemingly nowhere without needing a stage magician's training.

In the actual implementation to access Sleeve Space, the user would move a controller toward the area underneath the controller in the other hand while the two controllers face each other. By moving the first controller toward the area underneath the other controller past the translucent entry area, the user can access the contents stored within Sleeve Space.

3c.4.5 Sliding Door Space

3c.4.5.1 Trouser Space Bracelet

Upon equipping the Trouser Space Bracelet, the user will find a handlebar in front of their waist area. The user can grab and pull the handlebar outward to access Trouser Space, similar to pulling one's trousers' waistband outward to access the contents stored within that extradimensional space. The user can grab the handlebar and move it back to the starting location to close off the Trouser Space.

Trouser Space is located in an area that can be seen as inappropriate to interact with. While this could potentially be used for comedic effect in media[*], the implication of lewd actions may warrant additional care in implementation. One potential solution is not displaying the user's actions while engaging with the Trouser Space and other sensitive areas. Similar to hiding the interactors when engaged in private matters such as password entry, developers could hide the interactors when accessing Trouser Space. This solution, however, would only prevent the user's action from being seen by others in the virtual world but not in the real world.

3c.4.5.2 Lapel Space Bracelet

Upon equipping the Lapel Space Bracelet, the user will find a handlebar in front of their chest area, similar to lapels on a coat or a jacket. The user can grab and pull the handlebar outward to access the Lapel Space associated with that side. Grab the handlebar and move it back to the starting location to close off the Lapel space. In contrast to Sleeve Space and Cloak Space, where your hand simply needs to go toward that area, Trouser Space and Lapel Space require additional action.

---

[*] See Trouser Space in popular culture `https://tvtropes.org/pmwiki/pmwiki.php/Main/TrouserSpace`

3c.4.6 Drawer Space

Drawer Space is accessed by pulling the storage space out with a handlebar. It is similar to Sliding Door Space. The difference is that the storage space moves as the handlebar moves as it is attached to the handlebar. This has the benefit of giving the user control of where the storage space will be.

3c.4.6.1 Chest Space Bracelet

Upon equipping the Chest Space Bracelet, the user will find a handlebar in front of their chest area. The user can grab and pull the handlebar outward to access the Drawer Space. As the handlebar moves, the storage space and stored content will move along, similar to pulling a handlebar on a real-life drawer. The changing position can allow easier retrieval when using two hands for interaction.

3c.4.7 Simple Extradimensional Storage Space Summary

A wide variety of Simple Extradimensional Storage Space implementations is discussed to showcase the potential of utilizing the same physical space differently via the concept of extradimensional space. For simplicity, the different implementations are given names to help discussion. In practice, Cloak Space and Sleeve Space could be implemented as Sliding Door space like Trouser and Lapel Space and vice versa. However, requiring additional action is important for accessing these spaces located in certain parts of the body to avoid overlapping with objects in the world.

3c.5 Conclusion

The School of Spatial Sorcery is an exploration of spatial interactions using the creation of spells as a substitute for day-to-day interaction tasks. The design lenses of the Hyperphysical User Interface, Whole-body Interaction, and Extradimensional Space are utilized in the design and exploration of different spatial interaction techniques as well as concepts such as equipment and extradimensional space to manage those interaction techniques. Making use of these design lenses of Whole-body Interaction and Hyperphysical User Interface, Wearable and Handheld Equipment enable new abilities for the user. In particular, simple extradimensional space storage exemplifies the usage of all three classes of design lenses, allowing users to hyperphysically access storage space in other dimensions anchored near body parts. Wearable Equipment and Extradimensional Space Storage are further expanded and elaborated on in Chapter 4b and Chapter 6, respectively. The fantasy backdrop bridges the mental gap between what the user can do in real life and what the user can do in XR experiences filled with hyperphysicality. However, as hyperphysicality becomes more common and prevalent in XR, developers can adopt concepts presented here to develop new techniques or leverage these techniques as presented in their own work.

Chapter 3d

Virtual Equipment System

ch3:VES

3d.1 Virtual Equipment System Introduction

The concept of Virtual Equipment System (VES) encompasses a system in extended reality that deals with accessing and interacting with virtual equipment and the system's associated supporting subsystems. VES can be integrated and implemented across different devices and systems to be part of a universal user interface for different extended realities.

Virtual Equipment are virtual objects that fulfill specific roles or functions based on the user's needs. VES makes use of visual metaphors to guide users on how to interact with the equipment and the associated functionalities based on the look and feel of the equipment.

VES redefines what is traditionally considered **Personal Equipment** in several ways. First, we expand what constitutes Personal Equipment by expanding from worn objects in personal space to include objects that are attached to the user spatially, such as objects that reside in the user's peripersonal and extrapersonal space as well as non-physical space. The nature of the virtual objects in XR allows such association to be possible and greatly expands the potential space for storage. We adopt the term **Egocentric Equipment** to describe the equipment in these different spaces.

Figure 3d.1: Concept Art of a user in Mixed Reality utilizing Virtual Equipment. The user is grabbing Virtual Headphones by the ear to adjust volume.

fig:vesconcept

The nature of virtual objects also allows objects that would not be portable in real-life to be portable, further expanding Egocentric Equipment to include what would be traditionally considered Exocentric Equipment, such as a bookshelf or a garage workbench.

For interaction within VES, we defined and implemented object-centric gestures known as Equipment Gestures. Equipment Gestures serve as shortcuts for users to trigger pre-defined functionalities associated with a piece of equipment. Further, the object-centric gestures can be applied to not just the Equipment, but the Equipment's storage location. Combined with the lens of hyperphysicality, the user could reach for the same physical location through different directions to access different pieces of Equipment.

In short, VES is a system that provides XR users with an intuitive and useful interaction paradigm that redefines the relationship between users and equipment. Key Points

- Representation
    - Egocentric Equipment expanded to include Peripersonal and Extrapersonal Space
    - Egocentric Equipment can include traditional Exocentric Equipment and workspace
    - Equipment in alternate space (list-space, tab-space, etc.)
- Interaction
    - Multiple Egocentric Equipment can reside in the same Equipment Slot (same location)
    - Egocentric Equipment in the same physical space but in different dimensions
    - Accessing equipment not stored in physical space via multi-modal interaction
    - Equipment Gestures
        - Surface Gestures
        - Motion Gestures
    - Hybrid Interaction with other techniques
    - Multi-Modal Interaction
- Effect
    - Sensory Setting

  - Privacy Options
  - Avatar Presence
  - Self-Referential Modification

3d.2 Relation with the Design Lenses

The design lenses are used to great effect in the design of VES. This allows Virtual Equipment to have much greater capability than their real-life counterpart. The lens of Hyperphysical User Interface guides the design of VES on all fronts. In terms of representation, Virtual Equipment can float and follow the user. In terms of interaction, the user can perform a surface gesture or a motion gesture on Virtual Headphones to modify audio volume. In terms of effects, Virtual Headphones can be used in conjunction with a target to change the volume of that target. The combination of the lens of Whole-body Interaction and the lens of Hyperphysical User Interface led to the use of peripersonal and extrapersonal space and the semantic association of equipment to the human body (such as Virtual Headphones to hearing). Understanding the human body can lead to additional design opportunities, such as organizing Virtual Equipment around peripersonal space with respect to the arm in addition to peripersonal space with respect to the torso. Peripersonal space organized based on different body parts allows users more ways to re-use the same space around the body. Anything is possible with hyperphysicality, but Whole-body Interaction grounds us in practicality. The lens of Extradimensional Space is used to address the need for multiple different equipment at the same location. Equipment can be worn simultaneously in the same location and made accessible at a moment’s notice by utilizing the concept of switching dimensions.

3d.3 Key Qualities of VES

3d.3.1 Egocentric vs. Exocentric Equipment and Equipment Slot

Oxford Learner's Dictionaries define equipment as "the things that are needed for a particular purpose or activity." This definition is very broad and could refer to objects that are worn on a person as well as large, non-portable objects fixed to a location.

Looking at the term personal protective equipment, we see that it is defined as "clothing and equipment that is worn or used to protect people against infection or injury". Personal here may suggest that the equipment is worn on a person. In VR, the definition of 'wear' becomes muddied. While the user is physically wearing a VR headset, the user can be provided with goggles, headphones, microphones, and other equipment that exists and is worn virtually. These pieces of virtual equipment are based on the location of the headset. These objects follow the user around in the virtual space, but do not have any weight nor do they burden the user.

Virtual Equipment or Equipment generally refers to **Egocentric Equipment**, that is, objects that belong to and follow the user in the virtual space. This is in contrast to **Exocentric Equipment** (denoted as Exocentric Virtual Equipment or Exocentric VE when possible), which stays fixed to their location in the world. Using real-life as an analogy, Egocentric Equipment would be a kitchen apron worn on the user while Exocentric Equipment would be a kitchen oven. Another analogy featuring similar objects would be a shield carried by the user in the hand and a giant shield mounted to the floor. When the user moves, the shield carried in the hand would move with the user while the giant shield mounted to the floor would stay where it is mounted.

A study[75] by Khadka et al. explored the effects of Egocentric and Exocentric Virtual Object Storage Techniques and found that egocentric techniques improved task performance, accuracy, and reduced cognitive workload. The difference between their approach and our proposal is that their work involved comparing objects stored in the environment and objects stored on the user via tracked physical props, whereas we are working without physical props and also consider the equipment to be egocentric, not just the storage.

The portability of objects also changes dramatically when dealing with virtual objects. Weight only exists if it is simulated. This means that previously Exocentric Equipment can now become Egocentric Equipment. In a fantasy game, the giant shield from the previous example can float around and follow the user by virtue of magic, similar to what one may find in the Dungeons and Dragons spell Animated Shield[*]. In this case, the giant shield has transitioned from an Exocentric Equipment to an Egocentric Equipment.

Thus, egocentric in the term Egocentric Equipment only refers to whether the position of the equipment is affected by the user's position as it changes. In XR, the same Egocentric Equipment can reside in many different spaces. As such, we must provide the user with ways to organize these spaces based on psychology's understanding of the human brain.

3d.3.1.1 Mobile vs Stationary

3d.3.2 Personal, Peripersonal, and Extrapersonal Space

There are many ways to organize the space around the user. We look to proxemics, the study of human use of space, to find the relevant terminologies. As the Virtual Equipment System is primarily focused on dealing with a single user at the moment, we do not use the proxemics

---

[*] https://www.dndbeyond.com/magic-items/4571-animated-shield

terminologies regarding intimate distance, personal distance, social distance, and public distance as defined by Edward T. Hall with regards to proxemics in 1963[60]. In this work, we instead refer to neuropsychology's proxemics terminology for the space around the user. W. Russell Brain[17] initially described the existence of a grasping distance and walking distance. The concept of the brain creating a separate and distinct space surrounding the human body is researched in subsequent neurophysiological studies. Finally, Rizzolatti et al. first described the concept of peripersonal space in 1981[118].

In this work, we'll use simplified terminology. The space around the user is divided as shown below into personal space, peripersonal space, and extrapersonal space.

- Personal space: the area immediately bordering the body.
- Peripersonal space: the area that is within the distance of the user's reach.
- Extrapersonal space: the area beyond the user's reach.

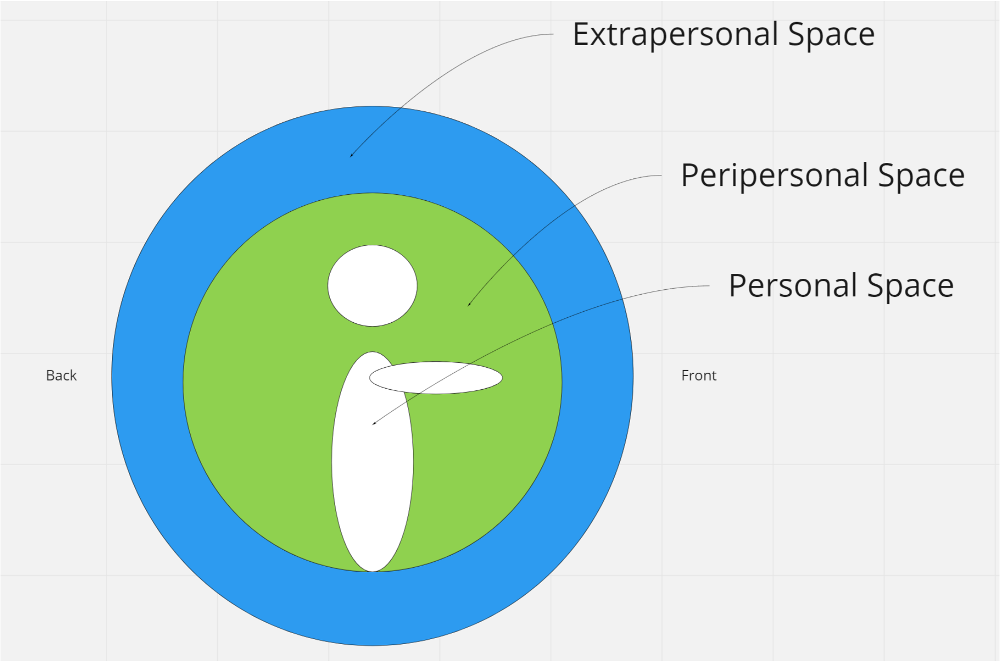


Figure 3d.2: Abstract Diagram showing personal space, peripersonal space, and extrapersonal space. Not to scale and not physically accurate.

fig:differentegocentricspace

These separations of space are divided because they trigger different areas of the brain and with different effects. Physical damage to the different regions of the brain affects the user's perception of the different spaces, giving neuropsychologists a gateway to better understand the human brain.

In this work, however, we use the terminologies to better describe the different types of Egocentric Equipment and utilize prior research findings as inspiration for the features in the Virtual Equipment System. While it is beyond the scope of this work, future works could investigate how the human brain responds to having Egocentric Equipment in these different spaces.

The Invited Review Work[37] by di Pellegrino et al. showed that peripersonal space can be centered around various body parts, such as centered around the hand, head, or torso (body trunk). This provides us with additional opportunities to increase our interaction space. For example, through the use of a gesture like stretching the left arm straight ahead, the user can switch from accessing peripersonal equipment centered around the torso to accessing peripersonal equipment centered around the left arm.

In "Processing of peripersonal and extrapersonal space using tools: Evidence from visual line bisection in real and virtual environments"[53] by Gamberni et al., we learn that a midpoint shift from the peripersonal space and extrapersonal space occurs for subjects in real life as well as in virtual reality. Furthermore, they confirmed that peripersonal space can be extended into extrapersonal space through the use of a tool. This has implications for how VES can be better designed to consider the differences that may exist in Peripersonal and Extrapersonal Space.

3d.3.3 Dimensions, Permissions, Alternate Realities

Multiple Egocentric Equipment can exist in the same physical location (Egocentric Equipment Slot) and different Egocentric Equipment can be retrieved based on different criteria, such as the direction in which the user accesses it, or the current equipment set. These changes can occur anytime and make interaction with different equipment very fluid.

3d.3.4 Active vs. Passive Participation

Active vs. passive participation addresses how much effort the user needs to apply the effect where needed.

Active Participation The user must make a conscious action involving the equipment for the effect to occur.

Handheld equipment commonly requires active participation, such as paintbrushes, pickaxes, swords, and wands.

Passive Participation Once equipped and/or activated, it requires no conscious action involving the equipment for the effect to occur.

In real life, this could refer to clothing, protection gear, a pacemaker, or a hearing aid.

Hybrid Participation Some equipment confers special effects when equipped, but the user can tweak the effect with effort. This is often related to equipment with directional effects. Examples include flashlights, headlamps, and shields.

3d.3.5 Standard vs Cosmetics

Some equipment may be purely for decorative purposes with no other effects.

Purpose	**Active**	**Passive**	**Hybrid**
Detection	Active Omnidirectional Sonar	Passive Sonar	Active Directional Sonar
Protection	Active Protection System in Tanks[†]	Armor	Shield in user's hand
Illumination	Torch	Floating Lantern	Flashlight in user's hand
Offense	Gun in user's hand	Dancing Sword[‡]	Sword in user's hand

Table 3d.1: Example of Active, Passive, and Hybrid Particip

tab:activeVsPassiveEquipment

While one could argue that looking fashionable has a great effect on other humans, cosmetic equipment typically refers to objects that do not confer meaningful interaction or effects within the computing systems. Cosmetic Equipment does not confer new ways to interact nor does it modify existing interactions by changing values used in interaction. When a piece of cosmetic equipment can have an impact on the system, then it is no longer purely cosmetics. For example, wearing a ski mask in a virtual bank could alert the guards.

Cosmetic Equipment is appreciated by gamers, which is why many games have introduced the ability to wear cosmetic equipment that is shown in place of the visuals provided by other equipment. These cosmetics may be known or implemented as vanity items, layered armor, character skins, and Transmogrification. They are sometimes placed in social slots or override slots which will allow the characters to simultaneously wear different equipment. The game character will gain the visual appearance of the cosmetic equipment while retaining the bonuses and effects provided by the normal equipment.

3d.3.6 VES Qualities Conclusion

In Table 3d.2, we provide the different qualities that make up a piece of Virtual Equipment. A piece of equipment can be described as egocentric or exocentric, depending on its spatial relationship with the user. If a piece of equipment is egocentric, then we can further delineate using the quality of personal, peripersonal, and extrapersonal. We can then describe the equipment based on whether it has some functionality or is simply decorative.

User-Centric	**Space**	**Functionality**	**Participation**
Egocentric	Personal	Standard	Active
Exocentric	Peripersonal	Cosmetics	Passive
	Extrapersonal		

Table 3d.2: List of equipment qualities in each category to describe equipment

tab:vesQualities

3d.4 Equipment

ch3:VESequipment

3d.4.1 Equipment Interactions / Gestures

Virtual Equipment can support many kinds of interactions to make them more versatile. While most XR experiences only utilize grabbing equipment and activating equipment with the equipment grabbed, the virtual nature makes it easy to support different kinds of interaction techniques from other areas (e.g., surface gestures from touchscreen devices).

3d.4.1.1 Grab

Grab is a fundamental interaction technique as it performs the role of object selection. The ability to grab a piece of Virtual Equipment allows subsequent interaction techniques that are unique to VES, and thus, it will be briefly described here.

As the user brings the motion controller close to a piece of Virtual Equipment, the user may be given feedback (haptic, audio, visual), suggesting that the user's motion controller has entered a hover state. The user can then use the trigger button to perform a surface gesture. Alternatively, the user can use the grip button to grab the Virtual Equipment, which can then aid the user in performing other interaction techniques (e.g., Alt Node or motion gesture, described below).

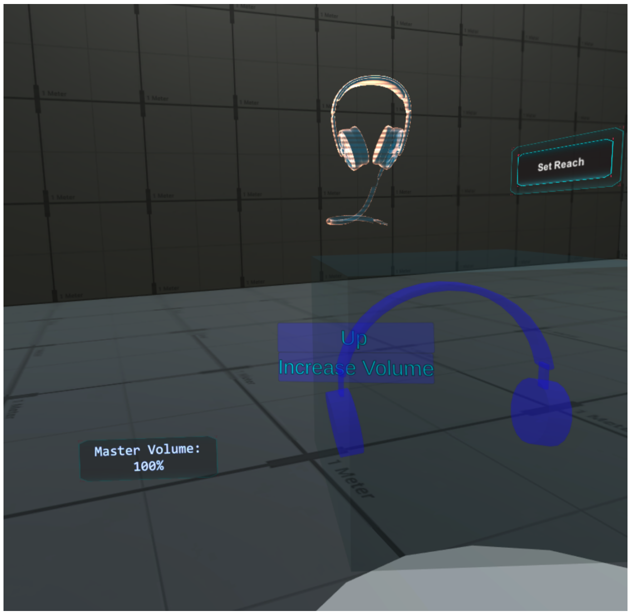


Figure 3d.3: This figure shows what happens when the user performs a Motion Gesture on the Virtual Headphones located at the right ear. Although the user cannot see the Virtual Headphones located at the ear, the user has visual feedback in the form of text showing the current audio volume, text showing performed gestures, and a mirrored copy (light blue) of the Virtual Headphones. Also portrayed are Virtual Headphones (dark blue) stored in the User's peripersonal space. fig:MotionGesture

3d.4.1.2 Equipment Motion Gesture

An Equipment Motion Gesture is performed when the user grabs a piece of Virtual Equipment and moves it in 3D space to perform 3D spatial gestures. Based on prior work with mobile devices[122, 44], we continue to use the terminology of Motion Gesture.

The difference is that the motion gesture applies to a virtual object. The motion of the virtual object may be tracked by physical hardware using accelerometers and gyroscopes (as in the case of a motion controller); it can also be tracked by computer vision and other techniques mentioned in Section 2.4.

In the basic implementation, the delimiter to determine whether the user has performed an Equipment Motion Gesture is based on the distance between a piece of Virtual Equipment and its assigned Equipment Slot. When the user grabs the equipment using the grip button and moves past a certain distance threshold away from the equipment slot, it will register as a motion gesture. To avoid false positives, additional conditions can be enforced to avoid false positives when the user is simply re-assigning equipment to a different slot. For example, it could require the user to re-assign equipment in the VES customization mode. Another solution would be to require the use of both the grip button and the trigger button to indicate a desire to perform an Equipment Motion Gesture.

Given that some Virtual Equipment is head-locked (such as Virtual Headphones) and cannot be seen by the user, VES has support systems to provide visualization and feedback to the player. These visualizations are placed in front of the users during the interaction and come in the form of a mirrored copy of the grabbed Virtual Equipment and text labels. This can be seen in Figure 3d.3 and is discussed in Section 3d.7

3d.4.1.3 Surface Gesture

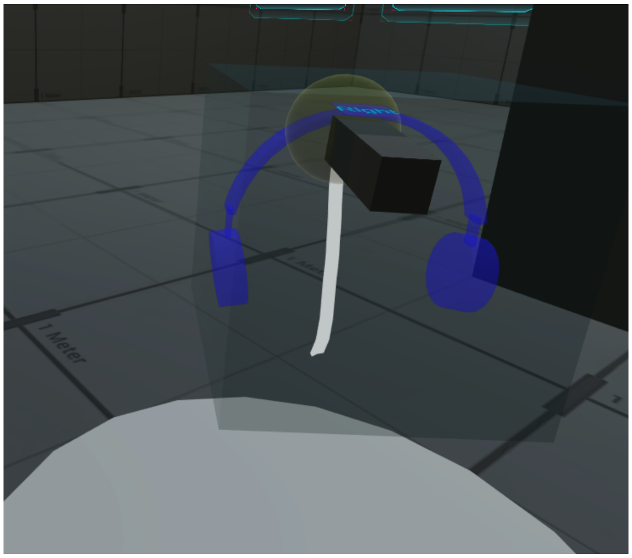


Figure 3d.4: Figure depicts the motion controller drawing up on the 'surface' of a piece of Virtual Equipment. The white trail indicates the starting position to the current position. fig:surfacegesture

An Equipment Surface Gesture is performed when the user's interactor draws on the surface of a piece of Virtual Equipment. In prior work[147, 44], it typically refers to drawing on the surface of a piece of hardware. For example, fingers drawing on the touchscreen of a smartphone or styluses drawing on a tablet surface. However, similar interactions with headphones[§] or a helmet[78] would also be classified as Surface Gesture.

In the current implementation, the user will push down the trigger button while the motion controller is hovering over Virtual Equipment to start a surface gesture. As the user moves the motion controller toward the Virtual Equipment, the user may feel haptic feedback in the form of a weak vibration, to indicate that the motion controller has entered the hover state. In the hover state, a translucent 3D volume (e.g., a cube) will appear to indicate the boundary where the surface gesture will be recognized. The user can hold down the trigger button to perform the surface gesture.

Just like Equipment Motion Gesture, VES has support systems to provide visualization and feedback to the player. These visualizations are placed in front of the users during the interaction and come in the form of a mirrored copy of the grabbed Virtual Equipment and text labels. In addition, there is a translucent 3D volume to show the boundary and a white trail shows the movement of the surface gesture as shown in Figure 3d.4.

3d.4.1.4 Tap Gesture

Tap Gestures (also Whack Gestures) can also be applied to Virtual Equipment. Traditionally, this refers to detecting taps or whacks from outside of a hardware device through the use of an accelerometer [121, 65].

[§] `https://helpguide.sony.net/mdr/wh1000xm3/v1/en/contents/TP0001703116.html`

3d.4.1.5 Alt Node

The user can grab any piece of Virtual Equipment and drop it on an object known as the Alt Node. Inspired by the alt key on a traditional PC keyboard, the Alt Node provides a modifier to existing interactions. For Virtual Equipment belonging to the Sensory Set, it would summon the associated setting menu (E.g., visual setting for Virtual Goggles, audio setting for Virtual Headphones).

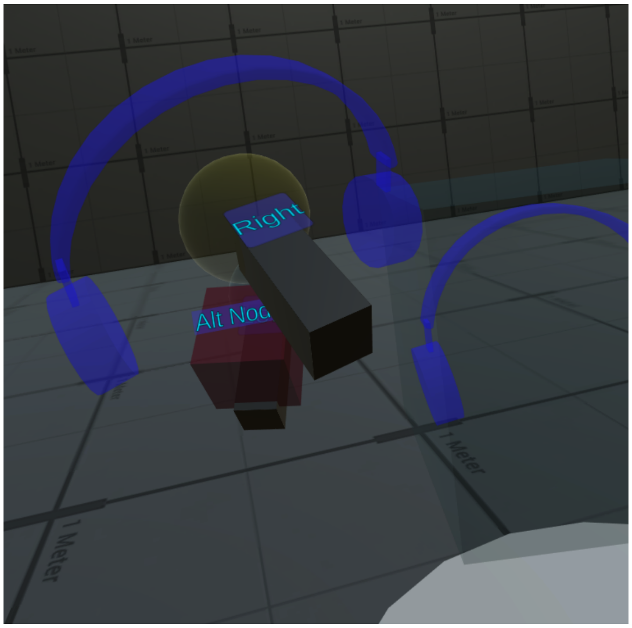


Figure 3d.5: The user dropping Virtual Headphones on the red Alt Node to access the assoc

fig:altNode

3d.5 Equipment Slots

Equipment Slots are storage locations for Egocentric Equipment. Equipment Slots (and their associated Equipment) can belong to personal space, peripersonal space, extrapersonal space, or an alternate space

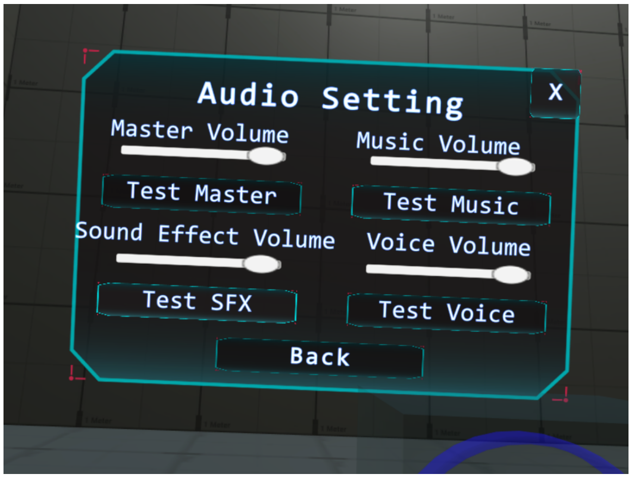


Figure 3d.6: An Audio Settings Menu with multiple volume adjustment options. It can be summoned by the push of the menu button on the motion controller or by grabbing Virtual Headphones and releasing it at the Alt Node.

fig:detailedMenu

3d.5.1 Equipment Slot in Personal Space

Equipment that is located immediately next to the user's body, typically physically touching the user. It may also be referred to as Personal Equipment Slot.

3d.5.2 Equipment Slot in Peripersonal Space

Equipment that is located within the user's reach. Peripersonal space can be centered around a body part, such as the head, torso, or arm. It may also be referred to as Peripresonal Equipment Slot

3d.5.3 Equipment Slot in Extrapersonal Space

Equipment that is located within the user's reach with a tool, such as a ray interactor. It may also be referred to as Extrapersonal Equipment Slot.

3d.5.4 Equipment Slot in Alternate and Extradimensional Space

This refers to any equipment that is stored in a location that is not currently accessible in the user's 3D space, even if the user moves or uses a tool for better reach. No amount of traversal in the 3D space will allow access to that equipment until it is made accessible. These types of equipment do not have a 3D position until they manifest, in which case the equipment itself would enter and belong to personal, peripersonal, or extrapersonal space. It may also be referred to as Non-Personal Equipment Slot or Alternate Equipment Slot.

Some examples may include

- Equipment that is stored and accessed using speech recognition
- Equipment that is stored and accessed using a 2-dimensional user interface
- Equipment that is stored in a different 'dimension' and not currently accessible

An object belonging to a different dimension

Occupied Space	**Egocentric Equipment**
Personal	Headphones
Peripersonal	Floating Weapons
Extrapersonal	Chandelier Figure 3d.7
Alternate	Equipment in a 2D Menu

Table 3d.3: Examples of Egocentric Equipment stored in tab:equipmentInDifferentSpace

3d.5.5 Equipment Slot Location Types

Given our categories of space, there are a few ways to place an equipment slot in those spaces with respect to the user's body.

1. Exact Location
    (a) Tracker

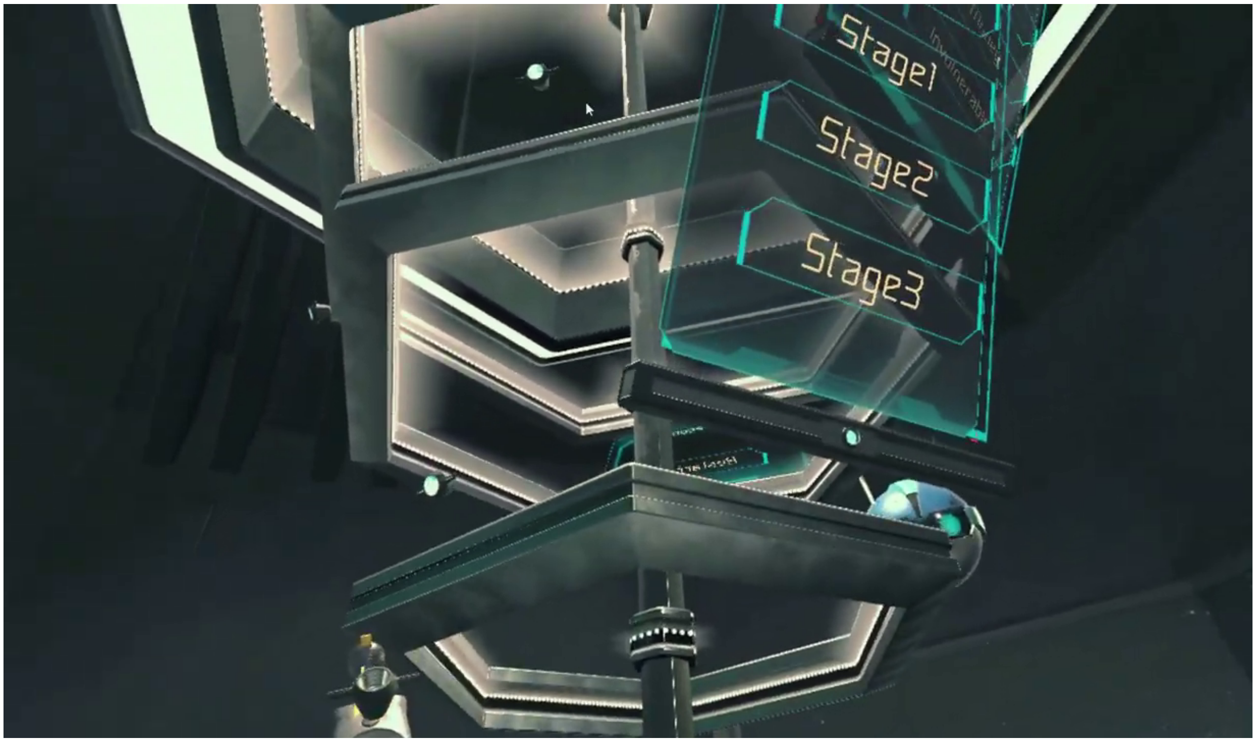


Figure 3d.7: A Chandelier located above the user's head. It is a storage for User Settings and other infrequently accessed menu options. fig:chandelier

    (b) Computer Vision

2. Derived Location
    (a) Simple Offset
    (b) Inverse Kinematics
    (c) Machine Learning Inferred

Exact Location - Tracker With hardware trackers, we can determine the position where a piece of equipment should be with high accuracy. In most VR setups, the user has access to hardware trackers in the form of a VR headset and a 6DOF controller in each hand. The VR headset provides us with spatial information of the headset in relation to the head. With the 6DOF controllers, we can also track the location of the two hands.

Additional hardware trackers could provide the exact location for other body parts, which can then in turn be used for equipment slots.

Exact Location - Computer Vision Computer vision is used in virtual reality for what's known as inside-out tracking technology. Examples such as the Oculus Rift S, Oculus Quest, and Vive Focus utilize the cameras embedded in the headset to track its position in 3D space. Computer vision technology can also determine the location of various body parts. Leap Motion has long been used to track fingers. Meta Quest also uses Computer Vision to track the user's hands with the same embedded camera on the headset.

OpenCV can also track users in 3D space using two or more cameras. As technologies continue to improve, having exact locations for the various body parts will be easier and easier.

Derived Location - Simple Offset From the location of the headset and the motion controllers, we have the position of the user's head and hands with respect to the headset and motion controllers.

We can add offsets to the headset position to get locations around the user's head with high precision. For example, locations for equipment by the user's eyes, ears, nose, and mouth, as well as other uncommon locations such as the temples, back of the head, and straight above the head.

Similarly, with the motion controllers in the user's hands, we can derive the location of the user's wrists.

We can use the motion controller position to derive estimates of the user's forearm position. However, the motion controller does not provide enough information to accurately predict the exact forearm location. Other techniques can alleviate the discrepancy, such as inverse-kinematics of the human body.

We can also use the headset position to derive the waist location using simple spatial offsets for a belt or the side of the hip for pockets. This estimation works well when the user is standing still and facing forward, but fails when the torso and the head are facing different directions or when the head tilts up or down.

Derived Location - Machine Learning Inferred We have mentioned ways to get the location of Equipment using Hardware trackers, however, it can also be possible to use machine learning to infer where the user imagines equipment to be located in their mind which can be different in the actual physical location[93].

3d.6 VES Customization

The design lens of Whole-body Interaction has led to VES making use of all the parts of the user's body. This creates the challenges that VES customization needs to represent spatial relationships of the equipment and the potential detailed options available for the equipment. The end result also has to be practical for the user. In other words, we needed to utilize Hyperphysical User Interfaces to solve these challenges.

Below is an overview of the different approaches and their pros and cons.

3d.6.1 2D List

The easiest way to implement customization in VES would be to leverage the tools and processes to make a list with 2D UI. While simple to implement and the current default in most VR experiences, this implementation cannot capture the spatial relationship between the different slots or equipment.

3d.6.2 Paper Doll System

A common solution to capture spatial relationships is the paper doll system, often found in role-playing games such as the Diablo game series[¶]. The paper doll system allows users to drag and place different equipment in an appropriate slot in order to change the in-game character's stats, cosmetics, or both. It can provide the user with a two-dimensional relationship between the equipment but not their three-dimensional relationships. If there are two objects in the same x and y position but with a different z, the paper doll system would not be able to easily show this to the user. For example, an item at the user's belly button, such as a belt buckle, and an item at the user's back, such as a holstered dagger.

3d.6.3 3D Point of View

To address this issue, we turn to 3D techniques. The first to come to mind would be to modify the equipment locations from the user's point of view. The user would enable customization mode, then the user would be able to adjust the equipment slots by using the grab button to move it to a location that better fits the user.

However, this approach is not without its drawbacks. The users are limited by their bodies. If an equipment slot is placed at a location that the user cannot reach, then the user would have no way of reaching the equipment slot to move it to a location that they can reach. This could be ameliorated using other hyperphysical user interface techniques such as reach-amplification [144].

The user's current field of view could also restrict what equipment the user can see at a time. This makes it difficult for the users to know all the equipment that is available at a glance.

[¶] `https://diablo.blizzard.com/en-us/`

2D/3D	Technique	Spatial Relationship	Con
2D	List	None	
2D	Paper Doll System	2D Restricted	
3D	POV	Full	
3D	Voodoo Doll	3D restricted	Does not reflect actual distance
3D	Voodoo Mannequin	Full, but without the user's body restrictions	Does not reflect how user would interact

Table 3d.4: Different ways for users to organize Virtual Equi tab:waysToOrganizeVES

3d.6.4 Voodoo Doll Technique

In modern-day pop culture, Voodoo Doll refers to an object of ritual magic, typically doll-shaped, where pins can be stabbed into in order to bring pain and discomfort to the person the object resembles. While this differs from its supposed origin in Haitian religion and Vodou magic, the term has entered popular culture and is used to describe this type of magical practice of the imaging magic or sympathetic magic family[12].

The Voodoo Doll Technique[110] can be adapted and used here to associate a humanoid figure with the user. In this particular usage, changes made to the voodoo doll in terms of Virtual Equipment can then be reflected upon the user. For example, a miniature hat may represent a hat for the user. When the miniature hat is placed on the voodoo doll representing the user, the associated hat will also be equipped on the user. When the user removes the hat from the top of its head, the voodoo doll's miniature hat will also be removed.

Voodoo Doll Mannequin is a similar technique, but a user-sized mannequin is used instead of a small doll to represent the user.

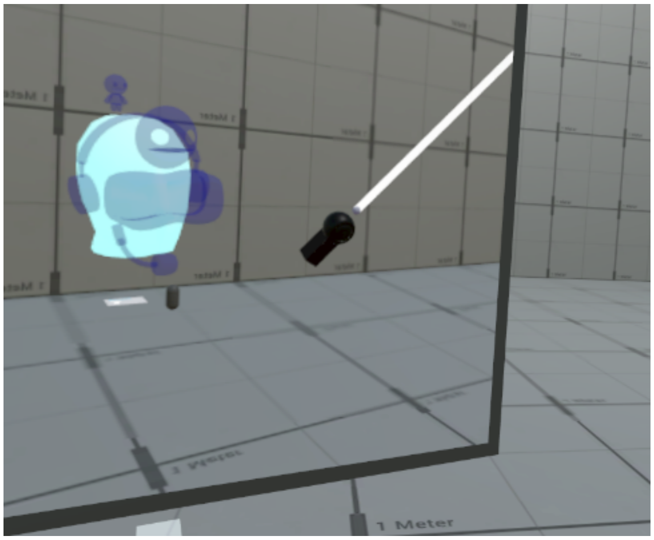


Figure 3d.8: A picture of Mirror Equipment in Action fig:ves-motion-gesture0

3d.7 VES Support Systems

ch3:VESsupport

A series of support systems were developed for the Virtual Equipment System to be intuitive to discover, learn, and use. These support systems do not deal with the placement of Equipment Slots or the Equipment directly but are useful in providing user feedback, organizing Virtual Equipment, or altering the ways to interact with Virtual Equipment.

3d.7.1 Mirrored Equipment & Gesture Tooltips

When users are interacting with certain Egocentric Equipment, they may not be able to see the piece of equipment. For example, when the user interacts with virtual headphones placed by their ears, they cannot see if their controller is hovering over the piece of equipment, no matter how they move their heads. For equipment stored in other locations where the user could turn their head to see, they may not want to move their head to look at it all the time.

Mirror Equipment provides a mirrored copy of the equipment that is shown in front of the user. This helps users with discovery and interaction with Virtual Equipment.

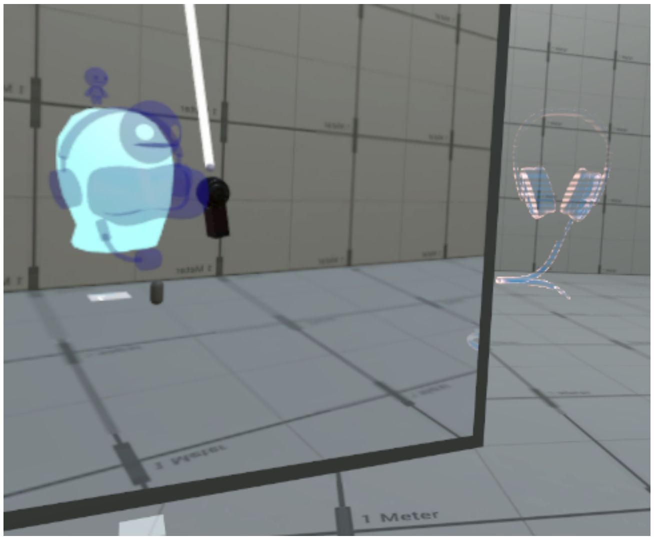


Figure 3d.9: User in front of a mirror that shows the various Virtual Equipment on the head in blue as well as one of the motion controllers

fig:ves-motion-gesture1

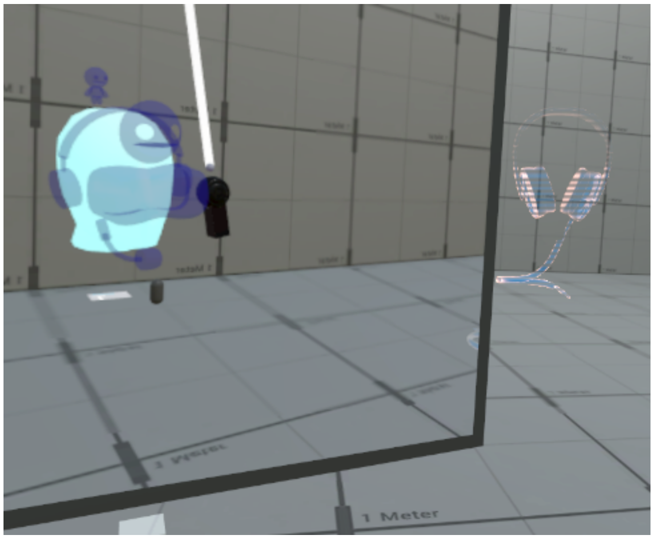


Figure 3d.10: User places the controller near the ear to interact with Virtual Headphones. Mirror Equipment Support System shows a holographic version of the headphone in front of

fig:ves-motion-gesture2

When the user moves their controller onto a piece of Virtual Equipment, they get the typical haptic feedback of weak vibrations to show that they have entered the hover state with an object. Aside from the standard feedback of vibrations, we also provide visual feedback with the Mirrored Equipment System.

When users hover their controller over a piece of Virtual Equipment, they will see a translucent copy of the Virtual Equipment in front of them. This allows users to know what they are interacting with and whether they should proceed from hovering to selecting the equipment.

Once users select the Virtual Equipment that they want to interact with, a text box known as the Gesture Tooltip will also appear by the Mirror Equipment to provide user feedback. The Gesture Tooltip will show which gesture is recognized from the users' motion with the Virtual Equipment as well as what that action this gesture corresponds to in this particular Virtual Equipment. For example, the up equipment gesture for Virtual Goggles may be to increase the screen brightness, while the up equipment gesture for Virtual Headphones may be to increase the audio volume.

The Gesture Tooltip could also show all the gestures and the associated functions to the user so that the user would know what a gesture would do without performing it. By default, this would be triggered after a period of inactivity from the user while in the hover state.

These support systems can be fine-tuned based on the developer's needs and user's preferences. For example, it could simply show the results of the current action without telling the user which gesture is identified.

3d.7.2 Context Layers

An Equipment Set is a collection of Equipment Slots and Equipment. Multiple Equipment Sets could be active at the same time, but for a different context. Within a context, there could also be multiple Equipment Sets that could be used for that context, but only one set would be used at a time. For this purpose, we introduce the concept of context layers in addition to Equipment Sets.

Chapter 3e

Extradimensional Space Storage

ch3:EDS

3e.1 Introduction

Storages in real life occupy physical space. In XR, virtual storages and virtual objects are not constrained by physical space in the same manner. This opens up the possibility of new designs based on the design lenses, particularly extradimensional space lenses. Inspired by works of fiction such as Bag of Holding from the tabletop role-playing game Dungeons & Dragons and the 4th-dimensional pocket from the Japanese comic series Doraemon, we envision a storage system that can store more items and items that are larger than the physical space occupied by the container. Furthermore, users can directly enter this storage space to see stored objects at a glance. This is done by separating the concept of storage from the container and linking the container to another dimension for storage. We refer to this technique as “Extradimensional Space Storage”.

Our work led to the identification of five core components of a storage framework: storage space, container, access space, stored items, and interactor. Various storage systems can be designed to fit different scenarios by varying the properties of these core components.

These core components are then applied to the inventory taxonomy by Cmentwoski et al.[30] We reinterpreted their inventory taxonomy using the five core components and introduced new codes.

3e.2 Examining Common Storage Models

XR storages that deal with virtual objects commonly involve these models of representing storage and items.

1. **Real World Model**: Items stored in their original sizes and placed in an appropriately sized container in the world.
2. **2D Layout Model**: Multiple items stored as their miniature versions (e.g., icons) in a 2D grid layout as backpacks, chests, etc.
3. **Single Symbolic Model**: A single item stored as its miniature and placed in a storage location.

3e.2.1 Qualities of Common Storage Solutions

We use the following terms to analyze these three storage solutions:

Spatial Relationship: Egocentric vs. Exocentric The spatial relationship of the storage system and its relationship with respect to the user is described using the terms egocentric and exocentric. This is similar to Egocentric Virtual Equipment and Exocentric Virtual Equipment. They also share many similarities, as a storage container may be a piece of equipment (e.g., backpacks, utility belts, or dining trays).

Egocentric Storage refers to storage with a spatial relationship that is fixed to the user. Egocentric Storage moves as the user moves. Examples of Egocentric Storage include holsters, pockets, messenger bags, etc.

Exocentric Storage refers to storage that is not fixed to the user location. Commonly, Exocentric Storage stays in the same location when the user moves. Exocentric Storage may be moved by other means, such as when it is attached to an entity that is different from the user. Examples of Exocentric Storage include closets, bookshelves, chests, torch sconces, etc.

With virtual storage, it's not hard to imagine the user picking up a virtual bookshelf storage and moving it around with the same ease as a small purse. Furthermore, the user could grab the bookshelf with one hand and reach for the books in another, turning the bookshelf from an Exocentric Storage into an Egocentric Storage. With larger storage spaces such as a virtual workshop (e.g., garage workshop), the user would likely have to set it down and move toward the desired item or otherwise use different interaction techniques. That is to say, egocentric and exocentric exist on a spectrum, and the terms are applied broadly for simplicity.

Manifestation: Original vs. Symbolic Manifestation deals with whether objects are modified in their manifestation when stored within the storage. **Original Manifestation** When objects are stored in this storage system, the objects are represented as is. It is stored without changes to its shape, color, and other visual qualities. In these virtual qualities, however, the functional qualities are often altered. For example, a virtual sword stored in a virtual storage will typically be unable to cut and slice other virtual objects. This is different from a real-life sword, which will cause damage to nearby objects unless stored within a special container such as a sheathe.

Table 3e.1: Examples of common storage models

examples_table2

Example	Spatial Relationship	Manifestation	Capacity
Weapon Holster	Egocentric	Original	Single
Torch Sconce	Exocentric	Original	Single
Closet	Exocentric	Original	Multiple
Backpack	Egocentric	Original	Multiple
Backpack with 2D Menu	Egocentric	Symbolic	Multiple
Chest	Exocentric	Original	Multiple
Chest with 2D Menu	Exocentric	Symbolic	Multiple

Symbolic Manifestation When objects are stored in this storage system, these objects are represented symbolically. This may be text, a 2D icon, or a miniature version of its 3D shape. Similar to the original representation, they often lose their functional qualities while stored.

Capacity: Single vs. Multiple Single and multiple refers to the quantity of stored items that the storage solution can handle.

Typically, storage solutions that can store multiple items use multiple storage slots. For example, a virtual backpack in VR may have a grid layout, with each grid being a storage slot. This makes it easier for users to grab the desired store item. The storage slots in a backpack can be considered as its own storage system or as part of a whole, depending on the mental encapsulation desired by the developer. Each slot can be viewed as a single-capacity storage solution. The backpack can be viewed as a multiple-capacity storage solution, with the multiple-capacity for stored items, provided through the use of storage slots. Alternatively, the backpack can be viewed as a multiple-capacity storage solution for storage slots, with each slot being able to handle one stored item.

3e.2.2 Comparing Common Storage Models

With the different terms, we can now examine the pros and cons of these models.

Real-World Model The Real-World Model is straightforward and easy for the users to learn. The interactions work similarly to real life. Users can use their interactors (e.g., hands, motion controllers, etc.) to grab the item. This allows users to utilize their hand-eye coordination skills, spatial senses, and possibly spatial memory.

2D Layout Model The 2D Layout Model is brought over from the WIMP computing paradigm. It typically compacts information in a much denser but organized configuration and thus uses less physical space. This means items can be retrieved easily physically. However, it is more difficult to utilize spatial memory, especially when items can be shifted around when adding and removing new items to the storage solution. Multiple pages can further add to the difficulty of remembering the location of a desired item[*], resulting in the user having to perform a visual search for the desired item.

Single Symbolic Model The Single Symbolic Model is similar to the Real-World Model, but the stored content is represented as symbols. Some examples are the wrist pockets in Half-Life: Alyx [54] or guns stored in holsters in many VR games. While objects are altered and occupy less space while in storage, users can still use their spatial sense and memory. That is because only one object is stored in the storage, and its spatial relationship does not change with respect to the storage. However, the spatial relationship between the storage and the user can still change. Furthermore, its limited stored item capacity means it should be reserved for storing and retrieving the most frequently used item only.

The three common models are common in XR games and experiences, particularly the 2D Layout Model, due to the wealth of resources established by the WIMP paradigm. Few examples, however, use the third dimension, spatial sense, and spatial memory for storage management.

---

[*] The author challenges you to recall the 3rd icon on your desktop

This restricts the interaction techniques that can be utilized in interacting with the storage system.

To bridge the gap between the three common but different cases of storages, we use the design lenses of Hyperphysical User Interface and Extradimensional Storage to create a new storage technique. This storage technique allows the user to easily switch between the different approaches. This storage would occupy less physical space than the items stored inside. It allows the user to enter the storage as a room to utilize their spatial senses and spatial memory if needed. This approach and series of techniques is referred to as Extradimensional Space Storage.

3e.3 Related Work

The work of Cmentowski et al.[29] illustrates a method to manage items in XR using a grid layout. This technique helps users manage their items efficiently since items can be placed in the grid structure based on where the user puts them. However, its disadvantages are also clear. It works like the typical inventory systems in non-VR games, using icons or models to replace the real item.

Additionally, according to the work of Wegner et al.[143], another approach allows users to place items on their body (e.g., at the waist). This approach brings spatial memory into the interaction. Also, it is easy to use because it works like the real world. However, its disadvantage is that the number of items it can hold is limited since there aren't many spots on the user's body that can be used to store items.

The game Fantastic Contraption motivated us to think about the design lenses of Extradimensional Space. Figure 3e.1 shows the user looking at the world inside the helmet (the dark world

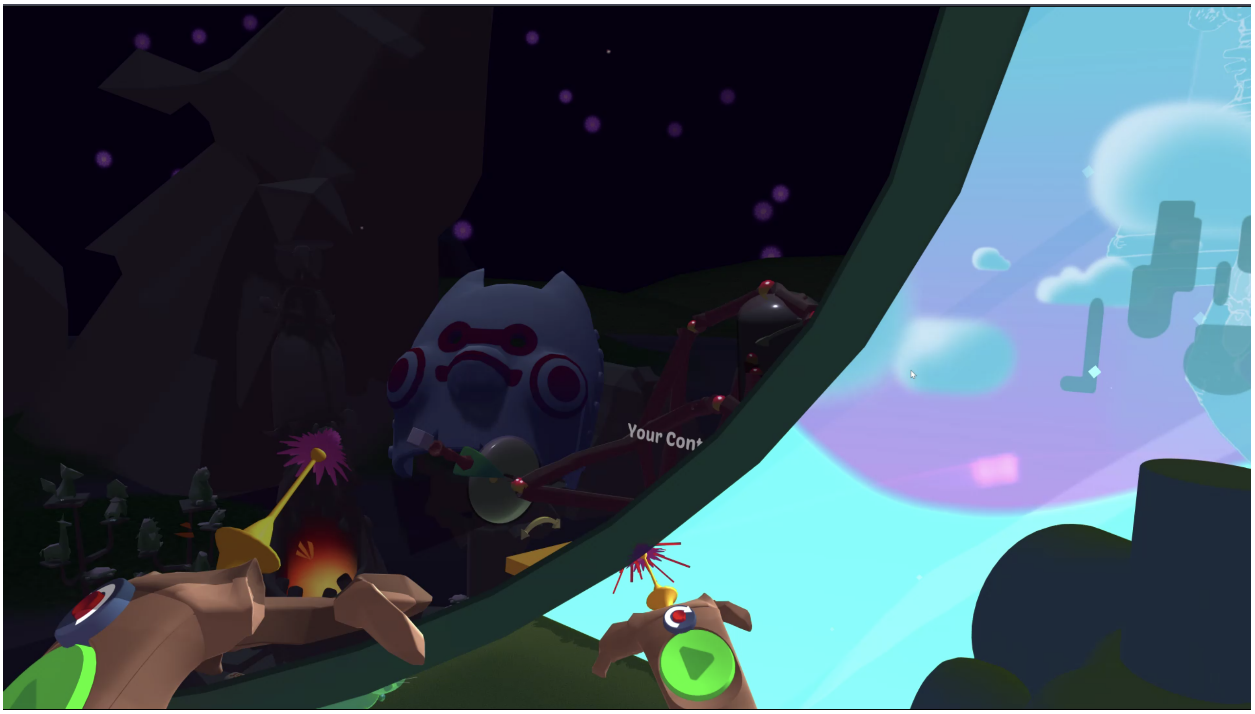


Figure 3e.1: User in the game Fantastic Contraption sees two different locations. On the left, the user's hand interacts with the world inside the helmet. On the right is the world the user and the helmet reside in.

fig:game

to the left) alongside the world that the user and their helmet reside in (the colorful world to the right). In this game, the player can put a helmet on the user's head to enter a different location in another dimension. That location functions as a settings menu and a level selection menu. When the player removes the helmet, the player's view will revert back to the original world. The helmet provides the user access to another space and other user interfaces without physically relocating.

According to Tavanti et al.[136], sustaining a general superiority of the 3D display for the chosen spatial memory task is feasible, even when compared to a 2D non-scrolling display. Thus, to bring spatial memory into XR, we combine the existing techniques of storage interaction and world interaction together. For some scenarios where efficiency is the top priority, the user will utilize the traditional class of storage interaction techniques. At the same time, users have the option of entering into a different world and fine-tuning their storage interaction techniques or making full use of their spatial sense to interact with objects in personal, peripersonal, or extrapersonal space from within the storage space, similar to how the user can enter another location in Fantastic Contraption.

3e.4 Extradimensional Space Storage

Extradimensional Space Storage (EDS Storage) refers to a storage system where the storage space is much larger than the space the storage container takes up. The extra space does not exist in the 3D world that the user occupies and no amount of movement in the 3D space will bring the user to that location. Thus, the storage exists in a separate dimension or a parallel universe, and we refer to it as extradimensional based on the popular culture definition. These and similar concepts, such as hammer space, are discussed in Section 2.5.

3e.4.1 Core Components of a Storage Framework

In our design and implementation of the Extradimensional Space Storage, we have identified five core components and their associated properties that make up a storage system. They are:

1. Interactor
2. Access Space
3. Container
4. Storage Space
5. Stored Item

Of the five core components, an implementation of an individual storage system (instance) would require three of the components: the container, the storage space within, and the access space. Stored Item and Interactor components are required to interact with the storage system and for the framework as a whole to function. They are not required for an individual storage system to function. For example, an empty box would have no stored items and no hands trying to grab items. However, the stored item and interactor are important for the empty box to function as a storage system. Otherwise, the empty box would be a visual cosmetics with no function.

In the case of EDS Storage, the storage instance comprises three components: the outside container, the inner Extradimensional Space used for storage, and the transitional space between normal dimensional and extradimensional space called access space. The access space is a physical manifestation of an interface to access the storage system.

Each of the five components can be represented differently and have unique properties that greatly affect how users interact with the storage.

3e.4.2 Taxonomy of Inventory Systems

The taxonomy proposed by Cmentowski et al. [29] can be linked to the five core components of a storage framework. Figure 3e.2 and Figure 3e.3 are figures modified based on the original figure. Different background colors are applied to the open codes, concepts, and building blocks to illustrate how the core components fit in and how they can be implemented. Further, new additions are added to the taxonomy, which is shown with a red border.

3e.4.3 Gestures Using Physical Objects

Of the five components, the interactor, container, and stored item exist as physical, tangible objects. However, they can benefit from the hyperphysical design lenses. For example, these physical objects can be used as interactors in motion gestures or surface gestures.

Given a tracked interactor (e.g., hands or controllers) and an interactable (e.g., containers or stored items), the system can produce different effects when the interactable is moved (as the tracked interactor moves). In other words, when a motion gesture is performed on the interactable. The interactable itself will not move for surface gestures. Instead, the interactor will be in contact with the interactable's surface and move to perform a surface gesture. For example, on a mobile phone, we recognize shaking the phone as motion gestures and swiping on the screen as surface gestures.

In XR, users aren't limited to dedicated hardware devices to perform spatial gestures on objects. As long as the virtual object is tracked (whether by computer vision, proxy via motion controllers, or other techniques mentioned in Section 2.4, the virtual objects can have their own spatial gestures. Just like the Virtual Equipment System in Chapter 3d, these storage solutions can incorporate spatial gestures such as motion gestures, surface gestures, tap gestures, and so on without needing hardware like accelerometers, gyroscopes, or touchscreens.

3e.4.4 Interaction Zones

In contrast to the previously mentioned tangible objects (containers, stored items, and interactors), the storage space and access space are not physical objects. They are areas in space that can define new behaviors for interactions with physical objects. Whereas performing a controller gesture of up might not do anything in one space, it might retrieve the last item placed in the storage when performed inside the access space.

We refer to these areas with different interaction opportunities as Interaction Zones. In other words, the interactor, container, and stored item can interact differently when inside the storage space (Storage Interaction Zone), inside the access space (Access Interaction Zone), or outside of either space (Default Interaction Zone).

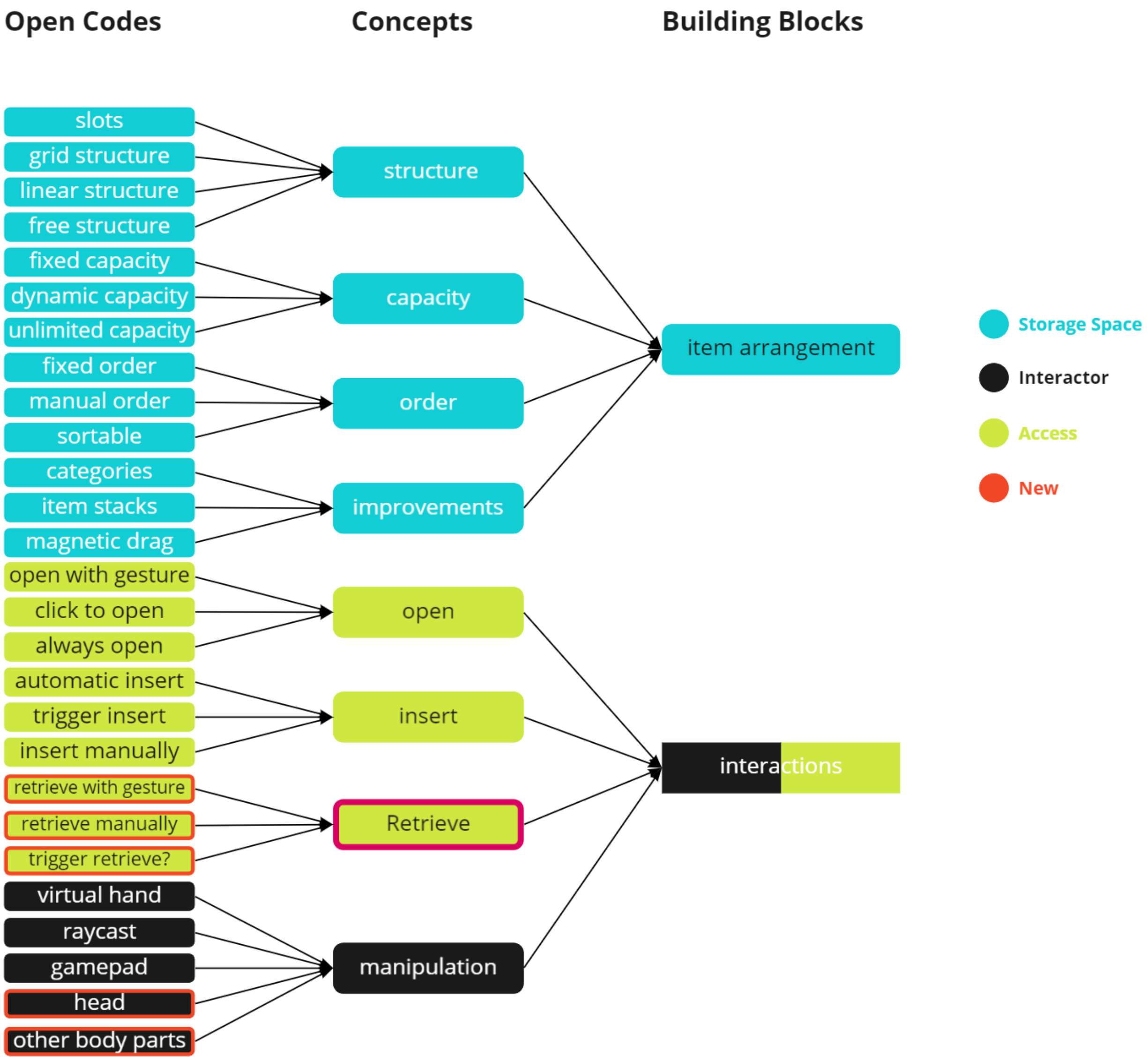


Figure 3e.2: Taxonomy of Inventory System Part I, modified to reflect the five core compo fig:taxonomy1

The interactions users can perform in an Interaction Zone are thus referred to as Zone Interactions. Zone Interactions may include the previously mentioned Motion Gesture and Surface Gesture that make use of interactables. However, the zone in which these gestures are performed can be used as another context to alter the outcome of the gesture as described in Chapter 3b. In the absence of an interactable, there are zone gestures. Zone Gestures are a subset of Zone Interactions that we focus on in EDS Storage. Zone Gestures are performed

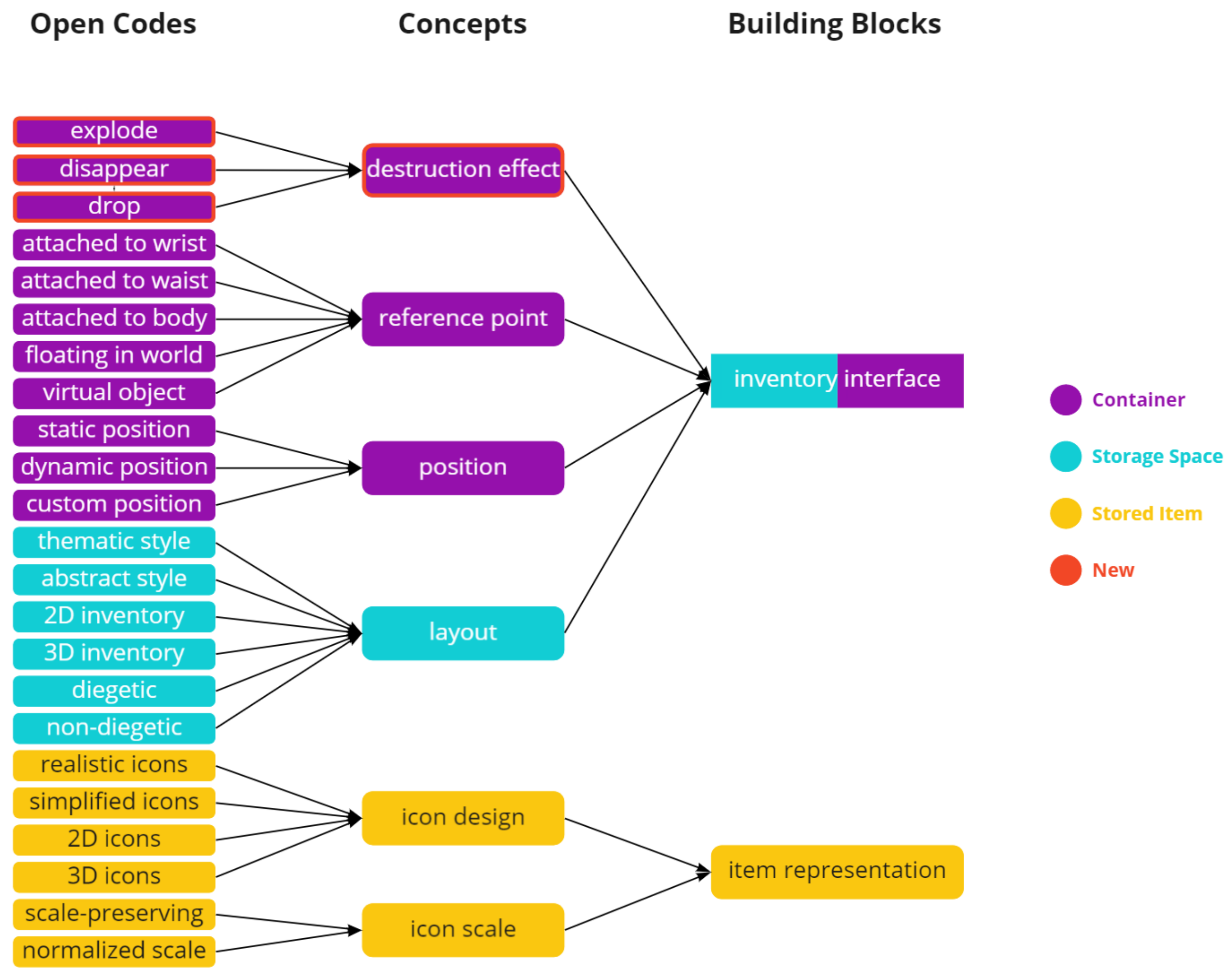


Figure 3e.3: Taxonomy of Inventory System Part II, modified to reflect the five core compo fig:taxonomy2

when moving the interactor without involving an interactable. Instead of having an interactable as the additional context, it only has the zone as the additional context.

3e.5 Core Components of Extradimensional Space Storage

In this section, we describe each of the five core components and their associated properties, emphasizing the implementation of EDS Storage. We discuss some of the most representative properties as well as special cases or alternate usages to illustrate the potential of this storage framework.

The examples we have mentioned in Table 3e.1 are actually properties belonging to different components. By analyzing the different properties of these different components of a storage framework, we can design and implement other storage solutions by manipulating these properties for different purposes.

3e.5.1 Interactor

An interactor is a component that interacts with the storage system, often a representation of the user actuators. While it is common to use the user's hands or to substitute VR controllers as interactors, it is also possible for other body parts to be used as interactors (e.g., head, feet, knees, etc)

3e.5.1.1 Interactor Properties

interactor_properties

1. Direct vs. Remote Interaction
2. Physics Collider
3. Motion Gesture
4. Surface Gesture

For example, when the interactor is inside the storage space, its motion gesture may be adapted to retrieve multiple items. When the interactor is inside the Access Interaction Zone, the motion gestures are replaced with a different set of motion gestures that allow the user to retrieve items known as Quick Access, described in detail later.

3e.5.2 Access

Access refers to a component that allows the interactors to utilize the storage system. In other words, it's the user interface of the storage system. In EDS Storage, the space between the inside of the container and the outside world serves as the physical manifestation of Access, known as Access Space. Access Space represents this transition space that allows users access to the extradimensional space storage. In this case, the user can access the EDS Storage with their hands or the whole body. Access Space is also called Entry or Exit in previous work.

Access Space could be a bag opening, a mirror surface, or even a tattoo. The concepts of Interaction Zones and EDS Storage allow the user to interact more within a small confined area, allowing these unusual objects to serve as both the container and the Access Space for storage. The user only needs to be able to move their interactors to the Access Space to interact with the stored contents.

3e.5.2.1 Access Properties

Access Properties (or Access Space Properties in the case of EDS) determine whether the storage is available for interaction. Access Properties also determine what happens when an interactor interacts with a physical manifestation of the access, such as an Access Space. A storage system can also have multiple different ways of accessing the storage, such as via different Access Spaces to gain entry or exit.

As Access Space is an area of space, it provides modifiers to interactions that interactors can perform. These new interaction opportunities are called Access Zone Interactions. Access Zone Interactions allow for new interaction opportunities or Quick Access and modification of its associated properties.

access_space_properties

1. Access Space Effect on Interactors (Figure 3e.4)
    - (a) Regular size
    - (b) Enlarged
    - (c) Non-linear mapping
2. Quick Access
    - (a) First In First Out (FIFO)
    - (b) Last In First Out (LIFO)
    - (c) Gesture-based
    - (d) Voice Recognition
3. Manual Management
    - (a) Grab spatially only
    - (b) Grab sockets only

3e.5.2.2 Access Properties on Different interactors

When the user enters the Access Space with interactions (such as the hand or the whole body), different Access Properties may apply to the interactors.

Figure 3e.4 illustrates three potential scenarios when accessing stored items. In the first case, the user's hands and arms are of their regular size. This means the user can only grab stored items near the Access Space, assuming the Storage Space is much larger. In the second case, the user's hands and arms are enlarged so that when the user's arm is all the way in the container, it will be just far enough to grab the item at the far end of the Storage Space. In the third case, the arm size remains unchanged, but the movement of the hand is modified by reach bounded, non-linear input amplification (such as the work of Wentzel[145]). This allows the

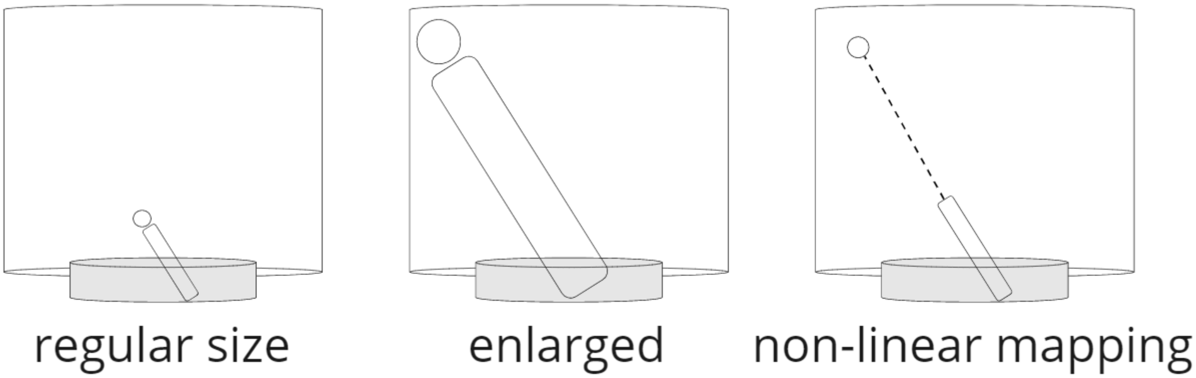


Figure 3e.4: Three different reach settings for users to access items stored in the storage space. The rectangular shape represents the users' arms. The circular shape represents the users' hands/control fig:arm

hand to travel a different amount of distance in the Storage Space as the hand moves in real world space, ultimately allowing the user to grab items at the far end of the Storage Space.

Similarly, while the user is outside and using Quick Access, the user may want to use different strategies or even multi-modality to access the items. By modifying the Access Properties for Quick Access, we can allow the users to use gestures, voice recognition, or other means to access stored items.

3e.5.3 Container

The container represents the part of storage that exists in the user's original dimension. It could be a bag, a cup, or even a pack mule. The container may follow physical rules in its original dimension but still contains hyperphysical and extradimensional storage space.

3e.5.3.1 Container Properties

Container Properties describe what happens to the container (and, optionally, its content) when the container is affected by other interactors.

1. Egocentric vs Exocentric

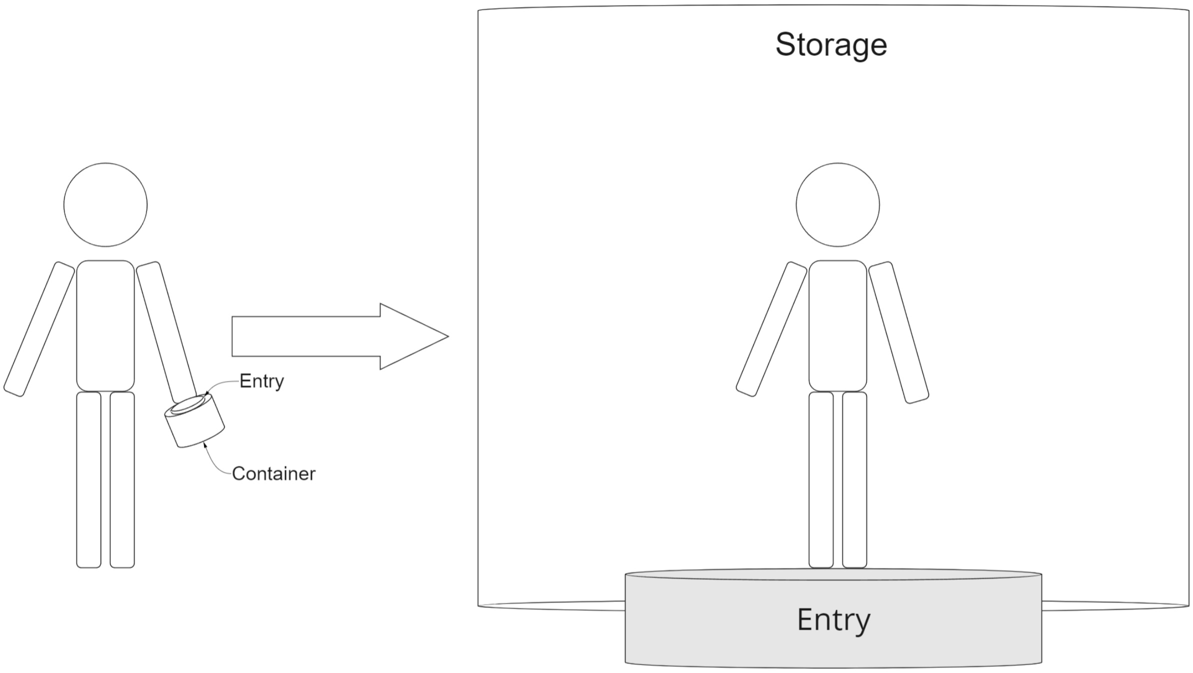


Figure 3e.5: The extradimensional space storage has a container the size of a handbag. Once the user enters the EDS storage, they may find themselves in a space large enough to stand inside (e.g., walk-in closet, garage, or even a whole world) fig:overview

2. Damage
3. Damage Propagation
4. Destruction
5. Weight
6. Motion Gesture
7. Surface Gesture
8. Tap Gesture

Egocentric vs Exocentric: This property determines the relationship of the container to the user as the user moves. It is egocentric if the container is attached to or follows the user when the user moves. It is exocentric if the container does not move when the user moves, such as when it is attached to a location or an object in the world.

Damage: This set of properties determines whether the container can be damaged, how it can be damaged, and the feedback to being damaged (such as visual, sound, haptic, etc).

Damage Propagation: This property determines what happens when the container is hit or otherwise damaged. The stored item could be unaffected, randomly damaged, equally damaged, etc.

Destruction: This property determines what happens to the items stored inside the storage space when the associated container is destroyed. The stored items could either disappear, be dropped on the floor, be ejected in an explosion, and so on.

Weight: This property determines what happens to the weight of the container When users store a new item in the storage system. The weight of the container could increase based on the item's weight, increase based on a fraction of the item's weight added, or remain unchanged.

Motion Gesture: This property determines the effect when the user performs a motion gesture. In other words, When the user grabs the container and moves it around to perform gestures such as shake, jerk, etc. For example, the user may turn the container upside down with the Access Space pointing down, then do a shake gesture to pour the contents of the EDS Storage out.

Surface Gesture: This property determines the effect when the user performs a surface gesture by drawing a gesture on the surface of the container. For example, the user may draw a lock symbol to lock the container and prevent it from being accessed.

Tap Gesture: This property determines the effect when the user performs a tap gesture on the container. For example, the user may tap thrice on the container side to sort the content within.

3e.5.4 Storage

Storage is the component that maintains information about the stored items in the storage. Storage may also be referred to as storage space, storage dimension, or storage world in other literature. In EDS, Storage is the Extradimensional Space that is connected to the user's dimension via the Access Space.

3e.5.4.1 Storage Properties

Storage Properties represent the forces that influence the storage space and, in turn, affect the stored items within. These forces could be based on the real-world laws of physics. Alternatively, they could be based on other hyperphysical laws that will affect how the user will use this storage. For example:

1. Gravity
2. Drag Coefficient
3. Oxygen
4. Temperature
5. Humidity

Gravity, drag coefficient, oxygen, temperature, and humidity are all examples of real-world properties that the storage space could have. These properties can then interact with any physical objects that enter the storage space. In the case of gravity, the storage space could have the standard 1G gravity, zero gravity, or other gravity values. The different gravity values could also affect objects differently so that normal gravity is exerted on the user while zero gravity applies to the stored objects.

In the case of drag coefficients, it would affect whether the object in motion will stay in motion or come to an immediate stop. This is particularly useful in combination with zero gravity so that stored objects would float in the air and stay in the air unless the user explicitly interacts with them.

Oxygen, temperature, and humidity are often important factors in our tabletop role-playing game inspiration of Dungeons and Dragons. Take oxygen, for example. Some Extradimensional Space Storage may contain oxygen, some may not. Oxygen levels may affect flammable items such as matches and torches. It may also affect whether a creature can stay alive inside the storage. The user may enter the extradimensional space to evade enemies but would have limited time before suffocating to death. This has high relevance in VR game scenarios.

Storage Properties also provide opportunities to implement features that a user may find useful in a work environment, allowing EDS Storage to be used beyond VR games. In this case, the storage serves as a tool to modify the properties of the stored items.

1. Sorting
2. Translation
3. Paint Job

For example, a container could function as a sorting machine. Like folders in a PC environment, the user can sort the items inside the storage based on parameters such as name, modified time, or size. Alternatively, the container could function as a translation machine.

Any items that contain text elements, such as documents, would be translated into a different language, depending on the container used.

3e.5.5 Stored Item

An item that is placed in a storage system is considered to be a stored item. Based on the current interaction zone, a stored item often takes on different properties or interactions. The stored item's own properties may also interact with the storage space's property.

1. Response to Oxygen
2. Response Gravity
3. Motion Gesture
4. Surface Gesture

The existence of oxygen in the storage space may affect the stored item, as mentioned in the Storage Properties. Different gestures can be used with stored items. An example of a motion gesture with a stored item would be a shake gesture on an item stored inside the storage space to send it outside. An example of a surface gesture would be to draw an X on an item in storage space to eject it outside.

3e.5.6 Storage Sockets and Storage Queues

Storage Sockets and Storage Queues are provided to aid the user in retrieving objects from outside of the storage space. These Storage Sockets and storage queues are not a core component of the Storage System. Instead, they are a limited and specialized version of the Storage System that is used to make interaction with the full EDS Storage easier.

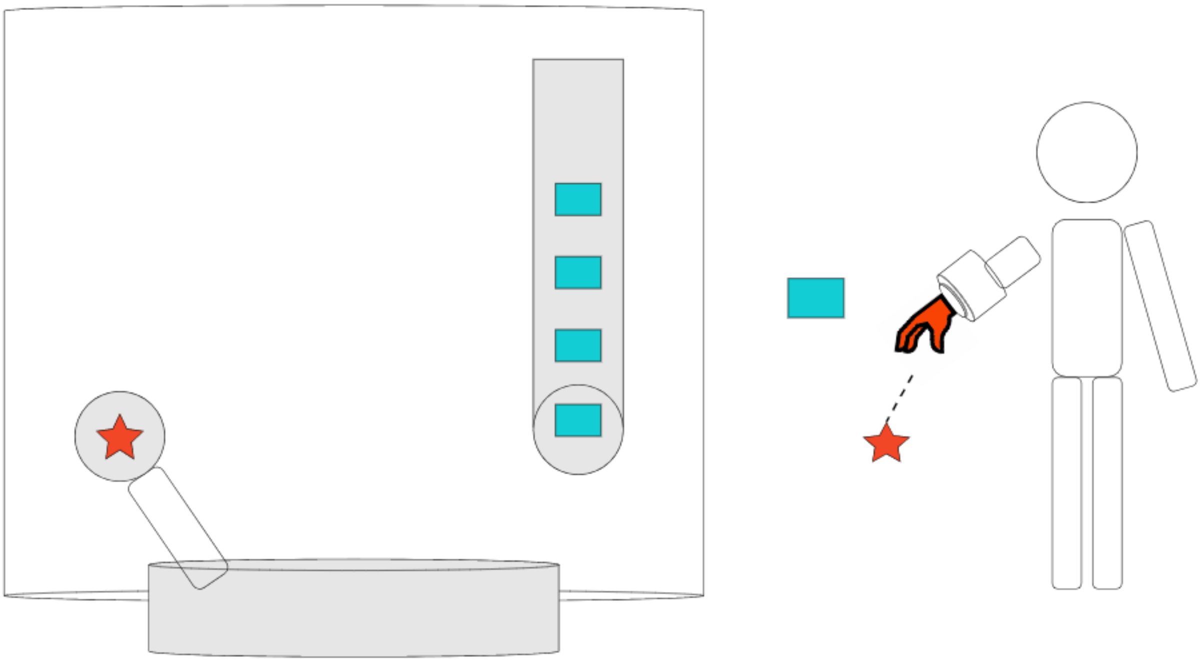


Figure 3e.6: Difference between storage sockets and storage queues fig:socket

Storage Sockets are able to hold a singular item in space. This guarantees an item to be in a specific spatial location for the user to retrieve. The wrist socket popularized by the game Half-Life: Alyx can be viewed as a Storage Socket that is attached to the user's wrist area. Certain storage sockets could also be associated with the user's gesture actions instead of a spatial location.

Storage Queues are an alternate and expanded version of storage sockets. It provides a track for users to place multiple items, with one end of the track functioning as a Storage Socket. When the user interacts with the storage queue from the outside, the user will grab the item at the end of the track. When an item is grabbed, the next item in the queue will fill in for the grabbed item. This allows the user to repeatedly retrieve multiple items from the same location by grabbing or using gestures.

H

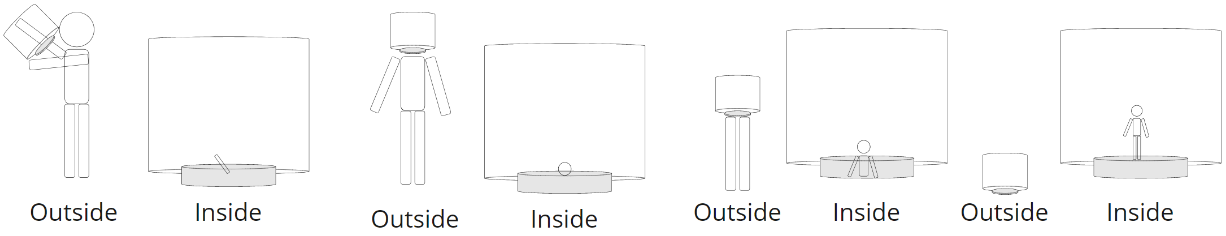


Figure 3e.7: Process when the user enters the storage space.

fig:enter

3e.6 Interaction With Extradimensional Space Storage

In addition to the typical storing and retrieving that a storage system has, EDS Storage has additional storage and retrieving interactions based on whether the user is inside or outside of the storage space. This section describes some examples of EDS Storage interactions.

3e.6.1 Entering and Leaving

The user's hands can enter the Extradimensional Space Storage through the Access Space in order to retrieve items. What sets the EDS Storage technique apart is that the user can enter the Extradimensional Space Storage as if it were a room.

By placing the container with the Access Space side facing the user over the user's head, the user can enter the Extradimensional Space Storage as if entering an attic, a closet, or a storage room.

As the container moves from the top of the user's head down, the user goes through different states of access. The first state is just outside of the head or partially over the head. The user can look into the container to see past the Access Space and then the Storage Space within. The user can get a better view by changing the angle between the head and the container, either by rotating the head or the container. The user can use the other hand to retrieve any item in sight. This allows a quick way to retrieve an item.

The user can then move the container further down to around the shoulder or further down to the waist. The user can leave the container there as if equipping the container on the head as a helmet or an article of a shirt covering the user to the waist. In this state, the user's head has entered the storage space, and rotation will affect what the user sees inside the storage space. With the container equipped, the user can use both hands to interact with the storage space or the outside world, depending on whether the hands are above or below the Access Space. In this state, other users would see the user with their upper body covered by the container.

Finally, the user can drag the container completely down to the feet. The user fully enters the Extradimensional Space Storage with a translucent view of the outside world at their feet. From the perspective of the outside world that the user came from, the only thing remaining is the container on the floor.

To leave the container, the user can grab the edge of the Access Space and pull up. Then the user will return to their previous position in the original world. Alternatively, if the user has access to teleportation interactors, the user can cast the teleportation ray through the Access Space to leave the Storage Space.

3e.6.2 Storing and Retrieving

When storing and retrieving items, the user can either be outside of the Storage Space or inside the Storage Space.

3e.6.2.1 Storing While Outside

The user can place an object at the opening of the container where the Access Space is to store it. Alternatively, the user can move the container to suck in any eligible storeable items. In this scenario, the user cannot decide where any items will be placed and stored. The new items will be stored in a previously defined sequence, such as LIFO, FIFO, or a customized sequence. If the user would like to place the item in a specific location, they can move the stored item into the storage space to a storage socket or any other location.

3e.6.2.2 Retrieving while inside

There are many ways for users to retrieve items from inside. Three ways are discussed here. First, the user can grab an object and bring it along with the user to the outside world. However, this method can only retrieve one item each time. Second, if the user wants to take more than one item, they can grab each object and drop it through the Access Space. These objects will show up in the outside world at the user's feet when the user leaves. Finally, users can perform a double grip action on any number of objects to tag or untag them. Tagged objects will have an outline applied to them. When the user leaves the container, the tagged items will all be retrieved from the EDS Storage.

3e.6.2.3 Retrieving while outside

subsubsec:ret_outside

For retrieving items from outside, the most basic way is to move a hand inside the Access Space and grab. At this moment, the user's hand will disappear in its original dimension. Also, only a user who owns the Storage Space is able to view and retrieve the items inside. If the Storage Space is very large, a non-linear mapping hand movement can be applied to the user's hand to make it easy to reach stored items. While the user's hand is in the Access Space, a floating minimized 3D model may appear in front of the user's head to help users retrieve items precisely. Similar to the world in miniature technique, the 3D miniature world helps players view the storage space more clearly [133]. Instead of a representation of the space that the user is in, however, it represents the world within the container. Players' interactors will be resized appropriately, and the interactors can move freely in the miniature world. The position of this miniature world can be adjusted as needed. Nevertheless, if the user does not select anything while putting their controllers into the access space, the Quick Access functions will be activated.

With Quick Access, the user can either grab the next item placed in the queue, take the first item in a FIFO or LIFO order, or say the object's name. Furthermore, the user can reach a hand inside and point in a specific direction to retrieve a pre-selected item, which is a form of gesture-based access. These methods are designed the maximize the efficiency of retrieval.

Finally, the user can also turn the container upside down and shake. As the user continues to shake, stored items fall out one by one in a predetermined sequence until the Storage Space is empty.

Figure 3e.8: Rendering Image of Extradimensional Space Storage fig:rendering

3e.6.3 Object Organization

Depending on the Storage Properties and how the user has arranged the Storage, objects inside the EDS Storage may be floating in the air, on the floor, or held in place by additional storage systems such as shelves or other constructs. The aforementioned constructs, such as Storage Sockets and Storage Queues, can further aid the user in organization in addition to storing and retrieving items. By allowing the user to place objects in 3D space, the user can rely on muscle memory or spatial memory to retrieve items.

Additionally, many sorting strategies can be applied if the user chooses to arrange their items automatically. For instance, items can be arranged by their categories, such as tools, weapons, etc.

3e.7 Conclusion

In this section, we discuss our proposed storage framework to expand the capability of existing VR storage solutions based on our design lenses of Extradimensional Space Storage. We designed and built a storage framework with five core components to create an Extradimensional Space Storage. We interpreted and matched the five core components with Cmentowski et al.'s Ground Theory analysis work and made some additions of our own. The five core components model provides us with an alternative perspective to think about and design a storage system. Each storage system component can have unique properties that can dramatically change how the user utilizes the storage. The properties also have the potential to transform these Extradimensional Space Storage that provides convenient access to your inventory to productivity tools. Our work results in a highly generalizable storage framework as well as its specialized implementation in the form of Extradimensional Space Storage. With

EDS Storage, users have the ability to retrieve and store items quickly and efficiently as with a normal storage container, but can also enter the container as a storage room to make use of different interaction techniques. As the concept of a 'Metaverse' is frequently mentioned and implemented, our approach has applications not only for games, but also for work and XR interactions.

Chapter 3f

Other Work

ch3:Other

3f.1 Other Related Work

I have collaborated with many master's students to explore the design lens presented. I'd like to briefly mention our collaborative work and how they contributed to this thesis.

3f.1.1 Machine Learning Explorations

With Aditiyan Jothi, Sloan Swieso, Mark Miller, and Andrew Zhao, we explored different ways machine learning could be used in virtual reality. Each was responsible for exploring a different area, resulting in three published works.

Mark was responsible for exploring whether Peripersonal Equipment Slots can be based on where users think they are rather than having a fixed spatial relationship with respect to the user's torso. This resulted in Virtual Equipment System: Toward Peripersonal Equipment Slots with Machine Learning [93].

Sloan explored whether motion gestures combined with machine learning can be used for text entry. This resulted in Toward Using Machine Learning-Based Motion Gesture for 3D Text Input[135].

Aditiyan explored whether it's possible for a machine learning model to predict the user's waist location with only the user's head and hand locations. The machine learning model is trained on data incorporating head, hands, and waist locations. This resulted in Toward Predicting User Waist Location From VR Headset and Controllers Through Machine Learning[69].

Andrew explored using machine learning for more complex motion gestures to be used for Equipment Gestures.

3f.1.2 Text Entry in Virtual Reality

A series of works is done on text entry in Virtual Reality based on the design lens of hyperphysicality and whole-body interaction.

The first of which is Punch Typing[161] with Ruoxi Jia. In Punch Typing, we explore the possibility of utilizing and adapting the common QWERTY keyboard layout to a 3D hemispherical layout. Then we explored using machine learning in combination with motion gesture for 3D Text Input as mentioned above[135]. Finally, we continued the exploration of Text Entry methods to create Flick Typing with Tian Yang [156, 155] to explore whether we can make use of 3D space with minimal positional change.

3f.1.3 Smart Home Control with Natural Language Processing

With Yu Hou, Yuan He, Da Cheng, and Huanpu Hu, we explored using spatial information from VR headsets and motion controllers in combination with natural language processing to control a Smart Home simulated with Virtual Reality as shown in Figure 3f.1. This has resulted in two works "Toward Using Multi-Modal Machine Learning for User Behavior Prediction in

Figure 3f.1: This figure depicts a Smarthome simulated in Virtual Reality. fig:smarthome

Simulated Smart Home for Extended Reality"[158] and "Using Multi-modal Machine Learning for User Behavior Prediction in Simulated Smart Home for Extended Reality"[159].

This work and its simple subject-verb-object model became the basis for the Spatial Interaction Model in Section 3a.1. The need to disambiguate between Egocentric Equipment and Exocentric Equipment led to the emphasis on additional contexts in the Gesture Taxonomy in Chapter 3b. Finally, the codebase became the foundation of the School of Spatial Sorcery in Chapter 3c.

3f.1.4 Computer Vision

With Pranavi Jalapati and Satya Naraparaju, we explored the possibility of combining spatial information from VR headsets and motion controllers with computer vision in the paper "Integrating Sensor Fusion with Pose Estimation for Simulating Human Interactions in Virtual

Reality"[67]. The intent is to use this technique to better track different body parts for evaluating different interaction techniques. However, for it to provide enough accuracy to be useful would require much more work and thus, it was not further pursued.

Chapter 4

Methods

ch4:Methods

Chapter 4a

Methodology for Gesture Taxonomy and Inventory Taxonomy

For the two taxonomies, I collaborated with master's students working with me at the USC GamePipe Labs. We utilized Grounded Theory Analysis[57]. In both cases, we used the Negotiated Agreement method, where we discussed the taxonomy dimensions in detail until we reached an agreement.

For the Gesture Taxonomy[165], I worked with Tian Yang. We started with open-coding on previous taxonomies dating from early linguistic work to modern HCI taxonomies. The detailed methodology can be found in Section 3b.3.

For the Inventory Taxonomy[158], I worked with Zhankai (Jackie) Ye. Previously, we have been independently building a more complex extradimensional space storage prototype that evolved from the simple extradimensional space storage from Chapter 3c and developing a storage system architecture. For the inventory taxonomy, we used hybrid-coding. We started with the only existing inventory taxonomy in VR[29]. Then, we incorporated elements from our storage system architecture and the extradimensional space storage prototype to provide new perspectives on the inventory taxonomy by Cmentowski et al.

Chapter 4b

Virtual Equipment System

ch4:VES

4b.1 VES Study Introduction

The Virtual Equipment System is a Hyperphysical User Interface system in Extended Reality as detailed in Chapter 3d. A Hyperphysical User Interface follows laws of physics that are different from our physical reality, such as having equipment in peripersonal space. While hyperphysicality is used as a lens of design, its impact on the resulting design is under-explored. In this section, we present the results from our evaluation of the Virtual Equipment System using a user experiment study and a modified version of the NASA-TLX questionnaire[63]. We compare different interaction techniques available within the Virtual Equipment System for adjusting audio volume. We also examine the effect of having the Virtual Equipment located in personal and peripersonal spaces. We perform statistical analyses on the collected data, such as the Kruskal-Wallis H Test. The results suggest that there are definitive differences among the different classes of techniques. It also shows that some classes of techniques are better in one aspect, such as duration required, while worse in another aspect, such as motion required. Due to the risk of COVID-19, user experiments were conducted with participants' own devices and at their respective homes. These limitations reduce the generalizability of the study. Even with the unorthodox data collection, the study has shown virtual equipment does not need to be restricted to personal space, demonstrated that different classes of techniques are useful for different contexts, and provided insights on how future user studies should be conducted.

4b.2 VES Study Evaluation

To evaluate VES, we primarily focused on the design lenses of Hyperphysical User Interface and Whole-body Interaction. For quantitative data, we track the movement of the user's headset and two controllers as substitutes for the user's head and hands in addition to task completion time. Given the many potential contexts of XR, a faster technique is not always the most desirable. For example, a user may not want to perform large sweeping gestures on a crowded train or speak aloud in public. For qualitative data, we use a modified NASA-TLX[63] Questionnaire to collect information about physical demand, mental demand, and preference.

Our user study compares the ability to adjust audio settings using the traditional 2D menu against experimental interaction techniques available in VES. The VES interaction techniques mentioned in Section 3d.4 are applied to different egocentric equipment slots located in personal space and peripersonal space.

4b.2.1 Research Questions

We seek to assess the different VES techniques with the following research questions (RQ):

- RQ1: How do the techniques compare in terms of efficiency?
- RQ2: How do the techniques compare in the effort required for each interaction technique?

- RQ3: What makes users prefer one technique over another?

4b.3 User Study

We performed a within-subjects user study followed by a questionnaire based on modified NASA-TLX. We evaluate the user on adjusting the audio settings with an equipment motion gesture, equipment surface gesture, alt node interaction, and menu button interaction. We also compare whether the user prefers having the audio adjustment equipment next to the ear in personal space, in other locations in the personal space, or in peripersonal space. The user is evaluated on motion gestures as well as surface gestures for Virtual Equipment stored in each space. In total, there are eight different types of tasks as shown in Table 4b.1.

Task Type	Shorthand	Description
1	Headphones + Motion	Personal Equipment (Ear) Motion Gesture
2	Headphones + Surface	Personal Equipment (Ear) Surface Gesture
3	Waist + Motion	Personal Equipment (Waist) Motion Gesture
4	Waist + Surface	Personal Equipment (Waist) Surface Gesture
5	Peri + Motion	Peripersonal Equipment Motion Gesture
6	Peri + Surface	Peripersonal Equipment Surface Gesture
7	Alt Node + Slider	Moving a slider on a 2D Menu for Audio, through the use of Alt Node
8	Menu Button + Sliders	Moving a slider on a 2D Menu for Audio, through a menu button on the VR Controller

Table 4b.1: Eight different types of tasks the user has to perform. The shorthand is shown to the user during the user study.

tab:8taskTypes

4b.4 Interaction Techniques

The Virtual Equipment System supports many different classes of gestures and interaction as discussed in Section 3d.4. We evaluate the effectiveness of some of the relevant interaction techniques as compared to traditional 2D menu interaction as well as against each other.

In this section, we discuss the actual implementation in the user study and the resulting limitations.

4b.4.1 Equipment Surface Gesture

The user can move the controller near Virtual Headphones stored by the user's ear, the user's waist, or in the peripersonal space. When the user pushes the trigger button and moves the controller, this draws a white line where the motion controller's tip is. As the user lets go, the white line is interpreted to determine which Equipment Surface Gesture it was. A line drawn from the bottom up is an up gesture, whereas a line drawn from the top down is a down gesture. The up and down gestures correspond to the increase or decrease of the system's audio volume.

4b.4.2 Equipment Motion Gesture

The user can grab Virtual Headphones stored by the user's ear, the user's waist, or in the peripersonal space. When the user grabs any of the Virtual Headphones and moves it up and down, this would trigger its Equipment Motion Gesture. The up and down gestures correspond to the increase or decrease of the system's audio volume, respectively.

4b.4.3 Alt Node

With Virtual Headphones grabbed, the user can move and drop the Virtual Headphones in the Alt Node located on the top of the left controller. The goal of the Alt Node is to perform the function of the Alt Key on a PC keyboard. When the Virtual Headphones are dropped into the Alt Node, the system will bring up the alternate function of accessing the associated menu.

In the case of Virtual Headphones, its associated setting menu is the audio setting menu. For other Virtual Equipment, the Alt Node will open different menus, such as the visual settings menu for Virtual Goggles and the audio recording settings menu for Virtual Microphone.

4b.4.4 Traditional Menu Button

In modern VR motion controllers, there are typically two types of menu buttons. One is reserved for accessing the Operating System and its menu, typically located on the right-hand controller. The other is the in-game or in-experience menu button, typically located on the left-hand controller.

In most VR experiences, the user would push the in-experience button to bring up a general menu. Using a ray emitted from the motion controller, the user can then navigate to the settings menu from the general menu by selecting the appropriate button. This is similar to using a mouse to click on a button. Then, from the settings menu, the user can select more specific settings menus such as audio, visual, and other categories.

In the study, the user has access to a full setting menu. To modify the master audio volume, the user would start by pushing the in-experience menu button to bring out the general menu. Then user the ray from the motion controller to select the setting menu button, then select the audio setting menu button, and then select the master audio volume slider.

Location	x	y	z
Personal Equipment (Ear)	0.15	0	-0.07
Personal Equipment (Waist)	0	-0.6	0
Peripersonal Equipment	0.2	0.2	0.6

Table 4b.2: Different Virtual Equipment's X, Y, Z p tab:DistancesBetweenEquipmentAndHeadset

4b.5 Egocentric Equipment in Different Space

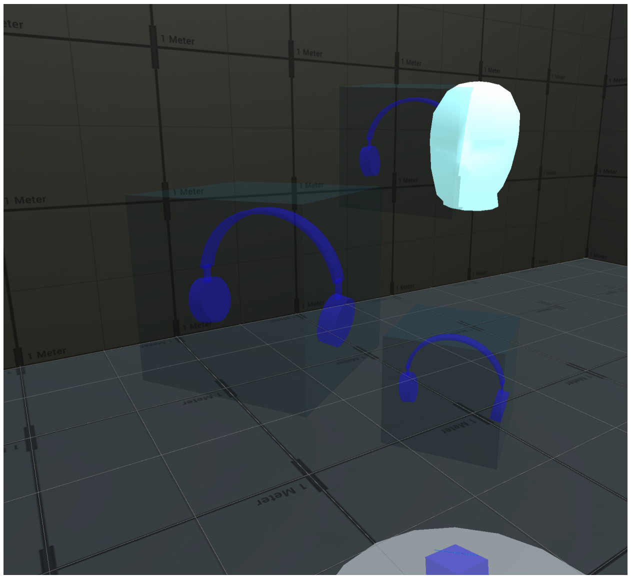


Figure 4b.1: Next to the cyan head is the Personal Equipment stored by the ear. To the very left is the Peripersonal Equipment stored at 1 o'clock with respect to the user's head. At the bottom is the Personal Equipment stored by the waist. fig:EquipmentInDifferentSlots

We placed the same Egocentric Equipment in three different locations. The first two are within Personal Space, and the third is within Peripersonal Space. This is shown in Figure 4b.1. The exact spatial relationship with respect to the headset is shown in Table 4b.2

The position offset values were pre-determined to roughly represent equipment in the respective areas. The only customization was how far away from the headset the Peripersonal Equipment was stored. The user had to calibrate their arm's reach by moving their arm to the front of the user, parallel to the floor, then select the "Set Reach" button with the ray interactor.

4b.6 Study Flow

Our VR experience includes simple text-based guidance so the users can go through the experiment themselves. This was made necessary due to the pandemic and remote testing.

4b.6.1 Tutorial Section

When the VR experience starts, a virtual panel on the users' right-hand side will guide them through the tutorial section. The guidance encourages the user to familiarize themselves with each interaction technique before moving on. The tutorial section can be moved through at the user's own pace, which is typically around ten minutes.

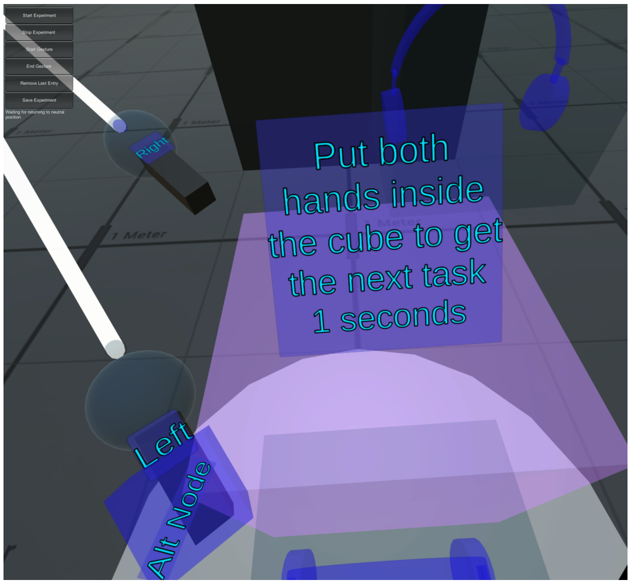
After the user finishes the tutorial section, they can move on to the experiment section.

4b.6.2 Pink Cube

In the experiment section, the guidance will show prompts for the user to perform a specific action when the user is ready. The user indicates readiness by putting both hands in a translucent pink cube as shown in Figure 4b.2.

The pink cube represents a neutral starting position and a way for the user to indicate that they are ready for the next task. To avoid giving any equipment too much advantage, it is placed 40 cm below and 20 cm in front of the user's headset. This roughly places the pink cube 49.5 cm away from the ear, 28 cm away from the waist, and 48.9 cm away from the peripersonal location.

Once both controllers are inside the pink cube, a timer will start counting down from 1 second to 0 seconds. If the user moves the controller away before it hits 0, the timer will revert to 1 second and the task will not start.



Waiting for returning to neutral position

Put both hands inside the cube to get the next task 1 seconds

Figure 4b.2: Translucent pink cube is placed in front of the user near the chest area. It is used to confirm the user's readiness.

fig:pinkcube

4b.6.3 Tasks

Once the timer finishes counting down to zero, a random task will show up in front of the user from the list of eight tasks. The user needs to adjust the audio volume using the interaction technique and equipment specified.

The eight tasks are described in short-hands to the user as shown earlier in Table 4b.1

After the user has performed each of the eight tasks once, the user will be asked to do a ninth task with any method they like.

Altogether, the 9 tasks are considered as a block. The process repeats until the user has completed 10 blocks. In total, the user would complete 90 tasks in the 10 blocks.

4b.6.4 Software & Hardware

We used the Unity3D game engine (version 2019.4.29f1), Unity's older Built-In XR system, and Unity's XR Interaction Toolkit to support VR devices from different vendors.

Our development device and targeted device is the HTC Vive. However, due to the COVID-19 pandemic, we asked volunteers to participate in the experiment remotely. As such, the experiment was conducted with a wide variety of different VR devices. The VR devices include HTC Vive, HTC Vive Cosmos, Valve Index, Oculus Rift S, and Oculus (now Meta) Quest 1 and 2.

The different devices and our implementation of the VR experience created minor discrepancies that may have affected the experiment results.

First, HTC Vive users only experience tactile feedback (vibration) in the form of small blips. Next, the HTC Vive's menu button corresponds to the Oculus family device's face button and not the menu button that Oculus users are familiar with. In addition, the different controllers have slightly different setups that resulted in the ray used to select the menu user interface having a slightly different orientation that we did not account for.

4b.7 Quantitative Data

For each task, we measure and tally up the time taken, cumulative positional movements for the head, cumulative positional movements for the VR controllers, cumulative rotational movements for the head, and cumulative rotational movement for the VR controllers.

Vive has a sub-millimeter precision in a static configuration[15]. However, we observed that the positional numbers only remained fixed before the third decimal place for a meter. To filter out the noise, we discard any detected changes below a millimeter.

We performed the same tests for rotational numbers and decided to discard any changes below 0.1f. After the experiment, all of the recorded data was saved to a comma-separated value (CSV) file.

4b.8 Qualitative Data

After the experiment, volunteers had to complete a post-experiment questionnaire. The goal was to get more insights into the Virtual Equipment System and help assess what areas can be further improved. They were first asked to evaluate Motion Gesture, Surface Gesture, and Alt Node Interaction in the following three areas: how mentally demanding the task was, how physically demanding the task was, and how much they liked completing the tasks using those specific interaction techniques. They are asked to choose a number between one (very low) and five (very high).

Then, they are asked to assess Personal Equipment by the ear, Personal Equipment by the waist, and Personal Equipment in the Peripersonal Space using the same three evaluation categories and rating system above. Finally, they are encouraged to share any thoughts specific to each interaction technique and the overall experiment in written form.

Chapter 5

Results

ch5:Results

5.1 Virtual Equipment System Results

ch5:VES

5.2 VES Results

Eleven participants (N=11, mean age: 27.1, SD: 4.43, male/female 9/2) volunteered for the experiment.

Due to the COVID-19 pandemic, the experiment instructions were sent to the participants. It was conducted remotely at their homes with the participant's own VR equipment.

The participants were emailed instructions to conduct the experiment. This includes a pre-experiment questionnaire, a post-experiment questionnaire, and a copy of the experiment build made using the Unity Game Engine.

5.2.1 Pre-Experiment Questionnaire Results

The participants self-reported having high familiarity with VR (mean of 4.09, lowest 3) on a scale of 1-5 between not very familiar to very familiar. This is expected as the volunteers have their own VR equipment. They also self-reported commonly using VR equipment (mean of 3.36, lowest 2) on a scale of 1-5 between never to very often.

5.2.2 User Study

For the eight interaction techniques, we performed statistical analysis and compared the different techniques within each criterion to find the best-performing method, respectively.

We removed any obvious outliers, such as when a task takes over a minute instead of a few seconds. Later interviews revealed that this is due to unexpected interruptions, such as family members needing their attention or when they revisited the tutorial instructions.

For statistical analysis, we check to see if our data follows a normal distribution by plotting Normal Quantile-Quantile Plots and using the Shapiro-Wilk Test. We found that our data did not, and thus, we used the Kruskal-Wallis test to compare the different techniques. We found that we had a p-value of 0.1481.

However, if we remove the first block of the data (out of the 10 blocks), the p-value changes to 0.0001244. This suggests that 1) despite the tutorials, participants were still learning in the first block, and 2) there is a significant difference between these techniques once we account for the learning that occurred in the first block. This can be seen in Figure 5.1.

For the following data, the data collection starts one second after the user has placed both motion controllers within the pink cube when the task is shown to the user. The data collection ends as soon as the user modifies the audio volume based on the technique specified in the task.

5.2.2.1 Duration

Duration is the time it takes to complete a task. This is shown in Figure 5.1 and Table 5.1. Personal Equipment (Ear) Surface Gesture has the lowest average value of 2.61 seconds, closely followed by Peripersonal Equipment Motion Gesture with an average of 2.96 seconds and other motion gestures, then by the rest of the surface gestures. The interaction technique that has the longest duration is the Controller Menu Button, averaging around 5.39 seconds. The Alt Node Button interaction barely beats the menu button with an average duration of 5.25 seconds.

This is in line with our expectation that having direct access to modifying volume through gestures would be much faster than going through a series of menus.

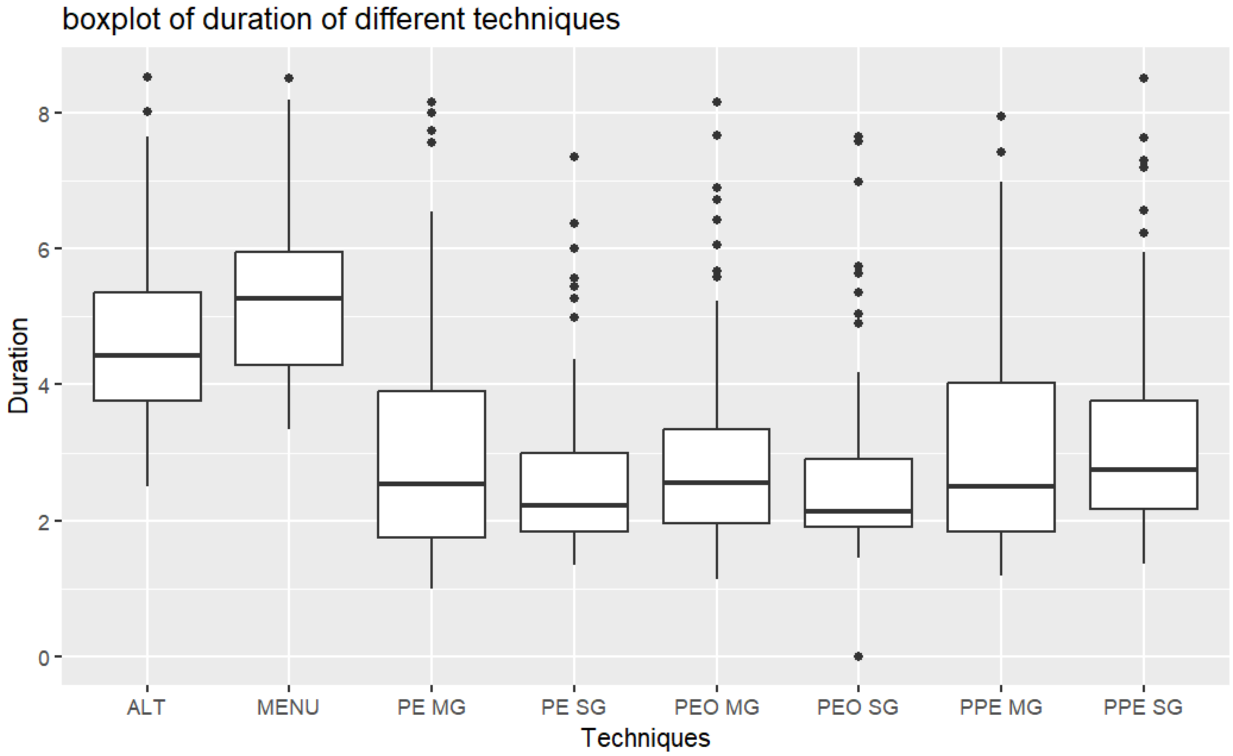


Figure 5.1: Box plot of duration to complete a task for each interaction techniqu fig:duration

Source	**Type**	**Duration (in seconds)**	**stdev**
Personal Equipment (Ear)	Surface	2.61	1.17
Peripersonal Equipment	Motion	2.96	1.55
Personal Equipment (Ear)	Motion	3.03	2.09
Personal Equipment (Waist)	Motion	3.10	1.78
Peripersonal Equipment	Surface	3.32	1.94
Personal Equipment (Waist)	Surface	3.37	2.42
Alt Node	Slider	5.25	2.25
Menu Button	Slider	5.39	1.34

Table 5.1: Task completion time for different interaction techniques in the Virtual Equipment System User Study

tab:duration

5.2.2.2 Accumulated Head Movement and Rotation

The head movement records how much the head has changed its position over the course of each task. We calculate the head position difference at each frame, adding up to the total distance traveled by the head. For the head rotation, we similarly calculate the head rotation difference at each frame, adding up to the total rotation angle by the head at the end of a task. This is shown in Figure 5.2, Figure 5.3, and Table 5.2

In Figure 5.2 and Figure 5.3, we note that despite the Personal Equipment at the waist being accessible without looking at it, participants ended up looking at it frequently. The techniques involving Personal Equipment at the Waist had among the most head movement and head rotation.

We also observed that both Personal Equipment (Ear) gestures involved the least head movement and among the lowest for head rotation. This makes sense as no amount of head movement will allow the user to see it, given that it is attached to the user's head.

Alt Node also has one of the highest head movements and head rotation, even though it's possible for the user to interact with the Alt Node by moving both hands into their field of vision or without looking at it.

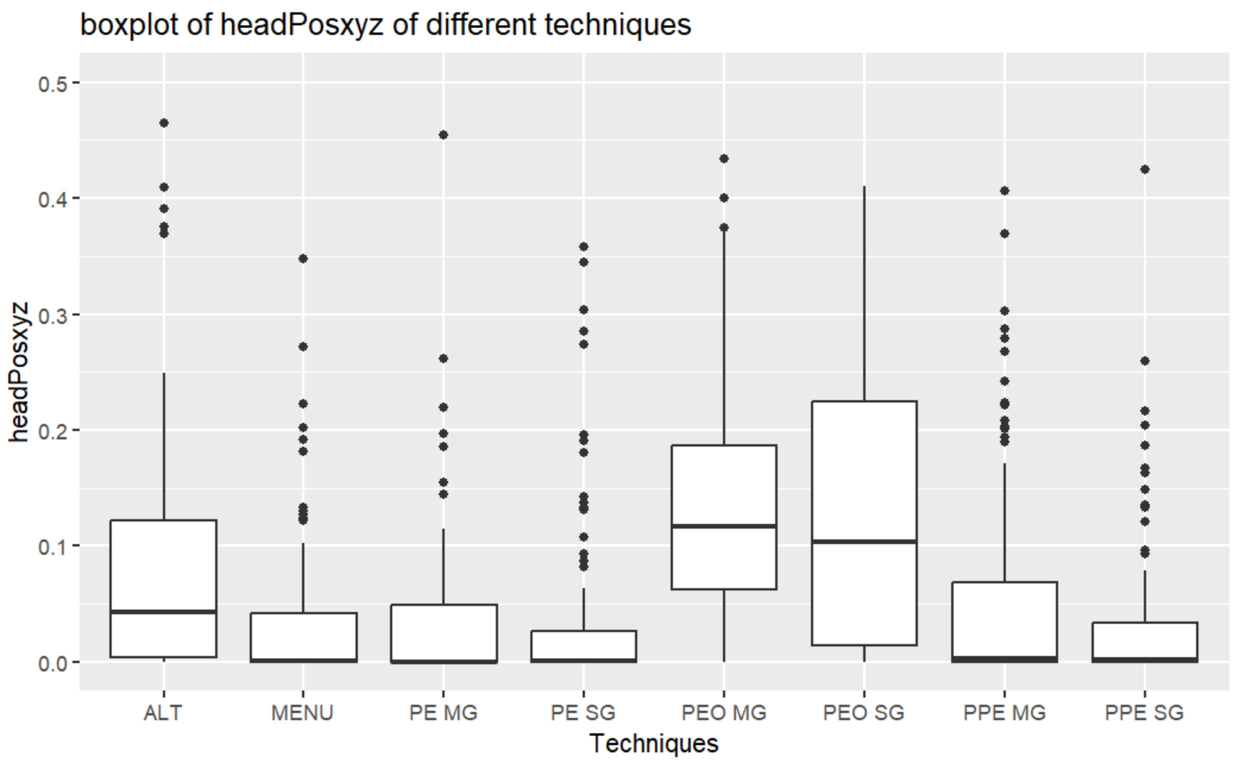


Figure 5.2: Box plot of head's accumulated XYZ position to complete a task for each inte fig:headposition

5.2.2.3 Accumulated Right-Hand Movement and Rotation

Looking at the accumulated position in terms of XYZ for the right hand in Figure 5.4, we see that the 2D menu technique requires the least amount of movement while the other techniques require much more. Despite the VE at the waist being closest to the hand starting location, the surface gesture and motion gesture (PEO SG & PEO MG) did not result in less movement when compared to VE in other locations.

5.2.2.4 Accumulated Left Hand Movement and Rotation

On the other hand, the left hand exhibited little difference in the accumulated position change in Figure 5.5. This is expected as the techniques have been implemented with the right hand

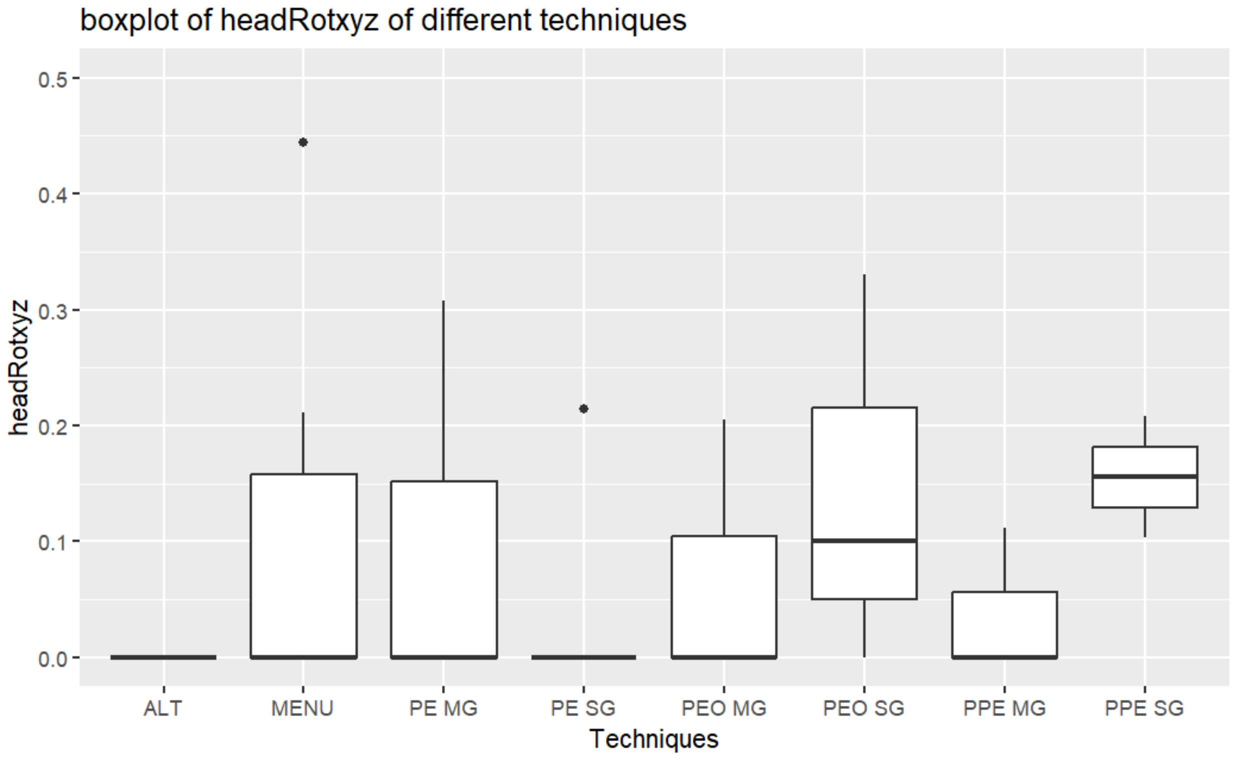


Figure 5.3: Box plot of head's accumulated XYZ rotation to complete a task for each inte fig:headrotation

in mind. The Alt Node is on the left hand, the personal equipment is by the right ear, and the peripersonal equipment is at the 1 o'clock position, which is also near the right hand.

5.2.3 Post-Experiment Questionnaire

We collected data through a post-experiment questionnaire using a modified NASA-TLX Questionnaire to understand how the users felt when using the different techniques.

We asked the participants to evaluate the four different interaction techniques (surface gesture, motion gesture, Alt Node + Slider, and Menu Button + Slider). We asked the participants to evaluate Virtual Equipment in the three different spaces (Ear, Waist, Peripersonal) for interaction. For each evaluation, we asked about physical demand, mental demand, and preferences, where the participants rate each on a 1-5 point scale from very low to very high.

Source	**Type**	**Accumulated Head Movement (in meters)**	**stdev**	**Accumulated Head Rotation (in degrees)**	**stdev**
Personal Equipment (Ear)	Surface	0.03	0.06	36.60	44.04
Personal Equipment (Ear)	Motion	0.03	0.06	40.85	53.18
Menu Button	Slider	0.04	0.07	37.78	44.78
Peripersonal Equipment	Surface	0.05	0.14	53.82	68.39
Peripersonal Equipment	Motion	0.07	0.12	62.78	70.19
Alt Node	Slider	0.08	0.11	85.69	54.66
Personal Equipment (Waist)	Surface	0.16	0.19	85.46	81.91
Personal Equipment (Waist)	Motion	0.16	0.18	90.88	75.60

Table 5.2: Head Movement and Rotation Values for one task using different interaction techniques in the Virtual Equipment System User Study tab:HeadMovementAndRotation

5.2.3.1 Interaction Techniques

The result of the participants' evaluations of the four different interaction techniques is shown in Table 5.3.

Motion Gesture is the least mentally demanding (1.55), whereas Surface gesture is the least physically demanding task (1.55). Motion Gesture is the most preferred interaction method (3.82). Surface Gesture is the second favorite interaction method (3.55), closely following the first place with a difference of only 0.27. Alt Node + Slider is the most mentally (2.27) and physically (2.45) demanding method. Interestingly, it is still slightly more preferred than the Menu Button + Slider, the least favorable out of all four.

An interesting result here is that despite being the technique that requires the least movement, the menu technique is not rated as the least mentally demanding. Motion Gesture is the most highly rated, perhaps due to the more developed feedback features with Mirrored Equipment.

5.2.3.2 Equipment Location

The results of the user's evaluation of different equipment locations are shown in Table 5.4.

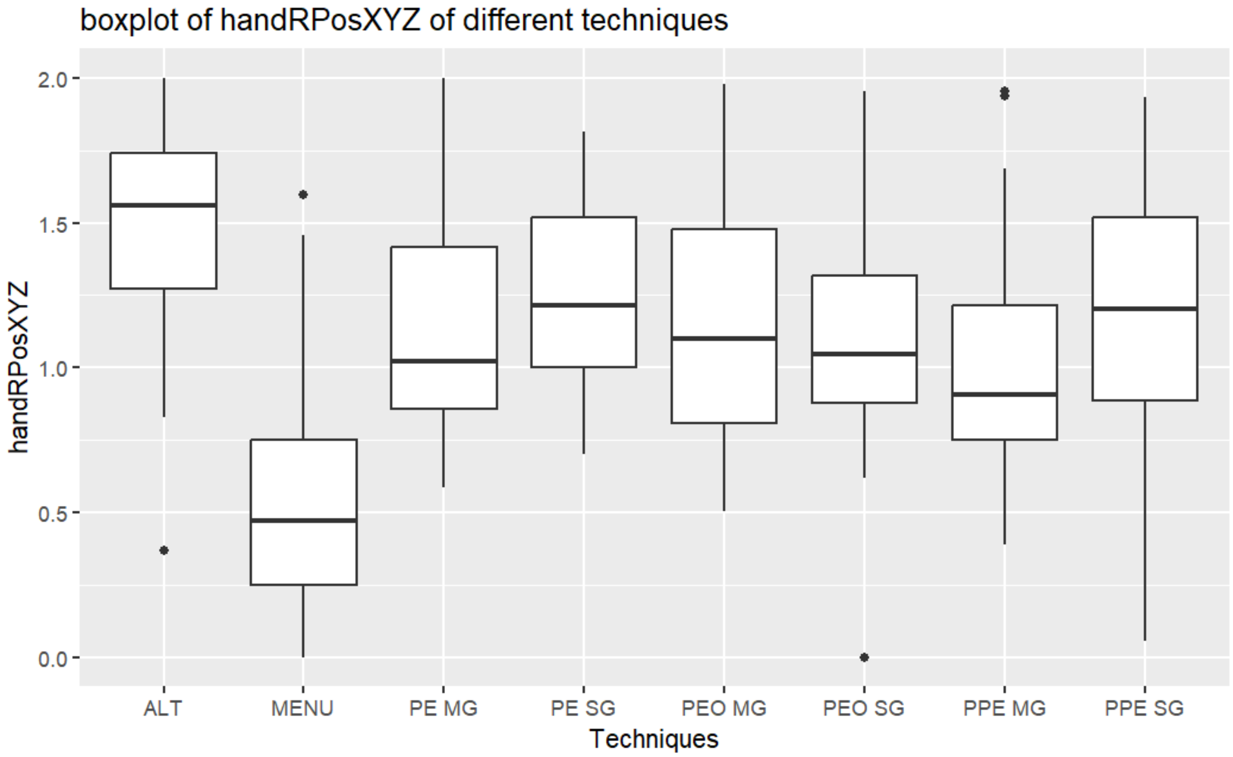


Figure 5.4: Box plot of right hand's accumulated XYZ position to complete a task for each interaction technique

fig:handrposition

Technique	Mental	Physical	Preference
Motion Gesture	1.55	1.82	3.82
Surface Gesture	1.73	1.55	3.55
Alt Node + Slider	2.27	2.45	2.45
Menu Button + Slider	2.09	2.00	2.27

Table 5.3: Mental demand, physical demand, and preferences for different interaction techniques in VES user study

table:techniques

Of the three locations, we can see that the waist position is the most demanding position both mentally (2.36) and physically (2.27). It is also the least preferred method. Of the other two, Virtual Equipment at the peripersonal location is the least mentally (1.55) and physically (1.82) demanding location. While Virtual Equipment at the ear location is tied in terms of how physically demanding it is (1.82), it is slightly more mentally demanding (1.82). However, of the three, the most preferred location is not the location with the least mental and physical demand. The ear location is more preferred (3.91) than the peripersonal location(3.55).

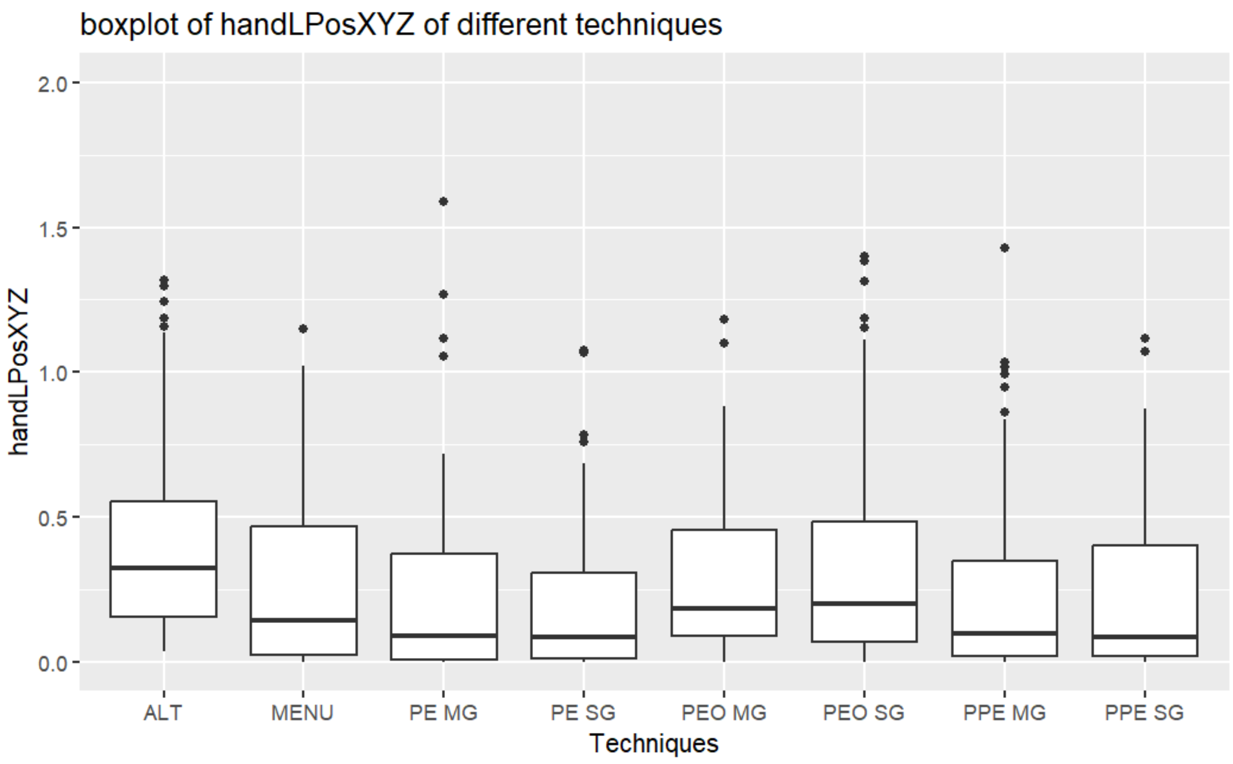


Figure 5.5: Box plot of left hand's accumulated XYZ position to complete a task for each interaction technique

fig:handlposition

Despite the VE at the waist being closest to the starting location, interacting with it resulted in the worst score in all three categories. While the ear location is rated to be more mentally demanding, it is preferred by the participants. This aligns with what we expect based on the Body Mnemonics [10, 9]. Users would prefer VE to be stored in locations where they can make mental associations. The preferred location for VE to adjust audio is next to the ear.

Location	**Mental**	**Physical**	**Preference**
Ear	1.82	1.82	3.91
Waist	2.36	2.27	2.45
Peripersonal	1.55	1.82	3.55

Table 5.4: Mental demand, physical demand, and preferences for Virtual Equipment stored in different locations

table:locations

5.3 Discussion & Future Work

Comparing the results to our original research questions, we can see those techniques that are the most efficient (RQ1) are not those that require the least effort (RQ2). Interestingly, the techniques requiring the least movement are not perceived as physically demanding.

We also see that surface gestures and motion gestures are not significantly different in terms of duration, position, or rotation change. This makes sense as the two classes of gestures are doing more or less the same motion, just with different prerequisites and buttons.

This experiment was designed to measure speed more than precision. The task is considered completed as soon as the user changes the volume in any direction. This was done to eliminate any difference that can result from the implementation of a given technique. For example, the motion gesture could also be implemented so that the gesture is determined based on the difference between the position at which the equipment is picked up instead of the position where the equipment is stored. The equipment could also be used as a slider handle so that the user can set the volume from min to max instead of making increments to the current volume. Potential future work comparing the precision of different techniques can provide new insights.

Precision aside, we would also like to examine variations of a given technique. In comparing the Alt Node and Menu Button, the Alt Node did not seem particularly advantageous. With the current implementation of the technique and of the experiment, users can easily navigate to the audio menu as the settings menu is the first item on the general menu and the audio menu is the first item on the settings menu. Without the need to read and navigate the 2D menu, many of the advantages of the Alt Node technique are lost. Despite this, some participants still recognize the potential for an Alt Node and overall prefer the Alt Node in spite of its higher mental and physical demand. Future work comparing VE for different sensory settings may shed new light on how mnemonic and association between the VE and functionality affect performance. For example, future work could investigate having the user interact with Virtual Goggles for visual settings, Virtual Headphones for audio settings, Virtual Microphone for audio recording settings, and so on.

The feedback we provide to the user also appears to be lacking from the questionnaire answers we have gotten. In particular, while we have visual feedback for the Motion Gesture that resides in the space by the user's ear and by the waist, we did not implement the same mirrored equipment feedback for the surface gesture. We also noticed that the user has much more trouble initiating a surface gesture than a motion gesture. Additional feedback, especially regarding spatial cues, could alleviate some of the difficulties users face.

We also noted that the user spends much time looking around when interacting with the Virtual Equipment at the waist or with the Alt Node when they should have been able to perform these actions without looking. One of the main advantages of our implementation of peripersonal equipment cited by participants was that it's in front of the user and thus fully visible. Future work can compare peripersonal equipment in different locations as well as VE with different visibility settings to compare VE in different spaces when users cannot see them directly.

Lastly, we did not ask for the user's dominant hand in the demographic questionnaire and did not design the experiment to support either hand. This can be easily addressed using hyperphysicality and would be something we support in future studies.

5.4 Limitations

In addition to some of the challenges mentioned in the discussion section, the biggest limitations come from the threats to internal validity. Due to COVID-19, the study was not conducted at a lab and participants were composed of volunteers who already have VR equipment at home, which raises concerns regarding subject selection bias. Conclusion Validity was affected due to having participants self-administer the user study with the guide of the program, impacting the reliability of treatment implementation. We also had a small sample, which suggests low statistical power.

5.5 Conclusion

Using the design lenses of Hyperphysical User Interface and Whole-body Interaction, we proposed and built a Virtual Equipment System for Extended Reality. Hyperphysicality provided us with new ways to categorize and think about Virtual Equipment in terms of different qualities, such as egocentric or exocentric Equipment, and locations in personal, peripersonal, and extrapersonal space. We then utilize Whole-body Interaction as a lens to aid us in evaluating user interaction with the Virtual Equipment System (e.g., tracking the head motion data). In this evaluation of the Virtual Equipment System, we have shown that there are indeed differences between the different classes of techniques and that different classes of techniques can be useful for different contexts. All of our proposed techniques are preferred over the traditional techniques even when they are not the techniques that involve the least physical movement. We have also gained valuable insights on directions for future Hyperphysical User Interface studies.

Chapter 6

Future Work

ch6:Future

This section is about the methods section of the EDS Storage Experiment. It was planned to contain information from an IRB-approved user study. At my thesis advisor Professor Zyda's suggestion, it is moved to the future works section to focus on the thesis and defense

6.1 Extradimensional Space Storage Introduction

In this study, we seek to compare different inventory systems in Virtual Reality (VR). In particular, we are interested in comparing inventory systems with Hyperphysical User Interfaces, that is, user interfaces that follow rules of physics that are different from the real world. Given that VR allows for inventory systems that cannot exist in our real physical world, we seek to understand the pros and cons of such systems.

Inventory systems are necessary for the continued evolution of VR toward richer, more diverse, and more complex VR experiences. They can provide the user with additional tools, options, and interactions. This allows the VR experience to move from featuring very limited ability options to one where users can easily transition to different contexts by using their inventory and accessing new abilities. Some examples where inventory would be useful would be intricate VR games where users are provided with various gameplay challenges that can be passed by different equipment or items. Another would be an all-purpose work environment where users can work on their 3D sculptures and transition smoothly to writing documentation.

To utilize an inventory system, users will use a variety of 3D selection and manipulation techniques as needed for the system. While there are many works on 3D selection and manipulation techniques in VR, few examine how the techniques can be used as a comprehensive solution (e.g., an inventory system). Often, a proposed series of techniques to interact with an inventory system might be tested in isolation and would not generalize to everyday scenarios.

On Google Scholar, searching "virtual reality" and "inventory system" in the title will get 6 results, with only 3 that are relevant toward a taxonomy of inventory systems for virtual reality games.[31] and "" I Packed My Bag and in It I Put…": A Taxonomy of Inventory Systems for Virtual Reality Games."[29] are two works by Cmentowski et al. that focus on building a taxonomy of an inventory system. While they proposed and built their own takes on different inventory systems, they have only evaluated them with a qualitative approach, but not with a quantitative approach.

"Game-Ready Inventory Systems for Virtual Reality"[98] is the third work. This work by Mußmann et al. also follows a qualitative approach with a Requirement Analysis.

In short, there exists a lack of research in the area of a comprehensive interaction solution in virtual reality. The existing research either does not evaluate techniques as a whole or focuses only on the qualitative aspects. This study seeks to bridge the gap by examining a suite of interaction techniques as a whole in the context of inventory systems and studying them both quantitatively and qualitatively.

6.1.1 Objectives

The goal of this study is to understand the user's ability to interact with hyperphysical inventory systems, specifically, in terms of retrieval tasks and storage tasks. Hyperphysical inventory systems are inventory systems utilizing a Hyperphysical User Interface, which is a user interface that follows laws of physics different from the real world. In this study, we use the terms storage system and inventory system interchangeably.

One such example is an Extradimensional Space Storage (EDS Storage). It is a storage system in which the storage space is larger than the physical dimension of the container. Users can interact with an EDS Storage by reaching into the container and grabbing an object that is stored in a space outside of the container space. Alternatively, the user can use it as a portal to enter the Extradimensional Space in which the actual storage space resides and make full use of that storage space. Such an EDS Storage system cannot exist in our physical world. Our motivation is to explore the pros and cons of some hyperphysical storage systems compared to traditional methods of storage interaction with the ultimate goal of providing VR developers with a better understanding of whether these systems and the associated interaction techniques are appropriate for their work.

6.1.2 Interaction Techniques

For this work, we'll compare the following four techniques. The first is the 2D Grid Inventory, which is the most common inventory system and serves as our baseline. Next, we have the Bag of Holding technique, which uses non-linear movement. The third is extradimensional space storage, where the user enters a different space altogether. Lastly, we have Flick Storage, which

Name	Shorthand	Description
2D Grid Inventory	2DUI	Grabbing item laid out on a 2D User Interface grid
Bag of Holding	Nonlinear	Grabbing items in a 3D volume where the item positions are a non-linear mapping from their actual positions in the Extradimensional Space storage.
Extradimensional Space Storage	EDS Storage	Entering an Extradimensional Space Storage and grabbing objects stored within as if one would in a storage room
Flick Storage	Flick	Accessing items stored on the surface of a sphere by rotating the user's controller

Table 6.1: List of different inventory techniques to be compared, their shorthand name, and a brief description

tab:EdsTechniquesComparison

is a variation of the Flick Typing Text Entry technique in [155]. Instead of choosing letters for text entry, the same flick interaction is used for storing and retrieving items.

6.1.3 Research Questions

In this study, we aim to investigate these research questions:

1. How do the techniques compare in terms of efficiency?
   (a) How do the different techniques compare regarding task completion time?
   (b) How do the different techniques compare regarding accumulated head movement required for task completion?
   (c) How do the different techniques compare regarding accumulated hand movement required for task completion?
2. How do the techniques compare in the effort required for each interaction technique?
   (a) How do the techniques compare in terms of mental demand?
   (b) How do the techniques compare in terms of physical demand?
   (c) How do the techniques compare in terms of temporal demand?
   (d) How do the techniques compare in terms of effort?
3. What makes users prefer one technique over another?
   (a) In what scenario does the user prefer one technique over another?
   (b) Why does the user prefer one technique over another?

6.2 Study

6.2.1 Recruitment

We recruited from the University of Southern California and made use of online recruitment such as the school mailing lists and various social media channels.

Recruitment Methods We used convenience sampling + snowball sampling.

We targeted USC students from the Viterbi School of Engineering, the School of Cinematic Arts, and the Iovine and Young Academy. These schools are more favorable to recruit from as they have courses related to VR. The participants will be more likely to have a wide range of participants with prior VR experiences.

Screening Tool We provided potential participants with an online screening survey to see whether they met the study inclusion/exclusion criteria.

Inclusion Criteria

- Speak, read, and comprehend English
- Age over 18
- Attend an in-person 90-minute user study
- Ability to stand for 40 minutes for the lab study
- Has full use of two hands
- Have 20/20 vision or corrected to 20/20 vision

Exclusion Criteria

- NOT color blind
- NOT pregnant
- Do NOT have a history of epilepsy seizures, or severe motion sickness
- Do NOT have a contact-transmitted disease (e.g., cold, flu, conjunctivitis or pink eye)

6.2.2 Experiment

Pre-Study We set participants up in a quiet environment with a pre-configured VR headset that has the task environment already set up. The participants were informed about the study and provided with the Informed Consent Form to read and agree to 1) participate, 2) be video recorded and data logged in the Virtual Reality lab study, and 3) audio recorded for the interview.

Study We performed a Within Subjects study. After participants consented, they did a warm-up in Virtual Reality to get familiar with VR. Then, they experienced each technique in a treatment, repeated four times total, once for each technique. During the treatment, they had a walkthrough of the technique, followed by tasks, and then a short break where they filled out the questionnaire. After all the treatments were completed, we conducted a semi-structured interview. Finally, the participant filled out a brief demographic survey, was debriefed, and was given compensation.

Study Flow

11 min	Informed Consent Form
11 min	Warm-Up (Wearing VR, VR Equipment Introduction, Safety)
40 min, 10 each	Treatments
2 min	Walkthrough & Tutorial
6 min	Tasks
2 min	Break & Questionnaire
20 min	Semi-Structured Interview
5 min	Debrief

- 11 + 11 + 4 * (2+6+2) + 20 + 5= 87 minutes in total

Treatments Participants were assigned one of the four techniques to start. Participants performed storage and retrieval tasks evenly divided between each technique, randomly ordered for counterbalancing.

Prior to beginning the tasks in each treatment, participants were given a walkthrough on the use of the technique. Following this tutorial, participants performed a warm-up exercise with the technique. These introduction and warm-up activities are expected to take 2 minutes to complete. Once participants gained a basic understanding of the technique, they were given a series of tasks. At the conclusion of a task, participants completed a post-task questionnaire (about cognitive load, see appendix) before moving on to the next task. Each task is expected to take 5-10 minutes to complete.

We made audio and video recordings of the participants as they completed their tasks. Their actions were captured by a data log.

Semi-Structured Interview Participants took part in a post-study semi-structured interview. The semi-structured interviews are expected to take 15-20 minutes to complete. We asked participants to rank the techniques and explain why the ranking was given. We also asked them what they would change about each technique if they can change anything about it.

We made audio recordings of the semi-structured interview.

Post-Study After the interview, participants filled out a demographic survey. We asked participants to fill out the survey at the end to avoid any bias to the previous components of the study. We debriefed, thanked, and compensated the participants at the end of the study.

6.2.3 Instrumentation

6.2.3.1 Questionnaires

For psychometric tests, we used a modified version of the NASA-TLX questionnaire[63] with a 5-point Likert Scale instead of the 7-point Likert Scale. We also added an additional question of "Preference: How much do you like using this interaction technique?" which will be rated between Love it and Hate it.

We also used the System Usability Scale (SUS)[18].

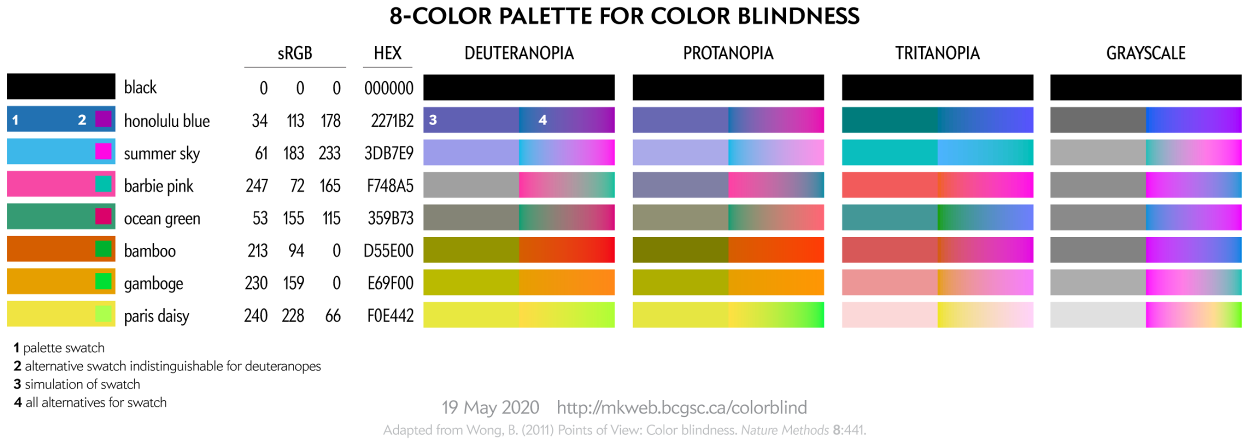
6.2.3.2 Qualitative Instruments

For the semi-structured interview that we conducted after the lab study, we asked open-ended questions on the different techniques they have tried in the lab study. We asked them to tell us about how they like each of the four techniques they have tried in the intervention. This is done by asking them which one is their favorite and why and proceeding down the list.

6.3 Implementation

Inventory To ensure the different inventory can be compared fairly, the inventory systems are designed to fit within a volume of the size 50 cm x 50 cm x 50 cm when possible. For the 2D User Interface, it is 50 cm x 50 cm by 5 cm. For the nonlinear technique, it is 50 cm x 50 cm x 50 cm.

Storeable Items Just like the inventory, storeable items (items that can be stored in the inventory system) are designed to fit within a volume of 10 cm x 10 cm x 10 cm. The shapes of these storeable items are selected for distinct silhouettes, as shown below.



8-COLOR PALETTE FOR COLOR BLINDNESS

		sRGB			HEX	DEUTERANOPIA	PROTANOPIA	TRITANOPIA	GRAYSCALE
	black	0	0	0	000000				
1 2	honolulu blue	34	113	178	2271B2	3 4			
	summer sky	61	183	233	3DB7E9				
	barbie pink	247	72	165	F748A5				
	ocean green	53	155	115	359B73				
	bamboo	213	94	0	D55E00				
	gamboge	230	159	0	E69F00				
	paris daisy	240	228	66	F0E442				

1 palette swatch
2 alternative swatch indistinguishable for deuteranopes
3 simulation of swatch
4 all alternatives for swatch

19 May 2020 http://mkweb.bcgsc.ca/colorblind

Adapted from Wong, B. (2011) Points of View: Color blindness. *Nature Methods* **8**:441.

Figure 6.1: 8-color palette from Martin Krzywinski, which is adapted from Bang Wong. The color palette can alleviate some impact of color blindness

fig:8colorpalette

Item Shapes

- Cube
- Sphere
- Triangular Pyramid/Tetrahedron
- Spring / Helix
- Torus / Donut
- Star Polygon
- 3D Square Cross

Item Color While color blindness is an exclusion criterion for the study, the colors of the storeable items are chosen to make them distinctive and reduce the impact of color blindness. We use the conservative 8-color palette for color blindness from `http://mkweb.bcgsc.ca/colorblind/palettes.mhtml#conservative-8-color-palette-for-colorbliness`, which is in turn adopted from Nature Method's Points of View: Color blindness by Bang Wong[148]. This is shown in Figure 6.1.

6.4 Study Data

The data that we will collect can be split into four categories: qualitative questionnaire data, quantitative lab study data, quantitative demographic information, and qualitative interview responses. We plan to apply the following analysis to the data: Qualitative Questionnaire data: We will ask each participant to fill in a modified NASA-TLX form and a System Usability Scale form for each Inventory System technique, both with a 5-point Likert Scale. After all experiments are completed, for each technique, we will calculate the average score for all questions, and convert it into a 0 100 scale. As we add one question regarding the preference for each technique in the NASA-TLX form, we will also calculate the average preference score for each technique, and make a comparison among them.

6.4.1 Quantitative Lab Study data

We will record the spatial position and rotation of the headset and two motion controllers used by each participant at each time frame during the experiments. After all experiments are finished, we will accumulate the position and rotation data, to get the total movement distance and rotation angles of participants' heads and two hands when using each Inventory System technique. We will run statistical analyses such as ANOVA if the data follows normal distribution or Kruskal-Wallis if the data do not follow normal distribution. We will also run paired t-tests or Wilcoxon signed-rank tests for post hoc analysis.

6.4.2 Demographic information

We may run statistical analyses on the data mentioned above with respect to the demographic information of the participants. This is to determine whether factors in the demographic information influence the experiment results. This may include whether the user is left-handed or right-handed, though the interaction technique will be designed to be usable by either hand.

6.4.3 Interview responses

As we will ask participants to give a ranking of the four techniques that they experienced in the experiment and their reasons, we will try to quantify the subjective factors that influence the user experience, the effort, and the efficiency of the tested Inventory Systems.

We will also ask participants about their suggestions for improving the four techniques so that we can summarize these suggestions and report them in our paper. We may run ground theory analysis or qualitative coding to understand better what participants look for in an inventory system.

Chapter 7

Conclusions

ch7:Conclusions

7.1 Introduction

The main objective of this thesis is to explore design principles of 3D interaction techniques in the domain of Extended Reality that advances in body tracking technology will enable.

A series of conceptual designs have been created and prototypes have been built to understand the design space. This ultimately led to the theory and systems presented in this thesis. In this final chapter, I'll briefly summarize the results of each work and discuss the findings of this thesis.

7.2 Statement of the Problem

As discussed in Chapter 1 and Section 3b.4, XR emphasizes certain qualities that other computing paradigms (WIMP, Mobile) do not place as much emphasis on. While it does not have exclusivity over those qualities, the emphasis greatly shapes its computing and interaction paradigm.

For example, advances in technology are making more of the user's body available to participate in interaction. This thesis prepares for the future by looking ahead and establishing design principles, guidelines, and models for how we could utilize these interactors if they were available.

The spatial nature of XR has also led researchers and developers to create and apply many interaction techniques and approaches to take advantage of what XR offers. 3D user interfaces and Natural User Interfaces have been utilized to make XR more intuitive, but at the cost of time and effort. Furthermore, the many situations and contexts that users may encounter in XR include what may be encountered in other paradigms and more, making it difficult for a one-shoe-fits-all approach. The user needs a Swiss Army Knife that can provide the right tool for the right job.

Hyperphysical User Interfaces make use of an alternative set of rules for developers and users to understand different interaction techniques. It allows new techniques to be developed which may be a better fit for XR applications. Furthermore, it can aid in managing and accessing the best interaction technique for the situation. However, hyperphysicality is an under-explored area where new hyperphysical techniques are often stumbled upon by serendipity. This thesis solves this problem by providing a framework that developers can use to understand existing hyperphysical techniques and to develop their own.

7.3 Model & Lenses & Taxonomy

The Spatial Interaction Model, Design Lenses, and Gesture Taxonomy serve as the theoretical foundation of this thesis.

The Spatial Interaction Model, as discussed in Chapter 3a, serves as the pin that ties all the concepts together. This model takes into account qualities that are prominent in XR as well as introducing hyperphysicality as a source of meaning to understand a spatial gesture.

The design lenses, on the other hand, provide important concepts and questions for interaction design. At the high level, there are the design lenses of hyperphysical user interface, whole-body interaction, and extradimensional space. Designers can apply these design lenses to the Spatial Interaction Model when choosing or creating interaction techniques.

A Gesture Taxonomy is also presented in Chapter 3b. The taxonomy is useful for classifying existing gestures or exploring new interaction techniques. In particular, the breakdown of the Nature dimension from previous work in Section 3b.8 is instrumental to understanding how hyperphysicality can be applied in the Spatial Interaction Model.

7.4 School of Spatial Sorcery & Virtual Equipment System & Extradimensional Space Storage

The School of Spatial Sorcery in Chapter 3c, Virtual Equipment System in Chapter 3d, and Extradimensional Space Storage in Chapter 3e are exploratory prototypes that are based on and implemented for this thesis.

School of Spatial Sorcery is an exploration of spatial interactions using the creation of spells as a substitute for day-to-day interaction tasks. This has allowed us to consider our theory work from the context and perspective of an in-game experience. The fantasy elements encourage both the developers and users to embrace hyperphysicality.

The Virtual Equipment System and Extradimensional Space Storage are two systems that came out of the School of Spatial Sorcery and facilitated the construction and discovery of design lenses. The Virtual Equipment System is a body-centric interface that uses equipment as interaction metaphors, primarily utilizing the design lenses of Hyperphysical User Interface and Whole-body Interaction.

As the Virtual Equipment System was applied to new contexts to solve different problems, the need to utilize the same physical space arose. The user needs not just more interaction techniques, but a way to manage interaction techniques, interactors, and interactables. This leads to Extradimensional Space Storage and highlights the need for Extradimensional Space to be singled out from hyperphysicality into its own design lens.

7.5 Final Thoughts and Future Development

It is inevitable that some interaction techniques will dominate a computing paradigm. However, XR has many different use cases that will require the use of non-dominant interaction techniques. The similarity of XR to real life means non-hyperphysical techniques will always compete to be the user's primary interaction technique. Instead of searching for a jack-of-all-trades interaction technique that would fit most situations, we should continue to develop specialized techniques using the design lenses offered here. With hyperphysicality, the sky is the limit to what our interactions can do. Meanwhile, whole-body interaction can ground us to what is desirable out of the infinite possibilities.

Lastly, and perhaps more importantly, we need to make it easy to access the best technique for the job at hand. Continued development of Extradimensional Space storage would work toward this goal, offering a repository for the many XR interactions that will best solve problems we face in daily life. Together, these design lenses will create a future, described by Arthur

C. Clarke's Third Law, with advanced interactions that will be truly indistinguishable from magic.

Bibliography

[1] Johnny Accot and Shumin Zhai. “More than dotting the i’s—foundations for crossing-based interfaces”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2002, pp. 73–80.

[2] Johnny Accot and Shumin Zhai. “More than dotting the i’s—foundations for crossing-based interfaces”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2002, pp. 73–80.

[3] Roland Aigner, Daniel Wigdor, Hrvoje Benko, Michael Haller, David Lindbauer, Alexandra Ion, Shengdong Zhao, and JTKV Koh. “Understanding mid-air hand gestures: A study of human preferences in usage of gesture types for hci”. In: *Microsoft Research TechReport MSR-TR-2012-111* 2 (2012), p. 30.

[4] Mike Alger. “Visual design methods for virtual reality”. In: *Ravensbourne. http://aperturesciencellc. com/vr/VisualDesignMethodsforVR_MikeAlger. pdf* (2015).

[5] Mike Alger. “VR interface design manifesto”. In: *Online video clip. YouTube* 6 (2015).

[6] Mike Alger. “VR interface design pre-visualisation methods”. In: *Online video clip. YouTube* 4 (2015).

[7] Eswar Anandapadmanaban, Jesslyn Tannady, Johannes Norheim, Dava Newman, and Jeff Hoffman. “Holo-SEXTANT: an augmented reality planetary EVA navigation interface”. In: 48th International Conference on Environmental Systems. 2018.

[8] Yangyi Ang, Puteri Suhaiza Sulaiman, Rahmita Wirza OK Rahmat, and Noris Mohd Norowi. “Swing-in-place (SIP): a less fatigue walking-in-place method with side-viewing functionality for mobile virtual reality”. In: *IEEE Access* 7 (2019), pp. 183985–183995.

[9] Jussi Ängeslevä, Sile O’Modhrain, Ian Oakley, and Stephen Hughes. “Body mnemonics”. In: *Physical Interaction (PI03) Workshop on Real World User Interfaces*. Citeseer. 2003, p. 35.

[10] Jussi Ängeslevä, Ian Oakley, Stephen Hughes, and Sile O'Modhrain. "Body Mnemonics Portable device interaction design concept". In: *Proceedings of UIST*. Vol. 3. Citeseer. 2003, pp. 2–5.

[11] Alissa N Antle, Theresa Jean Tanenbaum, Anna Macaranas, and John Robinson. "Games for change: Looking at models of persuasion through the lens of design". In: *Playful user interfaces: Interfaces that invite social and physical interaction* (2014), pp. 163–184.

[12] Natalie Armitage. "European and African figural ritual magic: The beginnings of the voodoo doll myth". In: *The materiality of magic: An artifactual investigation into ritual practices and popular beliefs* (2015), pp. 85–101.

[13] Rahul Arora, Rubaiat Habib Kazi, Danny M Kaufman, Wilmot Li, and Karan Singh. "Magicalhands: Mid-air hand gestures for animating in vr". In: *Proceedings of the 32nd annual ACM symposium on user interface software and technology*. 2019, pp. 463–477.

[14] Andrew Atkins, Serge Belongie, and Harald Haraldsson. "Continuous Travel In Virtual Reality Using a 3D Portal". In: *Adjunct Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology*. 2021, pp. 51–54.

[15] Miguel Borges, Andrew Symington, Brian Coltin, Trey Smith, and Rodrigo Ventura. "HTC vive: Analysis and accuracy improvement". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 2610–2615.

[16] Evren Bozgeyikli, Andrew Raij, Srinivas Katkoori, and Rajiv Dubey. "Point & teleport locomotion technique for virtual reality". In: *Proceedings of the 2016 annual symposium on computer-human interaction in play*. 2016, pp. 205–216.

[17] W Russell Brain. "Visual orientation with special reference to lesions of the right cerebral hemisphere." In: *Brain: A Journal of Neurology* (1941).

[18] John Brooke. "Sus: a "quick and dirty'usability". In: *Usability evaluation in industry* 189.3 (1996), pp. 189–194.

[19] Scott Bukatman. "17:: Some Observations Pertaining to Cartoon Physics; or, The Cartoon Cat in the Machine". In: (2014).

[20] Alessandro Carfì and Fulvio Mastrogiovanni. "Gesture-Based Human-Machine Interaction: Taxonomy, Problem Definition, and Analysis". In: *IEEE Transactions on Cybernetics* (2021).

[21] Christopher Carmichael, Marco Valdez Balderas, Bill Ko, Atiya Nova, Angela Tabafunda, and Alvaro Uribe-Quevedo. "Spring Stepper: A Seated VR Locomotion Controller". In: *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*. IEEE. 2020, pp. 346–350.

[22] Justine Cassell. “A framework for gesture generation and interpretation”. In: *Computer vision in human-machine interaction* (1998). Publisher: Cambridge University Press New York, pp. 191–215.

[23] Benjamin Cellini, Marioalberto Ferrero, and Jean-Michel Mongeau. “Drosophila flying in augmented reality reveals the vision-based control autonomy of the optomotor response”. In: *Current Biology* 34.1 (2024), pp. 68–78.

[24] Tuochao Chen, Benjamin Steeper, Kinan Alsheikh, Songyun Tao, François Guimbretière, and Cheng Zhang. “C-face: Continuously reconstructing facial expressions by deep learning contours of the face with ear-mounted miniature cameras”. In: *Proceedings of the 33rd annual ACM symposium on user interface software and technology*. 2020, pp. 112–125.

[25] Xiang Anthony Chen. “Body-centric interaction with a screen-based handheld device”. In: (2012).

[26] Xiang’Anthony’ Chen. “Body-centric interaction with mobile devices”. In: *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*. 2012, pp. 385–386.

[27] Xiang’Anthony’ Chen, Nicolai Marquardt, Anthony Tang, Sebastian Boring, and Saul Greenberg. “Extending a mobile device’s interaction space through body-centric interaction”. In: *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*. 2012, pp. 151–160.

[28] Xiang’Anthony’ Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott Hudson. “Around-body interaction: sensing & interaction techniques for proprioception-enhanced input with mobile devices”. In: *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*. 2014, pp. 287–290.

[29] Sebastian Cmentowski, Andrey Krekhov, and Jens Krüger. *"I Packed My Bag and in It I Put...": A Taxonomy of Inventory Systems for Virtual Reality Games*. 2021. arXiv: `2107.08434 [cs.HC]`.

[30] Sebastian Cmentowski, Andrey Krekhov, and Jens Krüger. “” I Packed My Bag and in It I Put… ”: A Taxonomy of Inventory Systems for Virtual Reality Games”. In: *2021 IEEE Conference on Games (CoG)*. IEEE. 2021, pp. 1–8.

[31] Sebastian Cmentowski, Andrey Krekhov, Ann-Marie Müller, and Jens Krüger. “Toward a taxonomy of inventory systems for virtual reality games”. In: *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*. 2019, pp. 363–370.

[32] Raphael Costa, Rongkai Guo, and John Quarles. “Towards usable underwater virtual reality systems”. In: *2017 IEEE Virtual Reality (VR)*. IEEE. 2017, pp. 271–272.

[33] Raphael Costa and John Quarles. “3D Interaction with Virtual Objects in Real Water”. In: *2019 11th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*. IEEE. 2019, pp. 1–7.

[34] Gray Crawford. “Developing Embodied Familiarity with Hyperphysical Phenomena”. PhD thesis. Carnegie Mellon University, 2019.

[35] Michael F Deering. “Making virtual reality more real: Experience with the virtual portal”. In: *Graphics Interface*. CANADIAN INFORMATION PROCESSING SOCIETY. 1993, pp. 219–219.

[36] Massimiliano Di Luca, Hasti Seifi, Simon Egan, and Mar Gonzalez-Franco. “Locomotion vault: the extra mile in analyzing vr locomotion techniques”. In: *Proceedings of the 2021 CHI conference on human factors in computing systems*. 2021, pp. 1–10.

[37] Giuseppe Di Pellegrino and Elisabetta Làdavas. “Peripersonal space in the brain”. In: *Neuropsychologia* 66 (2015), pp. 126–133.

[38] David Dobbelstein. “Near-body interaction for wearable interfaces”. PhD thesis. Universität Ulm, 2020.

[39] David Dobbelstein. “Unobtrusive interaction for wearable computing”. In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. 2016, pp. 203–207.

[40] David Dobbelstein, Tobias Arnold, and Enrico Rukzio. “Snapband: a flexible multi-location touch input band”. In: *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. 2018, pp. 214–215.

[41] David Dobbelstein, Philipp Hock, and Enrico Rukzio. “Belt: An unobtrusive touch input device for head-worn displays”. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 2015, pp. 2135–2138.

[42] David Dobbelstein, Christian Winkler, Gabriel Haas, and Enrico Rukzio. “PocketThumb: A wearable dual-sided touch interface for cursor-based control of smart-eyewear”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.2 (2017), pp. 1–17.

[43] Francine L Dolins, Kenneth Schweller, and Scott Milne. “Technology advancing the study of animal cognition: using virtual reality to present virtually simulated environments to investigate nonhuman primate spatial cognition”. In: *Current Zoology* 63.1 (2017), pp. 97–108.

[44] Ze Dong, Jingjing Zhang, Robert Lindeman, and Thammathip Piumsomboon. “Surface vs Motion Gestures for Mobile Augmented Reality”. en. In: *Symposium on Spatial User Interaction*. Virtual Event Canada: ACM, Oct. 2020, pp. 1–2. ISBN: 978-1-4503-7943-4. DOI: `10.1145/3385959.3422694`. (Visited on 07/19/2021).

[45] David Efron. “Gesture and environment.” In: (1941). Publisher: King’s crown Press.

[46] Barrett Ens, Aaron Quigley, Hui-Shyong Yeo, Pourang Irani, Thammathip Piumsomboon, and Mark Billinghurst. “Counterpoint: Exploring Mixed-Scale Gesture Interaction for AR Applications”. en. In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. Montreal QC Canada: ACM, Apr. 2018, pp. 1–6. ISBN: 978-1-4503-5621-3. DOI: `10.1145/3170427.3188513`. (Visited on 09/14/2022).

[47] Chao Feng, Nan Wang, Yicheng Jiang, Xia Zheng, Kang Li, Zheng Wang, and Xiaojiang Chen. “Wi-learner: Towards one-shot learning for cross-domain wi-fi based gesture recognition”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6.3 (2022), pp. 1–27.

[48] Chao-Kang Feng, Timothy K Shih, Hui-Huang Hsu, Chun-Hong Huang, Nick C Tang, Stanislav Klimenko, Valery Afanasiev, and Eugene Slobodyuk. “Space Walker: a Hyper Interaction Platform for Cosmonaut Training”. In: *GraphiCon 2006-International Conference on Computer Graphics and Vision*. 2006, pp. 22–26.

[49] Damon Young Florian‘Floyd’Mueller. “Five lenses for designing exertion experiences”. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM. 2017.

[50] Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. “ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures”. In: *Proceedings of the ACM international conference on interactive tabletops and surfaces*. 2009, pp. 165–172.

[51] Jann Philipp Freiwald, Oscar Ariza, Omar Janeh, and Frank Steinicke. “Walking by Cycling: A Novel In-Place Locomotion User Interface for Seated Virtual Reality Experiences.” In: *CHI*. 2020, pp. 1–12.

[52] Sofia Fröjdman. *User experience guidelines for design of virtual reality graphical user interfaces controlled by head orientation input*. 2016.

[53] Luciano Gamberini, Bruno Seraglia, and Konstantinos Priftis. “Processing of peripersonal and extrapersonal space using tools: Evidence from visual line bisection in real and virtual environments”. In: *Neuropsychologia* 46.5 (2008), pp. 1298–1304.

[54] Valve Games. *Half Life: Alyx*. [CD-ROM]. 2020.

[55] Thomas van Gemert, Kasper Hornbæk, Jarrod Knibbe, and Joanna Bergström. “Towards a Bedder Future: A Study of Using Virtual Reality while Lying Down”. In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023, pp. 1–18.

[56] Paul George, Indira Thouvenin, Vincent Fremont, and Véronique Cherfaoui. “DAARIA: Driver assistance by augmented reality for intelligent automobile”. In: *2012 IEEE Intelligent Vehicles Symposium*. IEEE. 2012, pp. 1043–1048.

[57] Barney G Glaser and Anselm L Strauss. *The discovery of grounded theory: Strategies for qualitative research.* Routledge, 2017.

[58] Jan Gugenheimer, David Dobbelstein, Christian Winkler, Gabriel Haas, and Enrico Rukzio. “FaceTouch: Touch interaction for mobile virtual reality”. In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* 2016, pp. 3679–3682.

[59] Jan Gugenheimer, David guDobbelstein, Christian Winkler, Gabriel Haas, and Enrico Rukzio. “Facetouch: Enabling touch interaction in display fixed uis for mobile virtual reality”. In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology.* 2016, pp. 49–60.

[60] Edward T Hall. “A system for the notation of proxemic behavior”. In: *American anthropologist* 65.5 (1963), pp. 1003–1026.

[61] Dustin T Han, Mohamed Suhail, and Eric D Ragan. “Evaluating remapped physical reach for hand interactions with passive haptics in virtual reality”. In: *IEEE transactions on visualization and computer graphics* 24.4 (2018), pp. 1467–1476.

[62] Shangchen Han, Beibei Liu, Randi Cabezas, Christopher D Twigg, Peizhao Zhang, Jeff Petkau, Tsz-Ho Yu, Chun-Jung Tai, Muzaffer Akbay, Zheng Wang, et al. “MEgATrack: monochrome egocentric articulated hand-tracking for virtual reality”. In: *ACM Transactions on Graphics (ToG)* 39.4 (2020), pp. 87–1.

[63] Sandra G Hart and Lowell E Staveland. “Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research”. In: *Advances in psychology.* Vol. 52. Elsevier, 1988, pp. 139–183.

[64] Philipp Hock, Sebastian Benedikter, Jan Gugenheimer, and Enrico Rukzio. “Carvr: Enabling in-car virtual reality entertainment”. In: *Proceedings of the 2017 CHI conference on human factors in computing systems.* 2017, pp. 4034–4044.

[65] Scott E Hudson, Chris Harrison, Beverly L Harrison, and Anthony LaMarca. “Whack gestures: inexact and inattentive interaction with mobile devices”. In: *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction.* 2010, pp. 109–112.

[66] Bashima Islam, Md Mahbubur Rahman, Tousif Ahmed, Mohsin Yusuf Ahmed, Md Mehedi Hasan, Viswam Nathan, Korosh Vatanparvar, Ebrahim Nemati, Jilong Kuang, and Jun Alex Gao. “BreathTrack: detecting regular breathing phases from unannotated acoustic data captured by a smartphone”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5.3 (2021), pp. 1–22.

[67] Pranavi Jalapati, Satya Naraparaju, Powen Yao, and Michael Zyda. “Integrating Sensor Fusion with Pose Estimation for Simulating Human Interactions in Virtual Reality”. In: *HCI International 2022–Late Breaking Papers: Interacting with eXtended Reality and Artificial Intelligence: 24th International Conference on Human-Computer Interaction, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings*. Springer Nature Switzerland Cham. 2022, pp. 74–87.

[68] Xianta Jiang, Lukas-Karim Merhi, Zhen Gang Xiao, and Carlo Menon. “Exploration of force myography and surface electromyography in hand gesture classification”. In: *Medical engineering & physics* 41 (2017), pp. 63–73.

[69] Adityan Jothi, Powen Yao, Andrew Zhao, Mark Miller, Sloan Swieso, and Michael Zyda. “Toward Predicting User Waist Location From VR Headset and Controllers Through Machine Learning”. In: *Proceedings of the 2021 ACM Symposium on Spatial User Interaction*. 2021, pp. 1–2.

[70] Adrien Jouary, Mathieu Haudrechy, Raphaël Candelier, and German Sumbre. “A 2D virtual reality system for visual goal-driven navigation in zebrafish larvae”. In: *Scientific Reports* 6.1 (2016), p. 34015.

[71] Maria Karam. “A taxonomy of gestures in human computer interactions”. In: (2005). ISBN: 0854328335.

[72] Victor Kartsch, Simone Benatti, Mattia Mancini, Michele Magno, and Luca Benini. “Smart wearable wristband for EMG based gesture recognition powered by solar energy harvester”. In: *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2018, pp. 1–5.

[73] Adam Kendon. “How gestures can become like words.” In: *This paper is a revision of a paper presented to the American Anthropological Association, Chicago, Dec 1983.* Hogrefe & Huber Publishers. 1988.

[74] Rajiv Khadka and Amy Banic. “Body-prop interaction: Evaluation of augmented open discs and egocentric body-based interaction”. In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE. 2019, pp. 1705–1710.

[75] Rajiv Khadka and Amy Banić. “Effects of Egocentric Versus Exocentric Virtual Object Storage Technique on Cognition in Virtual Environments”. In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE. 2020, pp. 205–209.

[76] Rajiv Khadka and Amy Banić. “Prop-based egocentric and exocentric virtual object storage techniques”. In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE. 2020, pp. 778–779.

[77] Rajiv Khadka, James Money, and Amy Banić. “Body-prop interaction: Augmented open discs and egocentric body-based interaction for exploring immersive visualizations”. In: *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*. 2018, pp. 327–332.

[78] Takumi Kitagawa, Yuki Yamato, Buntarou Shizuki, and Shin Takahashi. “A Viewpoint Control Method for 360 Media Using Helmet Touch Interface”. In: *Symposium on Spatial User Interaction*. 2019, pp. 1–2.

[79] Scott R Klemmer, Björn Hartmann, and Leila Takayama. “How bodies matter: five themes for interaction design”. In: *Proceedings of the 6th conference on Designing Interactive systems*. 2006, pp. 140–149.

[80] Luv Kohli. “Redirected touching: Warping space to remap passive haptics”. In: *2010 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE. 2010, pp. 129–130.

[81] Glenn E Krasner, Stephen T Pope, et al. “A description of the model-view-controller user interface paradigm in the smalltalk-80 system”. In: *Journal of object oriented programming* 1.3 (1988), pp. 26–49.

[82] Wallace S Lages and Doug A Bowman. “Walking with adaptive augmented reality workspaces: design and usage patterns”. In: *Proceedings of the 24th International Conference on Intelligent User Interfaces*. 2019, pp. 356–366.

[83] Joseph J LaViola Jr, Ernst Kruijff, Ryan P McMahan, Doug Bowman, and Ivan P Poupyrev. *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[84] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. “Lasense: Pushing the limits of fine-grained activity sensing using acoustic signals”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6.1 (2022), pp. 1–27.

[85] Jingyi Li, Ceenu George, Andrea Ngao, Kai Holländer, Stefan Mayer, and Andreas Butz. “An exploration of users’ thoughts on rear-seat productivity in virtual reality”. In: *12th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. 2020, pp. 92–95.

[86] Alex F Lim, Jonathan W Kelly, Nathan C Sepich, Lucia A Cherep, Grace C Freed, and Stephen B Gilbert. “Rotational self-motion cues improve spatial learning when teleporting in virtual environments”. In: *Proceedings of the 2020 ACM Symposium on Spatial User Interaction*. 2020, pp. 1–7.

[87] Paul Boguslaw Lubos. “Supernatural and comfortable user interfaces for basic 3d interaction tasks”. PhD thesis. Staats-und Universitätsbibliothek Hamburg Carl von Ossietzky, 2018.

[88] Chengwen Luo, Zhongru Yang, Xingyu Feng, Jin Zhang, Hong Jia, Jianqiang Li, Jiawei Wu, and Wen Hu. “Rfaceid: Towards rfid-based facial recognition”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5.4 (2021), pp. 1–21.

[89] Saif Mahmud, Ke Li, Guilin Hu, Hao Chen, Richard Jin, Ruidong Zhang, François Guimbretière, and Cheng Zhang. “PoseSonic: 3D Upper Body Pose Estimation Through Egocentric Acoustic Sensing on Smartglasses”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7.3 (2023), pp. 1–28.

[90] David McNeill. “Hand and Mind1”. In: *Advances in Visual Semiotics* (1992), p. 351.

[91] Christine M Mermier, Jeffrey M Janot, Daryl L Parker, and Jacob G Swan. “Physiological and anthropometric determinants of sport climbing performance”. In: *British journal of sports medicine* 34.5 (2000), pp. 359–365.

[92] Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. “Augmented reality: A class of displays on the reality-virtuality continuum”. In: *Telemanipulator and telepresence technologies*. Vol. 2351. Spie. 1995, pp. 282–292.

[93] Mark Miller, Powen Yao, Adityan Jothi, Andrew Zhao, Sloan Swieso, and Michael Zyda. “Virtual Equipment System: Toward Peripersonal Equipment Slots with Machine Learning”. In: *Symposium on Spatial User Interaction*. 2021, pp. 1–2.

[94] Sebastian Misztal, Guillermo Carbonell, Nils Ganther, and Jonas Schild. “Portals With a Twist: Cable Twist-Free Natural Walking in Room-Scaled Virtual Reality”. In: *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology*. 2020, pp. 1–3.

[95] Soumik Mondal and Patrick Bours. “Swipe gesture based continuous authentication for mobile devices”. In: *2015 International Conference on Biometrics (ICB)*. IEEE. 2015, pp. 458–465.

[96] Florian Mueller, Damon Young, et al. “10 Lenses to design sports-HCI”. In: *Foundations and Trends® in Human–Computer Interaction* 12.3 (2018), pp. 172–237.

[97] Florian’Floyd’ Mueller, Chek Tien Tan, Rich Byrne, and Matt Jones. “13 game lenses for designing diverse interactive jogging systems”. In: *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. 2017, pp. 43–56.

[98] Marc Mußmann, Samuel Truman, and Sebastian von Mammen. “Game-Ready Inventory Systems for Virtual Reality”. In: *2021 IEEE Conference on Games (CoG)*. IEEE. 2021, pp. 1–8.

[99] Brad A Myers. “A new model for handling input”. In: *ACM Transactions on Information Systems (TOIS)* 8.3 (1990), pp. 289–320.

[100] Hemal Naik, Renaud Bastien, Nassir Navab, and Iain D Couzin. “Animals in virtual environments”. In: *IEEE Transactions on Visualization and Computer Graphics* 26.5 (2020), pp. 2073–2083.

[101] Robin Neuhaus, Ronda Ringfort-Felner, Shadan Sadeghian, and Marc Hassenzahl. “To Mimic Reality or to Go Beyond?“Superpowers” in Virtual Reality, the Experience of Augmentation and Its Consequences”. In: *International Journal of Human-Computer Studies* (2023), p. 103165.

[102] Niels Christian Nilsson, Tabitha Peck, Gerd Bruder, Eri Hodgson, Stefania Serafin, Mary Whitton, Frank Steinicke, and Evan Suma Rosenberg. “15 years of research on redirected walking in immersive virtual environments”. In: *IEEE computer graphics and applications* 38.2 (2018), pp. 44–56.

[103] Sergi Olives Orfila. “Control systems in virtual reality video games”. In: (2021).

[104] Selcen Ozturkcan. “Service innovation: Using augmented reality in the IKEA Place app”. In: *Journal of Information Technology Teaching Cases* 11.1 (2021), pp. 8–13.

[105] W Park. “A multi-touch gesture vocabulary design methodolgy for mobile devices”. In: *Division of Mechanical and In-dustrial Engineering POSTECH* (2012).

[106] Randy Pausch, Jon Snoddy, Robert Taylor, Scott Watson, and Eric Haseltine. “Disney’s Aladdin: first steps toward storytelling in virtual reality”. In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 1996, pp. 193–203.

[107] Vladimir I Pavlovic, Rajeev Sharma, and Thomas S. Huang. “Visual interpretation of hand gestures for human-computer interaction: A review”. In: *IEEE Transactions on pattern analysis and machine intelligence* 19.7 (1997), pp. 677–695.

[108] Jeffrey S Pierce, Matthew Conway, Maarten van Dantzich, and George Robertson. “Toolspaces and glances: storing, accessing, and retrieving objects in 3D desktop applications”. In: *Proceedings of the 1999 symposium on Interactive 3D graphics*. 1999, pp. 163–168.

[109] Jeffrey S Pierce and Randy Pausch. “Comparing voodoo dolls and HOMER: exploring the importance of feedback in virtual environments”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2002, pp. 105–112.

[110] Jeffrey S Pierce, Brian C Stearns, and Randy Pausch. “Voodoo dolls: seamless interaction at multiple scales in virtual environments”. In: *Proceedings of the 1999 symposium on Interactive 3D graphics*. 1999, pp. 141–145.

[111] Domonkos Pinke, John B Issa, Gabriel A Dara, Gergely Dobos, and Daniel A Dombeck. “Full field-of-view virtual reality goggles for mice”. In: *Neuron* 111.24 (2023), pp. 3941–3952.

[112] Isabella Poggi. “From a typology of gestures to a procedure for gesture production”. In: *International Gesture Workshop*. Springer. 2001, pp. 158–168.

[113] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. "The go-go interaction technique: non-linear mapping for direct manipulation in VR". In: *Proceedings of the 9th annual ACM symposium on User interface software and technology.* 1996, pp. 79–80.

[114] Lisa Marie Prinz, Tintu Mathew, Simon Klüber, and Benjamin Weyers. "An overview and analysis of publications on locomotion taxonomies". In: *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2021, pp. 385–388.

[115] Francis KH Quek. "Toward a vision-based hand gesture interface". In: *Virtual reality software and technology.* World Scientific. 1994, pp. 17–31.

[116] Kasra Rahimi, Colin Banigan, and Eric D Ragan. "Scene transitions and teleportation in virtual reality and the implications for spatial awareness and sickness". In: *IEEE transactions on visualization and computer graphics* 26.6 (2018), pp. 2273–2287.

[117] B Rimé and L Schiaratura. *Gesture and speech, Fundamentals of nonverbal behavior, Edited Feldman RS.* 1991.

[118] Giacomo Rizzolatti, Cristiana Scandolara, Massimo Matelli, and Maurizio Gentilucci. "Afferent properties of periarcuate neurons in macaque monkeys. II. Visual responses". In: *Behavioural brain research* 2.2 (1981), pp. 147–163.

[119] Edgar Rojas-Muñoz and Juan P Wachs. "Magic: A fundamental framework for gesture representation, comparison and assessment". In: *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019).* IEEE. 2019, pp. 1–8.

[120] Thomas JL van Rompay, Sandra Oran, Mirjam Galetzka, and Agnes E van den Berg. "Lose yourself: Spacious nature and the connected self". In: *Journal of Environmental Psychology* 91 (2023), p. 102108.

[121] Sami Ronkainen, Jonna Häkkilä, Saana Kaleva, Ashley Colley, and Jukka Linjama. "Tap input as an embedded interaction method for mobile devices". In: *Proceedings of the 1st international conference on Tangible and embedded interaction.* 2007, pp. 263–270.

[122] Jaime Ruiz, Yang Li, and Edward Lank. "User-defined motion gestures for mobile interaction". In: *Proceedings of the SIGCHI conference on human factors in computing systems.* 2011, pp. 197–206.

[123] Claire Rusch, Eatai Roth, Clément Vinauger, and Jeffrey A Riffell. "Honeybees in a virtual reality environment learn unique combinations of colour and shape". In: *Journal of Experimental Biology* 220.19 (2017), pp. 3478–3487.

[124] Bhuvaneswari Sarupuri, Simon Hoermann, Frank Steinicke, and Robert W Lindeman. "Triggerwalking: a biomechanically-inspired locomotion user interface for efficient realistic virtual walking". In: *Proceedings of the 5th symposium on spatial user interaction.* 2017, pp. 138–147.

[125] Ehsan Sayyad, Misha Sra, and Tobias Höllerer. "Walking and teleportation in wide-area virtual reality experiences". In: *2020 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE. 2020, pp. 608–617.

[126] Jesse Schell. *The Art of Game Design: A book of lenses*. CRC press, 2008.

[127] Jonas Schjerlund, Kasper Hornbæk, and Joanna Bergström. "Ovrlap: Perceiving multiple locations simultaneously to improve interaction in vr". In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 2022, pp. 1–13.

[128] Marcos Serrano, Eric Lecolinet, and Yves Guiard. "Bezel-Tap gestures: quick activation of commands from sleep mode on tablets". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2013, pp. 3027–3036.

[129] Santhosh Simon, Ajith Prasad, and W Nelson. "Examining the potential of augmented reality to improve health and welfare of animals herded using virtual fencing". In: *Proceedings 1st Asian-Australasian Conference on Precision Pastures and Livestock Farming*. Vol. 6. 2017.

[130] Christian Sinnott, James Liu, Courtney Matera, Savannah Halow, Ann Jones, Matthew Moroz, Jeffrey Mulligan, Michael Crognale, Eelke Folmer, and Paul MacNeilage. "Underwater virtual reality system for neutral buoyancy training: Development and evaluation". In: *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology*. 2019, pp. 1–9.

[131] Anthony Steed, Tuukka M Takala, Daniel Archer, Wallace Lages, and Robert W Lindeman. "Directions for 3D user interface research from consumer VR games". In: *IEEE Transactions on Visualization and Computer Graphics* 27.11 (2021), pp. 4171–4182.

[132] Richard Stoakley, Matthew J Conway, and Randy Pausch. "Virtual reality on a WIM: interactive worlds in miniature". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1995, pp. 265–272.

[133] Richard Stoakley, Matthew J. Conway, and Randy Pausch. "Virtual Reality on a WIM: Interactive Worlds in Miniature". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '95. Denver, Colorado, USA: ACM Press/Addison-Wesley Publishing Co., 1995, pp. 265–272. ISBN: 0201847051. DOI: 10.1145/223904.223938.

[134] Ivan E Sutherland et al. "The ultimate display". In: *Proceedings of the IFIP Congress*. Vol. 2. 506-508. New York. 1965, pp. 506–508.

[135] Sloan Swieso, Powen Yao, Mark Miller, Adityan Jothi, Andrew Zhao, and Michael Zyda. "Toward Using Machine Learning-Based Motion Gesture for 3D Text Input". In: *Proceedings of the 2021 ACM Symposium on Spatial User Interaction*. 2021, pp. 1–2.

[136] M. Tavanti and M. Lind. *2D vs 3D, implications on spatial memory.* 2001. DOI: 10.1109/INFVIS.2001.963291.

[137] Jerald Thomas, Courtney Hutton Pospick, and Evan Suma Rosenberg. "Towards physically interactive virtual environments: Reactive alignment with redirected walking". In: *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology.* 2020, pp. 1–10.

[138] Bruce Tognazzini. "Principles, techniques, and ethics of stage magic and their application to human interface design". In: *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems.* 1993, pp. 355–362.

[139] Fereydoon Vafaei. "Taxonomy of gestures in human computer interaction". In: (2013). Publisher: North Dakota State University.

[140] Tijana Vuletic, Alex Duffy, Laura Hay, Chris McTeague, Gerard Campbell, and Madeleine Grealy. "Systematic literature review of hand gestures used in human computer interaction interfaces". In: *International Journal of Human-Computer Studies* 129 (2019), pp. 74–94.

[141] Lei Wang, Wei Li, Ke Sun, Fusang Zhang, Tao Gu, Chenren Xu, and Daqing Zhang. "Loear: Push the range limit of acoustic sensing for vital sign monitoring". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6.3 (2022), pp. 1–24.

[142] Penghao Wang, Ruobing Jiang, and Chao Liu. "Amaging: Acoustic hand imaging for self-adaptive gesture recognition". In: *IEEE INFOCOM 2022-IEEE Conference on Computer Communications.* IEEE. 2022, pp. 80–89.

[143] Konstantin Wegner, Sven Seele, Helmut Buhler, Sebastian Misztal, Rainer Herpers, and Jonas Schild. "Comparison of two inventory design concepts in a collaborative virtual reality serious game". In: *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play.* 2017, pp. 323–329.

[144] Johann Wentzel. "Reach-Bounded, Non-Linear Input Amplification for More Comfortable Virtual Reality". MA thesis. University of Waterloo, 2020.

[145] Johann Wentzel, Greg d'Eon, and Daniel Vogel. *Improving virtual reality ergonomics through reach-bounded non-linear input amplification.* 2020.

[146] Daniel Wigdor and Dennis Wixon. *Brave NUI world: designing natural user interfaces for touch and gesture.* Elsevier, 2011.

[147] Jacob O Wobbrock, Meredith Ringel Morris, and Andrew D Wilson. "User-defined gestures for surface computing". In: *Proceedings of the SIGCHI conference on human factors in computing systems.* 2009, pp. 1083–1092.

[148] Bang Wong. "Color blindness". In: *nature methods* 8.6 (2011), p. 441.

[149] Carole-Jean Wu, David Brooks, Kevin Chen, Douglas Chen, Sy Choudhury, Marat Dukhan, Kim Hazelwood, Eldad Isaac, Yangqing Jia, Bill Jia, et al. “Machine learning at facebook: Understanding inference at the edge”. In: *2019 IEEE international symposium on high performance computer architecture (HPCA)*. IEEE. 2019, pp. 331–344.

[150] Huiyue Wu, Weizhou Luo, Neng Pan, Shenghuan Nan, Yanyi Deng, Shengqian Fu, and Liuqingqing Yang. “Understanding freehand gestures: a study of freehand gestural interaction for immersive VR shopping applications”. In: *Human-centric Computing and Information Sciences* 9.1 (2019), pp. 1–26.

[151] Haijun Xia, Sebastian Herscher, Ken Perlin, and Daniel Wigdor. “Spacetime: Enabling fluid individual and collaborative editing in virtual reality”. In: *Proceedings of the 31st annual ACM symposium on user interface software and technology*. 2018, pp. 853–866.

[152] Zezhen Xu, Powen Yao, and Vangelis Lympouridis. “Virtual Control Interface: A System for Exploring AR and IoT Multimodal Interactions Within a Simulated Virtual Environment”. In: *International Conference on Human-Computer Interaction*. Springer. 2021, pp. 345–352.

[153] Jingzhou Yang and Karim Abdel-Malek. “Human reach envelope and zone differentiation for ergonomic design”. In: *Human Factors and Ergonomics in Manufacturing & Service Industries* 19.1 (2009), pp. 15–34.

[154] Jingzhou Yang, Karim Abdel-Malek, and Kyle Nebel. “Reach envelope of a 9-degree-of freedom model of the upper extremity”. In: *International Journal of Robotics and Automation* 20.4 (2005), pp. 240–259.

[155] Tian Yang, Powen Yao, and Michael Zyda. “Flick Typing: A New VR Text Input System Based on Space Gestures”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I*. Springer International Publishing Cham. 2022, pp. 379–392.

[156] Tian Yang, Powen Yao, and Mike Zyda. “Flick Typing: Toward A New XR Text Input System Based on 3D Gestures and Machine Learning”. In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE. 2022, pp. 888–889.

[157] Xiaoying Yang, Xue Wang, Gaofeng Dong, Zihan Yan, Mani Srivastava, Eiji Hayashi, and Yang Zhang. “Headar: Sensing Head Gestures for Confirmation Dialogs on Smartwatches with Wearable Millimeter-Wave Radar”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7.3 (2023), pp. 1–28.

[158] Powen Yao, Yu Hou, Yuan He, Da Cheng, Huanpu Hu, and Michael Zyda. “Toward Using Multi-Modal Machine Learning for User Behavior Prediction in Simulated Smart Home for Extended Reality”. In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE. 2022, pp. 688–689.

[159] Powen Yao, Yu Hou, Yuan He, Da Cheng, Huanpu Hu, and Michael Zyda. “Using Multi-modal Machine Learning for User Behavior Prediction in Simulated Smart Home for Extended Reality”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I.* Springer International Publishing Cham. 2022, pp. 94–112.

[160] Powen Yao, Vangelis Lympouridis, Tian Zhu, and Michael Zyda. “Interfacing with Sensory Options Using a Virtual Equipment System”. In: *Symposium on Spatial User Interaction.* 2020, pp. 1–2.

[161] Powen Yao, Vangelis Lympouridis, Tian Zhu, Michael Zyda, and Ruoxi Jia. “Punch Typing: Alternative Method for Text Entry in Virtual Reality”. In: *Symposium on Spatial User Interaction.* 2020, pp. 1–2.

[162] Powen Yao, Vangelis Lympouridis, and Michael Zyda. “Virtual Equipment System: Expansion to Address Alternate Contexts”. In: *International Conference on Human-Computer Interaction.* Springer. 2021, pp. 353–360.

[163] Powen Yao, Vangelis Lympouridis, and Michael Zyda. “Virtual Equipment System: Face Mask and Voodoo Doll for User Privacy and Self-Expression Options in Virtual Reality”. In: *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2021, pp. 747–748.

[164] Powen Yao, Shitong Shen, and Michael Zyda. “Virtual Equipment System: First Evaluation of Egocentric Virtual Equipment for Sensory Settings”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I.* Springer International Publishing Cham. 2022, pp. 131–149.

[165] Powen Yao, Tian Yang, and Michael Zyda. “Toward a Gesture System Architecture in Extended Reality Based on a Multi-dimensional Taxonomy of Gestures”. In: *International Conference on Human-Computer Interaction.* Springer. 2023, pp. 340–347.

[166] Powen Yao, Zhankai Ye, and Michael Zyda. “Virtual Equipment System: Toward Bag of Holding and Other Extradimensional Storage in Extended Reality”. In: *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I.* Springer International Publishing Cham. 2022, pp. 113–130.

[167] Powen Yao, Tian Zhu, and Michael Zyda. “Adjustable Pointer in Virtual Reality for Ergonomic Interaction”. In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW).* IEEE. 2020, pp. 828–829.

[168] Powen Yao, Tian Zhu, and Michael Zyda. “Designing Virtual Equipment Systems for VR”. In: *International Conference on Human-Computer Interaction*. Springer. 2020, pp. 137–144.

[169] Michael Zeilik and Stephen Gregory. *Introductory astronomy and astrophysics*. 1998.

[170] Daniel Zielasko and Bernhard E Riecke. “Sitting vs. Standing in VR: Towards a Systematic Classification of Challenges and (Dis) Advantages.” In: *VR Workshops*. 2020, pp. 297–298.