Faculteit Wetenschappen

School voor Informatietechnologie

master in de informatica

Masterthesis

Zerotraining: Extending zero gravity objects simulation in virtual reality using robotics as an encountered-type haptic feedback system

Maties Claesen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :

Prof. dr. Kris LUYTEN

2023
2024

De transnationale Universiteit Limburg is een uniek samenwerkingsverband van twee universiteiten in twee landen: de Universiteit Hasselt en Maastricht University.

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

Faculteit Wetenschappen
School voor Informatietechnologie

master in de informatica

Masterthesis

Zerotraining: Extending zero gravity objects simulation in virtual reality using robotics as an encountered-type haptic feedback system

Maties Claesen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :

Prof. dr. Kris LUYTEN

University Hasselt

Masterthesis proposed for the attainment of the degree of Master of Computer Science

ZeroTraining: Extending Zero Gravity Objects Simulation in Virtual Reality using Robotics as an Encountered-Type Haptic Feedback System

Author:

Claesen Maties

Promotor:

Professor Doctor Luyten Kris

Co-promotor: Professor Doctor Ramakers Raf

Supervisor(s): Verstraete Arno

Academic year 2023-2024

Abstract

Preparing astronauts for spacewalks on Earth is challenging due to the presence of gravity. This research, conducted in informal collaboration with the European Space Agency, presents a proof of concept for simulating zero gravity environments using virtual reality (VR). The developed system, named ZeroTraining, includes two key subsystems: ZeroPGT and ZeroArm. ZeroPGT is a VR application that simulates an Extravehicular Activity (EVA) environment, while ZeroArm is an encountered-type haptic feedback system that physically aligns a controller with virtual objects, allowing users to experience the sensation of handling zero gravity items. Several fundamental challenges were addressed, including the development of a custom IK FABRIK algorithm, establishing seamless communication between the VR headset and the robot arm, fabricating the end-effector and implementing hardware improvements to the robot arm. The system's feasibility was validated through a user experience study with 10 participants, revealing significant potential despite current robotic limitations and the need for further enhancements.

Acknowledgements

I would like to express my sincere gratitude to several individuals and teams whose support and expertise were invaluable throughout the course of this research.

First and foremost, I am deeply appreciative of my promoter, Professor Dr. Kris Luyten, and my co-promoter, Professor Dr. Raf Ramakers, for their guidance and unwavering support. My heartfelt thanks extend to my advisor, Researcher Arno Verstraete, whose insights and advice were instrumental in shaping this work.

I am also grateful to Postdoctoral Researcher Jaroslav Hruby for his assistance in establishing initial communications with the European Space Agency (ESA). Special thanks are due to Martial Costantini, Andreas Treuer, and Lionel Ferra at ESA, who provided me with the opportunity to engage with professionals in the field and generously shared their expertise during numerous informal meetings. I would like to particularly acknowledge Andreas Treuer for his ongoing support and valuable feedback throughout the project. His role as my informal contact person with ESA was instrumental in providing continuous guidance and insights following our initial discussions.

My appreciation extends to the Human-Computer Interaction (HCI) research team for their crucial contributions to this project. I am thankful to Postdoctoral Researcher Danny Leen, HCI PhD Researcher Mannu Lambrichts, PhD Researcher Dries Cardinaels, and PhD Student Tom Veuskens for their assistance with the modifications to the robot arm used in this study.

I wish to thank Jeroen Ceyssens for his support during the early stages of setting up the virtual reality (VR) and augmented reality (AR) hardware necessary for this research.

I also extend my gratitude to all the participants who took part in my User Experience Study, as well as those who expressed interest but could not attend due to the high volume of participants.

Finally, I would like to thank my friends and family for their unwavering support throughout this journey.

Each of these individuals and teams played a significant role in the successful completion of this thesis, and I am deeply grateful for their support.

Nederlandse Samenvatting

Deze thesis presenteert ons onderzoek naar het verbeteren van astronautentraining door middel van het simuleren van gewichtloze objecten met virtual reality (VR) en robotica. Dit onderzoek, uitgevoerd in samenwerking met het Europees Ruimteagentschap (ESA), richtte zich op het ontwikkelen van een proof of concept voor een systeem genaamd ZeroTraining, dat twee belangrijke subsystemen integreert: ZeroPGT en ZeroArm.

ZeroPGT is een VR-toepassing die we hebben ontwikkeld om extravehiculaire activiteiten (EVA's) in gewichtloze omstandigheden te simuleren, met speciale aandacht voor gereedschappen zoals de Pistol Grip Tool (PGT) die astronauten gebruiken. ZeroArm, ons encountered-type haptic feedbacksysteem, zorgt ervoor dat een robotarm fysiek wordt uitgelijnd met virtuele objecten in de VR-omgeving, wat een realistische ervaring van het hanteren van voorwerpen in gewichtloosheid mogelijk maakt. Deze subsystemen communiceren actief en zijn essentieel voor het ZeroTraining systeem, waardoor gebruikers gereedschappen kunnen zien, aanraken en voelen alsof ze in de ruimte zijn, wat de realiteitsgetrouwheid van de simulatie aanzienlijk verhoogt.

Tijdens ons onderzoek hebben we verschillende uitdagingen aangepakt, zoals de ontwikkeling van een aangepast inverse kinematica-algoritme gebaseerd op Forward and Backward Reaching Inverse Kinematics (FABRIK), het waarborgen van naadloze communicatie tussen de VR-headset en de robotarm en de optimalisatie van de hardwarecomponenten van de robotarm. We hebben de haalbaarheid van het systeem gevalideerd door een gebruikersonderzoek met 10 deelnemers, wat aantoont dat ons systeem, ondanks de huidige robotica beperkingen, veel potentieel heeft, hoewel verdere verbeteringen noodzakelijk zijn.

Onze belangrijkste doelstellingen waren om te onderzoeken of een robotarm met beperkte bewegingsvrijheid effectief interacties met gewichtloze gereedschappen kan simuleren en om het potentieel van het systeem voor astronautentraining te beoordelen. We hebben ook toekomstige verbeteringen besproken, zoals het versterken van de hardwarecapaciteiten en het verder verfijnen van de integratie tussen VR en robotica.

Samenvattend biedt ons onderzoek waardevolle inzichten in het gebruik van mediated reality voor astronautentraining en legt het de basis voor verdere ontwikkelingen in dit veld. Deze thesis fungeert als proof of concept en technologiedemonstrator, die de haalbaarheid van dergelijke simulaties valideert en richting geeft voor toekomstig onderzoek en ontwikkeling in samenwerking met ESA.

English Summary

This thesis presents our research aimed at enhancing astronaut training by simulating zero gravity objects using virtual reality (VR) and robotics. Conducted in collaboration with the European Space Agency (ESA), we focused on developing a proof of concept for a system called ZeroTraining, which integrates two key subsystems: ZeroPGT and ZeroArm.

ZeroPGT is a VR application we developed to simulate extravehicular activities (EVAs) in a zero gravity environment, specifically focusing on tools like the Pistol Grip Tool (PGT) used by astronauts. ZeroArm, our encountered-type haptic feedback system, physically aligns a robotic arm with virtual objects in the VR environment, providing users with a realistic sensation of handling zero gravity tools. Both of these subsystems, ZeroPGT and ZeroArm, actively communicate with each other and are integral components of the larger system we developed, called ZeroTraining. This combination allows users to see, touch, and feel tools as if they were in space, significantly enhancing the realism of the simulation.

In our research, we addressed several challenges, including the development of a custom inverse kinematics algorithm using Forward and Backward Reaching Inverse Kinematics (FABRIK) as a base, ensuring seamless communication between the VR headset and the robotic arm and fabricating and improving the hardware components of the robot arm. We validated the system's feasibility through a user experience study involving 10 participants, which revealed significant potential for our system despite current robotic limitations and further enhancements needed.

Our primary objectives were to explore whether a limited degree-of-freedom robot arm could effectively simulate interactions with zero gravity tools in space and to assess the system's potential to enhance astronaut training. Additionally, we discussed future avenues for improving the system, such as enhancing hardware capabilities and further refining the integration between VR and robotics.

In conclusion, our research provides valuable insights into the use of mediated reality for astronaut training and lays the groundwork for further advancements in this field. This thesis serves as a proof of concept and perhaps technology demonstrator that validates the feasibility of such simulations and offers potential directions for future research and development in collaboration with ESA.

Contents

Chapter 1

Introduction

1.1 Grounding the Research

When considering the environment for humankind, various factors come into play, influencing human behavior. The varying temperatures, ranging from freezing cold to scorching heat, significantly impact how individuals navigate their surroundings. The darkness or lightness of the surrounding ambiance could also play a role in this behavioral change. However, we rarely envision humans without gravity, given that life as we know it exists on massive planets with gravitational forces. There are adventurers who will brave this unexplored territory, notably astronauts.

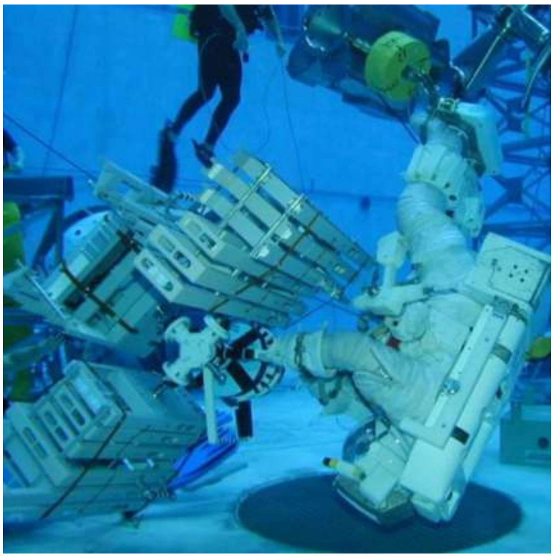
Preparing astronauts for space missions demands thorough dedication and should not be underestimated. Various aspects require training, ideally for longer durations than the actual tasks performed in space. While training predominantly occurs in Neutral Buoyancy Pools (NBP)[Age], these facilities can sometimes be insufficient, costly, or labor-intensive to set up. This is where virtual reality simulations prove invaluable. The European Space Agency (ESA) incorporates these simulations into their training regimen. Microgravity or zero gravity environments can induce or may need significant changes in human behavior. These changes may manifest in altered movement patterns for the astronaut at hand or a differing in handling behavior of objects in space. This thesis will specifically delve into one aspect of training: familiarizing astronauts with the behavior of zero gravity objects during spacewalks aided by a mediated virtual reality.

This thesis investigates a robotic solution aligned with the European Space Agency's (ESA) proposed approach from our initial formal meetings. The primary objective of those meetings was to contextualize the research within current industry challenges. Collaboration with ESA's astronaut training department identified critical issues, such as the"enchanted snake" problem, where tethered tools cause chaotic entanglements in zero gravity environments, impacting their usability. This metaphor illustrates the tools' unpredictable behavior. These findings guided the development of a robotic system for handling zero gravity tools. The researchers at ESA also appointed a person of contact to give further insights and recommendations given their interest in the project.

The system developed, named ZeroTraining, integrates two subsystems: ZeroPGT and ZeroArm, as illustrated in Figure 1.2. These subsystems are designed to work together to create an encountered-type haptic feedback system. An encountered-type haptic feedback system simulates robotic devices interacting with users to physically overlay a virtual object using haptic and force-feedback in VR. The ZeroPGT subsystem—where PGT stands for Pistol Grip Tool, a tool commonly used during spacewalks—operates on a head-mounted display (HMD) such as the Meta Quest 2. It manages all visual aspects, including the simulation of virtual zero gravity tools and the rendering of environments like the International Space Station (ISS). The

(a) Neutral Buoyancy Facility at EAC in Cologne, Germany[Age].

(b) The pre-familiarisation uses a makeshift EVA suit[Age].



(c) Fuglesang during EVA training in the Neutral Buoyancy Laboratory[Age].

Figure 1.1: A Neutral Buoyancy Pool used by ESA at EAC to train astronauts.

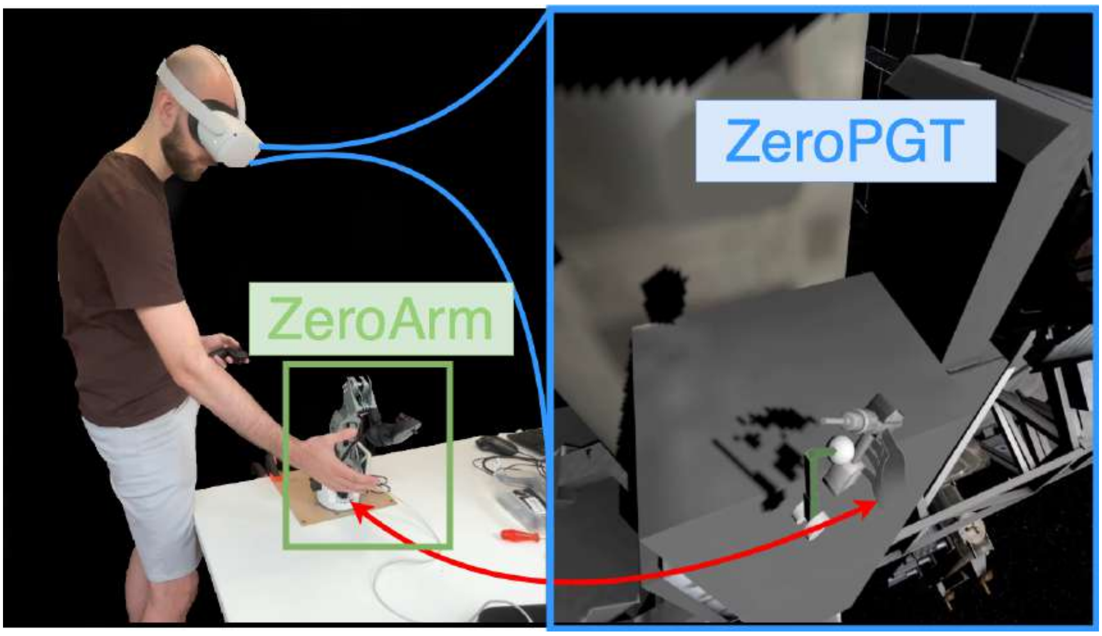


Figure 1.2: The ZeroTraining systems showing ZeroArm at the left side and ZeroPGT at the right side. ZeroPGT shows the view when wearing the VR headset. ZeroArm is the robot arm that holds a physical controller. Note that wired connection betweem both systems.

ZeroArm subsystem provides haptic feedback by positioning a physical object, akin to the virtual tool in ZeroPGT, at the location of the virtual tool. As users interact with the virtual tool, they experience haptic feedback from the physical object relocated by the robotic manipulator, allowing them to see, touch, and feel the tool's movement in 3D space.

Moreover, both the astronauts' extravehicular activities (EVAs) and the tools they use are tethered either to the spacecraft's hull or directly to the astronauts themselves. This precaution is rooted in Newton's third law, which states that for every action, there is an equal and opposite reaction. In the vacuum of space, where there is no air resistance or gravitational pull, any force applied to an object or tool will cause it to continue along a linear path indefinitely. Without tethering, objects could drift away from the space station, becoming unreachable and lost. Therefore, tethering ensures that tools and equipment remain within reach and that astronauts stay connected to the spacecraft, preventing potentially hazardous situations. Studying the interaction between a zero gravity tool and its tether is crucial, as it adds another layer of complexity beyond just understanding the tool's behavior.

1.2 Potential Future Avenues

Another potential application for mediated reality technology extends into consumer electronics, exploring both near-future and long-term possibilities. As the virtual reality (VR) sector continues to grow and advancements in augmented reality (AR) and extended reality (XR) technologies progress, attention is shifting to the next frontier. For widespread adoption, two key elements are crucial: high-quality XR hardware capable of supporting practical user applications and the development of these applications. For instance, while Apple's Apple Vision Pro [Appa] boasts advanced hardware capabilities, it currently lacks basic applications like YouTube, leading third-party developers to create alternative versions using available APIs. In contrast, Meta's Meta Quest [McM24][PCM23] has achieved greater success but remains primarily focused on gaming.

Drawing a parallel to the mature market of traditional smartphones like the iPhone, which has seen extensive development in both hardware and software, the VR industry is poised for similar growth. The evolution of this sector will likely involve additional hardware to enhance the user experience, similar to accessoiries seen in the smartphone market such as phone cases, smartwatches, laptops, and tablets, all integrated into a cohesive ecosystem. The VR industry, still emerging, has the potential to follow this trajectory as software and applications become more sophisticated. Mediated reality devices—including haptic suits, gloves, full-body tracking systems, omnidirectional treadmills, and robotics—are already part of the evolving VR landscape. However, the use of robotic arms interfacing with VR environments remains relatively rare in the consumer market.

The pricing spectrum for robotics is broad. On consumer platforms such as Amazon, robotic arms can be found for under $100, while high-end industrial robots, like CNC machines, offer exceptional precision for repetitive tasks. Additionally, human-robot interaction (HRI) is a growing field focusing on collaborative robots (cobots) designed to work alongside humans. This thesis presents a proof of concept using a robotic arm with constrained capabilities, including limited degrees of freedom, restricted ranges of motion, and suboptimal singularities. Despite these limitations, if the core functionalities of this approach prove effective, it could evolve with more advanced hardware. Conversely, if the approach shows limitations, it will help identify areas for improvement in the hardware, potentially making it a viable candidate in the mediated reality space. Using a simplified robotic arm in the initial stages is crucial for identifying deficiencies and bottlenecks. This research aligns with ESA's request to develop a proof of concept for a mediated reality robotic arm. As ESA progresses towards a comprehensive simulation, this thesis will provide valuable insights and groundwork for advancing their project.

In this context, even a minor improvement in physical realism is significant, as experiencing something physically present, albeit imperfectly, is preferable to having no physical feedback while using a VR headset. This is in contrast to the excellent pure virtual interactions provided by the latest VR headsets, which can simulate touches, collision responses, and grab interactions through gestures. Although these interactions lack the tactile sensation of a real object, they are still quite convincing to users.

As a side note, the European Space Agency (ESA) is developing an XR lab that features a simulation concept similar to the one explored in this thesis. They are considering the acquisition of a robot arm and have developed several algorithms for their simulation. Although their goal differs slightly—from enabling users to interact with virtual surfaces by moving a physical surface beneath their hand using a robot arm—their approach shares similarities with the thesis subject. In their system, the robot arm aligns with the surface normal to provide physical feedback when users push against a virtual surface, which simultaneously alters the virtual environment and moves the physical surface away. While this aspect is not the primary focus of the thesis, these developments are acknowledged and considered in the implementation of a simplified problem within this research.

1.3 The Evolution of Astronaut Training

This section provides an introduction to the framework that will underpin our research. The primary challenge with training an astronaut in this scenario is the absence of a zero gravity environment. While NASA has developed several more mature options such as Neutral Buoyancy Pools, this thesis also discusses other training simulations, including the use of virtual reality, particularly those still in the early stages of research. The main emphasis of this alternative approach is to introduce what is known as mediated reality. While mediated reality often involves extensive haptic feedback, it encompasses more than just this aspect. It may also incorporate elements such as the sense of smell, temperature, and other properties not provided by the mixed reality experience.

1.3.1 Immersion in Neutral Buoyancy Pools

One of the best simulations of zero gravity on Earth involves creating an environment where all forces acting on the body are neutralized. The challenge then lies in counteracting the force of gravity. One method is utilizing a surface beneath us to provide a force opposing gravity. However, this confines movement to a 2.5-dimensional space projected onto the surface. Another approach is submerging someone in a liquid where the body can float. Yet, if the buoyant force is too strong, the body rises to the surface, limiting movement to a 2.5D space relative to the liquid surface. The solution is to add weights to achieve neutral buoyancy, creating a fully three-dimensional space for movement, a principle well-established in scuba diving[Kni19]. However, challenges arise when applying this principle to simulate EVAs[Age]. Inside the spacesuit, the body isn't submerged, so gravity is still felt, leading to pressure points not experienced in zero gravity. Additionally, the motion of the body is not frictionless due to the liquid surrounding the spacesuit.

With 32 hours of EVA time in space, Swedish astronaut Christer Fuglesang[Age], as seen in Figure 1.1c, stands as the most experienced spacewalker in the European Astronaut Centre (EAC) as of the article's publication. Fuglesang sheds light on the extensive training regimen required for each spacewalk, stating, "For each specific spacewalk, there are several training units to be completed. One EVA run lasts around 5 hours, and the standard right now is that you spend five to seven times as long in the Neutral Buoyancy Lab (NBL) at Houston for each EVA, depending on the difficulty. In addition to that, you train a lot of contingency scenarios." Although we could delve deeply into this subject, we will refrain from doing so, as numerous trainings and simulations conducted in the Neutral Buoyancy Lab fall outside the scope of this thesis.

1.3.2 The Rise of Virtual Reality Training

Once the resources of Neutral Buoyancy Labs are exhausted, an alternative method for astronaut training becomes necessary. One approach involves creating a virtual world that replicates the conditions of zero gravity, utilizing tools like virtual reality headsets. However, a significant challenge arises: how can we accurately simulate the movement of objects and the human body as they would behave in a microgravity environment?

This first problem of creating the virtual world is adressed by NASA with a fancy system called Dynamic Onboard Ubiquitous Graphics (DOUG) [NASd]. The Virtual Reality Training Laboratory at Johnson Space Center has developed DOUG, a 3D viewing tool and graphic engine, which generates mission-specific scene configuration databases used in major simulations across NASA centers. These simulations encompass various activities. The activities mentioned encompass a wide range of training and simulation tasks crucial for space exploration. Simplified Aid for EVA Rescue (SAFER)[NASa] provides astronauts with a propulsion unit for emergency maneuvering during spacewalks. The Space Station Remote Manipulator System (SSRMS)[CM18], assists in various tasks from within the International Space Station (ISS) by supporting EVA activities using a robotic arm. Systems Engineering Simulation (SES)[Gar23b] ensures the functionality and integrity of spacecraft systems through comprehensive testing. For example a real-time, crew-in-the-loop engineering simulator for the International Space Station (ISS), Orion, lunar landing systems, lunar surface mobility, and other advanced concepts. Dynamic Skills Trainer (DST)[Gar23a] offers astronauts realistic simulations to hone their operational skills for space missions such as docking, ISS visiting vehicle capture or robotic arm operations. The Space Station Training Facility (SSTF)[Mar23] provides a hands-on environment for astronauts to familiarize themselves with ISS modules and equipment so they get used to operating these systems. Within the Virtual Reality Laboratory (VR Lab)[NASd], astronauts engage in immersive training and research activities using virtual reality technology. Finally, the Engineering DOUG Graphics for Exploration (EDGE)[NASb] tool aids engineers and scientists in visualizing and analyzing data for space exploration missions, contributing to the advancement of space exploration efforts.

Here's a detailed account of the NASA JSC Virtual Reality Lab (VRL) history[Pad+]. In this context, "JSC" refers to the "Johnson Space Center," a pivotal NASA facility in Houston, Texas, dedicated to astronaut training and spacecraft development. The Virtual Reality Lab was established in 1991 by David Homan, who devised a VR system suspended from the ceiling for zero gravity EVA training. In 1994, SAFER was introduced, followed by the development of DOUG in 2001. Notably, upgrades were made to the Charlotte[Swa+95] system, alongside the creation and flight testing of a novel intravehicular activity (IVA) robot aimed at automating routine tasks for crew members. This robot, tethered in six locations to create a 6DOF motion platform, could simulate surface movements and some controls in 3D space. By 2016, the lab adopted consumer-ready head-mounted displays (HMDs) in conjunction with game engines such as Unity and Unreal Engine 4, aligning with similar practices observed within the XR team at the European Space Agency (ESA), which also utilizes the Unreal Engine. The Active Response Gravity Offload System (ARGOS) represents the last system discussed here, designed to emulate planetary exploration by suspending the user from an overhead crane, providing realistic sensations of movement and reduced gravity. Another consideration is the increased walking involved in this simulation, requiring the provision of a moving surface for users to walk on. Additionally, the ARGOS facility includes various systems for mounting users, such as those designed to simulate interactions with for example handlebars.

1.4 Objectives and Rationale

The main objective of this thesis is to develop an encountered-type haptic feedback system by integrating a VR application with a robotic manipulator and facilitating robotic graphics. The first part of the thesis involves building a comprehensive background and exploring related work on this topic. Once this foundational knowledge is established, the thesis will describe the final application, "ZeroTraining," which is an overarching system comprising two subsystems: "ZeroPGT" and "ZeroArm." ZeroPGT focuses on the visual components and operates within a VR environment. This includes hand tracking, head tracking, controller input, and all visual aspects, typically facilitated through a Head-Mounted Display (HMD). ZeroArm handles the physical components, providing encountered-type haptic feedback using a robotic manipulator. This subsystem receives commands from ZeroPGT, translates these commands for the robot, and coordinates the robot arm. It also manages edge cases, such as singularities and low-level human safety considerations. We will detail the design of the entire system and illustrate its practical implementation. Following a comprehensive analysis, we will present the results of the User Experience Study conducted. We will describe the study setup and objectives. Since most participants are expected to lack experience with zero gravity environments and the specialized skills required, our study emphasizes their subjective experience of realism. This feedback will be used to evaluate the effectiveness and identify issues with our system in comparison to a baseline pure virtual version. Although this study is not intended as a pure comparison study, a baseline is included to account for the fact that most participants will need to become familiar with the primitive VR experience.

The main structure of this thesis will be composed as if it is a feasibility study considering many of these ideas and the technology surrounding it is inherently, innovative, and have little research. "How can a limited 4DOF robot arm be utilized to simulate realistic interactions with zero gravity tools in space, aiming to enhance the effectiveness of a VR simulation for astronaut training?". The implementation itself serves as a technology demonstrator, meaning that specific quantitative results are not the primary focus neither will we be proving or disproving this statement. The initial inquiry addresses whether such an implementation is feasible within the current timeframe and with the available tools and resources. Consequently, this thesis will elaborate on the potential implementation methods considered, the final method selected, the rationale for this choice, and the reasons for abandoning other methods. Despite its exploratory nature, this research is extremely valuable as it expands our understanding of the significant challenges in this domain. Numerous topics will be proposed for further research to explore potential solutions in greater depth. Additionally, as this is a proof of concept developed in collaboration with the European Space Agency and their forthcoming projects, many lessons learned could inform their future and current research endeavors.

To delve a bit deeper, this thesis will focus on a multitude of questions. In this paragraph we give a broader view on what types of topics we will be discussing using a set of questions. “Is it possible to imitate a zero gravity object on earth using a limited robotic manipulator?”, “Is an encountered-type haptic feedback using a robotic manipulator viable in terms of making a realistic simulation of this kind?”, “What avenues were considered not worth pursuing for future research?”, “What range or type of robot arm is required for this complex task?”, “Do users prefer a mature, purely virtual approach, or do they favor a constrained set of VR and haptics using a robotic manipulator for the current implementation?”, “What unforeseen problems arise when trying to create a realistic object simulator?”, “What software and hardware constraints most hinder the creation of an optimal solution?”, “How generic is the robotic manipulator solution, and what other problems can it address?”.

To conclude this section, we present a scenario illustrated in the comprehensive storyboard shown in Figure 1.3, where the ZeroTraining application demonstrates its effectiveness. Imagine an individual preparing for space travel, aiming to improve their proficiency in handling zero gravity objects, such as the Pistol Grip Tool (PGT). Simulating zero gravity on Earth, where gravity is ever-present, poses a considerable challenge. The ZeroTraining solution tackles this issue by integrating the ZeroPGT and ZeroArm subsystems. Together, these subsystems create an encountered-type haptic feedback system. As the virtual object in the virtual reality environment moves in a specific direction, the precisely aligned physical object, controlled by the robot arm, mirrors this movement at the same velocity. When a user interacts with the virtual tool, they receive direct tactile feedback, allowing them to touch, feel, or grasp the tool as if it were actually in zero gravity, thanks to the robot arm’s positioning of the aligned physical object. This interaction makes the behavior of zero gravity objects more intuitive, and with repeated training, enhances an astronaut’s competency in handling such objects in space.

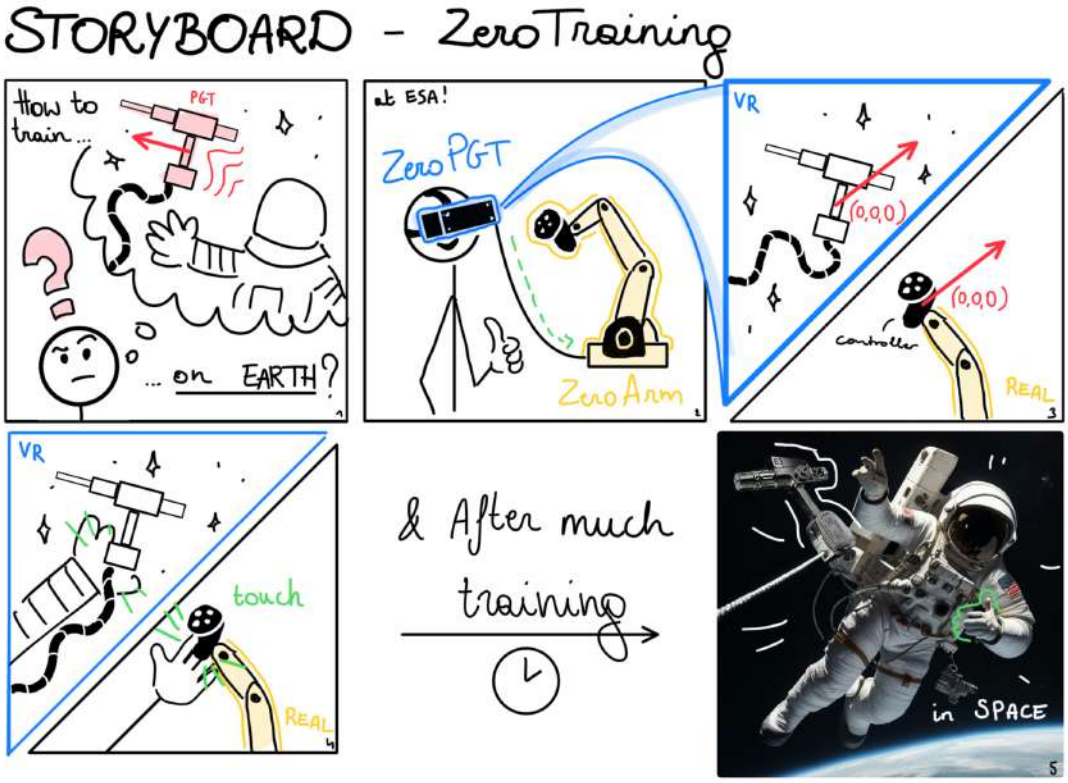


Figure 1.3: A storyboard illustrating the scenario where the ZeroTraining system is utilized. Note that the virtual tool and the physical robot arm are precisely aligned at the same location. The final frame demonstrates how the astronaut's experience with ZeroTraining enhanced their competence in space.

Chapter 2

Background and Related Work

This chapter provides a comprehensive overview of the foundational concepts and current advancements relevant to our research. We will explore the theoretical and practical aspects of key technologies and methodologies that underpin our study. We will start with a detailed comparison of virtual reality (VR) and augmented reality (AR). The subsequent section, 2.2, will explore different types of robotics and their current applications. Following this, section 2.3 will provide an in-depth examination of mediated reality, a key concept for our discussion. Finally, we will turn our focus to the research and best practices landscape in astronaut training, thereafter examining some specific VR/AR simulation methodologies.

2.1 Augmented Reality vs Virtual Reality

This literature study begins by comparing the use cases and differences between augmented reality (AR) and virtual reality (VR). This comparison is particularly relevant as both technologies were extensively utilized during the implementation of this thesis. Subsection 3.2.1 will delve into the specifics of these implementations, examining both the software and hardware aspects. We will evaluate their advantages, limitations, innovations, and shortcomings. While we will recap the fundamentals of these technologies, this section is not intended as an introductory course but rather as an analysis of the current state of the art.

2.1.1 Exploring Augmented Reality

The use of Augmented Reality (AR) for training offers several distinct advantages. It enhances real-world environments with digital overlays, thereby enriching user interactions without the need for complete immersion in a virtual world. This approach facilitates the integration of interactive, contextual information directly within the user's view, which can enhance learning and operational efficiency. AR supports the overlay of digital instructions, simulations, or data onto physical objects, thereby facilitating hands-on learning and immediate feedback. However, challenges remain in achieving seamless integration between digital elements and the physical world. Accurate spatial alignment and real-time tracking are essential to ensure that digital overlays align correctly with physical objects and movements. Additionally, achieving intuitive user interfaces and minimizing distractions are crucial for maintaining the effectiveness of AR-based training.

State of the Art Review

Some advantages will be explored of AR by examining the latest advancements in the field. Numerous sectors stand to gain from the integration of Augmented Reality (AR) technology, notably in gaming, educational pedagogies, medical practices, vocational training, and telecommunications infrastructure. In gaming, AR facilitates immersive experiences by integrating virtual entities within the user's physical milieu, thus diminishing the demarcation between virtuality and reality usually accompanied with other technologies like geolocation or haptic integration such as seen in Father.io[Fat] that creates an AR lasertag game with your smartphone. This results in a contraction of development timelines since it involves creating less of the virtual world compared to VR. Educationally, ZeusAR[Mar+21] serves as a tool for creating augmented reality serious games, superimposing digital content onto tangible environments to enhance experiential learning, thereby increasing learner engagement and facilitating knowledge assimilation.

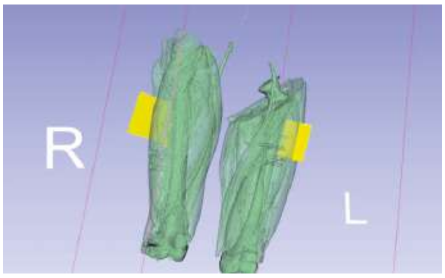
In the healthcare continuum, augmented reality (AR) has emerged as a transformative innovation with the potential to redefine medical education and patient management. For example, peer-learning applications of AR have enhanced neuroanatomy analysis for students, as demonstrated in research by O. V. Ravna et al.[Rav+22]. Additionally, O. George et al.[Geo+23] highlighted the benefits of AR in providing equitable learning opportunities through online dissections, reducing overcrowding in traditional settings. AR also shows promise in patient management; A. Charalambous et al.[CI20] explored its use in treating phantom limb syndrome, where AR glasses simulate the movement of a lost limb, facilitating therapeutic outcomes by allowing patients to control the digital representation of their limb through brain signals.

Surgical interventions are also already being tested using AR. It accords clinicians with instantaneous access to patient-specific data and imaging, alongside navigational aids during procedures, directly within their visual field, thereby augmenting clinical decision-making such as seen in the study by V.G. Grebenkov et al.[Aga+23]. Industrially, AR is instrumental in refining assembly operations, provisioning workers with enhanced visual directives, and amplifying productivity and operational efficacy allowing them to evolve to Industry 4.0[EB22].

Hardware & Software Support

Modern smartphones are equipped with essential hardware for augmented reality (AR), such as cameras and sensors. These components allow for the capture and display of the real world, with the added capability to modify it in real-time. Specifically, smartphones utilize sensors to determine their position in three degrees of freedom (3DOF). Advanced algorithms like Simultaneous Localization and Mapping (SLAM) enhance this by providing six degrees of freedom (6DOF), offering a more comprehensive understanding of spatial orientation. Simultaneous Localization and Mapping (SLAM)[con24d] is a fundamental technology in augmented reality (AR). It allows devices to understand the physical world by creating a map of their surroundings and accurately determining their position within that map. While the current smartphone technology is impressive, the experience is somewhat restricted due to the limited field of view, akin to peering through a small window. Nevertheless, incorporating this hardware for AR spectators to observe mixed reality (MR) users could enrich the experience. Despite MR devices typically requiring specialized equipment, this concept, termed 'Guilty Bystanders,' has been actualized by M. Cohen et al.[CM22] as seen in Figure 2.1.

To advance toward more specialized hardware, two distinct approaches have emerged. The first involves augmenting regular eyeglasses to create a dedicated augmented reality (AR) device. This lightweight option relies on external components such as batteries and computational power, while incorporating essential technologies like tracking and overlaying directly into the glasses. The second approach focuses on an integrated design, where all necessary components—batteries and compute power—are seamlessly built into a dedicated AR device. However, a significant challenge in AR technology lies in achieving effective overlaying. One straightforward solution involves using a prism mechanism to project digital content as a hologram while preserving the user's natural vision. Analogous to a telescope that employs mirrors, this approach faces a limitation: mirrors are not transparent, resulting in the digital screen being visible while potentially obscuring interesting visual content behind the screen. An alternative mechanism, based on waveguide technology, has recently gained prominence. Waveguide-based



(a) 3D models of the lower extremities simulating limb bony structures, muscles, vessels, nerves, and IIT of both thighs. The proposed planes of surgical access are shown in yellow[CM22].

(b) Fixing the marker to the installed magnets[CM22].

Figure 2.1: The setup for the AR surgical bystander research.

[Jak23] AR displays guide light through waveguides rather than prisms, offering a more compact and comfortable solution, thanks to economies of scale.

Overlaying technologies pose a significant challenge. Specifically, the eyebox[Cho+20] delineates the spatial region within which the human eye can perceive an image satisfactorily, adhering to precise criteria and thresholds. Decisions related to optical architecture play a pivotal role in shaping the system's volumetric properties. However, existing technologies fall short of achieving a sufficiently large field of view (FOV), thereby diminishing the practicality of augmented reality (AR) solutions. Consequently, most commercial AR headsets prioritize mixed reality (MR) experiences that incorporate elements of virtual reality (VR). Implementing this solution necessitates additional considerations. For example, capturing and displaying the real-world environment on screens introduces inherent limitations, including reduced color accuracy, lower resolution, and diminished brightness compared to natural vision.

Let's take a look at our current progress. The theoretical ideal display covers approximately 92%[Nie+22] of the visible spectrum. Meanwhile, the Apple Vision Pro achieves a similar coverage of the DCI-P3 color gamut[Pix24], which itself represents only about 53.2% [Wri19] of the colors discernible by the human eye. Our eyes rely on a specialized region called the fovea for perceiving sharpness. Determining the resolution discernible by humans is a complex task. Researchers, such as J. Hirsch et al., have investigated Nyquist Limits[HC89], although this topic lies somewhat beyond the scope of our current research. Nevertheless, it is essential to acknowledge their significance. In the consumer technology realm, seamlessness is highly valued. Apple introduced the term Retina display[con24c], which aligns with the pixel structure of the screen being imperceptible under normal usage conditions. Notably, the Apple Vision Pro display does not strictly adhere to this guideline and is not marketed as a true Retina display. Next, we address the concept of display brightness, which is commonly measured in units called nits [Woj23]. Nits are quantified as candela per square meter (cd/m^2). It is important to note that display brightness does not directly match the natural brightness of the real world. Achieving a level of brightness equivalent to that of the sun would require displays to emulate significantly higher nits than current consumer technology, which typically peaks at around 3000 nits.

Computer vision serves as the foundational framework for Augmented Reality (AR) applications, facilitating real-time environmental interaction. AR experiences are driven by three pivotal components. Firstly, feature detection and tracking are imperative, relying on methodologies such as feature matching[MNL99][Sar+20][Sun+21], keypoint extraction[Zha+22][Zha+22], and optical flow[BB95][FBK15] to discern objects, surfaces, and spatial relationships. The stability of overlays and the precision of virtual content alignment hinge upon robust feature tracking. Secondly, pose estimation assumes a critical role in accurately determining the position and orientation (pose) of the user's device relative to the environment. Pose estimation algorithms, notably Perspective-n-Point (PnP), precisely estimate the camera's position based on detected features, ensuring a seamless AR experience.

The development landscape for AR devices has been streamlined, particularly for mobile platforms, utilizing frameworks such as ARCore[Gooa] and ARKit[Appb]. These frameworks facilitate the deployment of AR applications and functionalities. In contrast, the development for specialized AR devices necessitates the use of robust game engines like Unity and Unreal Engine. Despite their capabilities, these engines do not intrinsically furnish interfaces for direct communication with the AR hardware. This gap is bridged by proprietary frameworks provided by AR device manufacturers, such as the XReal glasses[XRE], Microsoft HoloLens[Mic], and Magic Leap 2[Lea], which are tailored for integration with these game engines. While these proprietary solutions enable developers to access a range of sophisticated and sometimes exclusive features of the AR devices, the creation of a universally compatible application is contingent upon the adoption of a cross-device standard like OpenXR[Gro16].

2.1.2 Diving into Virtual Reality

The use of Virtual Reality (VR) for training offers several significant advantages. It eliminates the need for complex and expensive equipment setups, allowing users to immerse themselves in high-risk environments and gain valuable experiential learning. By incorporating digital twins, VR facilitates the development of muscle memory, enabling users to interact effectively with virtual objects. However, achieving high levels of haptic fidelity remains a key challenge. Accurately replicating attributes such as weight, texture, and temperature requires specialized VR peripherals. Additionally, ensuring precise alignment between physical and virtual environments is crucial, as any discrepancies can compromise the effectiveness of VR training.

State of the Art Review

In the context of virtual reality (VR) applications, gaming emerges as a prominent domain. As a widely adopted use case within mixed reality (MR), gaming captivates users due to its inherent immersiveness. Players can inhabit roles from beloved franchises or transcend real-world limitations. Noteworthy examples include superhero-themed games such as Megaton Rainfall[1], the lightsaber combat experience in Vader Immortal[2], and Half-Life: Alyx[3], a VR iteration of the iconic Half-Life series. These games excel in visual fidelity, physics simulations, and emotional resonance. However, our exploration extends beyond gaming alone.

The virtual reality (VR) technology has undergone another significant evolution, particularly in the realm of training methodologies. This advancement has facilitated the creation of immersive simulations, enabling professionals, including pilots, firefighters, and military personnel, to engage in high-fidelity scenario rehearsals devoid of real-world risks. For instance, prospective aviators[Pur23] can refine their aeronautical prowess by navigating intricate flight scenarios within a meticulously controlled virtual milieu. A virtual reality (VR) rendition of an Airbus A380 cockpit was meticulously crafted as seen in Figure 2.2a, incorporating virtual meshes programmed to simulate realistic haptic responses. The efficacy of this approach was evaluated by comparing the response times of two critical elements in the post-takeoff procedure. The analysis revealed a discrepancy of only 0.28 and 0.49 seconds, respectively, between a physical cockpit mock-up and the virtual design. These findings indicate that the VR model is sufficiently accurate and reliable for preliminary testing, underscoring its potential as a viable and scalable alternative to conventional simulator-based training. Similarly, firefighters[Ham+22], as seen in Figure 2.3, are empowered to emulate emergency responses encompassing exigent circumstances such as structural conflagrations, chemical hazards, and search and rescue operations. The availability of firefighter training remains constrained, primarily relying on real-world simulations fraught with inherent safety risks. Recognizing these limitations, a novel approach

[1]https://store.steampowered.com/app/430210/Megaton_Rainfall/
[2]https://www.oculus.com/vader-immortal/
[3]https://store.steampowered.com/app/546560/HalfLife_Alyx/

(a) Airbus VPT Cockpit Reconstruction[Pur23]. **(b)** Configuration of Quest Controllers on SenseGlove[Pur23].

Figure 2.2: Figures from the Enhancing Aviation Safety through Haptic Glove-Enabled Virtual Reality research.

Figure 2.3: The immersive virtual reality (IVR) used in comparison to the hot fire-live simulations (HF-LS) for the Firefighter Skilss Training research[Ham+22].

integrating virtual reality (VR) technology with artificial intelligence (AI) was devised. This VR system offers tactile feedback and enables trainees to familiarize themselves with personal protective equipment (PPE) and operational tools like hoses and nozzles, thereby enhancing preparedness for real-world firefighting scenarios. The group with limited prior experience in hot fire-live simulations (HF-LS) found the task to be more comparable to HF-LS, whereas the group with less actual fire exposure perceived the immersive virtual reality (IVR) stress level to be more reflective of real fire scenarios. Similarly, military personnel can refine tactical maneuvers and combat strategies safely using VR technology. This integration of VR enhances experiential learning, improves decision-making skills, and promotes safety awareness. Therefore, three methods[Har+23] of training for use-of-force decision-making were established to assess the situational awareness and judgmental skills of defense and security personnel. While 2D video training posed minimal challenges, live-fire and VR simulations yielded comparable results, albeit with slight variations.

The introduction of virtual reality (VR) technology has significantly transformed architectural visualization and design methodologies. Architects and designers can now create immersive virtual walkthroughs of structures before they are physically built[Sol+23], providing clients with an unprecedented ability to interact with and manipulate spatial environments, surpassing the limitations of traditional blueprints and 3D models[KK17]. Through VR, clients can explore each room, experiment with various lighting setups, assess material choices, and iteratively refine spatial layouts. This heightened interactivity not only enhances communication between architects and clients but also fosters a deeper understanding of architectural concepts and spa-

Figure 2.4: 3D city model. The developed model can be explored with VR hardware from an immersive egocentric perspective for the Urban Planning Scenarios research[Kei+23].

tial dynamics. Moreover, VR extends beyond individual buildings, emerging as a powerful tool for urban planning and landscape architecture[Kei+23] as seen in Figure 2.4. Urban planners can visualize proposed changes to urban landscapes or natural terrains, allowing stakeholders to assess the impact of development projects within a realistic context. By simulating the integration of new structures with existing environments, VR supports informed decision-making and promotes sustainable urban development practices. Ultimately, VR-driven architectural visualization accelerates the design process and improves the quality of built environments, ensuring they meet the needs and aspirations of current and future communities. In contrast, augmented reality (AR) is more applicable in scenarios where the physical environment already exists. For example, applications like the IKEA app[IKE17] allow users to overlay virtual furniture onto real-world settings, while tools like magicplan[mag] assist in creating floorplans.

Virtual reality (VR) has revolutionized the way people experience tourism and engage with cultural heritage sites around the world. Through immersive VR experiences, users can be transported to iconic landmarks, historical sites, and natural wonders without leaving the comfort of their homes[Gut10]. Imagine standing atop the Great Wall of China or strolling through the halls of the Louvre Museum, all through the convenience of a VR headset. These virtual tours offer not only a glimpse into distant places but also provide an opportunity for individuals with physical limitations to explore destinations they may otherwise never visit. However, some research highlights skepticism about the technology, particularly regarding its perceived effectiveness, despite a general willingness to try it[Pol+23]. Additionally, there is ongoing research into developing VR applications that can be used on-site at tourist locations, enhancing the visitor experience in real-time. Moreover, VR plays a crucial role in cultural heritage preservation by leveraging 3D scanning technology to digitally capture and reconstruct ancient artifacts and archaeological sites[Sel+20]. By creating virtual replicas of these cultural treasures, VR ensures their conservation and accessibility for future generations, even in the face of natural deterioration or human conflict. Beyond mere recreation, VR experiences in tourism and cultural heritage preservation foster a deeper appreciation for the world's diverse history and natural beauty, inspiring curiosity, empathy, and a sense of stewardship toward our shared global heritage.

Many applications require capturing real-world objects and digitizing them for various purposes. Significant research efforts have been devoted to this area, with recent developments focusing on techniques such as Pointformer[Pan+21], which uses a transformer architecture to enhance point cloud coherence and feature extraction. Additionally, advancements in full room/environment capture involve generating 3D models from sequential still frames supplemented with 3D spatial information. Neural Radiance Fields (NeRF)[Gao+23] have been prominent in this field, providing a method for synthesizing novel views of complex scenes by modeling the volumetric scene function using neural networks. In contrast, Gaussian Splatting[Ker+23] as seen in Figure 2.5, has emerged as a competitor, offering higher quality and greater flexibility through the utilization of 3D Gaussian representations.

(a) A rasterized image of a Guassian Splatting respresentation where every gaussian of the bicycle is rasterized as fully opaque[HS23].

(b) The normal Guassian Splatting method where the bicycle image is rasterized with non opaque Guassians[HS23].

Figure 2.5: A representation of Guassian Splatting as seen on the Github Page from Tyler Sehon[HS23].

Hardware & Software Support

This subsection delves into the fundamental hardware and software components that form the backbone of virtual reality (VR), influencing its evolution and widespread acceptance. Tracing its origins to its inception, VR has fascinated users by offering immersive experiences capable of transporting them into simulated environments. This exploration aims to elucidate the pivotal role played by hardware and software in shaping the trajectory of VR technology, thereby revolutionizing the paradigm of human-computer interaction.

We could start by looking at an early development of VR, such ad Google Cardboard[Goob], which was an introduction to VR for many users. Smartphones were repurposed into virtual reality (VR) devices through the integration of peripherals like lenses and head-mounted containers. This configuration splits the smartphone display into two screens, facilitating the presentation of stereoscopic images to each eye. While the device's internal inertial measurement unit (IMU) sensors enable basic three-degree-of-freedom tracking, achieving six degrees of freedom may necessitate the use of external cameras[BRH18], albeit not commonly implemented. This research, later manifested in technologies like WorldSense[Gooc], serves as the foundation for standalone VR headsets, which will be discussed shortly. These headsets utilize internal cameras to achieve inside-out tracking, enabling a remarkable level of precision in positional tracking. Nonetheless, notable drawbacks to smartphone VR include the insufficient display resolution to mitigate the screen door effect (SDE) and limited support for external controllers and hand tracking in most applications. Some of these problems were addressed; for example, much research to this day still focuses on the reduction of SDE[Ngu+20] by physically shaking the display to visually reduce the gaps between pixels. Additionally, commercial solutions like Gear VR[Sam21] have been developed, which entail dedicated smartphone containers and additional VR controllers.

Virtual reality (VR) headsets, crucial tools in immersive technology, can be classified into two main categories: standalone and tethered devices. Standalone VR headsets, representing one branch, encompass integral components such as the processor unit, battery, and tracking systems within the device itself. Notably, these headsets commonly incorporate inside-out tracking mechanisms, leveraging onboard cameras to ascertain their position within a three-dimensional (3D) space. This autonomous setup renders standalone headsets highly convenient, offering users freedom of movement without tethering constraints. Examples of these types of headsets include Meta Quest 2[McM24], Meta Quest 3[PCM23], Pico 4[PCg23], etc. The quality of these devices in terms of hardware has increased dramatically in the recent years. The software has steadily grown but has not reached its full potential considering its limited adoption compared to smartphones.

Conversely, the other branch comprises tethered VR headsets, distinguished by their tethering to external computing devices, typically high-performance personal computers (PCs) equipped with advanced graphics cards. This tethering enables tethered headsets to undertake graphically intensive tasks, including rendering high-quality visual content and executing compute-intensive processes such as eye tracking and hand gesture recognition. Unlike standalone counterparts, tethered VR headsets utilize outside-in tracking methodologies, where external cameras positioned in the environment track markers affixed to the headset, facilitating precise spatial localization using triangulisation. Some examples of such tethered devices are Varjo XR-4[Car23], HTC VIVE Pro 2[PCM21a], Valve Index[PCM21b].

Despite their computational prowess, tethered VR headsets are subject to certain limitations, notably in terms of mobility and setup complexity. Tethering to an external PC restricts users' freedom of movement, necessitating a designated physical space conducive to VR experiences. Moreover, the setup process involves configuring the external tracking cameras and ensuring optimal coverage of the VR play area, which may deter users seeking seamless, plug-and-play experiences. Consequently, while tethered VR headsets excel in performance and feature-rich capabilities, their usability may be compromised by these inherent constraints.

Additionally, it's plausible that VR developers may utilize in-house development kits to facilitate the design, implementation, and testing of VR games, catering to specific project requirements. Nonetheless, mainstream 3D game engines like Unity Engine and Unreal Engine play a pivotal role in democratizing VR development by providing robust support for virtual reality features. Their widespread adoption underscores their significance in advancing VR technology and accessibility to developers and enthusiasts alike. For the development of Half-Life: Alyx, Valve used a suite of tools known as the Half-Life: Alyx Workshop Tools. These tools are part of the Source 2 engine, which is Valve's proprietary game engine. Support for development of smartphone-based VR experiences is currently limited due to constraints commercial viability and the mediocrity of the final application. While dedicated frameworks like OpenXR aim to unify virtual reality software development. Dedicated frameworks such as OpenXR have emerged to address this challenge, offering compatibility across various VR and AR platforms and their differing cross-platform APIs. However, the landscape remains akin to that of augmented reality, warranting a concise treatment in this discussion.

2.2 The World of Robotics

In this section, we embark on a comprehensive exploration of robotics, starting with a foundational theoretical analysis. This analysis entails defining key terminology from an engineering perspective and discussing fundamental concepts such as sensors, kinematics, singularities, envelopes, etc. We then delve into the diversity of robotic manipulators, outlining various types of robot arms used across commercial and industrial domains. This overview sets the stage for our subsequent discussion on implementing zero gravity tools simulation using robotics.

2.2.1 Robotics Theory Unveiled

The essence of robotics lies in the integration of mechanical and software engineering, where physical hardware embodies the form of a robotic mechanism mimicking human anatomy. In "Modern Robotics"[LP17] by K. Lynch and F. Park, a robotic arm is defined as a system comprising interconnected rigid bodies, or links, joined together via joints, facilitating relative motion. However, without actuation, this system remains akin to a limp limb. Activation of these joints, typically through electric motors, enables coordinated movement, allowing the robotic arm to exert forces on objects, thus serving as a versatile tool. We will use the lightweight definition of a robotic arm given by the book called "Modern Robotics"[LP17] to further explain and frame this field of study.

Core Concepts

Link chain setups are mainly categorized into two types: open-chain and closed-chain. In an open-chain serial setup, a stationary base initiates the sequence, while an end-effector, often equipped with a gripper or other tool, acts as the final link engaging with the environment. Although other setups exist, they are beyond this thesis's scope. Actuators consist of various electrical motors such as DC motors, AC motors, stepper motors, or shape memory alloys. The weight distribution, with motors typically being the heaviest, is crucial. The base actuator must support all subsequent parts, whereas the end-effector actuator only needs to support the external object being manipulated. Additionally, it's important to acknowledge that some actuators may exhibit imperfections. Ideally, they should minimize slippage and backlash, defined as the rotation available at the output without input motion. For example, consider a gear system with two gears: a large input gear and a smaller output gear. In the absence of backlash, rotation of the large gear should promptly translate into motion of the small gear. However, the presence of backlash may cause a delay or "play" in the movement of the small gear relative to the input rotation. Additionally a braking mechanism may be included to stop the robot or maintain a stationary posture.

Due to the imperfect movement of robots in a 3D space, it is crucial to verify that the commands sent to actuators result in the correct displacement. Therefore, it's necessary to measure the joints' location and sometimes their velocity. This can be accomplished using various sensors or by incorporating sensing capabilities within the actuators to determine the rotational position of the links. Additionally, vision-only cameras can be utilized for inside-out or outside-in tracking, as discussed in section 2.1.2. The configuration of a robot system refers to the position of every point on the robot. In a 2D plane, a robot can be described using three variables: the x and y coordinates and its orientation. This concept extends to a 3D plane, requiring six variables: the x, y, and z coordinates and three components that define the link's orientation. The number of variables needed to define a rigid body's position and orientation is known as its degrees of freedom (DOF). A link that can rotate and move freely in a 3D space has six degrees of freedom. The DOF of an open-chain robot can be determined by summing the DOFs provided by each link in the system. Furthermore, robots utilize various types of joints. The most common ones are revolute (rotational) and prismatic (translational) joints. Other types, such as ball joints, also exist. A ball or spherical joint can be considered a special case of two revolute joints connected by a link of zero length[MS22].

Kinematics

In the domain of robotics, kinematics plays a pivotal role in enabling precise tool manipulation within zero gravity contexts. Central to kinematic analysis are forward kinematics and inverse kinematics. Forward kinematics entail determining the spatial configuration of the end-effector solely based on the joint parameters of the robotic arm, abstracting from force considerations. Conversely, inverse kinematics tackle the inverse problem, deducing the requisite joint parameters to achieve a specified end-effector configuration. Common methodologies for forward kinematics encompass employing transformation matrices or geometric algorithms to propagate motion across articulated joints, while inverse kinematics frequently leverage iterative numerical techniques such as the jacobian transpose[Wel93] or pseudoinverse[Bus04] to iteratively refine joint parameters until the desired end-effector state is attained. These implementations are advantageous because they inherently support joint limits; however, they can be challenging to work with.

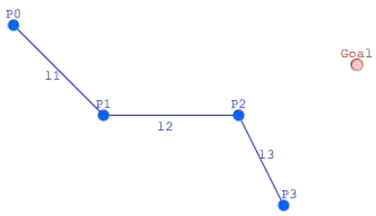
FABRIK (Forward And Backward Reaching Inverse Kinematics)[AL11] represents a notable algorithmic framework for addressing inverse kinematics challenges in robotics without recourse to rotational representations or matrix operations. This approach could be used when the resulting end-effector has a tiny delta from the specified input end-effector meaning it is a heuristic algorithm. This methodological approach iteratively refines joint configurations to approximate a result, yielding realistic poses within a minimal number of iterations. Operationally, FABRIK entails two primary phases: the forward propagation phase and the backward propagation phase. In the former, the algorithm initializes with the base joint configuration and progressively updates joint positions to propagate the end-effector toward the target position through linear interpolation. This phase serves to spatially align the end-effector with the desired target, albeit potentially deviating from the prescribed orientation. Subsequently, the backward propagation phase commences, wherein FABRIK systematically adjusts joint positions from the end-effector back to the base of the robotic arm. Each iteration computes the positional error between the current joint configuration and the target position, iteratively redistributing this discrepancy along the kinematic chain until convergence is achieved. Noteworthy is FABRIK's efficiency in rapid convergence, typically requiring only a few iterations to attain realistic joint configurations. Leveraging iterative refinement based on positional discrepancies, FABRIK navigates the kinematic chain adeptly to realize the specified end-effector pose, rendering it amenable to real-time implementation in robotics simulations. This method is also beneficial for generating smooth motions, as it utilizes previous points and moves the joints by a minimal amount. This approach ensures temporal stability, in contrast to other methods that often result in problematic and jarring transitions between solutions.

Let's explain how FABRIK works using a simplified 2D example using three points and two links. This explanation can easily be used in 3D environments with more joints and links. The simplest case of FABRIK is when the goal is out of reach for the total length of all the links. The solution could then be given by pointing every link towards the goal and essentially stretching out the link chain as seen in figure 2.6b. This could also result in the link chain reaching the goal if the total length is equal to the distance between the base and the goal point. When we consider a less trivial case like seen in figure 2.6a, the algorithm goes as follows. First, the backward propagation begins by positioning the end-effector, P3, at the target location, creating a new point P3'. This often breaks the constraint of the link length between P3' and P2. To address this, a vector is formed between P3' and P2, and P2' is placed along this vector such that the distance between P3' and P2' equals l3. This process continues, adjusting each point up to P0, which is repositioned to P0' as seen in figure 2.6d. Since P0 is the fixed base, it must remain stationary. Therefore, the second phase, called forward propagation, begins. In this phase, P0' is moved back to its original position P0 and is called P0", and the process of adjusting the points to maintain link length constraints is repeated, propagating the adjustments forward until all points are correctly positioned while respecting the constraints. Hence a new point for P1' is created called P1" as seen in figure 2.6e. Due to limitations such as floating-point precision and computational time, the algorithm uses a threshold (delta) to terminate when the end-effector is within an acceptable margin of error from the target. This approach ensures that the solution is sufficiently accurate while managing computational constraints. Finally, when the delta is reached, a result can be seen in figure 2.6f.

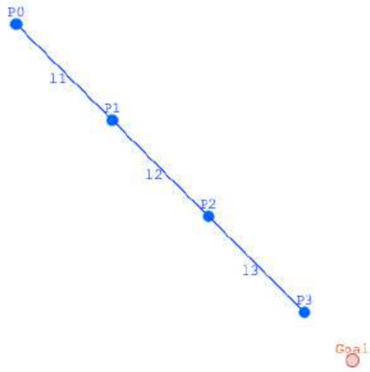
Spaces and Singularities

A robot's configuration denotes a comprehensive specification detailing the positional attributes of every constituent element as seen in chapter 2 of "Modern Robotics"[LP17]. The number of real-valued coordinates needed to describe this configuration defines the robot's DOF. The configuration space (C-space) is an n-dimensional space that represents all possible configurations of the robot. Each unique configuration of the robot corresponds to a specific point in this C-space.

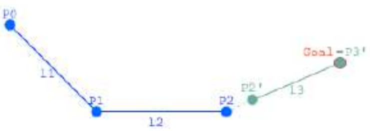
The task space and workspace constitute fundamental constructs governing the spatial capabilities and operational boundaries of robotic systems. The task space delineates the spatial requisites essential for executing a designated task. For instance, in scenarios involving planar tasks like writing on a sheet of paper, the task space typically manifests in two dimensions, corresponding to the surface area of the medium. Conversely, tasks necessitating three-dimensional articulation, such as object manipulation or assembly, extend the task space into three dimensions. In contrast, the workspace characterizes the admissible range of configurations attainable by the end-effector of the robotic arm. It encompasses the volumetric domain within which the robotic system can operate proficiently, constrained by mechanical limitations and kinematic



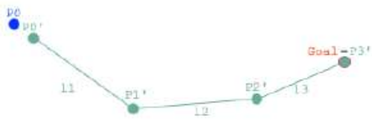
(a) The original FABRIK setup with the lengths denoted by lx and the goal.



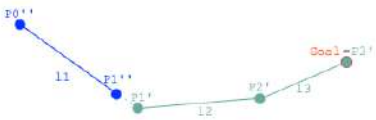
(b) The link chain is now completely stretched pointing towards the goal but the total length cannot reach the goal.



(c) The backward propagation started by moving P3 to P3' and adding P2' on the vector between P3' and P2.



(d) The point P0' is derived from the base point P0.



(e) The point P1" is a result of the forward propagation phase created in line of the vector between P0" and P1'.

(f) The final result of the FABRIK algorithm.

Figure 2.6: A link chain in different configurations according to the FABRIK algorithm.

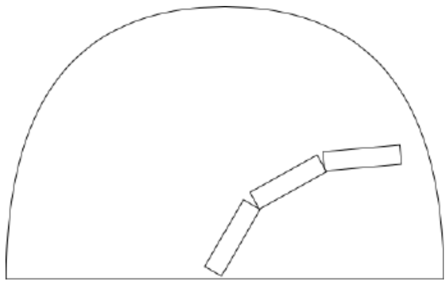


Figure 2.7: A robot's configuration space in 2D.

constraints. A comprehensive understanding of both the task space and workspace is imperative for devising robotic solutions tailored to specific applications, ensuring optimal maneuverability within prescribed spatial constraints while fulfilling desired task objectives.

The task space and workspace are distinct entities from the robot's configuration space (C-space). A point in either the task space or the workspace may correspond to multiple configurations of the robot, rendering it insufficient for fully determining the robot's setup. For instance, in the context of an open-chain robot with seven joints, the positional and orientational parameters of its end-effector fail to uniquely specify the robot's configuration. The introduction of additional degrees of freedom enhances the robot's flexibility, particularly evident in the specification of the angle of attack, denoting the direction from which the robot arm approaches the designated position such as seen in the research by D. Arathorn [Haq+15]. Certain points within the task space may remain inaccessible to the robot, exemplified by scenarios where, for instance, the x-coordinate exceeds the cumulative length of all links. Conversely, as per formal definition, every point within the workspace is theoretically reachable by at least one robot configuration.

As claimed by A. Sil [ZHL22], singularities in serial robotic manipulators denote configurations wherein the robot's mobility is restricted along at least one axis, underscoring their significance in refining contemporary control and motion planning paradigms. These singularities can manifest through various mechanisms, including inter-link collisions, collisions between a link and its external environment, or when joint limits are reached, necessitating a full rotation to restore orientation. One viable strategy for mitigating these singularities entails employing Monte Carlo simulations, a methodology expounded upon by T. Stejskal et al. [SSO22]. Furthermore, Stejskal et al. propose an innovative technique for inverse task computation, predicated on the stochastic mapping of the robot's workspace to identify points proximal to the target trajectory, obviating the need for computationally intensive inverse kinematic calculations. Such a robotic manipulator visualizer was also briefly designed during the creation of the ZeroTraining software as seen in section 4.2;

Human-Robotic Interaction (HRI) investigates the dynamics between human agents and robotic systems, emphasizing the nuances of human-robot collaboration and communication. Intention-based systems, a novel user-centric paradigm, discern and act upon user intentions, enhancing interaction with robotic agents for both actively interacting and passively supporting. Unlike its counterpart, Human-Computer Interaction (HCI), which primarily concerns itself with digital interfaces, HRI encompasses a tangible dimension, where robots engage in physical interactions with the environment. This domain is characterized by a heightened reliance on artificial intelligence (AI) and computer vision algorithms to facilitate seamless human-robot exchanges.

Additionally, within the realm of HRI research, a burgeoning field focuses on intent-based robotics, which scrutinizes how robotic agents decipher and respond to human intentions[ZD23]. The goal of the given research done by Wendemuth et al. [Wen+18] was to present and define a new class of user-centered assistance systems known as Intention-Based Anticipatory Interactive Systems (IAIS). This requires an expanded sensor array and a multi-modal approach for comprehensive user input, facilitated by sensor fusion techniques. Considerable attention is devoted to various facets, such as holistic bodily intentions that may indicate pedestrian locomotion for autonomous vehicles [Gol+20] or upright posture inferred solely from a pressure-sensitive floor design [Li+20]. Additionally, localized intentions are investigated, encompassing hand gestures [Che+20], upper limb actions including shoulder and upper arm movements for rehabilitation employing exoskeleton robotics [Liu+19], and even lower limb motions pertinent to the development of a single-leg knee joint assistive robot with motion intention recognition [MKH19].

2.3 Mediated Reality: Bridging the Physical and Digital Worlds

Mediated Reality (MR) integrates real-world elements with virtual reality (VR) and augmented reality (AR) to create a seamless blend of physical and digital experiences. It enhances user immersion by combining physical cues and sensory inputs with digital data. We will specifically explore how adding haptic feedback to VR and AR can enrich this experience by providing tactile sensations that align with virtual interactions. In this section, we will systematically select and analyze significant haptic implementations, followed by a comprehensive review of pertinent literature addressing similar issues to our study.

2.3.1 Reach Redirection and Haptic Retargeting

Reach redirection in virtual reality (VR) refers to the technique of altering the user's perception of their own movements within a virtual environment, typically to enhance immersion or mitigate spatial limitations. Research done by E. J. Gonzalez et al.[Gon+22] explores methods to dynamically redirect the user's reaching motions so that they align with the virtual scene, even when physical constraints or space restrictions would otherwise hinder realistic interactions. This is achieved through sophisticated algorithms that adjust the trajectory of the user's hand movements in real-time, creating a seamless experience where users perceive themselves interacting naturally within the virtual world. This can be seen in Figure 2.8a.

Additionally, Haptic Retargeting researched by A. Barrio et al.[Azm+16] addresses the challenge of integrating tactile feedback in VR environments where users interact with numerous virtual objects, which complement reach redirection by adjusting the tactile feedback users feel during interactions. By strategically manipulating users' visual and proprioceptive cues, the framework enhances the perceived realism and coherence of interactions, contributing to a more immersive virtual experience. This approach not only optimizes resource utilization by minimizing the need for multiple physical props but also demonstrates the potential to revolutionize how VR systems can efficiently simulate complex, multi-object interactions. As VR technologies continue to evolve, the application of Haptic Retargeting promises to expand the possibilities for interactive and engaging virtual environments across various domains, from training simulations to entertainment and beyond. This can be seen in Figure 2.8b.

2.3.2 Haptic Turk and Mutual Human Actuation

Haptic Turk[Che+14] introduces a groundbreaking approach to creating motion platforms by replacing traditional motors and mechanical components with human participants. In this system, a group of individuals, termed "actuators," manually generate forces and movements that correspond to virtual reality experiences. By lifting, tilting, and pushing the participant's body, these human actuators provide physical sensations that enhance the immersion of VR interactions. This innovative method not only allows for a more accessible and mobile solution compared to conventional motion platforms but also opens new possibilities for collaborative and social engagement in VR experiences. By using simple, rhythm-based instructions delivered through mobile devices, Haptic Turk coordinates the efforts of the actuators, ensuring that the timing and force applied are in sync with the virtual scenario. This approach demonstrates the potential for crowd-sourced haptics, making complex motion feedback systems more affordable and widely deployable, while also adding a unique human element to the immersive experience. This can be seen in Figure 2.8c.

A continuation on this is Mutual Human Actuation[CMB17] explores a new way to make virtual reality experiences more immersive by using people, rather than machines, to create force feedback. Unlike previous systems that required several people to act as "human actuators" for a single user, this new approach involves pairs of users who provide force feedback to each other without realizing it. For instance, one user might be reeling in a virtual fish, while another is battling a storm—each feels the forces caused by the other's actions, enhancing the realism of their experience. By synchronizing these interactions through shared props, the system makes virtual worlds feel more lifelike without needing complex machinery. This innovation could make high-quality virtual reality experiences more accessible and enjoyable. This can be seen in Figure 2.9a.

2.3.3 HapticPanel: Render Haptic Interfaces in Virtual Reality for Manufacturing Industry

HapticPanel[Deu+21] is another significant piece of research aiming to introduce haptic feedback for VR applications. It specifically focuses on the manufacturing design of machine control panels in VR. The visual component of the design is realistically represented to the user through a head-mounted display (HMD). The physical aspect is provided by a 2D surface that can orient itself in front of the user. This 2D surface includes multiple physical controls such as dials, knobs, sliders, and more. The user's hand is tracked using an external tracking system. When the user attempts to interact with a physical interface by reaching for it in the virtual world, the 2D surface slides underneath the hand, allowing the user to interact with the control panel. This design makes the control panel completely rearrangeable and modularly designable. The conclusions indicated that the speed of this system was sufficiently fast for touch interfaces, making it a promising design tool. This type of interface is very akin to robotic graphics or encountered-type haptic feedback. This can be seen in Figure 2.8d.

2.3.4 Thor's Hammer: Propeller-Induced Force Feedback

Propeller-based force feedback systems, such as Thor's Hammer[Heo+18], represent a significant advancement in the field of ungrounded haptic devices. By utilizing propellers to generate force feedback, these systems can produce continuous and multidirectional forces without requiring a fixed structure, making them ideal for mobile and untethered applications. Thor's Hammer, specifically, demonstrates how propeller propulsion can be effectively employed to create 3 DOF force feedback in virtual reality environments. The device is capable of generating forces up to $4N$ in any direction, which enhances the realism of VR interactions, such as simulating the weight of objects or the resistance of water flow. Despite the challenges associated with latency and noise, the ability of propeller-based systems to deliver strong and precise force feedback without grounding offers a promising avenue for further research and development in haptic technology. This can be seen in Figure 2.8e.

2.3.5 UltraBots: Large-Area Mid-Air Robotically Actuated Haptics

Traditional haptic devices, such as controllers and wearables, often face challenges like limited shape-rendering capabilities and complex setup requirements. Ultrasound-based haptics present a promising alternative, as exemplified by Ultraleap's technology, which utilizes ultrasound transducers for mid-air haptic feedback and shape rendering. However, a significant drawback is their constrained interaction area and fixed positioning, restricting their application versatility in VR environments. Moreover, the high cost of ultrasound transducers poses scalability challenges, making expansive coverage economically impractical. Despite these limitations, ultrasound haptics hold substantial potential for advancing VR development by enhancing tactile interactions beyond current capabilities.

To address the limitations of existing ultrasound haptic systems, the paper named UltraBots[FFS22] introduces a novel approach that combines ultrasound transducers with robotic actuation for large-area haptic feedback. By integrating ultrasound transducers onto mobile tabletop robots like Sony Toio, UltraBots can dynamically extend the haptic interaction area based on user hand and body movements. This innovative method not only enhances the precision and flexibility of haptic feedback across larger spaces but also mitigates the inherent constraints of stationary ultrasound setups. Furthermore, UltraBots demonstrates scalability by supporting multiple robots simultaneously, thus enabling comprehensive haptic sensations for both hands in VR interactions. This approach is envisioned to revolutionize various application domains including medical training, workspace simulations, educational tools, and interactive entertainment, showcasing the transformative potential of large-area ultrasound haptics in shaping the future of immersive technologies. This can be seen in Figure 2.8f.

2.3.6 HapticBots: Multiple Shape-changing Mobile Robots

In the realm of haptic technology, the concept of distributed encountered-type haptics introduces a novel approach known as distributed encountered-type haptics. This research is heavily researched in the paper called HapticBots[Suz+21]. This method employs coordinated robots capable of adjusting their position and shape dynamically to simulate various objects and surfaces distributed within a space. A key innovation lies in its ability to cover expansive and adaptable interaction areas efficiently, facilitating complex interactions that support two-handed manipulation. Each robot, designed with simplicity and modularity in mind, can scale effortlessly to increase the number of touch points and expand coverage, enhancing the versatility of haptic interactions. Furthermore, the system's portable and deployable form factor is pivotal, utilizing lightweight tracking mechanisms integrated into a simple mat or leveraging inside-out hand tracking technologies like those found in the Oculus Quest HMD. This setup ensures flexibility in deployment across different horizontal surfaces without the need for complex external tracking systems.

Beyond its structural advantages, distributed encountered-type haptics enables sophisticated haptic interactions with unique affordances. By employing concurrent lateral motion, the system can render continuous surfaces seamlessly, avoiding spatial aliasing common in traditional pin-based shape displays. Real-time tracking of user hand movements guides the robots to maintain constant contact, adjusting their orientation and height dynamically to match virtual geometries. Moreover, coordinated behaviors among multiple robots enhance the system's capability to simulate large objects or multiple virtual interactions with a reduced number of physical entities. This capability extends to rendering graspable objects, empowering users to physically manipulate and reposition robots as tangible inputs within the virtual environment, thus expanding the scope of interactive possibilities. This can be seen in Figure 2.9b.

2.3.7 Force Sensing and Zero Gravity Motion Simulation

The research by Hu et al.[Hu+21] investigates zero gravity motion simulation technology for spacecraft on-orbit service missions using industrial robots. By employing a BP neural network, the study establishes a force prediction model utilizing robot pose, acceleration, angular velocity, and angular acceleration as inputs, with data from a force/torque sensor on the robot wrist as the output. The model achieves accurate force sensing throughout the robot's workspace, enabling zero gravity motion simulation with a maximum force sensing error of 39.8N and a maximum torque sensing error of 18.7Nm for a 100kg load.

To evaluate the feasibility and human-machine efficiency of installing and replacing a 60kg antenna on the China Space Station, it is essential to simulate the on-orbit installation conditions on the ground, creating a zero gravity state for the antenna. Testers perform the installation according to on-orbit procedures, ensuring the process runs smoothly and recording the exerted operational force. This assessment determines whether the required force falls within the normal range that astronauts can apply in orbit, thereby verifying the practicality and safety of the operation. This can be seen in Figure 2.9c.

2.3.8 X-ARM: Exoskeleton combined with Extended Realities to train Future Astronauts

The X-aRm project[Bar+23] introduces a cutting-edge arm exoskeleton combined with Extended Reality (XR) technologies to train future astronauts, addressing the inadequacies of current training tools for microgravity environments. This innovative system enhances the immersion experience by providing multimodal stimuli and realistic force feedback, facilitated by a meticulously redesigned exoskeleton emphasizing robustness, comfort, and responsiveness. The exoskeleton, powered by three custom-designed Brushless DC motors and integrating two passive degrees-of-freedom, enables bilateral communication with the virtual world, allowing trainees to experience forces encountered during typical Extravehicular Activities (EVAs), such as pushing and pulling from handrails in microgravity. This approach effectively replicates the movements and constraints of wearing a spacesuit in real-time, supported by gravity compensation technologies. Consequently, training facilities utilizing the X-aRm technology are anticipated to offer greater immersion, flexibility, scalability, customization, and safety, while reducing the need for supervision and occupying less space. This can be seen in Figure 2.9d.

2.3.9 Collaborative Robots

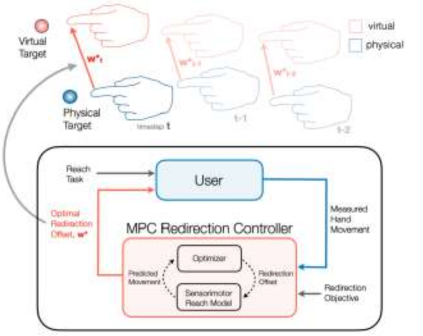
In the study by Adriaensen et al. [Adr+22] explore how integrating collaborative robots (cobots) into industrial environments can be made safer and more effective by incorporating systems thinking methods. Traditionally, cobot safety has focused on controlling kinetic energy and ensuring physical separation between humans and machines. However, this approach can be limited when applied to the increasingly complex tasks that modern cobots are designed to handle. By introducing systemic safety analysis methods, such as the System-Theoretic Accident Model and Processes (STAMP), the Functional Resonance Analysis Method (FRAM), and the Event Analysis of Systemic Teamwork (EAST), the study demonstrates how these methods can address the broader socio-technical interactions between humans and cobots. These approaches highlight the importance of understanding the distributed nature of cognitive processes in cobot operations, allowing for more comprehensive risk management and safer implementation of these technologies in industrial settings.

2.4 Extraterrestrial Practices

The realm of outer space presents humanity with an environment unlike any other, where the laws of physics reign supreme and the challenges of exploration are uniquely daunting. Among the myriad complexities encountered in space, extravehicular activities (EVAs) stand out as pivotal moments in astronaut missions, offering both unparalleled opportunities and formidable obstacles. The practices of outer space encompass a broad spectrum of activities and protocols designed to ensure the safety, efficiency, and success of these ventures beyond Earth's atmosphere.

This exploration delves into the practices of outer space, with a particular focus on EVAs and the management of zero gravity tools during spacewalks. As astronauts venture into the void, they are tasked with navigating an environment devoid of the familiar constraints of gravity, where tools and equipment must be carefully managed to prevent mishaps and ensure mission objectives are met. Understanding the intricacies of these practices is essential for comprehending the realities of space exploration and the measures taken to mitigate risks and optimize performance.

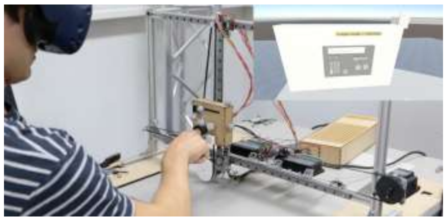
Throughout this exploration, we will examine the protocols, technologies, and training method-



(a) Overview of the proposed Model Predictive Control approach to reach redirection. As the user reaches to a target (top), the optimal virtual hand ofset is computed at each timestep based on the redirection objective[Gon+22].

(b) 1 for 3 Illusion - User touches 3 different virtual cubes while in reality the same cube is touched each time[Azm+16].

(c) Haptic turk allows producing motion experiences anywhere anytime. Here, the suspended player is enjoying an immersive hang gliding game. The four actuators create just the right physical motion to fill in the player's experience[Che+14].

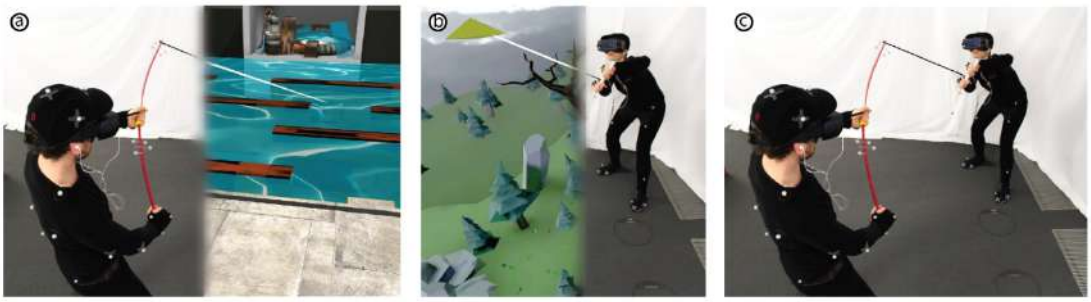


(d) Virtual Reality (VR) allows simulation of machine control panels without physical access to the machine, enabling easier and faster initial exploration, testing, and validation of machine panel designs[Deu+21].

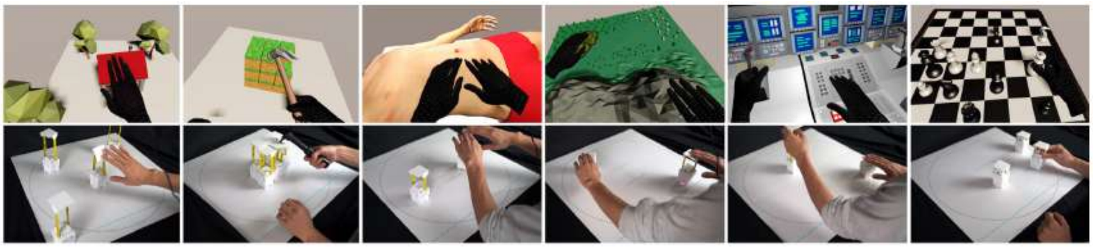
(e) Thor's Hammer held in a user's hand[Heo+18].

(f) System Design to enable 2D translations of ultra-sound transducers using Sony Toio tabletop robots[FFS22].

Figure 2.8: The research studied in this section with the respective captions from the original sources.

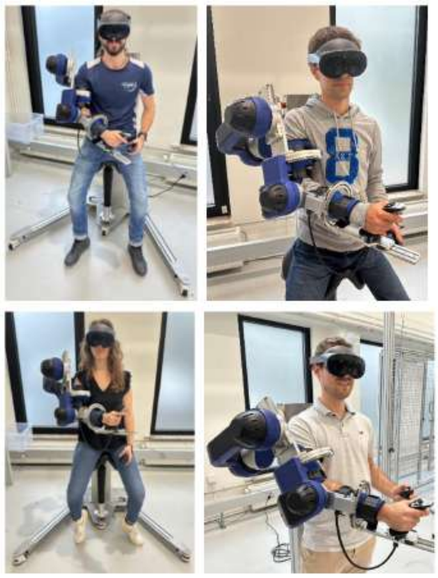


(a) (a) This user, alone in his virtual world, is trying to pull a huge creature out of the water. He feels how the creature is struggling and pulling on his fishing rod. (b) At the same time, this other user, also alone in her virtual world, is struggling to control her kite during a heavy storm, which is whipping her kite through the air. (c) While users' experiences of force might suggest the presence of a force feedback machine, Mutual Turk achieves force feedback instead using shared props that transmit forces between users. The system orchestrates users so as to actuate their prop at just the right moment and with just the right force to produce the correct experience for the other user[CMB17].



(b) HapticBots render encountered-type haptics for a range of VR applications[Suz+21].

(c) Human-robot interaction in zero gravity state[Hu+21].



(d) Integrated X-aRm system with different candidates validating comfort, adjustability and portability[Bar+23].

Figure 2.9: A continuation of the research studied in this section with the respective captions from the original sources.

ologies employed to facilitate safe and effective extravehicular activities. From the meticulous planning and preparation required before a spacewalk to the execution and management of tasks in the unforgiving vacuum of space, we will delve into the challenges and innovations shaping the practices of outer space. By shedding light on these essential aspects of astronaut missions, we aim to deepen our understanding of humanity's ongoing quest to explore the cosmos and push the boundaries of our knowledge and capabilities.

2.4.1 The Intricacies of Extravehicular Activities (EVAs)

An extravehicular activity (EVA)[Das05] entails an astronaut exiting a spacecraft while in space to perform various tasks[con24a]. Space, in this context, is characterized by the absence or negligible presence of gravity, either due to the lack of significant mass or the equilibrium of forces resulting in microgravity conditions[NAS95]. Moreover, the absence of breathable air is an additional requirement for it to be called an EVA. This necessitates astronauts to rely entirely on supplementary life-support systems. Therefore, astronauts undertaking EVAs must be equipped with spacesuits or other suitable vehicles to ensure their safety and survival outside the spacecraft. Here the term extravehicular mobility unit (EMU) was established combining such a spacesuit and a life-support system[Das05]. Whilst NASA sometimes exclaims that EVAs are always executed in orbit above earth[NASc], this is not necessarily an all encompassing definition. An EVA can be conducted in Earth's orbit, the orbit of any other planet or moon, or even on the surface of any other celestial body.

2.4.2 Classifying Types of EVAs

This implies the existence of various types of EVAs, which indeed is the case. One such type is the Stand-Up EVA (SEVA) [MG10], which involves limited mobility as the astronaut is only partially exposed to the hazardous environment. In a SEVA, the spacecraft features a dedicated open hatch through which the astronaut's upper body can extend. This configuration allows the astronaut to conduct activities outside the spacecraft without the risk of drifting away uncontrollably. The success of SEVAs paved the way for the development of full external EVAs. The training methods of these types of EVAs will be provided in the subsequent Subsection 1.3.

Another notable Extravehicular Activity (EVA) involved the utilization of a Hand Held Maneuvering Unit (HHMU), a compact device designed to be gripped in one glove, enabling astronauts to navigate in the weightlessness of space. In contrast, a simpler approach utilizing tools like Velcro patches, foot restraints, and handrails was implemented as an alternative concept. This led to the development of two distinct types of mobility support systems, informally categorized in this study as tethered and untethered EVs (Extravehicular Crewmembers). Untethered crewmembers rely on internal propulsion mechanisms during spacewalks, whereas tethered crewmembers utilize their own physical exertion while connected to the spacecraft.

Typically, EVAs undergo thorough advance preparation and planning, with ongoing monitoring and potential adjustments during execution. Consequently, many tasks are predetermined. Noteworthy research, such as NASA's MINERVA [Dea+17] project, has focused on preplanning such missions. MINERVA facilitated scheduling and monitoring of extravehicular activities on planetary surfaces through user-centered science operations software, drawing upon insights from the BASALT science program. This program explored surface habitability on Mars compared to earthly environments like Hawaii's East Rift Zone and Idaho's eastern Snake River Plain. Fieldwork during simulated Mars missions, including EVAs, was conducted, featuring annotation of findings, geographical marking, detailed monitoring graphs, and dynamically adjusted timelines accounting for communication latency between Earth and Mars. Additionally, tools like SEXTANT were incorporated which detailed three dimensional maps and paths for planetary traversal. While these tools were deemed helpful, better integration was identified as necessary. Research emphasizes simplicity and integration over functionality, as evident in studies on EVA evolution. This perspective is supported by findings from research such as "The

Evolution of EVAs.” [MG10].

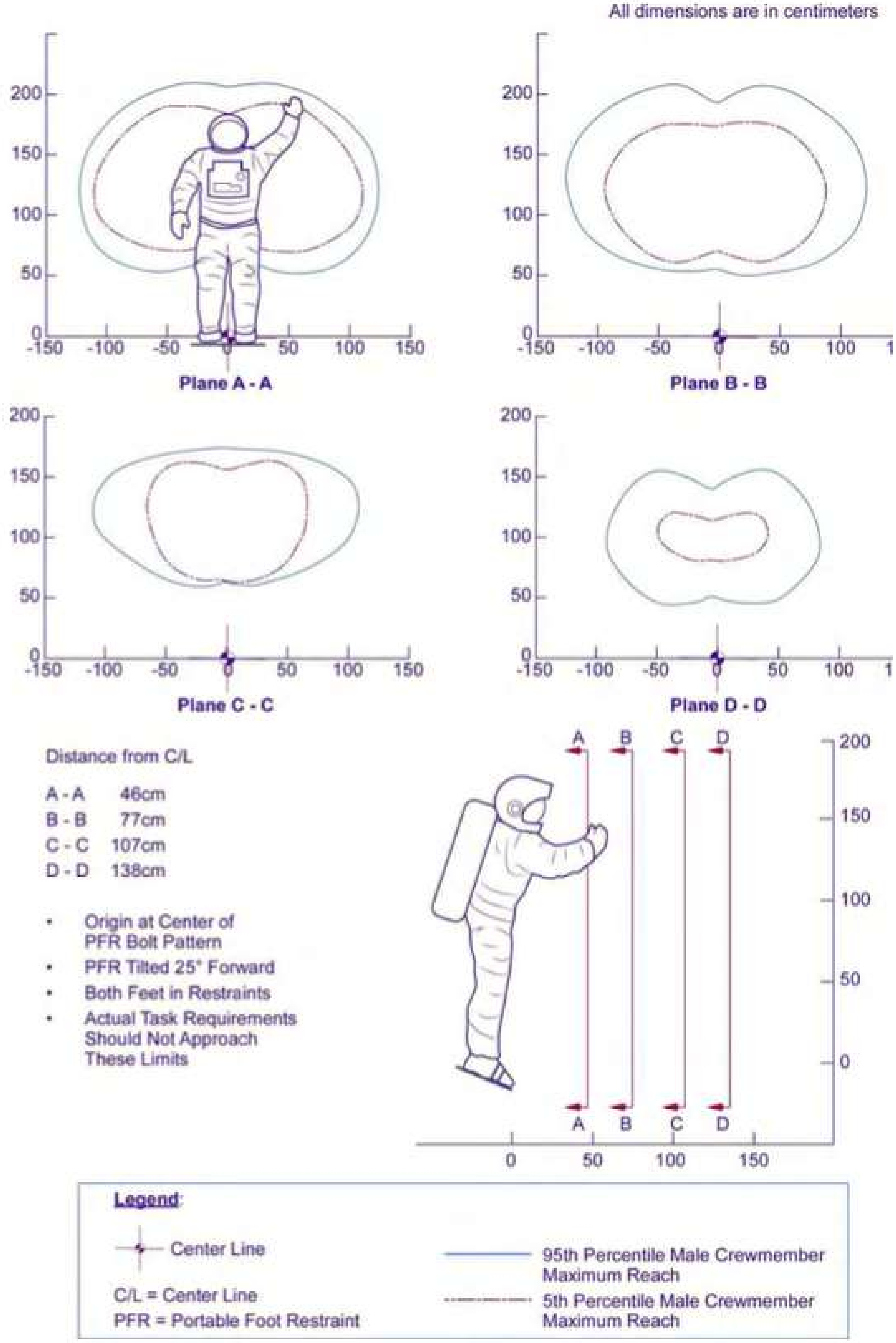
2.4.3 The Extravehicular Mobility Unit (EMU)

For an EVA suit to be as easily manageable as possible, it needs to fit the body snugly. These suits, known as Extravehicular Mobility Units (EMUs)[con24b], not only minimize barriers to interaction with the environment but also offer life support, environmental protection, and communication support between EVs or intelligence teams. Typical spacesuits, like those used during the Apollo missions, comprise various components such as visor shielding against UV light and liquid cooling systems. However, it’s important to acknowledge that these suits remain bulky and heavy, weighing approximately 110-150kg in total. Additionally, operating pressure, which stands at a formidable 4.3 psi, presents a strenuous challenge for astronauts. Looking ahead, there’s a continuous quest for improvement. About two decades ago, discussions were already underway to enhance EMU designs, aiming for easier maneuverability during tasks in space [JSN06]. This underscores the ongoing commitment to refining spacesuit technology to better support astronauts in their extraterrestrial endeavors.

An entire write-up of the EVA suit can be seen in the deprecated section 14 of the MAN-SYSTEMS INTEGRATION STANDARDS[NAS95] and in section 11 of the HUMAN INTEGRATION DESIGN HANDBOOK (HIDH)[NAS+10]. Adjusting the pressure levels of an EVA suit offers distinct advantages. Higher pressures can eliminate the need for extended prebreathe periods, reducing the risk of decompression sickness, while also providing a safety buffer between operational and emergency pressures. Conversely, lower pressures have been shown to decrease the forces exerted on the suit, lowering pressure loads and overall bulk, thus enhancing mobility in soft space suit designs. When the suits are pressurized, the decrease in reach ranges from 2% to 45%, with the most significant reduction observed in vertical downward reach when wearing the pressurized D-suit, which was not formally defined in the HIDH. For instance, the STS EMU glove design is engineered to maintain crewmembers’ skin temperature within a comfortable range while enabling dexterity comparable to heavy work gloves. Despite these advancements, achieving optimal grasp retention and force requirements remains a challenge. Nonetheless, lower pressure configurations enhance maneuverability, allowing astronauts to operate standard handles, knobs, toggle switches, and buttons with relative ease during EVAs.

Section 11.3.4 explains the ‘Suited Reach and Range of Motion’ [NAS+10]. This is a function of anthropometry of a crewmember. The Center of Disease Control and Prevention (CDC) defines it as “Anthropometry[Dis] is the science that defines physical measures of a person’s size, form, and functional capacities. Applied to occupational injury prevention, anthropometric measurements are used to study the interaction of workers with tasks, tools, machines, vehicles, and personal protective equipment — especially to determine the degree of protection against dangerous exposures, whether chronic or acute.” Anthropometric dimensions for suited crewmembers depend on the specific suit and its response to pressurization. Typically, basic anthropometric dimensions are given for shirtsleeve conditions. In $1G$, suit dimensions that are sufficient may not be optimized for $0G$, where crewmembers “float” inside the suit, potentially alleviating or creating new pressure points. An intriguing visualization is also provided, showing the envelope of the reach of motion given you are inside the spacesuit. It displays the 95th and 5th percentile for male crewmembers for both maximum side and forward reach. This can be seen in Figure 2.10.

When you’re inside an EMU, maneuvering with your lower body during $0G$ EVAs can be challenging. Typically, your feet are secured in foot restraints, so this aspect won’t be taken into account during this research. The helmet also restricts the FOV of the astronaut. This is a balance between protection and visibility. Most importantly, the critical areas of vision are not obstructed inside the helmet. This is defined in figure 11.3-1 [NAS+10]. It is interesting to provide a model of the helmet in VR but the main bottleneck of FOV will most likely be the headsets FOV instead of the helmet model to be used.



C/L = center line; PFR = portable foot restraint

Figure 2.10: Maximum forward reach envelope [NAS+10].

The HIDH[NAS+10] outlines various environmental factors to be considered during a spacewalk. Firstly, it emphasizes the pronounced temperature fluctuations. Additionally, it underscores the elevated radiation levels in the absence of atmospheric shielding. Lastly, it highlights the imperative need for protection against microdebris. Consequently, these factors necessitate a substantial augmentation in the layers of protection for astronauts' gloves.

2.4.4 Tools and Techniques in Zero Gravity

Tools are indispensable for executing a spacewalk, where astronauts venture into perilous environments. However, due to the significant resource allocation required for these missions, exploratory endeavors are infrequent. Consequently, most missions are purpose-driven. While human hands serve as the innate toolset provided by evolution, they often fall short of meeting the objectives of an Extravehicular Activity (EVA). Thus, this section delves into the types of tools utilized, their handling procedures, and their key attributes. Additionally, we'll explore the distinctions between tools used inside and outside a space station.

Intravehicular Items

It is important to compare the outside use of tools with how they are used inside the spacecraft. Inside a spacecraft, the management of tools is a meticulously planned and executed process. Velcro is a ubiquitous feature, serving as the primary means of securing tools and equipment in the microgravity environment. Unlike on Earth, where gravity keeps tools in place, Velcro[NAS07] ensures that tools remain securely attached to designated surfaces, preventing them from floating away and becoming potential hazards to crew members or sensitive equipment. This facilitates systematic organization. Additionally, the number and variety of tools available onboard spacecraft far exceed those used outside. This is primarily due to the less time-constrained environment inside the space station. Consequently, there's a greater emphasis on supporting a comfortable living environment, making amenities like eating utensils, clothing related items, hygiene products, and smart devices ever-present.

Interestingly, not all tools are tethered to specific locations within the spacecraft. While tethering is common practice for critical tools, others may be left untethered but strategically positioned for easy access during routine tasks. When a person inside the space station wants to use an item, they simply remove it from the wall where it was attached with Velcro. Once detached, the item becomes free-floating. Tethering these items isn't necessary because the worst-case scenario is that they might bump against the inner walls of the space station. While items cannot be completely lost, they can be misplaced inside the space station. This approach minimizes the risk of clutter and entanglement while maintaining efficiency and accessibility for crew members.

Moreover, the availability of fluids and other consumables onboard spacecraft is essential for various maintenance and repair tasks. From lubricants to cleaning solutions, these fluids are carefully stored and managed to ensure their safe and efficient use in the microgravity environment. Specialized containers and dispensing mechanisms are designed to prevent spills and leaks while facilitating precise application, thereby minimizing waste and maintaining a clean and functional workspace for crew members. These measures are preventative, but what if there's a need to remove free-floating fluid that has already escaped from its container within the environment? One method involves the use of absorbent materials such as towels or wipes to capture and absorb floating fluids. These materials can then be carefully maneuvered to soak up the fluids without spreading them further. Additionally, astronauts may use specialized suction devices or vacuum systems to remove floating fluids from the air or surfaces within the spacecraft. These devices can effectively capture and contain the fluids for proper disposal or recycling. These devices are described in section 7.12.4 under the title housekeeping tools in the HIDH[NAS+10].

Tool carriers and transfer devices within the spacecraft are designed with careful consideration according to the MSIS[NAS95] for the retention of small parts and hardware. These carriers feature mechanisms that securely hold these items in place while ensuring they remain visible for easy retrieval when needed. Moreover, the restraint of tools during translation is of utmost importance. To prevent detachment during movement, tools are firmly secured within the carrier or transfer device with sufficient force, ensuring they remain in place and readily accessible for astronauts during their tasks. Additionally, to mitigate the risk of inadvertent tool disassembly during installation, use, removal, or transportation, specific measures are incorporated into the design to prevent unintended disassembly, enhancing the overall safety and efficiency of tool handling in space.

Interestingly, section 8.2.3 of the HIDH[NAS+10] highlights the necessity for space stations to accommodate both $0G$ and $1G$, and possibly higher, environments. This means that in certain situations, things like ceilings are accessible or inaccessible depending on the state of the environment. To delve deeper, they assert that the design of the space station should also facilitate the transition between these states. This is why lose items in space are not just manually placed in a singular location in space. While technically feasible, employing such a strategy would be extremely impractical due to the likelihood of displacement caused by any initial velocity imparted upon them, either by placement or by accidental bumping. This is particularly problematic given that crew mobility heavily relies on the translation of their arms and hands, as well as pushing off nearby surfaces with their feet. As a second thought, when a space station is consistently launched from Earth, the presence of loose tools would undoubtedly create chaos during this transitional phase. Additionally, operating in a zero gravity environment presents its own challenges.

While increased movement mobility makes loose items more accessible, it also diminishes the control over one's body. Hence, there are numerous types of restraints and mobility aid locations available, both inside and outside the station. Examining such restraints is intriguing because at first glance, they occupy hands that could otherwise handle IV items. However, past experiences, as outlined in section 8.4.3 of the HIDH[NAS+10], reveal that, similar to on Earth, once individuals become adept at moving in the environment, they primarily use their feet. Hands are then utilized for carrying items or grasping these restraints for body orientation, speed, and stability control. While increasing the number of items on walls is a possibility due to humans requiring less space for movement, having dedicated kick surfaces remains highly valued.

Extravehicular Items

While many of these guidelines are applicable to extravehicular items used during a spacewalk, we will initially focus on delving into the specific guidelines outlined by NASA regarding tools during an Extravehicular Activity (EVA). The following guidelines are given by the HIDH[NAS+10] directly starting at page 812 of section 9 named tools. Standardization is crucial in ensuring efficient maintenance procedures. This entails adopting a common measurement system, whether English or metric, to facilitate consistency across tools. A minimal tool set is paramount for maintainability and reconfigurability, promoting ease of use and reducing operational complexities. By employing a minimum set of tools shared among different systems, maintenance tasks can be accomplished without the need for an excessive array of unique tools. Moreover, incorporating multipurpose and multi-size tools into the toolkit addresses unforeseen requirements effectively, enhancing versatility and adaptability in handling various maintenance scenarios. This comprehensive approach not only streamlines training, operations, and support requirements but also optimizes system functionality and resource utilization.

Tool and tool stowage labeling and identification requirements are crucial aspects of space missions [NAS+10], ensuring efficient organization and operation. Prominent labels should accompany each tool within the stowage container or kit, especially if the tool is not readily recognizable. This aids astronauts in quickly locating and identifying the necessary equipment during tasks. Moreover, tools should be tracked using an automated inventory control identification system, facilitating accurate inventory management and ensuring tools are accounted for at all times. Additionally, for Extravehicular Activities (EVA), it's imperative to ensure

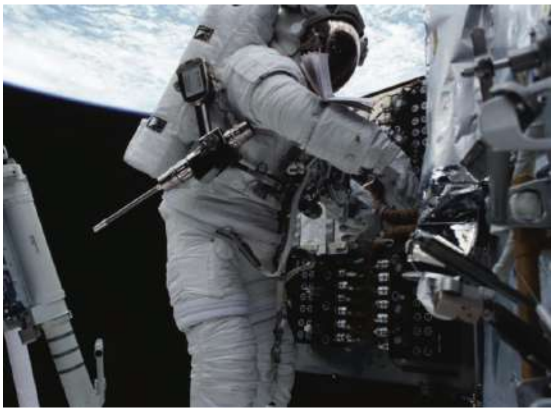


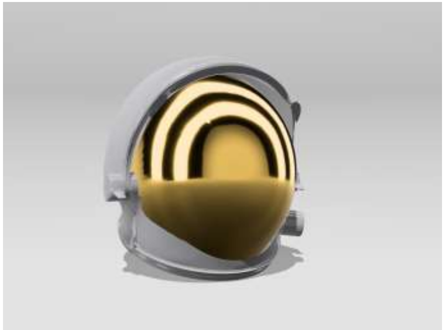
Figure 2.11: Astronaut Richard M. Linnehan, the Pistol Grip Tool secured at his side, works on Hubble during Servicing Mission 3B in 2002. Credit: NASA [NAS20]

compatibility between IVA and EVA. Any IVA tools that are not EVA-compatible must be clearly labeled and identified as such to prevent errors and ensure the safety and success of spacewalk missions.

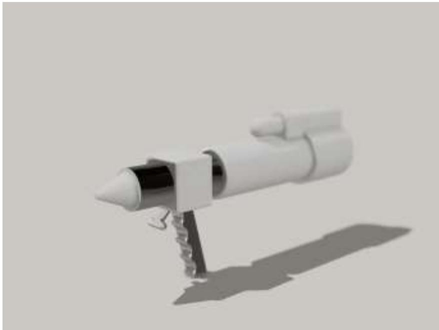
Ensuring a secure grip is paramount in tool design, especially considering the varying conditions under which they are used [NAS+10]. Hand grip surfaces must be meticulously crafted to accommodate both bare hands and EVA gloved hands without compromising the integrity of the latter. Brush-type motors, notorious for generating electromagnetic interference and potential ignition risks, should be avoided in power tool design. Historically, many hazards associated with tool usage have been shrugged off as inevitable aspects of the job, with users left to shoulder the burden of protection. However, there's a pressing need to proactively mitigate these risks, whether through design alterations or clear warning labels, to enhance safety standards across the board.

Although the number of tools and items available inside the space station is too extensive to be covered in this thesis, the tools used outside the station are much more manageable to account for considering their limited amount of them. Some of the most critical tools are available as 3D models on the NASA government website[GN]. Once the search tool is adjusted to filter for tools, several key tools can be identified. One notable example is the Pistol Grip Tool (PGT)[NAS20] as seen in Figure 2.11. The invention of this tool marked a significant milestone in space technology, revolutionizing the use of power tools by astronauts during extravehicular activities (EVAs). Developed by Paul Richards, a Hubble program engineer who later became an astronaut, the PGT was first used during Servicing Mission 2 in 1997, where it played a crucial role in servicing the Hubble Space Telescope. Featuring computer-controlled precision, the PGT provided astronauts with unparalleled control and versatility, offering customizable settings for torque, speed, direction, and other parameters tailored to specific tasks. Its introduction transformed EVA procedures, establishing it as the standard power tool for subsequent Hubble servicing missions and a mainstay for both the International Space Station and Hubble missions.

Additional tools include a hammer, a wrench, and a ratchet, all of which have direct counterparts on Earth and well-defined purposes. However, the most intriguing tool is the grease gun, which plays a crucial role in tribology[con24e], a field of study that becomes particularly significant in space, where it is referred to as space tribology. These tools can be seen in Figure 2.12 with the grease gun visible in Subfigure 2.12b. Lubrication is essential in preventing damage or wear between moving surfaces by introducing a third body with low shear resistance, which can include adsorbed gases, reaction films, or liquid/solid lubricants. Detailed discussions and examples of space tribology can be found in the book "Mechanics & Materials Science/The Mechanics and Materials Science Series" within the chapter on "Space Tribology"[JJ00]. This



(a) An astronaut helmet model uploaded by NASA Ames Research Center[GN].



(b) A grease gun used for applying oils and grease[GN].

(c) A hammer uploaded by NASA Ames Research Center[GN].

(d) The tethering system model with a rope [GN].

Figure 2.12: Some additional tools found on the NASA website[GN] which showcase the real world counterparts.

chapter delves into various engineering aspects, such as the enhancement of wheel bearings using lubricants, which are vital for the positioning of solar arrays. Slip rings[BGB][Sys20] act as electrical collectors, transferring current from stationary wires to rotating devices and are commonly used in space applications, where effective lubrication is crucial. Kalogeras et al. (1993) highlighted that excessive electrical noise, often caused by surface contamination, is a common failure mode in slip rings, underscoring the importance of selecting the appropriate lubricant to mitigate this issue. Several of these tools may require maintenance, making the grease gun an essential component.

During spacewalks, the efficient organization and accessibility of tools are critical for mission success and astronaut safety. Typically, only one tool is used at a time, making it essential that other tools remain nearby for easy switching. Toolboxes or tool stowages[NAS+10] effectively address this need. Proper tool arrangement within containers, such as through precise foam cut-outs, helps prevent tools from sticking or binding to the surrounding surface; however, alternatives to foam may be necessary for long-duration missions. The implementation of tool placement labels, such as different colored foam layers, aids in quickly identifying missing tools, thereby enhancing both efficiency and safety during spacewalks. Discussions with the European Space Agency (ESA) have emphasized the necessity of training in the use of toolboxes. Unlike tools attached to stationary surfaces, tools on dynamic surfaces like toolboxes present unique challenges, such as the "enchanted snakes" problem, discussed in detail in the introduction1. These tethers can lead to entanglement, presenting unpredictable challenges even for well-trained but inexperienced astronauts.

Section 9.7.3.2 of the HIDH [NAS+10] focuses heavily on so called equipment restraints. In the design of tethers, several guidelines were considered to ensure a consistent and streamlined experience: all tethers should adhere to a common method of attachment, necessitating the use of standardized interlinking. Each tether hook and receptacle is securely provided to the item requiring them. Additionally, the interlinking mechanism should be transparent, ensuring clarity on whether the hook is locked or unlocked, even in nighttime conditions. The concept of minimum and maximum loads is approached in a natural manner. The minimum load should support items under expected and normal working conditions, while the maximum load should account for a crew member attempting to dislodge the item. This means that astronauts should be able to disengage such tethers by force when necessary, without pulling the entire mechanism off the surface to which it was attached. There are also some general guidelines for equipment restraints given. Each restraint must be operable by hand, without the need for tools, and manageable by either the left or right hand. It should be manipulable without direct visual attention, allowing for quick management once muscle memory is established. Additionally, restraints must be adjustable and tightly fitting to prevent loosening due to environmental factors, thus ensuring that restrained items are not damaged. A common design should be employed for most restraints for consistency and ease of use.

Let's consider some of the tethers used during IVA and EVA. Temporary transparent plastic or netting Stowage Bags, Gray Tape, Cable Restraint Clips, Bundling Wrap Assembly, Velcro, Straps with Snaps, Metal and Elastic Bungee Springs with Snaps or Flat Hooks, EVA Tethers, Rubber Bands and Other Devices. Less durable tethering methods, such as Temporary Stowage Bags, Gray Tape, and Velcro, are unsuitable for EVA due to their susceptibility to temperature fluctuations. It must be acknowledged that Velcro is extensively utilized inside the space station, but it also has its drawbacks. Over time, it weakens due to the textile-based loops snapping[Bfe23] with each use, and it also tends to accumulate an unfavorable amount of dust and debris. Although more structurally robust restraints like snaps are theoretically more effective during EVA, aligning them proves challenging due to the lack of dexterity. Cable Restraint Clips were specifically designed for use in these environments, featuring adhesive backing and a steel spring-loaded pass-through. However, engineering did not anticipate the cables popping out of the seat track too easily, which has led to their disfavor among the ISS crewmember community. Finally we consider the use of EVA Tethers which are either fixed-length, adjustable and retractable. They are made to be able to withstand extreme environments and the most critical situations. They could however also be used using IVA.

Finally we take a look at section 9.13 [NAS+10]. It's important to note that most designs are also influenced by training considerations. Terms like acquisition, retrieval, retention, and transfer are crucial to bear in mind. Human-centered design aims to minimize errors and simplify training and procedures. In software design, it's understood that designers aren't always the end users, so prioritizing "intuitive" design can be misleading; user research is key. Emphasis is placed on preventing major errors over minor ones, as the payoff balance is significantly better. While these principles align with software design, delving deep into them isn't necessary here. Nonetheless, it's crucial to acknowledge their alignment with overall values.

2.4.5 Related Research Works related to EVAs

Virtual Reality (VR) offers the advantage of complete control over the user's visual experience, a capability not feasible in a neutral buoyancy environment (NBL). However, an NBL environment provides a more realistic experience due to the full offloading of limb weight. An intriguing area of research explores the combination of VR with neutral buoyancy. The re-

Figure 2.13: VR headset integrated with a full-face diving mask, featuring head tracking via an underwater camera system and a physical handrail replicated in VR[Age23].

search done by C. Sinnott et al. [Sin+19] focuses predominantly on the psychological aspects, given its nature as a master's thesis in psychology. The integration of innovative technologies with psychological insights presents compelling findings. For instance, the study adapted a SCUBA mask into a head-mounted display, albeit not as advanced as current VR devices, utilizing smartphone displays and 3D-printed lenses. Additionally, a jetpack locomotion feature was introduced for simulated Extra-Vehicular Activities (EVA) similar to the SAFER system mentioned before, replacing hand-tracking with a sealed controller attached to the scuba suit. The experiment was conducted in a standard swimming pool, utilizing SCUBA gear to achieve neutral buoyancy, a common practice in SCUBA diving. Several challenges arose, such as the degradation of Bluetooth and other wireless communication methods underwater. To address this, the researchers connected the controller to the Android phone using a watertight wire. Additionally, communication between the experimenter and the user was established by connecting the phone's headphone jack to an external microphone above water. To prevent the user's face from tilting upward due to excess air inside the SCUBA mask, weights were added to the headset. However, the use of rudimentary 3D models and simulations may have contributed to the reported user discomfort, aligning with findings of nausea during baseline tests conducted on the surface. These symptoms comprise manifestations such as heightened stomach awareness, increased salivation, perspiration, and generalized discomfort.

Ongoing research in this domain is specifically being pursued by the European Space Agency (ESA) [Age23]. For instance, an OSIP study demonstrated underwater VR astronaut training, featuring sophisticated hardware incorporating underwater tracking markers to enable comprehensive 6-degree-of-freedom (6DOF) movement, alongside a physical handrail replicated in the virtual environment. Nonetheless, detailed documentation regarding this research remains limited. This can be seen in Figure 2.13.

A noteworthy study investigates the use of digital twins in astronaut training[BCF19]. A digital twin is a virtual representation of a real-world entity, enabling simulation testing and monitoring. It allows for realistic visualization training and data analysis. The study focuses on modules for European Space Agency (ESA) training, particularly on extravehicular activity (EVA) simulations. Three scenarios are explored: replacing an ISS battery, repairing an exterior leak, and retrieving a sample with the Canadarm. The implementations utilize virtual reality (VR) for an immersive experience. Oxygen consumption was measured through temperature, pulse rate, and heart rate. It was observed that users' breathing patterns and heart rates varied depending on whether they were tethered, suggesting similar brain activation as in real-life scenarios.

Chapter 3

ZeroPGT: Virtual Reality EVA ISS Showcasing a Zero Gravity PGT

3.1 ZeroTraining: A complete Encountered-type Haptic Feedback System

Having thoroughly explored the background and related work, we are now ready to discuss the implementation of the ZeroTraining system. This system integrates a virtual reality environment with realistic haptic feedback, allowing users to both see and feel objects within the virtual space. It achieves this by linking the virtual reality component with a physical robot arm, which provides encountered-type haptic feedback. The robot arm translates within a 3D space and moves a physical object to correspond with a virtual object. When users reach for the virtual object, they can simultaneously feel the physically aligned object, enhancing the haptic realism of the application.

The system is divided into two intuitive main subsystems, which are detailed in the this chapter and the next chapter named ZeroPGT and ZeroArm respectively. The ZeroPGT chapter covers the virtual aspects of the system, including the development of 3D models, lighting, and physics rendering, along with the associated settings. Additionally the ZeroPGT application also facilitates the IK software to commands the ZeroArm subsystem. In contrast, the ZeroArm chapter addresses the physical components, focusing on the robotic manipulator and its controller, which facilitates the operation of the zero gravity Pistol Grip Tool. This chapter also provides an in-depth explanation of the inverse kinematics algorithm, including a custom version developed for this application (see Section 4.3). Additionally, it details hardware modifications and extra modules created using additive manufacturing techniques, such as 3D printing.

ZeroArm is specific to the robot arm, while ZeroPGT is designed to be compatible with any generic robot arm but is optimized for the robot arm used in ZeroArm. Both systems were primarily developed and implemented by the researchers during this thesis. Any components that are extensions or sourced from other materials will be cited accordingly. We will begin with the ZeroPGT subsystem, which is the Unity application shown in Figure 3.2. This application operates similarly to many VR applications, functioning as a standalone system that utilizes controllers and hand tracking to simulate a zero-gravity environment. ZeroPGT will be transformed into ZeroTraining through the integration of the ZeroArm hardware and software, which is the key innovation of this research. ZeroArm can be seen in Figure 3.3.

Many early design decisions for ZeroPGT were made with the integration of ZeroArm in mind, resulting in their components being finely tuned to each other. However, in discussing both chapters, it is essential to highlight the interdependence between the ZeroPGT and ZeroArm subsystems. We will place greater emphasis on the ZeroArm chapter due to its innovative solutions and the substantial time and effort invested. This approach clarifies that the ZeroTraining system comprises both the ZeroPGT and ZeroArm subsystems, which are interconnected and exchange information to support a haptic feedback system. This system's robustness allows it to be tested by participants, as will be detailed in subsequent chapters.

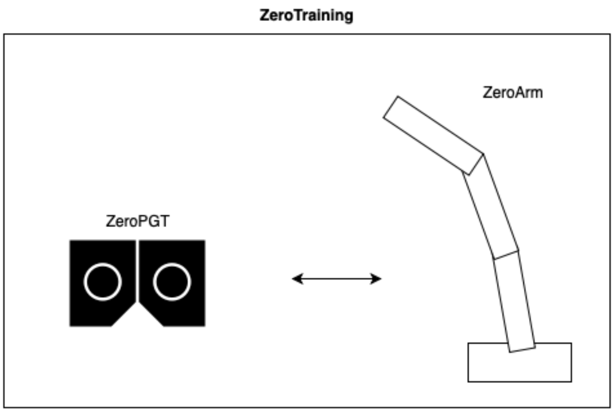


Figure 3.1: ZeroTraining as a simplistic diagram showing both subsystems ZeroPGT and ZeroArm.

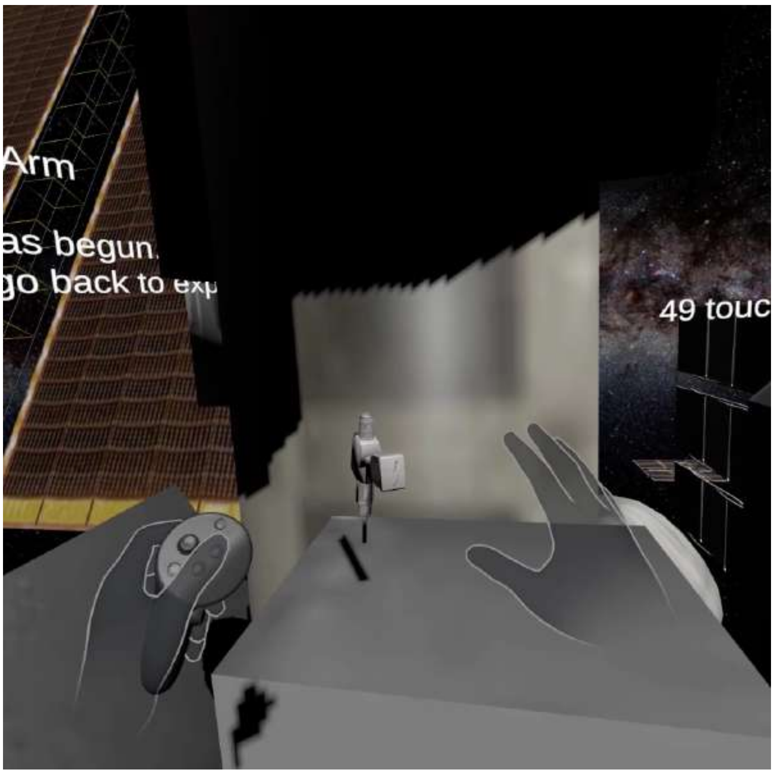


Figure 3.2: ZeroPGT in practice using the Meta Quest Pro Controller and the hand-tracking feature seen from the participants perspective without the use of a rope.

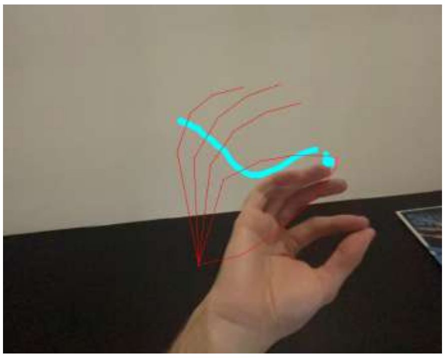
Figure 3.3: ZeroArm in practice using the Meta Quest 2 and the Arduino Braccio++ robot arm seen from an external viewpoint.

3.2 ZeroPGT: Purely Virtual Zero Gravity Pistol Grip Tool Simulation

The Zero Gravity Pistol Grip Tool (ZeroPGT) is an innovative application designed to simulate the conditions of stationary extra-vehicular activity (EVA) in space, where a zero gravity PGT hovers and can be manipulated by the user, specifically for astronaut training. By integrating this tool into a virtual reality (VR) environment, we create a highly immersive training experience. This section explores the implementation of ZeroPGT featuring an International Space Station (ISS), realistic lighting and physics and an accurate 3D model of the Pistol Grip Tool used during EVAs. The subsequent chapter will expand upon this simulation by incorporating the ZeroArm tool.

3.2.1 Experimental Path of the MR Platforms

This section is closely related to Section 2.1, which discusses the differences between augmented reality (AR) and virtual reality (VR). In the initial phases, both AR and VR were extensively used, with mobile applications being tested to evaluate hand-tracking capabilities in toolkits like ARKit and ARCore featured in figure 3.4a. However, these toolkits lacked maturity, often relying on outdated hand-tracking algorithms and not utilizing the LiDAR scanner available on



(a) A handtracking demonstration using an iPad.

(b) The 'enchanted snakes' problem portrayed in a PSVR application.

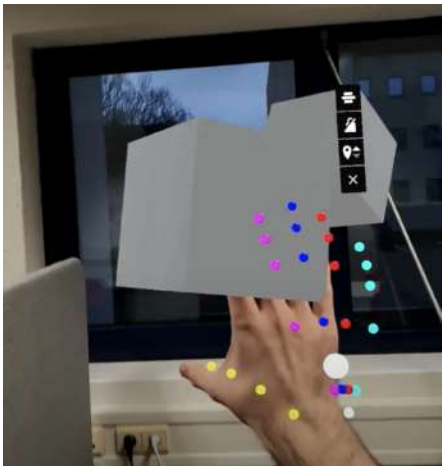
Figure 3.4: The earliest design concepts.

newer mobile devices. Additionally, a VR application was developed using the PSVR headset to demonstrate how a zero gravity object might behave with user input as seen in figure 3.4b. This application drew inspiration from the 'enchanted snake' problem, where tools were attached to a surface using uncontrollable microgravity ropes. However, these rope simulations were merely tethers that couldn't collide with each other or with the zero-gravity toolbox. The grab interactions were pointer-based, making them less realistic. Despite these limitations, the erratic behavior of the floating tools suggested that this type of simulation was on the right track.

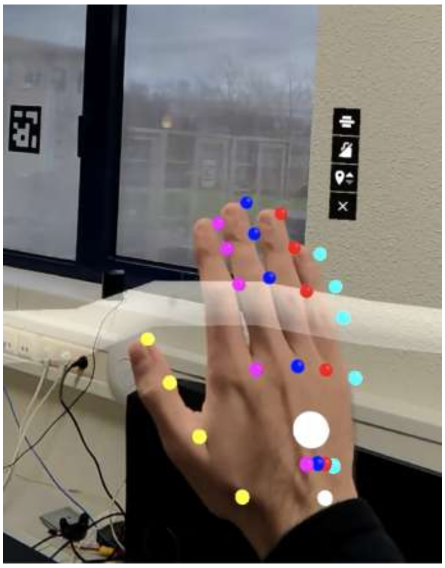
A decision had to be made regarding the most suitable platform for this stage of the design process. Although we were still exploring options, we chose to use the Braccio++ robot arm and decided on the Magic Leap 2 AR headset as the current platform. This choice was particularly viable in the early development stages, as the interactions primarily involved overlaying a single zero-gravity object without the need for a complete virtual environment. Additionally, the ability to see the robotic manipulator during development was crucial to ensure the safety of both the equipment and the user. Key base functionalities, such as the simulation of a zero-gravity object and rope, as well as communication between the robot arm and the HMD, were established at this stage (see Figure 3.5). Improvements and code could be relatively easily transferred to a VR headset like the Meta Quest, which also supports Unity as its game engine. Some adjustments were necessary to support the specific SDKs of these devices, enabling their unique features within Unity. The decision not to continue with the Magic Leap 2 for the final implementation was due to the need for a more immersive virtual environment to effectively integrate the haptic feedback system. A fully virtual environment would enhance the illusion of interacting with a virtual world rather than a physical robotic manipulator, providing a more sophisticated haptic experience.

3.2.2 3D NASA models and the Realistic Environment

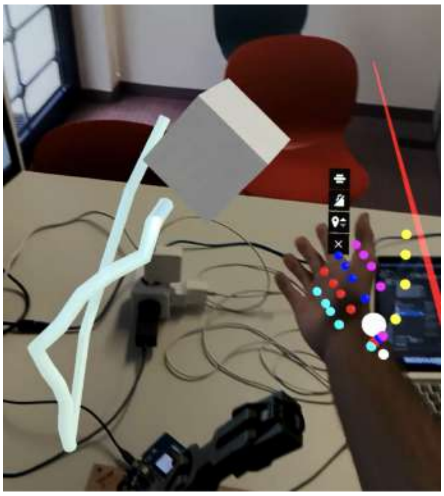
One of the first important components of creating a virtual world is that it feels believable so that the user can focus on the skill at hand without being interrupted by jarring and unimmersive behavior of the system. When creating a realistic simulation, it is important that the user feels somewhat present in the environment. Therefore, it is crucial to use relatively realistic 3D models that are preferably also used in the real world. Since this is a proof of concept, we don't perform the same tasks as we would during a real-life Extra-Vehicular Activity (EVA). The focus here is solely on realistic aspects such as lighting, models, physics, and abstracted interactions with the environment. This approach makes sense because the participants testing our proof of concept are neither experienced astronauts nor astronauts in training. Additionally, they are likely not regular users of virtual reality.



(a) An iteration of handtracking, rigidbody collisions and a $0G$ object.



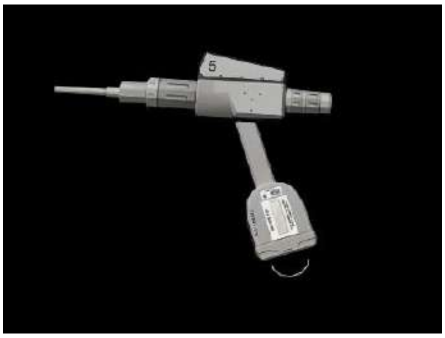
(b) An iteration where the hands can interact with a $0G$ virtual rope.



(c) The interaction of the $0G$ virtual object moves the unaligned physical robot arm (visible at the bottom) to the corresponding position using IK while a $0G$ rope tethers the object.

Figure 3.5: The design iterations of the simulation of the 0G environment using the Magic Leap 2 headset.

(a) The pistol grip tool (PGT) model features custom textures and can function as a screw with adjustable dynamic torque [GN].



(b) The pistol grip tool (PGT) model with original textures[GN].

Figure 3.6: The 3D models available on the NASA website for the Pistol Grip Tool.

The first aspect considered during the creation of this simulation was the 3D models used. After some research, many of the real-world objects depicted in Figure 3.6 used during EVAs on the International Space Station were found on NASA's website[GN]. However, the quality, fidelity, and age of these models varied greatly. Consequently, considerable effort was made to support multiple types of 3D model formats like .usdz, .gltf, .3ds, .obj, .fbx, and others. Additionally, various importing settings were adjusted in Unity, particularly for separating the mesh and material details. Although many custom-designed objects were initially created for this task, they were eventually replaced with the more realistic models available online.

The most important model in this simulation is the Pistol Grip Tool (PGT) shown in Figure 3.6a and 3.6b, which will be used during the main interaction with the user. The model used in our final application uses the original textures as depicted in Figure 3.6b. This model initially had some problems importing the materials, hence the custom textures given to this model seen in Figure 3.6a. The tool can be easily scaled to its original size and can be mounted using a rope simulation at the base which will be discussed in the following section. It has a natural grabbing point, making it easier for users to align with their probable grasping behavior. Additionally, the PGT is intuitive to use because many participants can infer the required gestures from its design and visual cues. The tool does not require complex animations to be defined by the programmer, enhancing its realism and compatibility with VR controllers. For example, the trigger of the virtual PGT can be pressed using the trigger on traditional VR controllers. The surface of the tool is intentionally varied rather than uniform. This design choice is crucial because one of the main goals of our simulation is to convey the sensation of weightlessness in the tool. If the chosen object were too simple, like a cube or a ball, many complex interactions could be lost due to the simplicity of the surface normals.

The previous sections focused on smaller objects within the virtual environment. Equally important is creating an immersive experience where the user feels present in a realistic setting. To achieve this, a 3D model of the International Space Station (ISS) was imported from the NASA website [GN], as shown in Figure 3.7. Although this model is not an exact replica of the real ISS due to its lower fidelity, it was selected because it is the most accurate model available from a reliable source. While alternative versions were considered, they were deemed unsuitable as many third-party models lacked the accuracy needed to maintain a convincing presence. Additionally, the skybox and lighting settings were adjusted to enhance realism. The skybox features the Milky Way galaxy[1], as shown in Figure 3.8, emphasizing the stars in the

[1] `https://assetstore.unity.com/packages/2d/textures-materials/milky-way-skybox-94001`

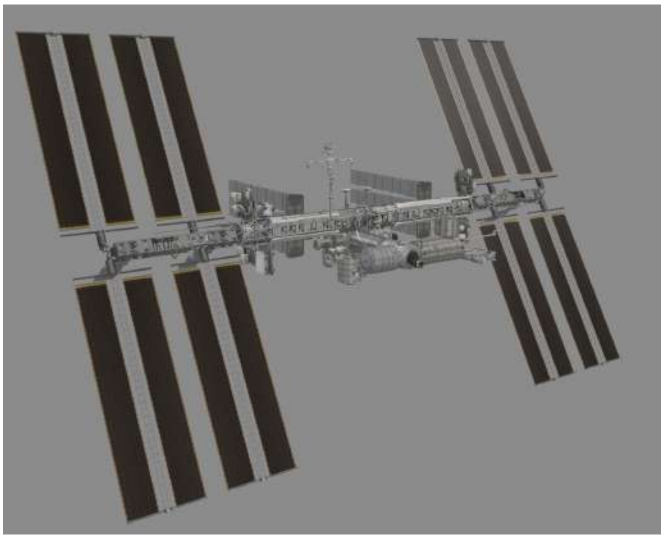


Figure 3.7: The International Space Station Model used in the ZeroTraining System[GN].

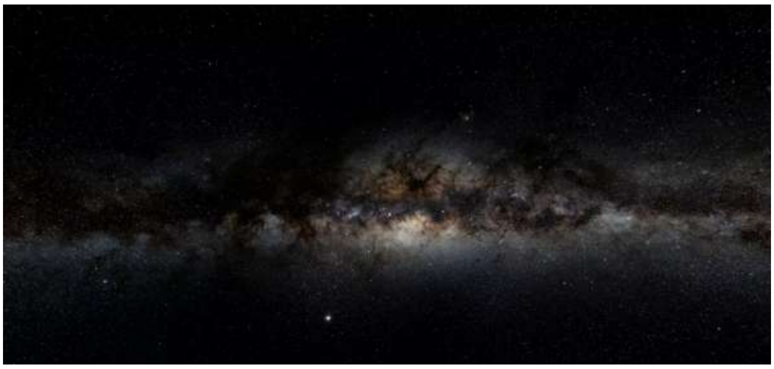


Figure 3.8: The Milky Way skybox used in the ZeroTraining System (see footnote 1).

background. Lighting was configured with a directional light behind the user, creating harsh shadows typical of the lack of ambient light in space. To ensure optimal performance on the Meta Quest 2, shadow casting by dynamic objects was minimized.

3.2.3 Virtual Physics and Rope Simulations

We adjusted several settings in Unity to enhance the accuracy and stability of our physics simulations. The documentation of these settings were suggested by a physics tutorial[Tut22] and the documentation of these settings can be found online[Tec]. Unity's Default Solver Iterations define the number of solver processes that run on each physics frame, managing interactions like joint movements and contact between overlapping Rigidbody components. Increasing the iterations from 6 to 25 helps reduce jitter in demanding configurations or when a non-default 'Time.fixedDeltaTime' is used. Similarly, we increased the Default Solver Velocity Iterations from 1 to 15. This setting determines how many velocity processes a solver performs per physics frame, enhancing the accuracy of exit velocities after collisions. This adjustment is especially useful if jointed Rigidbody components or Ragdolls move excessively post-collision. We also enabled the Adaptive Force option. This feature improves the realism of force transmission through stacks of objects, which is typically disabled by default. Next, we changed the friction model to One Directional Friction. This model, while requiring more solver iterations than the patch friction model, applies friction in alternating tangent directions, offering a simpler but less accurate approach compared to the two-directional model. For this model to function correctly with Articulation bodies, the Solver Type must be set to Temporal Gauss Seidel. Thereafter, the Temporal Gauss Seidel solver was selected. This solver provides better convergence, handles high-mass ratios more effectively, minimizes energy introduced during penetration corrections, and enhances the stability of joints. We change the fixed timestep in the time category to 0.01 instead of 0.02. Lastly we of course set the gravity to zero in multiple locations of the Unity setup. Sometimes this can be done in the settings of the rigidbody itself. Sometimes this can be done in the solver component as we will see in the following paragraphs for the rope simulation. It is important to note that the movement of the astronaut itself will not be simulated in this environment and the interactions with the tool itself are limited to touch. This does not mean however that the limited set of interactions provided are not less important neither easier to implement.

With the enhanced fidelity of our physics settings, we began by integrating Rigidbody physics into our system. This was accomplished using Unity's built-in physics engine by adding Rigidbody components and appropriate colliders to the necessary objects. We chose not to use mesh colliders due to their lower quality of physics, based on our experience, and instead opted for a combination of box, sphere, and capsule colliders. Initially, we assigned simple box colliders to tools in our scene to begin the development process. As development progressed, we moved to a compound collision setup, where multiple colliders, potentially of different types, are added under a parent object to create a more complex collision boundary. The first tool to receive this setup was the PGT, as illustrated in Figure 3.9. Adding colliders alone is insufficient; they need to interact with other objects for collisions to occur. Thus, user interaction was essential. One of the initial implementations involved adding a box collider to the controllers. This setup allowed the user to move the controller, which had an approximate box collider, facilitating the exertion of force on the PGT tool when the user pushed the tool within the environment using the controller.

Looking ahead to our ZeroArm implementation, the user will interact with a physical controller aligned with a virtual PGT, using their hands to grab the controller. To ensure consistency for user experience studies, we aimed to make both simulations as similar as possible. Therefore, we added sphere colliders to the finger segments of the hand-tracked hand mesh to simulate some sort of response. Initially, the palm did not have a collider, but due to the necessity for users to grab objects, a palm box collider was subsequently added. The colliders added to the right hand are visible in Figure 3.10 because their respective mesh renderers are enabled. Normally, these spheres and the box for the palm are not visible. An improvement could have been to use long cuboid colliders that align more accurately with the finger segments. This adjustment would prevent objects from passing through the user's fingers, which can happen with the current sphere colliders. This addition would enable the purely virtual version to have a collision response with the virtual PGT. In contrast, the encountered-type haptic feedback version does not include a collision response.

The next box collider added to the scene was for the ground.A detailed mesh was not required for this implementation, as its primary purpose was to prevent the tool from becoming lost and to ensure that the robot arm in the ZeroArm subsystem did not accidentally collide with the table. Additional scripting was included to ensure the tool remained within the user's interaction area during system execution. In a zero gravity environment, there is no air, meaning no velocity dampening force and no gravity to pull objects down. According to Newton's first law of motion, an object in motion will stay in motion until acted upon by an external force. Therefore, an object could drift away from the user if a force is applied. To enhance user experience, several scripts were introduced, albeit at the cost of some realism. One such script was the "Nudge Towards Envelope" script. This script ensures that if the tool leaves the user's work envelope, a new velocity is applied to return the tool to the envelope, keeping it within the user's reach.

Next, we will consider the rope simulations, which are crucial for realistic environments such as during an Extravehicular Activity (EVA) where tools are often tethered to prevent loss.

(a) The box colliders associated with the screw of the PGT tool.

(b) The box colliders associated with the handle of the PGT tool.

Figure 3.9: The PGT tool has two subobjects for the collider setup, namely the the ScrewCollider and the HandleCollider.

Figure 3.10: The sphere and box collider for the right hand when the mesh renderers are enabled.

Tethering significantly alters the behavior of zero gravity objects. The rope applies a force to the object, and the object applies a force to the rope. The rope also interacts with itself, and these interactions are vital because users may tug on or accidentally hit the rope. We utilized the Obi Rope[Met17], a Unity extension designed for extensive rope and cloth simulations. This rope simulation, as previously seen in Figure 3.5b, is based on particles interacting with the environment. Although these particles are not visible, a mesh representing the rope overlaid on these particles is visible. Numerous iterations were conducted to enhance the realism of the rope simulation, adjusting parameters such as thickness, flexibility, shear force, bend force, and bend constraints. To attach an Obi Rope to an object, we used the Obi Particle Attachment component, which links to a GameObject via one of the control points on the Obi Rope. Physics interactions with other objects were facilitated by adding an Obi Rigidbody component to those objects, along with the necessary basic Rigidbody and collider components available in Unity. While the specific guidelines and settings adjustments for achieving expected behavior are detailed in the Obi documentation[Met], they are crucial for ensuring accurate simulations.

To ensure the Obi Rope extension is compatible across various platforms, several key steps were undertaken during the system's development. The extension demonstrated seamless functionality on MacOS and Windows-based Unity game and device simulators, while required additional support for less conventional platforms such as Android-based AR and VR headsets. The underlying issue, as suggested by the documentation, stemmed from an incorrect backend configuration. To address this, we transitioned from the legacy Oni backend to the Burst backend, which involved integrating additional packages. Starting with Obi 5.6, Burst has become the default backend. Burst is implemented in high-performance C# and leverages Unity's Burst compiler and job system. It supports all platforms that can execute jobs compiled by the Burst compiler, and like Oni, it operates entirely on the CPU while employing multithreading and SIMD (Single Instruction, Multiple Data) techniques to enhance performance.

The packages required for the use of the Burst backend are:

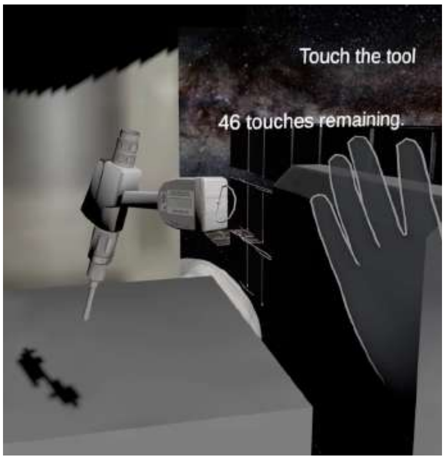
- Burst 1.3.3 or newer
- Collections 0.8.0-preview 5 or newer
- Mathematics 1.0.1 or newer
- Jobs 0.2.9-preview.15 or newer

3.2.4 Simultaneous Hand and Controller Tracking

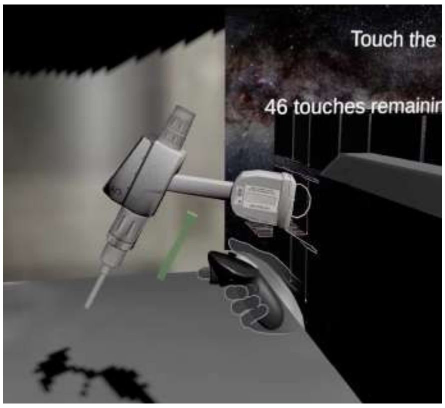
This application requires two essential aspects: realism and precision. Firstly, realism is crucial as users want to interact with the virtual environment in the most natural way possible. Using their real hands is the best method for this interaction. Fortunately, the Meta Quest 2 VR headset, used in the final implementation, features an excellent built-in hand tracking system. It is natural, easily customizable in Unity, and works well with a wide range of users without requiring extensive setup. Secondly, precision is necessary for this application. Users need to progressively familiarize themselves with the virtual environment, starting with the passthrough system, exploring the physics with the microgravity PGT tool, and finally testing the differences between the untethered and tethered PGT tool. Since VR is relatively unfamiliar to most participants, it is beneficial to structure the application in stages. Users should advance through these stages by pressing a button on a controller rather than using hand gestures. This approach avoids false positive triggers that might occur with hand gestures, preventing abrupt transitions that could hinder the user's exploration of the system. The user should be able to quickly switch between hand tracking and controller interaction, depending on which method is most suitable for the task at hand. However, this decision is not left to the user. Instead, the switching of input methods is managed by an instructor who is on standby during the system's use.

The solution to this issue is simultaneous hand and controller tracking, which the Meta Quest 2 headset fortunately supports. This feature automatically detects whether two controllers, one controller, or no controller is in use, and it enables hand tracking for the hands without controllers. However, this feature is somewhat hidden in the documentation and requires specific hardware for the Meta Quest 2 headset, which we used in our final system implementation. Simultaneous tracking is only supported when using the Meta Quest Pro controllers. These controllers are sold separately or bundled with the Meta Quest Pro. After some experimentation, we discovered this solution. We also adjusted software settings to make the physical controllers visible or invisible depending on the situation. For example, the left controller, which serves as the interface controller and is not used for realistic interactions with the environment, is always shown to the user when placed on the floor. When the user unintentionally displaces this controller, it can be easily found without needing instructor intervention. The right controller, however, is used later to be mounted on the end-effector of the robot arm (as discussed in Section 4.5). This controller remains invisible when not held by the user to prevent revealing too much information about the robot arm's actions and location during system operation. Revealing this information would break immersion and presence. Figure 3.11 provides a simple demonstration of how the dynamic switching of visuals operates. While the visuals are not displayed to the user, both the controller and the hand are continuously tracked. The left controller and hand, which remain visible at all times, are shown in Figure 3.12.

The integration of simultaneous hand and controller tracking significantly was introduced, leveraging the Meta Quest 2 headset's capabilities. This feature automatically detects whether two controllers, one controller, or no controller is in use, and it enables hand tracking for the hands without controllers. The Meta Quest 2 requires Meta Quest Pro controllers, which support simultaneous tracking and are either sold separately or bundled with the Meta Quest Pro. Through experimentation, we optimized the system to adjust software settings for dynamic visibility of physical controllers. Specifically, the left controller, which functions as the interface controller and does not interact with the environment. It is used for stage progression and settings configuration during the system's use. It hence should always be visible to facilitate user interaction and prevent misplacement. Conversely, the right controller, intended for mounting on the robot arm's end-effector as discussed in Section 4.5, remains invisible when not held. This approach prevents the disclosure of the robot arm's actions and location, preserving immersion and presence during operation. Figure 3.11 illustrates the dynamic visual switching mechanism, while Figure 3.12 demonstrates the consistent visibility of the left controller and hand, even as tracking continues for both controllers and hands.



(a) The user approaches the virtual PGT with their hand and their hand is being tracked.



(b) The user grasps the virtual Pistol Grip Tool (PGT) and feels the controller in their hand. For visual clarity, the locations of the controller and the virtual tool are deliberately positioned at a significant distance from each other.

Figure 3.11: A demonstration of the simultaneous hand and controller tracking.

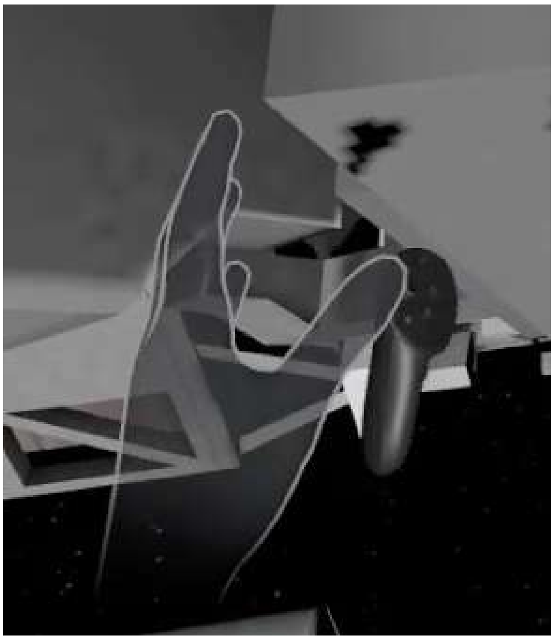


Figure 3.12: The left hand and the left controller are always simultaneously tracked and always shown.

Chapter 4

ZeroArm: Encountered-Type Haptic Feedback using a Robotic Manipulator

This chapter will cover various physical considerations dependent on the ZeroArm subsystem, including the design of the end-effector, an analysis of the Braccio++ robot arm, modifications made to the arm, and the iterative process for selecting an appropriate inverse kinematics library for relocating the end-effector. While a primary focus is on physical aspects, software-related details will also be included due to the integrated nature of the systems. The software running on the Arduino Braccio++ robot arm will be briefly described as well.

4.1 Analysis of Braccio++ Robot Arm and Modifications

This section begins by outlining the characteristics and advantages of the Braccio++ robot arm. While many of these characteristics are detailed in the official documentation, additional insights based on extensive development with this arm will also be provided. Modifications were made to the robot arm during this research, some of which were necessitated by the requirements of our ZeroTraining system, while others were quality-of-life improvements. A summary of the modifications made will be presented in the following subsections.

4.1.1 Arduino Braccio++ Robot Arm

The Arduino Braccio++[1], the latest iteration of the Tinkerkit Braccio robot, is designed specifically for higher education applications, including engineering schools, university institutes of technology, and advanced high school and college science programs. This robotic arm offers versatile assembly options, making it suitable for tasks such as object manipulation. The Arduino Braccio++ allows students to apply physical concepts through activities like lifting, placing, and rotating items, addressing topics such as motion, forces, torque, gear ratios, stability, and payload weight. Notable advancements in this version include the addition of a Braccio Carrier with an LCD screen, new RS485 servo motors, and an overall enhanced user experience. The structural components are made from EcoAllene, a sustainable plastic derived from recycled polylaminate food cartons, ensuring that all plastic parts are eco-friendly, recyclable, and easily replaceable. The kit also includes an e-learning platform with step-by-step instructions, lessons, and additional educational resources. These materials guide users through projects and the use of various hardware components, including the display, joystick, buttons, and intelli-

[1] `https://store.arduino.cc/products/bracciopplusplus`

gent motors. This curriculum provided a quick start and a fundamental baseline for the final implementation.

Central to the Arduino Braccio++ are the new RS485 servo motors, which integrate seamlessly with the Arduino Nano RP2040 Connect board for programming and communication. These smart servo motors facilitate bidirectional data exchange, adhering to industrial communication standards. Bidirectional data exchange is defined such that the Arduino Braccio++ robot arm can both send and receive information. For example, the smart servo motors in the arm can receive commands from the Arduino Nano RP2040 Connect board to perform specific movements. At the same time, these motors can send feedback data back to the board, such as their current position or any encountered errors. This two-way communication ensures precise control and monitoring of the robotic arm.

The robot arm used in this study, provided by the research center, offers several distinct advantages despite its simpler design compared to industry-grade alternatives. It is more approachable for programming, especially for individuals familiar with Arduino software and hardware. The lightweight nature of the arm facilitates ease of setup and demonstration. Furthermore, the arm is highly modifiable; the end-effector can be redesigned and iterated upon using 3D printing technology, allowing for rapid and efficient modifications. The process of swapping end-effectors is also notably simple. A major reason for choosing this robot arm is its relatively high speed. Although higher speeds may reduce reliability, accuracy, and precision, they are crucial for applications that require rapid movements, such as simulating objects in a zero-gravity environment. This application aims to facilitate physical interaction with virtual objects within a VR headset. While the robot arm's speed is prioritized, accuracy can be enhanced using additional software techniques, as discussed in [Döl+23][Wij+08][Azm+16] and detailed in Subsection 2.3.1.

Significant advancements were achieved in the development of the robot arm, particularly in addressing challenges related to reliability and user interaction. The design process led to notable improvements in the arm's performance, despite the inherent complexities. The enhanced reliability of the robot arm was a key focus, with extensive testing contributing to a more robust and dependable system. This iterative testing phase, while initially slowing implementation, ultimately refined the arm's operational stability and functionality. The potential for the robot arm to reach its operational limits was effectively managed, thus safeguarding the integrity of the proof of concept evaluation and validation. Additionally, the incorporation of sophisticated software for human-robot interaction marked a significant enhancement. This software, usually designed for more complex collaborative robots or "cobots," ensures that the arm operates safely and securely in proximity to human users. By preventing potential collisions, the system mitigates risks and enhances user safety. While the current implementation of these safety protocols may not match the complexity found in high-end cobots like the UR30[2] and the Elephant Robotics MyCobot 320 M5[3], it nevertheless sets a high standard for safety and interaction in the design of collaborative robotic systems. This approach underscores a commitment to improving user safety and operational reliability within the context of evolving robotic technology.

4.1.2 Tethered but Loose Carrier Board

To enhance the stability and safety of the robot arm, a significant design improvement was implemented concerning the Braccio++ Carrier board. Originally, the Carrier board, which houses the Arduino board, was connected to the robot arm but not directly mounted. This

[2] `https://www.universal-robots.com/products/ur30-robot?utm_source=Other&utm_medium=referral&utm_content=Qviro&utm_campaign=HQ_HQ_QviroTest2023&qviroGID=GA1.2.1138961693.1723971964&utm_leadsource=Referral&utm_term=Qviro`

[3] `https://shop.elephantrobotics.com/collections/mycobot-pro-320/products/commercial-and-economic-six-axis-collaborative-robot?ref=QVIRO&utm_source=Qviro&utm_medium=paid&utm_content=myCobot+320+2022-m5&utm_campaign=&qviroGID=GA1.2.1138961693.1723971964`

(a) The grip ratchet clamp used to keep the robot arm in place on the table.

(b) The zip tie brace aleviating the stress on the Micro-USB port on the board by transferring the stress to the mounting port underneath the Arduino Board.

Figure 4.1: The grip ratchet clamp and the zip tie brace.

configuration risked potential damage to the arm, adjacent objects, or users due to the Carrier board being dragged during base rotation.

To advance the design, we introduced a 3D-printed Carrier Board Mount, securely affixed to one side of the robotic arm. This modification provided more stable support for the Carrier board, improving the overall robustness of the system. It is important to note that the internal cables connecting the logic boards remain vulnerable to being tugged by nearby objects. Although some iterations of the robot arm included modifications for internal cable management, these improvements did not significantly enhance performance and were ultimately abandoned. These internal cables link various subsystems of the robot arm. External cable management will be addressed in the subsequent subsection.

4.1.3 External Cables Damaging the Physical Ports

In the development of the Braccio++ robot arm, significant advancements were made to improve the durability of external cables, particularly the command feed connection. The command feed, which utilizes a Micro-USB cable connected to the Arduino board mounted on the robot arm, is critical for system operation. Enhancements were implemented to address potential issues related to cable management and stability. One key contribution involved the extension of the command feed cable through the use of a USB female-to-USB male extension cable adapter. This modification provides the necessary flexibility to adjust cable length according to user movement without compromising the cable's integrity or risking entanglement. Additionally, the design of the cable mounting point was improved to better accommodate the dynamic movements of the VR headset user. A novel system was developed to manage the forces exerted on the Micro-USB port by using two zip ties to anchor the cable to the underside of the board mount. This strategic positioning ensures that any pulling force is redirected to a more stable location, thus reducing stress on the crucial port and enhancing the overall reliability of the connection essentially working as a brace system as seen in Figure 4.1b.

4.1.4 Abrasion of Plastic Servomotor Horns

The development of the ZeroPGT application led to significant improvements in the robot arm's payload capacity and joint responsiveness. During the design phase, it was observed that the arm's joints exhibited reduced responsiveness at certain angle ranges, which was traced to a defect in the horn gear. These components are depicted in Figure 4.2 and Subfigure 4.2b. The malfunction was caused by wear and tear, which was exacerbated by the arm's custom end-effector holder and the relatively heavy controller, pushing the limits of the arm's maximum weight capacity. A more detailed explanation of the defect can be seen in Figure 4.3.

To address these performance challenges, a series of enhancements were explored. Initial approaches included reducing the weight of the end-effector, limiting the joint angle range in the software, and 3D printing new plastic horns. While these methods offered temporary relief, the most impactful enhancement involved replacing the plastic horn with a metal horn, as illustrated in subfigure 4.2c. This metal horn significantly improved the arm's responsiveness and stability with only minor adjustments required to the overall setup. Despite slight misalignments, the metal horn's installation proved to be efficient, enhancing the arm's mechanical robustness and operational reliability.

4.2 Robotic Manipulator Visualizer and Envelope Calculator

During the design of our inverse kinematics algorithm, some software tools were designed by the research team to help aid to development of this system using software visualisers. The first visualiser, can be used to check the configuration of the robot arm using a set of input angles. Those angles are all configured for the respective joints and then mapped using a specified number of segment lengths. The final configuration is then plotted on a 3D graph as seen in Figure 4.4. The next visualiser is heavily inspired by the monte carlo methods described by T. Stejskal et al. [SSO22] in Section 2.2.1. The parameters of this visualiser could be adjusted so that the configuration envelope could be digitally seen as shown in Figure 4.5. Fundamentally it uses the previous visualiser to calculate the end-effector location using forward kinematics. Then a high amount of different robotic manipulator configurations are tried in either a regular or a stochastic method to plot the working envelope.

4.3 Choosing an Inverse Kinematics Algorithm

This section explores the development of a custom inverse kinematics (IK) solution within the ZeroPGT application, addressing the limitations of existing Arduino IK libraries. The custom solution is discussed in detail, a.o. the optimizations and improvements that were made. Potential enhancements for future iterations of this implementation are discussed.

The development of our custom inverse kinematics (IK) solution represents a significant advancement in addressing the limitations of existing libraries. The prevalent challenges with traditional IK libraries—such as outdated versions, inadequate support, and poor documentation—prompted us to design a bespoke solution independent of the ZeroArm platform. We adopted the FABRIK algorithm, a modern approach with extensive Unity extensions offering diverse functionalities and complexities. Our implementation utilized the FastIK extension[4], which, despite its challenges in integrating joint limiters, provided a robust foundation for our needs. We enhanced FastIK by incorporating custom modifications to effectively manage joint limits, overcoming the constraints typical of pure FABRIK implementations. This modification marks a significant shift from the previous approach of running IK software on the Arduino arm, as it enables the IK calculations to be performed directly within the Unity-based ZeroPGT application. As a result, the alignment of the world space between ZeroArm and ZeroPGT is no longer necessary, simplifying the integration process and improving overall system efficiency.

The joint angle limitations were designed for the specific Braccio++ robot arm in mind. This was designed as follows. In an inverse kinematics algorithm the end-effector and the target need

[4] https://assetstore.unity.com/packages/tools/animation/fast-ik-139972

(a) The servomotor with the output shaft.

(b) The degraded plastic horn mounted on the given servomotor.

(c) The replacement metal horn.

Figure 4.2: The components in relation to the abrasian experienced. The abrasion of the plastic servomotor horn resulted from the material difference between the servomotor shaft and the plastic horn, compounded by extensive use.

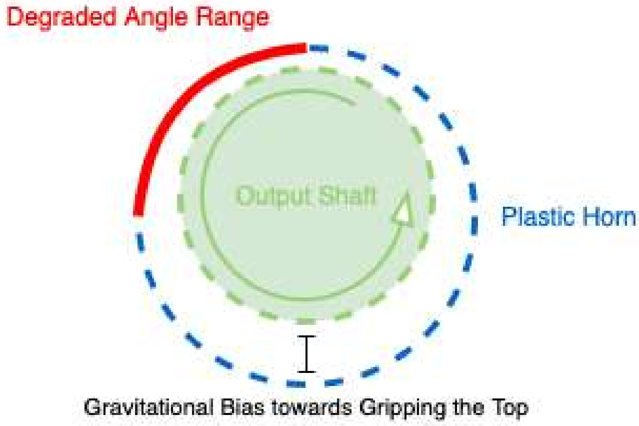


Figure 4.3: The shaft rotates the horn, which subsequently moves the associated joint and link. As the shaft turns, the degraded angle range positions itself at the top of the plastic horn. Gravity introduces a slight bias towards gripping at this top position. Consequently, when the output shaft enters the degraded range, it slips continuously due to the worn condition of the horn.

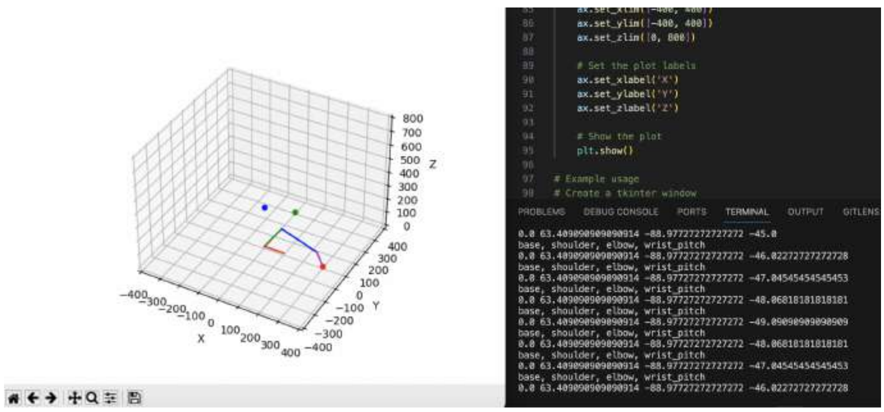
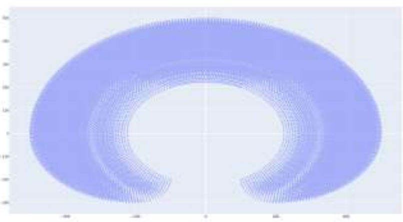
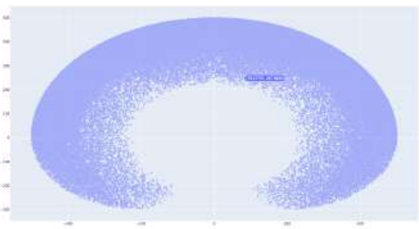


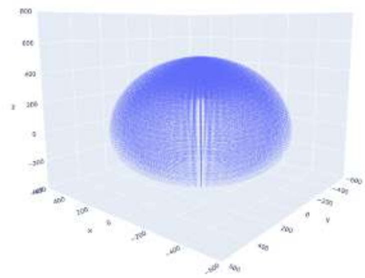
Figure 4.4: The configuration visualiser where it dynamically can be altered using a simple UI. The red rod shows the base rotation. The green, blue and purple rods show the joint segments. The red, green and blue dots show the points at (200, 0, 0), (0, 200, 0) and (0, 0, 200) respectively.



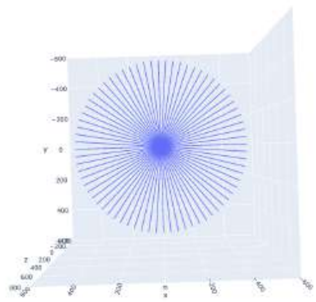
(a) The 2D envelope plot using a regular pattern.



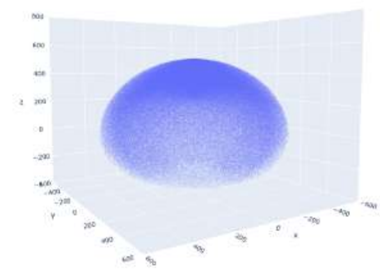
(b) The 2D envelope plot using a stochastic pattern.



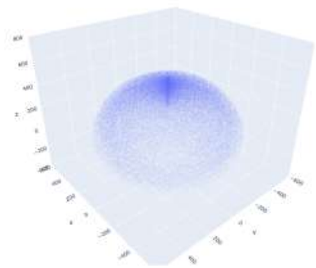
(c) The 3D envelope plot using a regular pattern as seen from the side.



(d) The 3D envelope plot using a regular pattern as seen from the top.



(e) The 3D envelope plot using a stochastic pattern with a high density of end-effector locations.



(f) The 3D envelope plot using a stochastic pattern with a low density of end-effector locations.

Figure 4.5: The 3D and 2D envelope visualiser with different settings.

Figure 4.6: The respective joint angles and their limits for the Arduino Braccio++ Robot Arm [Ard].

to be in the approximatly same position at end of the iteration. The FABRIK algorithm would give correct joint angles if all the joint types were of a ball-and-socket joint type. This however is not the case because the Braccio++ robot arm has a setup seen in figure 4.6[Ard] where certain joints are revolute joints with different rotational axis. Thereby the FABRIK algorithm is limited as follows. In our system, we initialize the rotation axis for the robotic joints by projecting the target's position onto the floor plane, which we define as the XZ plane. This projection is achieved using the 'Vector3.ProjectOnPlane' method, which calculates a vector on the floor plane by removing the vertical (Y) component of the difference between the target's position and the robot arm's position. The resulting vector, 'this.targetProjectedToFloor', is then normalized to ensure it has a unit length. To determine the precise angular orientation required for the robot arm to face the target on the horizontal plane, we compute the arctangent of the ratio of the X and Z components of this projected vector using 'Mathf.Atan2'. This function returns the angle in radians, providing an accurate measure of the rotational adjustment needed for the robotic joints to align with the target effectively. This approach ensures precise and efficient targeting by the robotic arm within the designated plane. The final angle is then used to input the joint angle orientation of the base seen in listing 4.1. The next step is obtaining all the joint angle orientations once the base is rotated toward the target. This is done disregarding the potential twist applied to the segments and just calculating the angle between the current segment and the next segment seen in listing 4.2.

Listing 4.1: Base rotation angle from XZ plane projection

```
...
this.targetProjectedToFloor = Vector3.ProjectOnPlane(
    Target.position - RobotArm.position,
    Vector3.up
).normalized;

Mathf.Atan2(
    this.targetProjectedToFloor.x,
    this.targetProjectedToFloor.z
)
```

```
    ...
```

Listing 4.2: Rotation angles from joint positions

```
float ShoulderRotation = Vector3.Angle(
    ShoulderSegmentVector,
    desk.transform.position
);
float ElbowRotation = Vector3.Angle(
    ElbowSegmentVector,
    ShoulderSegmentVector
);
float WristRotation = Vector3.Angle(
    WristSegmentVector,
    ElbowSegmentVector
);
```

Next, we need to make sure that the angles provided by the ZeroPGT code also correspond to the possible angles seen in 4.6. This is done through a simple conversion done in ZeroArm seen in the example listing 4.3. This is C++ Arduino code that runs on the the robot arm itself. Do note that some precision is lost because the entire range of 360° is mapped to the reduced range of 315° from the robot arm itself. This deadzone was slightly circumvented by mapping the 315° to 360° range to the edge of the deadzone. This problem however is further explained in Section 5.3.5.

Listing 4.3: Convert To Braccio Joint Angles

```
float convertToBraccioBase(float angle) {
      float min = -180;
      float max = 180;
      float braccioRange = 315;

      float fromMinusToZero = angle - min;
      float clamped = fromMinusToZero > braccioRange ? braccioRange :
          fromMinusToZero;
      float reversed = braccioRange - clamped;

      return reversed;
}
```

4.4 Command Interface and supported custom libraries

Having established a method to control the robotic manipulator, the next logical step is to develop a command interface. This interface transmits commands from ZeroPGT to ZeroArm via a wired connection. We will first analyze the design choices of the overall system, followed by a detailed examination of the internal command structure and its processing.

4.4.1 Bridging the Gap Between ZeroPGT and ZeroArm using an AAR

Research initially focused on connecting Unity game and device simulators on MacOS and Windows directly to the robot arm, enabling command transmission from the developer's device rather than a dedicated HMD. This approach appeared straightforward using the Unity extension Ardity [Wil21], which provides access to COM ports on these operating systems, facilitating command transmission to the robot arm. However, establishing serial communication with dedicated HMDs proved more challenging. Both the Magic Leap 2 and Meta Quest 2 are Android-based headsets, and the Ardity extension did not support these platforms, resulting in failed connections and command transmission to the robotic manipulator.

The initial design phase investigated a wireless connection method, including a setup where the Meta Quest 2 headset connected remotely to an access point configured on the Arduino Board. The ZeroPGT application communicated with the Arduino Board through HTTP requests to a designated IP address. However, this method exhibited latency issues, with command delays occasionally exceeding one second, making it unsuitable for rapid movements. Consequently, the approach shifted to a wired connection, which was ultimately chosen for its reliability. The Meta Quest 2 headset's capability to send debug logs to a developer device facilitated the use of a bash script to interface with the robot arm. Despite this, latency concerns persisted, with delays consistently above one second. Various alternative connection methods were explored but did not provide satisfactory results and are thus not detailed further in this section.

To identify the root of the problem with the Ardity extension, it was discovered that the application only supported desktop operating systems because they provided specific access to COM serial port interfaces within the C# Unity environment. Android, however, does not expose this interface in C#. To address this, a small Java library was developed to expose the serial interface on Android-based HMDs. This library was created by reverse engineering the USB Serial Terminal application[Mor19] and its GitHub repository[Kai], demonstrating that connectivity to the Arduino robot arm was achievable when sideloaded onto an HMD. Further development involved consulting the official Unity documentation on Android Archives (AAR)[Unia] and Java Archives (JAR)[Unib], as well as the Android Studio Developer Documentation[Stu]. Informal tutorials also assisted in refining the library[Voi21][Keo18]. The final system dynamically switches behaviors based on the environment using C# preprocessor directives. The *Write* function from the Java library is shown in code listing 4.5, and it is invoked in C# Unity as depicted in code listing 4.4. While additional setup and USB serial communication code is present, it is too detailed to include in this thesis. The application flow is illustrated in Figure 4.7.

Listing 4.4: The call to the AAR in C# in the UsbSerialCustom class

```
...
void Start()
{
    InitializePlugin("com.hoho.android.usbserial.util.UnityUsbSerial");
}

void InitializePlugin(string pluginName)
{
    unityClass = new AndroidJavaClass("com.unity3d.player.UnityPlayer");
    unityActivity = unityClass.GetStatic<AndroidJavaObject>("currentActivity
        ");
    _pluginInstance = new AndroidJavaObject(pluginName);

    if (_pluginInstance == null)
    {
        Debug.Log("Plugin Instance Error");
    }
    _pluginInstance.Call("receiveUnityActivity", unityActivity);
}

public void Write(string line)
{
    if (_pluginInstance != null)
    {
        _pluginInstance.Call("Write", line);
    }
}
...
```

Listing 4.5: The Java Write Function for the HMD to robot arm communication.

```
public void Write(String line) {
        try
```

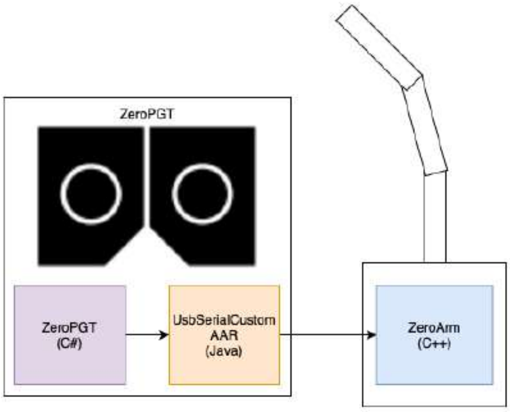


Figure 4.7: The overal flow of the commands from ZeroPGT to ZeroArm.

```
        {
            System.out.println("Writing to Robot Arm");
            port.write((line+"\n").getBytes(), WRITE_WAIT_MILLIS);
            port.setBreak(true);
            Thread.sleep(100);
            port.setBreak(false);
        }
        catch (Exception e)
        {
            System.out.println("Exception Writing line to Robot Arm");
        }
}
```

4.4.2 Internal Command Structure

Now that communication between the Head-Mounted Display (HMD) and the robot arm is established, we will examine the command structure of the commands being sent. The final angles for the robot arm are computed using the customized FastIK and need to be transmitted to the robot arm. There are two versions of the command available. The first version involves a complete reconfiguration of the robotic manipulator. The command, which is a string sent from ZeroPGT to ZeroArm, is structured as *B0 S0 E0 W0 T0*. In this structure, the letters represent the joint specifiers (*B* for base, *S* for shoulder, *E* for elbow, *W* for wrist pitch, and *T* for wrist roll), and the numbers following these letters indicate the angles in degrees. This command format requires conversion as detailed in code listing 4.3. In the example command, all joints (base, shoulder, elbow, wrist pitch, and wrist roll) are set to 0°. During system operation, these angle specifiers can be adjusted within the limits of the Arduino Braccio++ joint angles. If an angle outside this range is specified, it is clamped to ensure it remains within the allowable range. This approach not only simplifies debugging but also prevents potential damage to the robot from invalid angle inputs.

The second version involves a partial configuration change, where only a subset of the joint specifiers from the previous command is sent to the robot arm. Unspecified joints remain in their last known positions, while the joints with specified angles are adjusted accordingly. This partial configuration capability allows the robot to handle multiple commands concurrently. For instance, if a command changes the base orientation from 0° to 180°, the robot will move towards this target angle. During this motion, another command, such as *S90*, can simultaneously update the shoulder joint to 90°. This capability allows for flexible adjustments during complex motions. In simpler cases, if a new command attempts to change the base orientation while the base is already moving to a target angle, the commands are queued. This queuing behavior is managed by the Braccio++ library integrated with the robotic manipulator.

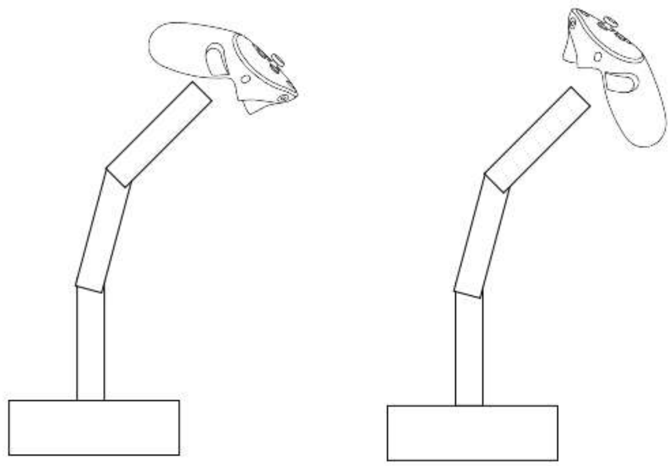


Figure 4.8: The left shows the orientation of the wrist roll 0°. This however would imply that the controller is consistently upside down. Therefore this wrist roll is flipped to −180° to achieve the right side up orientation seen on the right.

Additional supplementary commands are also defined. The reset command, denoted as *R*, configures the robot arm to a position similar to *B0 S0 E0 W0 T0*, as previously described. However, a modification is made to the *T* (wrist roll), which is set to 180° to position the controller more naturally, as illustrated in Figure 4.8. Earlier designs considered adjusting the wrist roll based on context, such as rotating it towards the user to facilitate controller handling or aligning it with the zero gravity PGT simulation in the HMD. These options were ultimately discarded to avoid adding unnecessary complexity for the user. Additionally, visual realignments of the virtual PGT with the robotic manipulator controller were considered but will be further analyzed in Section 4.6 and specifically Subsection 4.6.2. However, implementing these orientation changes and visual realignments would have been cumbersome and could significantly reduce the realism and usability of the system.

The reset command is triggered automatically at each startup of the Arduino code or manually via an application interfacing with ZeroArm. Additionally, a button on the robot arm's Carrier Board is assigned to this command. Specifically, the joystick functions as this button; when pressed, it virtually activates the reset button displayed on the LCD screen. This reset button can be used in emergencies to move the robot arm to its home position and pause command reception for a specified duration. Although it was considered to trigger the reset command when the robot arm crossed a path where the participant's hands might be, this solution proved impractical. Computing such paths within the ZeroPGT environment was complex, and frequent activations could negatively impact the application's realism. A more practical solution was implemented: the instructor can manually disable the ZeroArm functionality if it is deemed too dangerous to proceed.

The next command specifier is the angular velocity, which can be set using commands such as *A0*. Manipulating angular velocity is crucial during development, as reducing it can minimize the risk of damaging the robotic manipulator, especially when the system's code is incomplete. However, dynamically adjusting velocity during system operation is also important to maintain realism. If the virtual PGT moves at a different rate than the physical controller, the experience may feel less realistic. Additionally, the controller does not always follow the exact path of the virtual PGT due to the robot arm's blind spots and its need to perform a 315° turn when reaching the limit of its angular range. Higher velocities could help cross these singularities more quickly. Despite this, increasing angular velocity comes with risks, including reduced system longevity and potential harm to participants. Therefore, the decision was made to forgo dynamic velocity adjustments in favor of a consistently lower speed setting. This approach fosters greater trust during the User Experience Tests and ensures a more controlled, albeit slower, simulation that still achieves a high level of realism.

4.5 End-Effector Fabrication

A significant portion of the implementation effort was devoted to the end-effector design. This design focuses on creating a mechanism for the controller's end-effector that includes an attachment module capable of securely attaching and detaching objects during system operation. Although this seamless attaching and detaching mechanism is currently utilized mainly for development, repair, maintenance, and setup, it holds considerable future potential. This section will first detail the design levels for both the robot end-effector and the tooling, which incorporate the seamless attachment mechanism. Subsequently, we will examine the various tools considered for this system.

4.5.1 End-Effector and Tooling Iterations

The robotic manipulator in the final implementation did not necessarily require an object at the end-effector. If the system had incorporated VR space extensions with encountered-type haptic feedback, the addition of a toolless end-effector would have significantly enhanced the haptic experience. However, we decided that incorporating tooling would offer greater versatility, as it would allow for faster design modifications. Specifically, male and female modules could be designed independently if an attaching mechanism were implemented, making the system more flexible by allowing tools to be attached and detached during operation. Note that the documentation of this design is not presented chronologically but has been reorganized for a more coherent reading experience. In reality, the design of these components was developed and iterated simultaneously.

The first module to be discussed is the end-effector design, named the grip latch 4.9. Our primary goal was to enable attachment of objects to the end of the robotic manipulator's last link. Fortunately, a simple screw was included with a plastic module that could secure an object in place while allowing it to rotate via the wristroll servomotor. The gripper servomotor, shown in Figure 4.6, which originally facilitated the gripping mechanism in the as-packaged tooling, was not used in this research. A 3D model was created based on the plastic grip latch to allow additional items to be attached. In subsequent design iterations, the end-effector module was modified to be easily removable. Earlier designs had obscured the screw by using superglue to attach another part over it. To address this, an offset bridge was introduced. This bridge allowed other items to be affixed using superglue while preserving access to the screw location. Additionally, the offset bridge improved the alignment of the attachment mechanism, centering many of the final toolings.

4.5.2 A Seamless Attaching Mechanism

The next module to discuss is the tooling grip. Our design was heavily influenced by the idea of integrating a virtual controller into this module. This addition would provide the system with valuable data, such as location tracking and hand-on-controller detection, and in future iterations, enable the tooling to be detached and used similarly to a virtual Pistol Grip Tool (PGT) due to its pistol-like design. Although the servomotors already detect their position using PWM signals, expanding the custom library to include bidirectional communication was necessary, as discussed in section 4.4.1. While both solutions had merits, the controller input offered additional benefits in various areas. In the final design, however, only the hand-on-controller detection functionality was utilized. Two main designs are considered for this module: a controller mount for the Magic Leap 2 and a controller mount for the Meta Quest Pro. These designs were created by scaling the 3D models of the controllers and using Boolean operations

(a) The 3D-printed grip latch, which rotated according to the wrist-roll servomotor, could be modified to accommodate additional attachments.

(b) The top part of the image shows the grip latch, including the screw and plastic insert. The bottom part displays a female magnet holder with a magnet inside.

Figure 4.9: The fabricated grip latch.

to form their holders. Each mount includes a circular hole and a flattened area to attach the seamless attachment mechanism.

In designing the end-effector and tooling modules, it was crucial to incorporate a seamless interconnection mechanism. Initially, we employed a basic interlocking system using pegs and holes in both the male and female components of the modules. While straightforward, this approach proved insufficient as it did not ensure a secure connection during the robot arm's movement. Hence we developed a more robust system. Given the limited engineering experience at the time, we relied heavily on a trial-and-error approach. Achieving the right balance of friction to allow for seamless sliding while ensuring a secure fit was a fine-tuning process. Multiple versions of the design, with slight adjustments in dimensions, were printed and tested, sometimes in the same print run. An early consideration was a clicking mechanism to enhance safety. However, it was eventually abandoned due to the need for additional user interaction—a two-handed operation—which would compromise the mechanism's seamlessness. Additionally, implementing such a complex mechanism was challenging given the level of experience in product design.

Another mechanism considered was Velcro. While it was simple and easy to apply to the end-effector and controller mount modules, it had several significant drawbacks. First, the Velcro was adhered to the modules using superglue, which proved unreliable. Additionally, the adhesive strength of Velcro diminished with use. The final system devised is a magnet system with two neodymium magnets for each module. Small magnet holders were 3D printed to securely house the magnets and could be attached with superglue. The rigidity of the magnet holders improved the reliability of the adhesive bond. Neodymium magnets also retained their strength longer than Velcro, making magnetic degradation negligible. A small design iteration was made to enhance the magnetic forces by minimizing the gap between the magnets. In an early iteration, the barrier was set at 0.16mm, corresponding to the layer thickness of the 3D printer. This design suffered from stringing artifacts that reduced the barrier integrity, preventing the magnets from directly touching for around 300 uses. In a newer version, the barrier thickness was increased to 0.32 mm in later versions and printed with a support structure. This adjustment

Figure 4.10: The male and female magnet holders are shown in the image. The red components are the holders, with the top part representing the male version and the bottom part representing the female version. For scale, a single metallic magnet is included in the images, although each holder contains two magnets for proper functionality.

improved the solidity of the barrier while maintaining flexibility for emergency tool removal. Both these parts are visible in Figure 4.10.

4.5.3 Different Toolings and their Characteristics

For this research, the tooling selected was primarily a controller mount tailored to the HMD in use. However, other tooling options were considered, each with its own advantages and disadvantages. One initial idea was to use a styrofoam ball as the end-effector item. This choice would have been advantageous due to its light weight, which is well within the payload capacity of the robotic manipulator, and its isotropic nature, allowing the robotic arm and the participant to approach it from any angle without affecting performance. Despite these benefits, using a styrofoam ball would significantly reduce realism. The primary drawback was that users would not interact with a more realistic tool, such as one used in an Extra-Vehicular Activity (EVA), which could detract from the immersion and training value of the simulation.

Another consideration was to create a more realistic artifact by replicating the virtual Pistol Grip Tool (PGT) in the real world. A 3D model of the PGT was decomposed for 3D printing and then reassembled using superglue. Although the model was highly realistic, its weight rendered it unmanageable for the robotic manipulator, making it an impractical option. To address this, a lighter version of the realistic tool was developed. The new design used the 3D printed PGT as a guide to cut out shapes from styrofoam for the lighter PGT version. The reduced rigidity was addressed by applying a papier-maché shell and securing the parts with toothpicks. While this approach allowed for easier replacement of interconnected parts, the final design was not used due to time constraints, the reduced work envelope of the robotic manipulator and the sytrofoam tool's fragility. Both of these physical tools could be seen in Figure 4.12.

(a) The controller mount, indicated in red, now holding the Meta Quest controller.

(b) The controller mount for the right Meta Quest or the Meta Quest Pro controller with a male magnet holder superglued to the end.

(c) The controller mount, now holding the Meta Quest Pro controller.

(d) The controller mount slanted on its side to see the magnet hole for the controller mount and the male magnet holder.

Figure 4.11: The controller mount seen from multiple angles with different configurations.

Figure 4.12: The 3D printed PGT at the top left and the styrofoam PGT at the bottom right. The 3D printed PGT includes a slide and lock magnet mechanism for easy lateral magnet removal.

4.6 The Virtual Robot Arm and its Alignment

One of the crucial aspects of this project is aligning the virtual robot arm with the physical robot arm on the table. Proper alignment is essential for providing an accurate and immersive experience. This section briefly outlines the methods considered for alignment and explains the chosen approach.

4.6.1 Different Alignment Methods

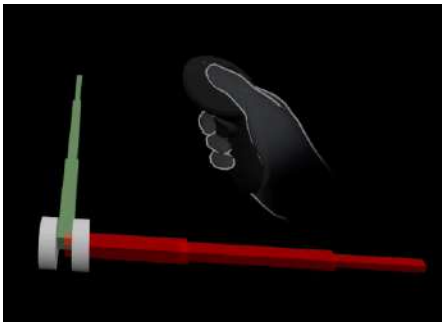
The first alignment method considered involved using the coordinate system of the Arduino robot arm, as described in subsection 4.3. This approach would require aligning the robot arm's coordinate system with the ZeroPGT Unity application's coordinate system using OpenCV. However, this was circumvented by implementing the inverse kinematics (IK) methodologies directly within the ZeroPGT Unity application, thus avoiding the need for this alignment. The next alignment task was to match the physical robot arm with the virtual robot arm. Given that the physical robot arm is stationary and mounted, alignment could theoretically be achieved by adjusting its position to match the virtual robot arm. However, we opted to adjust the virtual robot arm instead, as this approach allows for easier modifications of height, rotation, and overall location compared to moving the physical robot arm.

The first alignment method considered was the automatic alignment. This could be achieved by mounting a Heroku marker to the physical robot arm and than scanning said marker with the HMD with the onboard cameras. Whilst this would have been the easiest for the user, the implementation of this would have taken a considerable amount of time and was replaced for a rudemantery alignment method explained in the further paragraphs. This implementation itself is not all that complex but the integration with the customized FastIK algorithm and its conversion would have to be generalised for any random location in the coordinate system. The dynamic virtual moving of the FastIK chain link was however not supported. We therefore agreed that because of the limited set of time, the alignment would have to be done manually by the user with the fact that it could potentially impact the performance of the robot arm accuracy. This would essentially not move the chain link considering this would use the recalibration feature provided by the HMD.

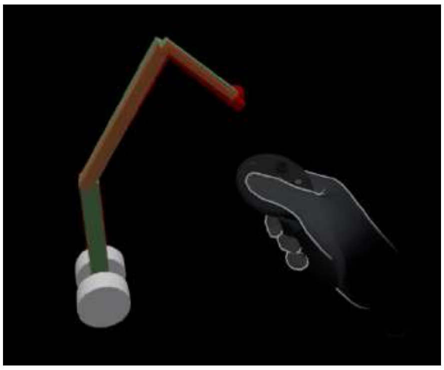
The manual alignment procedure is detailed in Section 5.2.2, where the execution of the ZeroPGT application is described. The alignment process begins with the user pressing the trigger on the right controller and touching the table, with the tip of their controller, where the robotic manipulator is mounted. This action aligns only the height of the virtual robot arm with the physical robot arm as seen in Subfigure 4.13a. The next stage involves calibrating the position by using the recalibration feature of the home button, as outlined in Section 5.2.2. The link sizes and scale of the robot arm are preconfigured to match those of the physical robot arm. Subsequently, the user aligns the rotation of the robot arms using the recalibration functionality. This is facilitated by the fact that both the virtual and physical robot arms are oriented towards a common location along the z-axis, making any discrepancies in rotation easily visible to the user as seen in Subfigure 4.13b.

After completing the alignment stage, participants proceed to the Movement Test stage. In this stage, they manipulate the inverse kinematics using the red sphere on their right controller, as illustrated in Subfigure 4.13c. This allows them to control the robotic manipulator, understand its work envelope, learn about static rotational velocities, and recognize its limitations. The virtual robot arm aligns its end-effector with the red virtual sphere, guided by the right controller, while the physical robot arm mirrors this alignment. The passthrough system of the Meta Quest 2 headset, which is dynamically activated during this phase, facilitates this alignment. The passthrough view will be deactivated when the simulation progresses to the ISS stage. Just before reaching the ISS stage, the robot arm is halted to allow for easy mounting of the right controller onto the robot arm using the seamless attaching mechanism.

(a) Height Alignment: The height of the robot arm is adjusted by pressing the trigger. The height of the virtual robot arm then follows the height of the right controller. The tip of the controller needs to touch the table on which the physical robot arm is mounted.



(b) Rotation Alignment: The green robot arm represents the virtual robot arm, while the red robot arm, now shown for demonstration purposes, denotes the physical robot arm. During the actual alignment process, the physical robot arm is visible through the passthrough view and hence the red robot arm is not there. Users can observe any angle discrepancies and use the home button to align the virtual robot arm rotationally with the physical robot arm.



(c) Movement Test: Both the virtual (green) and the physical (red) robot arm follow the red target at the end of the user' right controller.

Figure 4.13: Each participant performs the manual alignment themselves. Note that the Meta Quest does not permit recording of the passthrough view due to privacy concerns, resulting in images with a black background. In practice, participants will see the real world through the passthrough view, which aids in accurately aligning the virtual and physical objects. Hence additional visuals are provided to enhance the clarity of the figures such as the red robot arm.

4.6.2 Nudge Towards Rotation Envelope

To enhance the accuracy of haptic feedback in the simulation, crucial software alignment improvements have been implemented. A key development involved aligning the virtual PGT with the physical controller, addressing a critical misalignment where the fixed rotation of the physical controller led to discrepancies in feedback, as illustrated in Figure 4.8. To achieve better alignment, the virtual PGT was adjusted to correspond with the natural alignment of the robot arm's end-effector. This adjustment balances the need for realistic feedback with the alignment of the virtual and physical components. Additionally, the alignment of the virtual PGT is dynamically adjusted based on the proximity of the user's hand to the end-effector, thereby optimizing the user's interaction experience while maintaining system realism. The alignment was governed by an exponential decay formula, which is given by

$$0.1 \times \exp\left(\frac{1}{\text{distance} - 0.1}\right)$$

where the distance variable represents the distance between the end-effector and the closest tracked hand. This ensures that the closer the hand is to the end-effector, the full 180° rotation is achieved with an offset of 0.1. Specifically, this means that the full rotation is completed when the hand is at a distance of 10 cm or less from the end-effector. However when the physics project settings were changed, discussed in Section 3.2.3, this function did not work as well as before and hence was replaced by a piecewise function as seen in listing 4.6. This piecewise function works as follows. When the distance of the hand from the end-effector is less than $20cm$ or 0.2, the full rotation gets achieved (in this case a hardcoded value as 360°). If the value is in between 0.2 and 0.5 or $20cm$ and $50cm$, a linear decrease from 180° to 0° gets followed.

Listing 4.6: Base rotation angle from XZ plane projection

```
return
    (x >= 0.0f && x <= 0.2f) ?
        360.0f :
        (x > 0.2f && x <= 0.5f) ?
            180.0f - ((x - 0.2f) * 600.0f) :
            throw new ArgumentOutOfRangeException("x", "Value must be
                between 0.0 and 0.5.");
```

It should be noted that this nudge rotation of the virtual PGT does not affect the rope simulation or any other physics simulations. Consequently, the rope endpoint connected to the virtual PGT remains stationary on the virtual PGT, as if it were not rotating, while the tool itself visually rotates. This approach prevents the physics from becoming jarring when interacting with the virtual PGT. Additionally, the mesh of the right hand is disabled when it approaches the end-effector. This measure was implemented to avoid noticeable misalignment discrepancies that occurred when the hand mesh was still rendered. While haptic retargeting could be considered a potential solution, hiding the hand mesh while preserving functionality was considered a sufficient compromise.

Chapter 5

User Experience Study for Validating ZeroTraining

Having thoroughly explored the implementation of the ZeroTraining system, it is now essential to evaluate and validate it. This study aims to evaluate the effectiveness and user experience of ZeroTraining through a structured User Experience Test (UET). By systematically gathering and analyzing user feedback, we can uncover key insights into how the system performs in real-world scenarios. This chapter details the purpose and goals of the study, the design of the setup and questionnaire, and the methods employed to analyze the data collected. The outcomes will inform potential refinements and optimizations, ensuring ZeroTraining meets the needs and expectations of its users.

5.1 Defining Objectives and Vision

It is crucial to first define the objectives of this User Experience Study. The main argument could be that this study involves a comparative analysis of two methods to determine which is better based on factual evidence. However, this is not our primary aim. Instead, our focus is on assessing the feasibility of achieving an encountered-type haptic feedback system. To reiterate, an encountered-type haptic feedback system seeks to enhance the realism of a virtual environment by incorporating external forces, such as those provided by our robotic manipulator to create some sort of haptic feedback, in line with the virtual world. In essence, the goal of this study is to evaluate how well this system improves realism, rather than comparing different methods. Participants will use their knowledge of how vision and tactile feedback interact to assess the system. The performance of an interaction is not a focus of this study. We are not concerned with metrics such as speed, efficiency, creativity, or other benchmarks. The focus is on determining whether the simulation effectively integrates with the participant's sensory experience, rather than evaluating the participant's performance.

At first glance, this might seem challenging because, upon closer examination, it becomes clear that most participants likely have no experience with a zero-gravity environment. Is it fair to ask them to assess the realism of an application designed to emulate such an environment? This issue is somewhat highlighted during the tests themselves. We also anticipate that some participants might make assessments based on their assumptions about zero gravity rather than focusing solely on their actual experience. Despite this, it appears that valuable insights can still be obtained without full knowledge of weightlessness. There are many other aspects of realism that users might perceive.

Firstly, the visual aspects of the environment—such as lighting, shadows, texture detail, 3D model accuracy, skybox detail, and potential limitations of the VR headset like field of view and screen door effect—are interesting yet relatively less significant for this study. More crucial for our study are the rigidbody simulations. Participants might focus on questions like: How do the collisions work? Does hand tracking function adequately? Are the objects moving in a realistic manner, even in zero gravity? Does my virtual hand align with my real hand? Does the object I'm supposed to feel match what I see in front of me? Although these questions are related to the zero-gravity environment, they can still be answered abstractly, even if participants lack knowledge of how objects should behave in zero gravity.

The fact that the object is in zero gravity may simplify the analysis of this encountered-type haptic feedback system. In a virtual environment with gravity, the simulation would need to be orders of magnitude more complex. It would have to account for the object's weight, its fall when dropped, and potentially the surface it lands on, to give some examples. In zero gravity, however, an object only maintains the velocity imparted by the user if it spawns as stationary, making extreme velocity adjustments unnecessary—this was a key design decision. In a gravity-affected environment, the object's acceleration due to gravity would create more challenging conditions, especially for simpler hardware like our robotic manipulator. Additionally, if an object does not behave as expected in zero gravity, users might attribute discrepancies to their unfamiliarity with such an environment, even virtually, which could aid in their suspension of disbelief. Any such discrepancies should however be reported. The final results of this study could provide valuable insights into whether this proof of concept demonstrates the feasibility of the project on a larger scale, based on the results achieved with lower-grade hardware.

5.2 Designing the Study: Setup and Questionnaire

Designing an effective study to evaluate the ZeroTraining system requires careful consideration of various elements, from the overall setup to the specific questions that will elicit meaningful feedback. The goal is to create a framework that not only captures the essence of user interactions but also provides insights into their experiences and perceptions. This section outlines the design process, starting with the selection of relevant interactions, followed by the participant screening process, and culminating in the development of a comprehensive questionnaire. Each step is crafted to ensure that the data collected will be both reliable and insightful, guiding future improvements to the system.

5.2.1 Setup

First let us start with explaining how this User Experience Study is set up. What we first need to consider is what the user should be able to do in our virtual environment. The interaction itself should not be extremely complex as it would prevent the user from exploring the system itself. It should also be of an exploratory nature as too much repetitiveness would make it mundane and uncreative. The participant should be asked and able to do the task in multiple ways while simultaneously learning how well the system works and what realistic aspects are satisfactory. We will also not directly define a stringent hypothesis. This is because in the timespan provided, we are not capable of proving or disproving such a hypothesis using the required amount of participants and a statistical analysis. Therefore we will define a somewhat lenient objective of these tests similar to the objective readable in Section 1.

The chosen interaction involves touching an object in the virtual environment, specifically the virtual PGT as previously mentioned. Users will first enter an exploratory phase, where they can investigate the capabilities of the zero-gravity virtual tool. They will then be asked to perform a series of touches on the tool. After touching or feeling the tool, users will move their hands away, causing the tool to respawn at a different location and orientation. They will then touch the tool again. This interaction was selected because it is inherently simple yet accommodates a variety of scenarios. These scenarios include the tool's spawning location relative to the user, the tool crossing dead zones, its different rotations, the various angles of attack by the participant, and the testing of the robot arm's singularities, such as positioning

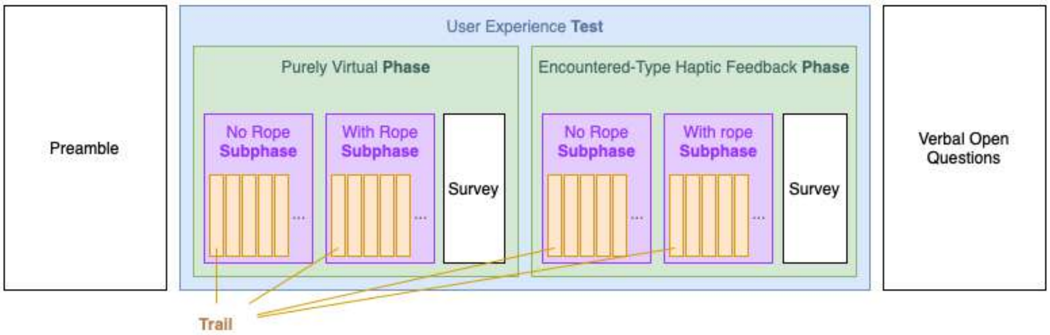


Figure 5.1: The terminology and denominations of a single UET including the preamle and verbal open questions.

the tool directly above the base of the physical robot arm. Additionally, the distinction between a rope and a ropeless version will be considered. The rope is simulated in zero gravity, and the user's hand will interact with it through virtual collisions. This setup allows the user to influence the object without directly touching it.

5.2.2 Execution of ZeroTraining

The execution of ZeroTraining will be conducted in two main phases, each consisting of two subphases and further divided into several trials. The primary phases are: the purely virtual phase, where no robotic manipulator is used and collision response is enabled, and the encountered-type haptic feedback phase, which involves the use of a robotic manipulator and does not utilize collision response. The subphases, which are digitally similar, include a ropeless subphase and a rope subphase simulated in $0G$. As described in Section 3.2.4, the application operates in stages for its setup. The trials, which are the smallest unit of our test, are detailed in the following sections. This structure is visualized in Figure5.1. Participants are not required to complete a fixed number of trials. Instead, the instructor will determine the number based on factors such as time, comfort of the user and exploration done the participant. Typically, participants complete between 10 and 25 trials, with some extreme cases reaching up to 50 trials. To reduce the learning effect, the order of the phases is randomized. Without randomization, participants might prefer the purely virtual phase if they learn about features like collision response and then find them missing in the robotic manipulator phase, potentially leading to disappointment. By employing a Latin square design, where half of the participants start with the purely virtual phase and the other half with the robotic manipulator phase, we mitigate this first impression bias.

All the tests were executed in around one hour and fifteen minutes giving the users about the same amount of time of exploration with the system. The preamble included the signing of a consent form seen here[1], the clarification what we will be doing on a high level such as not a performance review but an evaluation of our system, the controller explanation of the buttons we will be using outside of VR and a simple UI explanation of what our application app icon looks like. Additionally the external camera that will be recording is indicated, the fact that a screen recording is made will be explained and a reminder that the halting of the UET can be requested at any time without penalty.

[1] https://docs.google.com/document/d/1Gm4jHLuo8A-EKkMQHQvfFmKIUPFaGw26qiYVXPxN_k4/edit?usp=sharing

Purely Virtual Phase

The first thing to teach the user is how to press the home button. This button is crucial as it will be used extensively during the application. It allows the user to access the home screen to quit the application and to recalibrate the virtual environment. Recalibrating lets the user recenter the world, which will be very useful later. Both interactions are respectively activated by pressing the button to access the home screen and by holding the button to recalibrate the environment.

Next, the user places the headset on their head and adjusts it for comfort and clear vision. They will start at the Meta Quest 2 home screen, outside of our application, and can get acquainted with the headset's basic black-and-white passthrough system. The next step is to guide the user to open the application on their own, using the button information they learned earlier. This simple task is important for several reasons. It helps the user get comfortable with basic UI interactions, which is a more manageable first step than immediately diving into a complex physics simulation with hand tracking. It also trains the user to open and close the application, a necessary skill for later stages.

Once the user opens the application, a text panel appears on the left side of their screen, indicating the current stage and providing additional status information, such as hand positions. This textual information is primarily intended for development purposes and to assist the instructor. Given its abundance, it may be cumbersome for users to remember. Therefore, the preferred method of communication is to provide information directly to the user, with the textual information serving as a backup if needed.

The a subsection of the alignment, namely the height alignment, is then completed as described in Section 4.6. In the next stage, the user will enter a purely virtual environment: the passthrough view will be disabled, and the Milky Way skybox and the ISS will be displayed, as shown in Figures 3.8 and 3.7. The PGT tool will also be visible, floating in front of them. During this exploratory stage, the instructor will take the right controller from the user, and the system will automatically switch to hand tracking for the right hand. The user's tracked right hand can then interact with the PGT tool by touching it. Realistic collision responses will be simulated as the colliders on both the PGT tool and the hand interact, with these colliders illustrated in Figures 3.9 and 3.10.

Once the user is done exploring the interaction and is sufficiently comfortable with the simulation, they can move on to the testing stage. This stage is set up as follows. The PGT tool will spawn in a random location with a random orientation as a stationary object. The user will then have to touch the PGT tool. The text, visible to the right of the participant, will give additional instructions like 'Touch the tool' and 'Move away from area'. So once the participant has touched the tool the text will say 'Move away from area' which implies that the user will have to remove both their hands from the work envelope of the simulation. There are many reasons for the user to have to do this interaction which will be explained in the next paragraph and in the following chapter. Once the user has done this interaction, the PGT tool will oncemore spawn in a random location. The participant will hence have to touch the tool again. Every time such a trail is executed, the 'Touches remaining' counter decrements. The participant will be asked to do a certain amount of touches.

The user will be asked to remove their hands from the work envelope for several reasons. First, this approach allows the user to perceive the interaction with the PGT tool for as long as needed. It is even possible for the user to touch the tool multiple times, try to grab it, etc. Once the user is satisfied with the current trail, they can remove their hands from the envelope. This method is more beneficial than using a timer because the user would not be able to control the timer intuitively, leading to a potentially jarring reset effect. Additionally, requiring the user to move their hands away helps prevent the tool from spawning too close to or, in extreme cases, inside the user. Although this spawning behavior might seem unnatural, segmenting the interaction into trials is necessary to evaluate the realism of the environment effectively.

The location and orientation of the PGT tool are controlled by setting the random number generator's seed to the same value each time the program restarts. This ensures that different versions of the test are comparable and allows the developer to create a test with specific extreme cases by selecting the appropriate seed. Finally, asking the user to move away from the tool ensures that they travel a significant distance before the tool respawns. This prevents users from simply holding their hand near the previous tool location and then translating to the next spawn location, thus experiencing a variety of angles and interactions with the PGT tool and thoroughly testing the system.

The user now has gone through a single run of the ZeroPGT training. The user will be asked to close the application and to reopen it. The alignment is then done again, but many alignment settings will be saved between phases. Once the user now hits the exploration phase, they will be asked to enable the rope attached to a floor and the PGT tool. Once the user has explored this interaction sufficiently, they will perform the exact same testing stage as before, only now it includes a more erratic zero gravity rope, very similar to the "enchanted snake" problem addressed in the introduction of this thesis 1.

The user will complete a single run of the ZeroPGT training and then be asked to close and reopen the application. A subsection of the alignment process, namely the height, will be repeated, though many alignment settings will be preserved between phases. In the exploration phase, the user will be instructed to enable a rope, which is attached to both the floor and the PGT tool. After thoroughly exploring this interaction, the user will undergo the same testing stage as before, but with the added complexity of a more erratic zero-gravity rope, akin to the "enchanted snake" problem discussed in the introduction of this thesis 1.

Encountered-type Haptic Feedback Phase

To begin the encountered-type haptic feedback phase, a full alignment must be completed. This alignment always precedes the stage where the robotic manipulator is in use. Since we are now discussing the encountered-type haptic feedback, modifications were made to both the exploration and test stages. In this phase, collision response is disabled because the colliders attached to the hand are turned off. This change is intended to prioritize the user's feeling of the controller in their hand without interference from virtual collisions. However, collision detection for the rope simulations remains enabled, as no encountered-type haptic feedback is provided for the zero-gravity rope. Since the system cannot automatically detect when a real haptic touch occurs, the user must manually indicate, by pressing the left trigger, that they have found, felt, touched, grabbed, or otherwise encountered the physical controller. Additionally the "Nudge Towards Rotation Envelope" is also introduced here, as described in Section 4.6.2.

Now it also becomes apparent why users are asked to remove their hands from the work envelope each time they touch the tool. This practice is essential for preventing accidental collisions with the robotic manipulator while it moves to the starting location of the next trial. Maintaining this requirement across both stages helps to preserve consistency, reducing the need for users to relearn different procedures. Consequently, this approach minimizes the adjustment time and allows users to focus more on perceiving the realism of the simulation.

The subphase where a rope is enabled is reintroduced for two reasons. First, during an EVA (Extravehicular Activity), most items are tethered to prevent loss, making this the most realistic scenario to include. Second, the rope affects the velocity of the virtual tool when starting a new trial. The inherent instability of the rope means the tool does not remain stationary but instead exhibits realistic motion. We found during development that this motion translates well to the movement of the robotic manipulator. This approach leverages our system's strengths, closely emulating the sensation of feeling the object move through space in sync with its visual movement. Hence, in this phase, a participant can track the object as it moves through space, in contrast to the previous phase where the object remained stationary.

5.2.3 Screening

Before the tests commenced, all participants completed a screening survey. This survey included demographic questions designed to inform the results and to exclude participants who might provide irrelevant data. Out of the 20 participants contacted, 17 completed the survey. The survey included the following questions, with brief explanations provided where necessary to clarify their relevance. Although the survey was conducted in Dutch, the questions are translated into English here, with the Dutch versions in parentheses. The survey was intentionally kept brief and somewhat informal to lower the entry barrier for potential participants.

- First name (Voornaam)
  This question had an open textfield where they could enter their first name for practical purposes.
- Sex (Geslacht)
  - Male (Man)
  - Female (Vrouw)
  - I would rather not say (Zeg ik liever niet)
- Age (Leeftijd)
  - 18- (jaar)
  - 19-24 (jaar)
  - 25-34 (jaar)
  - 35-44 (jaar)
  - 45-54 (jaar)
  - 55-64 (jaar)
  - 65+ (jaar)
- Do you often use a Virtual Reality headset? (Gebruik je vaak een Virtual Reality headset?)
  - Very often (Heel vaak)
  - Often (Vaak)
  - Sometimes (Soms)
  - Almost never (Bijna nooit)
  - Never (Nooit)
- Do you have visual impairments that could affect your ability to interact with the application? (Heb je zicht afwijkingen die je mogelijkheid om met de applicatie te interageren kunnen beïnvloeden?)
  - No (Nee)
  - Contacts/Glasses (Lenzen/Bril)
  - *Other* [user specified]
- Do you have physical impairments that could affect your ability to interact with the application? (Heb je fysieke afwijkingen die je mogelijkheid om met de applicatie te interageren kunnen beïnvloeden?)
  - No (Nee)
  - *Other* [user specified]

Questionnaire	Rejection Reason
SUS[Bro96]	Too general while being less all encompassing than the PQ.
MEC-SPQ[Vor+04]	Focused solely on spatial presence with irrelevant questions.
IPQ [SFR01]	Too general while being less all encompassing than the PQ.

Table 5.1: The additional questionnaires considered and why they were rejected.

The remainder of the survey provided details about when and where the tests would be conducted and allowed participants to specify their preferred timeframes. Out of the 20 participants invited to take the test, most were eligible. Ultimately, 10 participants completed the test, as significant qualitative results were obtained within the limited timeframe.

It is noteworthy that 90% of the participants who completed the test were aged 19-24, with one participant aged 25-34. Additionally, a little over half (60%) of the participants were male, while the remaining were female. No participants with significant visual or physical impairments were tested. A diverse range of individuals from various fields participated, including computer scientists, industrial engineers, film animators, chemists, and pharmacists. This diversity may influence the results, as many participants had varying expectations for the system. While many experts evaluated the system prior to these User Experience Tests (UETs), the participants in this study are primarily students. To see the full conducted survey[2], see the Google Form.

5.2.4 Questionnaire

A questionnaire was developed to be administered after each phase of our test. This questionnaire is derived from multiple sources and includes a consistent set of questions, though they may vary slightly depending on the context of the phase the participant is in. This facilitates comparing aspects such as presence and realism across different phases.

The primary focus of the questionnaire was on assessing realism. To this end, we employed the Presence Questionnaire [WS98], which is widely recognized for its comprehensive evaluation of realism in VR [Gon+21]. Research indicates that the Presence Questionnaire evaluates three key dimensions: perceived environment realism, involvement, and presence. In contrast, the System Usability Scale (SUS) [Bro96] evaluates only two dimensions: perceived environment realism and presence. This distinction is important because our research emphasizes the extension of haptic feedback rather than the fidelity of the virtual environment. Flexibility in the questionnaire was crucial, allowing for customization while maintaining standardization. Note that the subpart related to 'sound' was excluded, as it was not implemented in our setup. Although we considered various other presence questionnaires, as detailed in Table 5.1.

The second crucial questionnaire used was the Avatar Embodiment Questionnaire [GP18]. Based on the research, we focused on the most pertinent aspects: body ownership, agency, motor control, and tactile sensations. These aspects align closely with the objectives of our study. We aimed to evaluate whether the Meta Quest 2 effectively supports body ownership, spatial awareness, agency, and motor control. Establishing this baseline interaction is essential for assessing the effectiveness of the extensions made to our system. Aspects such as external appearance and responses to external stimuli were considered less relevant, as they primarily relate to the participant's self-perception and their reactions to virtual objects perceived as threats to their real body. To view the specific questions used in these questionnaires, we recommend consulting the cited research sources directly.

We also included an assessment of motivation and enjoyment using the Intrinsic Motivation Inventory (IMI) [RD85]. This assessment helps gauge how effectively the simulation engages users, as frustration could indicate issues with the simulation. Given the length of the survey, we excluded irrelevant subscales, specifically Perceived Choice and Relatedness. Perceived Choice

[2] `https://forms.gle/9ZejLM7CUtQ7Mefd7`

was deemed unnecessary since the task is exploratory by nature, and Relatedness, which pertains to interactions with other individuals, is not applicable here. The remaining subscales are Interest/Enjoyment, Perceived Competence, Effort/Importance, Pressure/Tension, and Usefulness/Value. Additionally, we included standardized task assessments such as the NASA Task Load Index (NASA-TLX) [RQ05], specifically the RAW-TLX variant, which omits the pairwise comparison hierarchical task analysis. This helps evaluate subjective Mental workload (MWL). We also briefly noted that participants should have some experience with similar tasks, as most people have considerable familiarity with interacting with objects in daily life. These aspects of the questionnaire, however, are considered supplementary. To reduce bias, all questions were randomized for participants. We strongly recommend reviewing the questions from the questionnaire, particularly the Avatar Embodiment questions. The selection of relevant questions for our study is particularly noteworthy and can be seen in our questionnaire. These questions included blanks that needed to be filled in, which could significantly influence the overall results of this research. To see the full User Experience Study questionnaire[3], see the Google Form used for both phases.

There is an ongoing discussion about how presence questionnaires are not robust enough to evaluate such an subjective metric[Sla04], [Uso+00], suggesting that we need to move away from questionnaires and adopt other methodologies as mentioned in [Gon+21]. Hence the second way to validating our system was via a set of open ended questions. These open questions were derived by the researchers themselves and were deliberately set up to be as open as possible. These open questions were then answered by the participant at the end of the test when both phases are completed. It is also important to realise that this information is highly subjective. This however is extremely important because of the inherent nascent stages of this type of research.

- Which positive and negative aspects were notable during the purely virtual version? (Welke positieve en negatieve dingen waren opmerkelijk bij de virtuele versie?)
- Which positive and negative aspects were notable during the robot arm version? (Welke positieve en negatieve dingen waren opmerkelijk bij de robot arm versie?)
- Were there significant positive and/or negative differences between the 'without rope' and 'with rope' versions for both the virtual and the robot arm versions? (Waren er grote positieve en/of negatieve verschillen tussen 'zonder touw' en de 'met touw' versie voor beide de virtuele en de robot arm versie?)
- Were there significant positive and/or negative differences between the virtual and the robot arm versions? (Waren er grote positieve en/of negatieve verschillen tussen de virtuele en de robot arm versie?)

5.2.5 Pilot Study

This pilot study focused mainly on refining the instructions given by the instructor, the understandability of the questionnaire and the action that can be taken to improve the comfortability of the user. This study also clearly focussed on flaws in the technical design preventing the viability of this tests. These flaws however did not seem to be very apparant nor improvable within the period of this research because of the reliance on additional hardware like the Meta Quest 2. The entire ZeroTraining system was iterated upon and showcased informally to a varied number of experts before this test was conducted. Because of the limited amount of participants that need to perform this test, the pilot study included a singular testparticipant.

Height and Positional Alignment Instructions

The first test evaluated whether the instructions provided during the alignment stage were sufficiently informative for users. Since the Meta Quest 2 does not support recording the

[3] `https://forms.gle/adDPf9v88JGCAo1i8`

passthrough system, demonstrating the setup to participants was not possible. Thus, the clarity of the instructions became crucial. Previously, the alignment process was left somewhat ambiguous, which was found to be inadequate during the pilot test. To address this, the first instruction was revised to specify that users should press the tip of the controller against the table while adjusting the height. This modification significantly improved clarity and ensured that each participant achieved optimal alignment. Additionally, virtual landmarks were introduced to simplify positional alignment. The virtual robot arm now features two cylinders that mimic the base of the physical robot arm. Users are required to align these cylinders precisely with the base of the robot arm, by slotting them inside of the base. These cyclinders can be seen in Figure 4.13.

Translations for Questionnaire

The initial version of the questionnaire, available only in English, posed some challenges for participants in terms of interpretation. This led to an increased number of meta-questions regarding the survey itself. To address this issue, we introduced a Dutch translation in addition to the English version, aiming to reduce the cognitive load on participants. Although adding a translation can introduce some cognitive Demand, it significantly improved comprehension for native Dutch speakers by allowing them to cross-reference and understand the questions more effectively. While this addition could potentially affect the overall reliability of the survey, we deemed this trade-off acceptable, given that the study was not focused on extensive statistical analysis.

The same approach was applied to the open-ended verbal questions. Note that the original English question is included in parentheses following the Dutch translation of each question 5.2.4. Since these questions were posed verbally, they were asked in the participants' native language, Dutch, to allow them to express themselves freely without the need to translate their thoughts into English. For practicality, both positive and negative aspects were addressed in the same question, as participants in the pilot study naturally transitioned between discussing positive and negative aspects. However, in the written form, these aspects are separated to facilitate easier post-analysis.

Comfortability and Tether Management Improvements

To enhance participant comfort and manage tethering issues, several adjustments were implemented. The instructor actively monitored and addressed tethering concerns, such as preventing the tether from becoming caught on the grip ratchet clamp or getting stuck beneath the base of the robot arm. Additionally a USB extender was used to make the length of the cable tethering the participant to the robot arm essentially invisible in VR. Special accommodations were made for participants with a fear of heights, including the option to use a chair or to immediately opt out of the simulation if they felt uncomfortable. Additional measures were taken to improve participant comfort, such as managing temperature, defogging the VR headset, assisting with putting on the VR headset and making head strap adjustments and simplifying the button mapping explanations provided during the preamble.

5.3 Deeper Dive into the Study Results

First, we will present the general results from the questionnaires. We will briefly review the provided graphs and analyze key metrics such as averages, medians, and standard deviations. These metrics will be explained in a concise manner. Note that the results will be categorized into the purely virtual and robotic arm versions; however, this categorization is not intended for comparison purposes. Instead, the results will be used to evaluate the feasibility of the technology demonstrator. A detailed statistical analysis is not within the scope of this feasibility study. Therefore, the empirical results will inform the final implementation discussions, but no hypotheses will be tested or confirmed based on statistical methods.

5.3.1 NASA Task Load Index

We will be starting with the NASA-TLX version, namely the RAW-TLX version. This questionnaire focused on assessing the general Demand of the participants such as Mental, Physical, Temporal Demand and also Effort, Perfomance and Frustration level. The questionnaire included six questions whereby these are seen in Figure 5.2. We can see that in general the Demand for this test was relatively low for both the purely virtual and the robot arm version. We can also see that both the Effort and the performance is highly variant between the participants. The frustration levels were also very low with the robot arm version having a slightly more frustration bound to it.

If we delve deeper into the data, we observe that both the average Mental and Physical Demands are slightly higher during the robot arm phase compared to the virtual phase. This increase is expected since participants interact with a physical object, requiring more cognitive Effort and repeated physical contact. The Mental Demand is particularly elevated during the robot arm phase, as reflected in the higher average scores, although participant responses varied, with some ratings at 3 and 4. Conversely, the Temporal Demand is slightly lower during the robot arm phase, suggesting reduced time pressure. This decrease could be attributed to the absence of object relocation by the robotic manipulator and the lack of physical interaction with an object by the participant during the virtual phase, which allowed participants to complete this phase more quickly if desired.

Regarding Physical Demand, the median is lower than the average, indicating a right-skewed distribution, as depicted in Figure 5.2. Responses are clustered at the lower end of the scale (1 and 2) with a slight spread towards higher values, indicating low Physical Demand overall. Additionally, the standard deviation for Mental Demand in the robot arm phase is higher, implying more varied responses and less consensus among participants. In contrast, the Temporal Demand shows a smaller standard deviation, indicating that responses are more closely clustered around the mean, with participants showing more consistent experiences. Responses to the robot arm phase are distributed between 2 and 5, suggesting varying levels of Effort, whereas the virtual phase responses are predominantly at 1 and 2, indicating lower overall Effort.

For performance, pure virtual responses range from 1 to 4, with a slight concentration around 3 and 4, indicating moderate performance levels. In comparison, robot arm responses are more concentrated around 4, suggesting generally higher performance. We can also see that the performance for the robot arm version was perceived as higher by the participants. Finally, frustration levels are low in both phases, with the robot arm responses primarily at 1 and 2 and the pure virtual responses highly concentrated at 1, indicating very low frustration. Both conditions exhibit right-skewed distributions with very low standard deviations, particularly in the pure virtual phase.

5.3.2 Intrinsic Motivation Inventory

The statistical analysis of the IMI test reveals notable differences between the pure virtual and robot arm phases as seen in Figure 5.3. In terms of Interest/Enjoyment, participants rated the pure virtual environment significantly higher, with most scores clustering around 8/10, indicating a more engaging and enjoyable experience compared to the robot arm phase, where scores were more varied. This could be because the robot arm version was not always extremely reliable and hence misalignment would break the immersion. For Pressure/Tension, which is plotted from no tension to maximum tension, the robot arm setting resulted in higher reported tension, with scores predominantly around 4/10. This suggests that the robot arm environment may have induced greater stress or pressure with some participants, perhaps because of the fact that the robot arm would make noise whilst being invisible to the user. This could make some users wary and hence more tense. Regarding Perceived Competence, participants felt more capable in the pure virtual environment, with most scoring slightly lower than 8/10, compared to a wider range of scores in the robot arm phase. When examining Effort/Importance, the

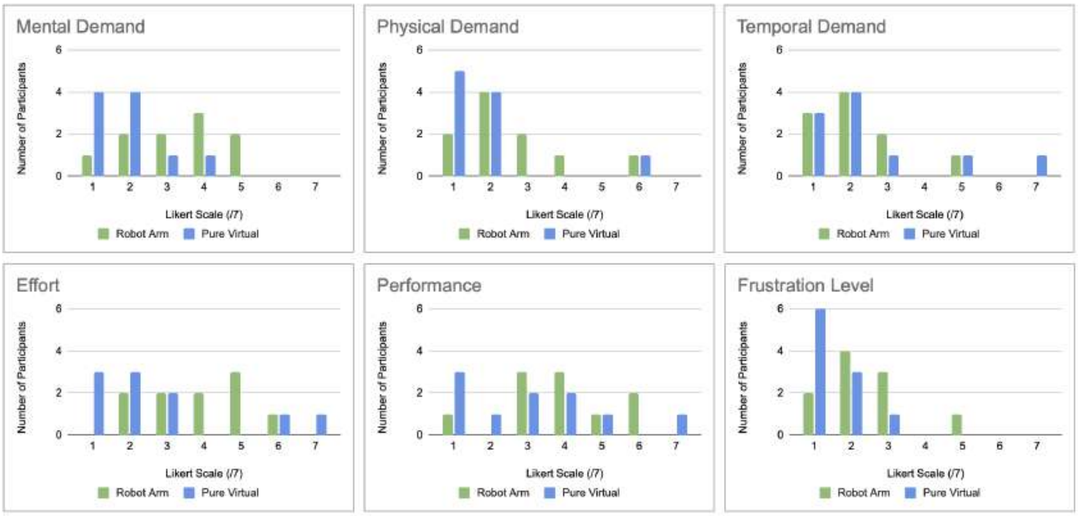


Figure 5.2: The results from the six NASA-TLX questions are plotted in histograms, with the x-axis representing the Likert scale from 1 to 7 and the y-axis showing the number of participants who selected each score.

Measure	Mental Demand	Physical Demand	Temporal Demand
Average			
Pure Virtual	27.14	27.14	37.14
Robot Arm	47.14	37.14	31.43
Median			
Pure Virtual	28.57	21.43	28.57
Robot Arm	50.00	28.57	28.57
Standard Deviation			
Pure Virtual	14.21	21.77	27.93
Robot Arm	19.11	21.51	17.56

Measure	Effort	Performance	Frustration Level
Average			
Pure Virtual	40.00	44.29	21.43
Robot Arm	55.71	55.71	34.29
Median			
Pure Virtual	28.57	42.86	14.29
Robot Arm	57.14	57.14	28.57
Standard Deviation			
Pure Virtual	29.97	28.13	10.10
Robot Arm	19.58	21.77	16.77

Table 5.2: The average, median and standard deviation of measured Likert Scale converted to a scale on 100 for both the pure virtual and the robot arm phases for the NASA-TLX questions.

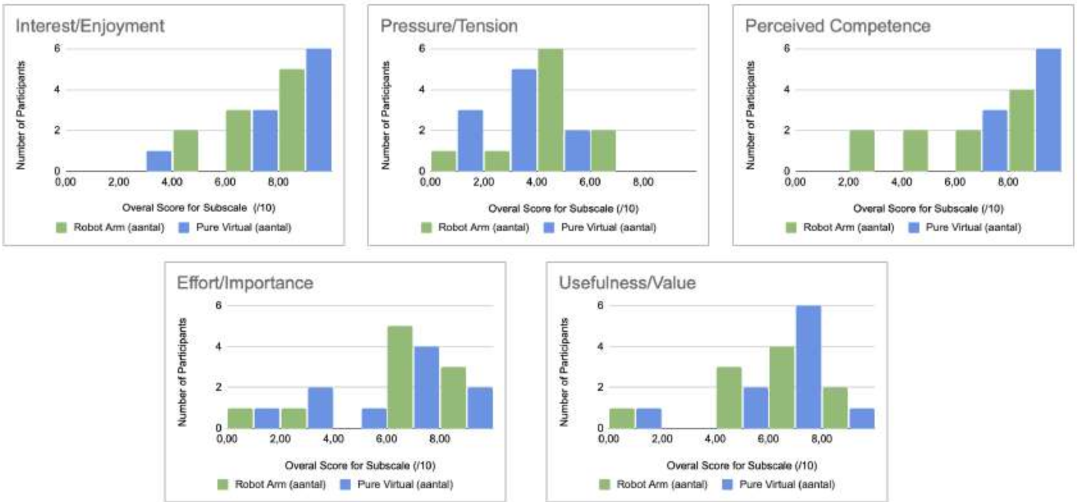


Figure 5.3: The results of the 21 questions from the IMI are plotted in histograms, with the x-axis representing the overall score for each subscale and the y-axis showing the number of participants who selected each score.

robot arm phase required a more consistent middle-range Effort, while the pure virtual condition showed a broader distribution of Effort levels. Finally, participants viewed the pure virtual environment as more Useful/Valuable, with higher scores typically around 8/10, reflecting its immersive and relevant nature. These results will be discussed in more detail in the following sections; however, a trend is already evident. In summary, the pure virtual condition was found to be more engaging, valuable, and associated with higher perceived competence, whereas the robot arm condition offered a more consistent Effort level but was perceived as less enjoyable and valuable, whilst not being far behind the purely virtual version. If these results were to be combined into a single system, we can also conclude that all these results are favourable for our system. One interesting thing to note in table 5.3 is the high standard deviation of the Effort. This observation may stem from the fact that some participants achieved high performance with minimal Effort, yet reported exerting maximum Effort during the tests. Consequently, this discrepancy might have led to some confusion in their responses to related questions.

5.3.3 Presence

The graphs 5.4 present participant responses across six different subscales: Possibility to Act, Realism, Quality of Interface, Self-Evaluation and Performance, Possibility to Examine, and Haptics. For the Possibility to Act, the pure virtual phase shows a higher concentration of participants scoring in the upper range (8/10), suggesting enhanced interactivity compared to the robot arm phase, which displays a wider spread of scores, centering around 4/10. Regarding Realism, the robot arm phase scores predominantly between 6/10 and 8/10, indicating a more realistic experience. On the other hand, the pure virtual phase sees a more uniform distribution of scores, with some falling lower. The Quality of Interface sees both phases achieving high marks, especially around the 8/10 score, but the pure virtual phase slightly outperforms with more scores at this high level. The interface was hence not a detriment to our scores found. For Self Evaluation and Performance, both phases show high scores with a slight edge in the robot arm phase for perceived better performance. In the Possibility to Examine, participants in the pure virtual phase frequently scored around 8/10 and higher, suggesting better exploration capabilities. This can be connected to the fact that less manipulation of the PGT was possible because of the lack of a collision response in this phase. Lastly, the Haptics subscale reveals that both phases were well-received, though the pure virtual phase had a marginally more consistent positive feedback. Again, some misalignment with the virtual PGT and the physical robot arm

Measure	Interest/Enjoyment	Pressure/Tension	Perceived Competence
Average			
Pure Virtual	7.69	4.76	6.86
Robot Arm	7.86	2.90	8.50
Median			
Pure Virtual	7.98	5.00	7.14
Robot Arm	8.21	2.86	8.57
Standard Deviation			
Pure Virtual	1.43	1.78	2.43
Robot Arm	2.08	1.11	0.89

Measure	Effort/Importance	Usefulness/Value
Average		
Pure Virtual	6.93	6.33
Robot Arm	5.86	6.38
Median		
Pure Virtual	7.86	6.90
Robot Arm	6.07	6.67
Standard Deviation		
Pure Virtual	2.53	1.97
Robot Arm	2.55	2.02

Table 5.3: The average, median and standard deviation of measured Likert Scale converted to a scale on 10 for both the pure virtual and the robot arm phases for the IMI questions.

configuration probably made it a little less well received. The overall analysis highlights that while the robot arm phase is seen as more realistic, the pure virtual phase is preferred for its interactive qualities and interface.

5.3.4 Avatar Embodiment

The analysis of the Avatar Embodiment questionnaire reveals that participants felt a stronger connection and ownership over their avatars in the pure virtual phase compared to the robot arm phase. This suggests that the virtual environment better facilitated the illusion of the avatar being the participants' own body. Additionally, participants reported a higher sense of control and agency during the pure virtual phase, indicating that they felt more in command of their movements and actions. This could be contributed to the fact that the collision response in the pure virtual phase seems extremely realistic in addition to hiding the hand meshes in the robot arm phase. It appears that the variability in alignment, particularly in the robot arm phase, was significantly influenced by user error and negatively impacted certain participants. However, some participants reported a moderately positive tactile experience, suggesting that while the system has potential, its reliability needs improvement. The alignment between the virtual PGT and the controller is used as an indicator for tactile sensations because the Avatar Embodiment Questionnaire, which was specifically designed to assess these aspects, included questions with blanks that was filled in by the research team. Such questions included: "It seemed as if I felt the touch of the virtual Pistol in the location where I saw the virtual Pistol. ". This approach ensured that the responses were directly linked to the participants' experiences of tactile feedback and alignment for this system. However, tactile sensations were more variable in the pure virtual phase, with some participants experiencing strong feedback and others less so. This is very interesting since the pure virtual phase of course did not have any haptic feedback involved an purely demonstrated a visual input to the user. While the robot arm phase generally provided weaker but more consistent tactile feedback, it does show potential. Overall, the pure virtual phase appears to enhance the embodiment experience, particularly in terms of body ownership and agency, while the robot arm phase offers a more

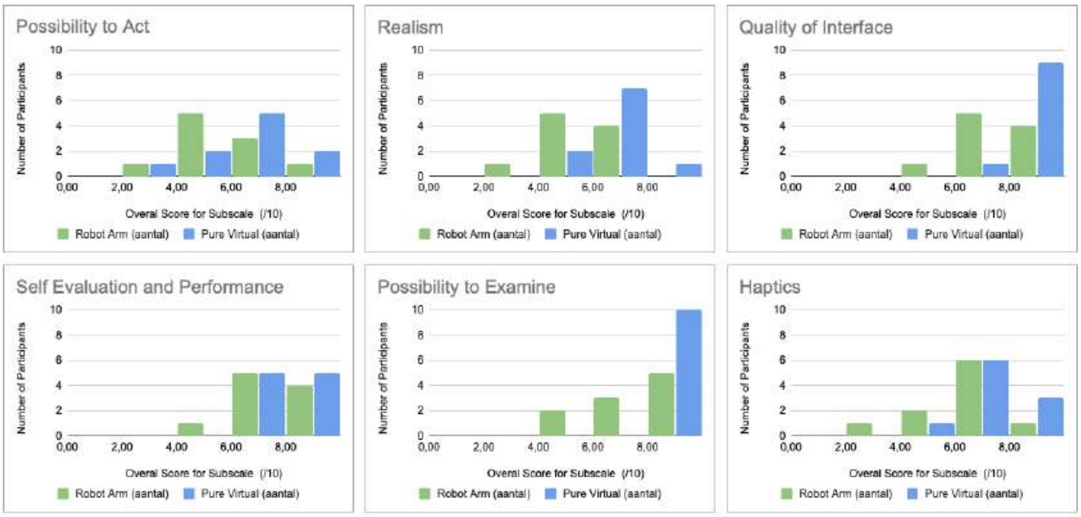


Figure 5.4: The results of the 16 questions from the PQ are plotted in histograms, with the x-axis representing the overall score for each subscale and the y-axis showing the number of participants who selected each score.

Measure	Possibility to Act	Realism	Quality of Interface
Average			
Pure Virtual	6.57	6.71	8.81
Robot Arm	5.79	5.88	7.33
Median			
Pure Virtual	6.43	6.90	9.05
Robot Arm	5.36	5.95	7.38
Standard Deviation			
Pure Virtual	1.68	1.08	0.79
Robot Arm	1.63	1.56	1.03

Measure	Self Eval. and Perform.	Possibility to Examine	Haptics
Average			
Pure Virtual	8.57	9.29	7.50
Robot Arm	8.00	7.57	6.50
Median			
Pure Virtual	8.21	9.29	7.14
Robot Arm	7.86	7.86	6.79
Standard Deviation			
Pure Virtual	1.01	0.75	0.91
Robot Arm	1.20	2.03	1.60

Table 5.4: The average, median and standard deviation of measured Likert Scale converted to a scale on 10 for both the pure virtual and the robot arm phases for the PQ questions.

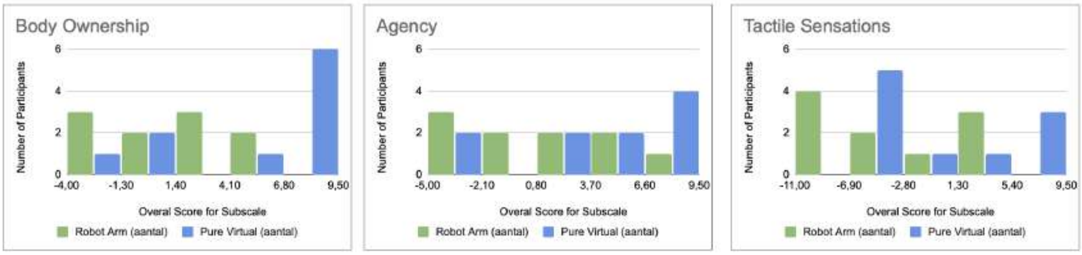


Figure 5.5: The results of the 10 questions from the Avatar Embodiment Questionnaire are plotted in histograms, with the x-axis representing the overall score for each subscale and the y-axis showing the number of participants who selected each score. Note that the x-axis is not restrained to being /10 yet can also have negative numbers.

Measure	Body Ownership	Agency	Tactile Sensations
Average			
Pure Virtual	4.5	3.2	-0.9
Robot Arm	1.1	1.2	-3.6
Median			
Pure Virtual	6	3	-3
Robot Arm	0.5	0.5	-4
Standard Deviation			
Pure Virtual	4.77	4.92	4.98
Robot Arm	3.54	4.24	5.25

Table 5.5: The average, median and standard deviation of the measured subscales for both the pure virtual and the robot arm phases for the Avatar Embodiment questions.

modest embodiment experience.

5.3.5 Open Questions Results

In this section, we discuss the responses to the open-ended questions. We also explore explanations for why certain aspects were preferred less in the robot arm version. Both positive and negative aspects for each question are compiled into separate subsections.

Aspects of the Purely Virtual Phase

In the responses regarding the positive aspects of the virtual version, several key points were highlighted. Participants appreciated the detailed and realistic environment such as the lighting, the 3D models, the physics itself, particularly noting that the virtual space station resembled the ISS. The virtual hand and its interaction with objects like the drill were praised for their realism and responsiveness, providing a convincing sense of zero gravity. Many felt that the application offered greater control compared to the robot arm phase and allowed for a more immersive experience. The freedom within the virtual environment and the good calibration of the virtual hand were also mentioned positively.

However, the negative aspects noted included issues with the virtual pistol's behavior and grip, which was described as erratic and unresponsive. Many participants instinctually tried to hold the object in their hands. This behavior however was not implemented since it would fall outside of the scope of our tests. The erratic nature also could sometimes be attributed to the limitations of the physics engine. Additionally the erratic nature of the rope was also very unknown to the participants which can be attributed to a human being's lack of involvement with zero gravity ropes. We could hence point back to the "enchanted snake" problem mentioned by ESA throughout this thesis. Participants found it sometimes difficult to interact with the pistol and noted that it often moved away unexpectedly or appeared to disappear. This was because of the despawning and spawning nature of the trails. The lack of physical properties and the inability to physically grasp objects detracted from the realism, making some interactions feel less intuitive and engaging. Additionally, the absence of visible body parts like the hiding of the right hand, which was done to improve immersion as stated earlier, and the constraints on movement because of the wired version were also cited as areas for improvement.

Aspects of the Robot Arm Phase

In the responses regarding the positive aspects of the robot arm version, several key advantages were noted. Participants appreciated the physical feedback provided by the robot arm, which allowed them to feel and interact with objects more concretely. They could feel the contours of the tool in their hand. They could touch the top of the tool and even sometimes press the trigger of the PGT when it was found. Users also noted that the robot arm itself could help locate the tool, whilst not being extremely realistic. The calibration was generally smooth, and the ability to physically grasp the objects was highlighted as a significant benefit. The tactile sensations and the physical presence of the robot arm made the experience less monotonous and provided a more intuitive sense of where objects were located. The movement of the robot arm was also seen as beneficial for learning how to interact with the environment. One of the best experiences noted by a couple of participants was during the rope subphase. This was because the object was moving and hence this movement could be tracked with their hand. The robot arm hence followed this path pretty well particularly when the movement of the tool was not that fast.

However, there were notable negative aspects as well. Issues included erratic movement of the robot arm when hands were removed from the work envelope, and difficulties with the robot arm's accuracy and responsiveness. Participants experienced disjunctions between the intended and actual movement of the robot arm, which sometimes caused confusion and inefficiencies in locating objects like the pistol. They felt that in certain areas of the robot arms work envelope, the misalignment made it almost impossible to find the tool itself. They then proceeded to wiggle their hands around in the approximate area, until they hit something. The worst case example was when the alignment was completely off and the tool could not be found which happened very rarely. This was because of certain singularities being tested, such as when the virtual PGT was in line with the base of the physical robot arm and hence needed to curl around itself to be able to reach this position. The presence of cables and the risk of damaging the robot arm added to the challenges, and the audible noise of the arm was distracting. Concerns were also raised about safety and the physical proximity of the arm, which sometimes led to accidental bumps or feelings of discomfort. Again, in the worst case scenario the robot arm misbehaved by making a load noise or by pushing the participant in their thigh, which again happened very rarely. Certain participants also pointed out that the "Nudge Towards Rotation Envelope" script was jarring and not easily understandable from the user's perspective. It appeared to completely disrupt the experience for some users.

Some significant delays with the relocation mechanism were also seen in terms of the instructors perspective. The participant then would note that the robot arm would not work properly, but the instructor would then see that the robot arm was still just traveling to its destination. A great example of this problem is given in Figure 5.6. The base rotation of the robot arm is 315° which means that a subset of the 360° is hence a deadzone as mentioned in Section 4.3. When the virtual PGT is at the far right side of the deadzone, as seen in Figure 5.6a, some inherent misalignment takes place. Then the virtual PGT tool crosses the deadzone, it needs to make a complete revolution to get to this point. Also when the virtual PGT entered the deadzone from the right and then briefly left the deadzone again, back from whence it came, the robot arm is could in the worst case scenario make two full revolutions worsening the delay significantly. This problem could have been fixed in software by changing the Braccio++ Library and allowing the deadzone to be crossed by the robot arm, but then more problems would have arisen with the tethering of the cables after making one or more crossings of the deadzone. The cables could then be completely wrapped around the robot arm stopping the system in its tracks.

Significant delays with the relocation mechanism were observed from the instructor's perspective. While participants noted that the robot arm was not functioning properly, the instructor would see that the robot arm was still traveling to its destination. A notable example of this issue is illustrated in Figure 5.6. The base rotation of the robot arm is 315°, creating a deadzone within the 360° range, as discussed in Section 4.3. When the virtual PGT is positioned at the far right side of the deadzone, as shown in Figure 5.6a, inherent misalignment occurs. If the virtual PGT tool crosses the deadzone, it requires a full revolution to reach this point. Furthermore, if the virtual PGT enters and exits the deadzone from the right, the robot arm could, in the worst-case scenario, make two full revolutions, significantly worsening the delay. This issue could have been addressed in software by modifying the Braccio++ Library to allow the robot arm to cross the deadzone without necessitating a full revolution in the opposite direction. However, this solution could have led to other problems, such as the cables becoming wrapped around the robot arm. Repeated crossings of the deadzone could cause the cables to become entangled, potentially compromising the system's integrity and halting its operation.

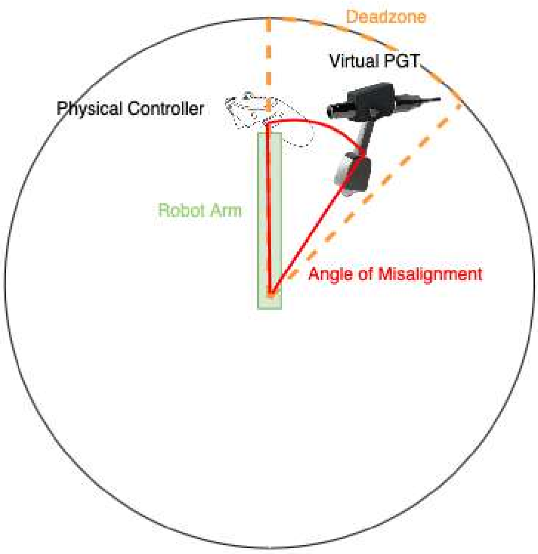
5.3.6 Aspects of Ropeless and a Rope

There were varying opinions on the differences between the versions with and without a tether, for both the virtual and robot arm phases. For the robot arm phase, many participants observed little difference between using a tether and not using one. With a rope it also made it easier to locate the object itself after spawning. On the other hand, the tether sometimes added constraints to physical freedom, making movement feel less natural because the collisions of the hand with the rope were sometimes erratic. Overal many people did not experience a huge amount of difference between both version. This is only logical, considering the ropeless subphase also had a system that nudged to virtual PGT tool towards the center, essentially working as an invisible tether. However, occasionally a difference was felt where the object could be physically tracked using the hands of the participant. This is a moment where the potential of this application could truly be noticed and where this system actually shines. However we did realise that this moment was few and far between.

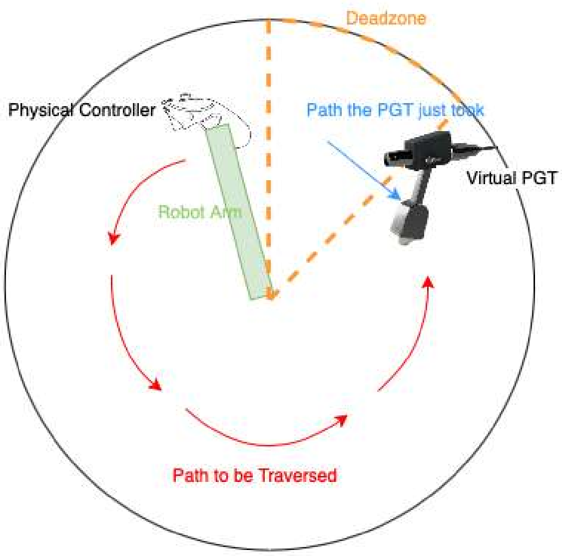
5.3.7 Pure Virtual vs Robot Arm

Participants generally found the virtual version preferable in terms of ease of use and freedom from physical constraints. The virtual environment allowed for more intuitive interaction without the need to search for or manage a physical tool. Additionally, the feedback in the physical robot arm version was appreciated, but the virtual version provided a less stressful experience, particularly when using the virtual untethered setup as the tethered setup was somewhat erratic in both phases. There was also a preference for the virtual setup's flexibility, as it avoided the limitations imposed by the robot arm's fixed movements and alignment issues.

Conversely, the robot arm version presented challenges related to alignment and calibration. Participants noted that the physical robot arm sometimes made interactions less fluid and was dependent on user adjustments. Issues such as not being able to freely decide the robot arm's movements and the irritation of having the hand or arm move unexpectedly detracted from the experience. The virtual version was generally favored for its superior alignment and ease of interaction, despite some limitations with realism.



(a) The virtual PGT is on the far right side of the deadzone whilst the robot arm is stuck at the far left side of the deadzone.



(b) The virtual PGT crosses the deadzone and hence the robot arm needs to make a full revolution.

Figure 5.6: The problems with the Deadzone as seen from a topdown view.

Chapter 6

Discussion and Conclusions

6.1 Feasibility Analysis

It became apparent that the physical prototype was perceived to be significantly inferior to its virtual counterpart in several key areas. Firstly, the physical setup suffered from various unresolved issues, such as the presence of a deadzone, discrepancies between the contours of the controller and the virtual PGT tool, and alignment errors that were influenced by user variability. Furthermore, the cables restricting participant movement added another layer of complexity and limitation. In addition to these issues, the robot arm itself presented a range of inherent challenges. The arm's limited work envelope and accuracy problems hindered its effectiveness. Notably, the rotation feature was not applied to the tool itself; instead, only the tool's location was adjusted. Problems related to weight distribution across different configurations and the mechanical vulnerability of the arm further compounded the difficulties. While some of these problems are theoretically addressable, such as the alignment issues, the limited timeframe of this project constrained the extent to which they could be resolved. Other issues, like integrating collision response during the encountered-type haptic feedback phase, present even greater challenges due to their complexity and the current state of technology.

The issues identified might suggest that this feasibility study was unsuccessful, especially given that many users preferred the purely virtual implementation. However, I contend otherwise. The fact that the scores for both the robot arm implementation and the virtual version were relatively close indicates that the robot arm approach is closer to a viable solution than to being deemed infeasible. Expert feedback and consultations throughout the research highlighted that addressing these problems would be extremely challenging and potentially beyond the scope of the given timeframe. The occasional success of the implementation, even with a limited headset like the Meta Quest 2 and a relatively low-end robot arm such as the Braccio++, demonstrates a level of potential that exceeds initial expectations. While refining the fundamentals was a significant undertaking, it was not insurmountable. The issues encountered are complex but not impossible to resolve. To achieve a truly effective solution, however, an extended period of development—potentially many times longer than the duration of this study—would be necessary. Nonetheless, the potential for a breakthrough is evident if such an effort is made.

For instance, when the cables were not obstructing movement, and the robotic manipulator was correctly aligned with the virtual PGT tool, the system performed remarkably well. The tool floated at a slow enough pace to allow the rotational joints to keep up, followed a simple path, and avoided crossing any deadzones while staying clear of the robot arm's singularities. Additionally, with the controller positioned appropriately between the robot arm and the participant, the system demonstrated excellent functionality. Achieving this excellent performance required meeting a multitude of conditions, aimed for by the researchers. Many participants may have only experienced these optimal conditions occasionally. Consequently, considerable refinement is necessary to achieve a higher level of consistency and ensure that these impressive results are replicable more frequently.

6.2 Guidelines for a Robotic Manipulator

Considering the limitations of the robotic manipulator used in this research, it would be beneficial to establish guidelines for designing a more optimal robot arm based on our findings. A primary constraint is the reach of the robot arm, which depends on the length of its link segments: longer segments increase the work envelope and reach. A significant issue is the payload capacity, which directly affects the flexibility of the system. Higher payload capacity allows for more sophisticated end-effector toolings and enables features like an automatically switching end-effector. For instance, one degree of freedom could be sacrificed to allow dynamic switching between different end-effectors, potentially supporting multiple toolings simultaneously. However, a large payload capacity is not essential because small objects are not typically very heavy; their mass can be simulated by the robot arm resisting movement, and their volume can be emulated through the arm's articulation.

Another important design consideration is the degrees of freedom (DOF) supported by the robot arm. This aspect involves a balancing act, as increasing the number of DOF makes the control algorithms more complex. For example, CNC machines use advanced path-planning algorithms to prevent link collisions, and the complexity of these algorithms increases with the sophistication of the end-effector tooling. However, there are significant advantages to having more DOF. With three DOF, the robot arm can determine the 3D position of an object. Expanding to six DOF allows for precise control over both the position and orientation of the object, enabling realistic positioning as seen in the ZeroPGT subsystem. Additional DOF can further enhance the system by adjusting the angle of attack, ensuring that the robot arm does not obstruct the user's interaction and creating a more immersive experience.

The connection between the ZeroPGT and ZeroArm systems is currently implemented via a wired connection. However, this wired approach has several limitations. Firstly, the user is tethered to the environment, which reduces realism. Secondly, the cable is connected to the Carrier Board, which rotates with the robot arm's base. This can cause operational issues if the cable wraps around the arm during its movement. To address these issues, a potential solution could be to avoid direct cable connections to the moving robotic manipulator and hence make sure that the cable cannot be manipulated by the normal movement of the robot arm. Alternatively, a wireless connection could be implemented, provided that it overcomes technical challenges such as latency.

It is highly recommended that the next robot be a collaborative robot (cobot), designed to interact directly with users in its environment. Research [Adr+22] indicates that preventative measures often involve reducing the distance between the user and the robotic manipulator. However, advancements in robotics have led to the development of safer robot arms that can operate effectively at closer distances.

6.3 Future Improvements and Future Work

Several improvements could be made to our current ZeroTraining system, yet it is also worth exploring potential innovations for future development. One possibility is incorporating methods for haptic retargeting and reach redirection. This enhancement would enable the robot arm to more accurately reenact virtual objects or support multiple objects simultaneously by aligning its movements with the user's reach. Another potential improvement involves adjusting the virtual PGT to more closely match the actual location of the controller. This would bring the PGT, as an accuracy measure, closer towards the controller instead of the controller coming closer to the virtual PGT. Implementing these systems could significantly enhance accuracy.

However, determining the optimal conditions and relative importance of these improvements requires thorough testing and further research, as different systems may need to be emphasized more strongly in specific situations.

Another potential enhancement for the robot arm is incorporating a mechanism to detect if a joint has skipped by sensing its 3D location, integrated into the ZeroPGT subsystem. Although the servomotor supports a PWM signal, the inherent instability of the arm means that this alone may not be sufficient. Therefore, improvements in weight distribution and overall build quality of the robotic manipulator could be valuable areas for future development. Designing a robotic manipulator from scratch specifically for this system might also be an intriguing avenue to explore as it could incorporate sensors that directly plug into the ZeroTraining system.

The next proposed area for future work is somewhat more advanced but still closely related to our ZeroTraining system. One potential enhancement is the integration of a seamless detaching mechanism. This would allow participants to detach the tool from the robot arm and use it as if it were an actual PGT. Haptic feedback and sound could be implemented to simulate the trigger press. Although features like haptics during trigger presses are already included in the system, they have not yet been utilized, suggesting that their implementation would not be overly complex. To address this, once the user holds the controller, additional haptic feedback could be provided by the robotic manipulator to enhance realism. The robot arm could then position itself to the slot where the PGT needs to be used, allowing the participant to slot it back into place. Furthermore, when the user attempts to use the tool, the robot arm's wristroll rotation could be activated to simulate torque, providing a more authentic experience. However, creating a realistic and user-friendly detaching mechanism presents challenges. Currently, detaching the controller from the robot arm requires an uncomfortable amount of force and must be applied in line with the last link, which is not readily apparent in the VR application. Thus a multidirectional detaching mechanism could be designed to support this behavior.

The robotic manipulator could also be used to emulate surfaces that the user can touch. For example, when the user approaches a virtual wall, the robot arm could follow the surface normal based on the user's hand position, simulating the sensation of the wall being at a specific location. This concept aligns with the goals of the European Space Agency (ESA), which is interested in developing systems where, when a force is applied to a surface, the virtual world appears to move around the user's perspective. This would simulate the experience of pushing an extremely heavy object, such as the International Space Station. In this scenario, the force applied to the space station would be negligible, with the primary perceivable reaction being the astronaut's own movement. Additionally, the robotic manipulator would move the emulated surface away from the astronaut to further enhance the realism of the interaction.

If the participant is holding the controller in their hand, it is important to consider whether they should be able to drop it, and what would happen if they do. One of the earliest and most intriguing ideas explored was to allow the participant to drop the controller, but to avoid depicting this action in the virtual environment to maintain the zero-gravity illusion. To address this, a retrieval mechanism could be implemented to reattach the controller to the robot arm. For instance, the controller could be tethered by a rope that allows the robot arm to retrieve it from a neutral hanging position. Alternatively, a manual retrieval system similar to the one used in the Haptic Turk research [Che+14] could be employed, where an instructor physically picks up the fallen controller and remounts it, bypassing the need for an automated system.

Incorporating a collision detection and response system into the encountered-type haptic feedback system presents significant challenges. One possible solution is to suspend the controller using multidirectional springs at the end-effector of the robot arm. When the user interacts with the controller, the springs would compress and allow the controller to move gently out of the way, simulating a collision. The system would detect this compression and adjust the robotic manipulator's movement to reflect the new velocity of the virtual tool so that the decompression is not felt by the user.

Currently, a similar effect was attempted by placing the controller in a stationary position and measuring any movement caused by user interaction. However, this approach is limited by the rigidity of the stationary robot arm and hence abandoned. Designing an end-effector suspension with multidirectional springs would address this rigidity issue. Additionally, integrating a 360° rotational capability for the controller would be a complex yet intriguing area for future development. While adding velocity to the controller is relatively straightforward, simulating torque would require sophisticated collision detection and the use of an inertia tensor. These aspects represent promising directions for future research and development.

Lastly, incorporating a physical representation of a zero-gravity rope using the robotic manipulator presents a significant challenge. The concept involves creating a physical simulation of a rope that behaves as if it were in a zero-gravity environment. This would require the robotic manipulator to accurately simulate the dynamics of a rope, including its tension, movement, and interaction with the user. Given the current system's limitations, adding such a feature is far from straightforward. The robotic manipulator would need to be capable of replicating the physical properties of a rope, including its ability to stretch, slack, and interact dynamically with the user. Simple extensions or modifications to the existing system do not appear to be sufficient to support this feature effectively. Developing a solution would likely require a complete redesign or significant enhancement of the robotic manipulator to handle the complexities involved in simulating a haptic zero-gravity rope.

6.4 Self-Evaluation and Critique

This undertaking was certainly not without it's challenges. I underestimated the extent to which this project required a solid understanding of hardware design, industrial engineering, and physics. Consequently, I had to either refresh my knowledge in these areas or, in some cases, learn them from scratch. The ambition—and, at times, the audacity—of tackling such a daunting task was not lost on me. In the early stages of my research, I spent time brainstorming futuristic ideas, which, while keeping me engaged, also led to unrealistic expectations about resolving fundamental issues within the given timeframe. However, it is common in the initial phase of research to explore broad ideas before narrowing down to a more achievable focus. This was particularly evident in this project, as it was a subject of my own design rather than one proposed by advisors as viable.

I consider myself an "out-of-the-box" thinker, with an innate ability to find innovative yet sometimes unconventional solutions to problems. However, this inclination can occasionally lead me to focus on less obvious solutions and spend excessive time on trivial issues. For instance, my initial idea to design a custom metal horn as a replacement for the plastic horn on the servomotors, though ambitious and potentially a better fit for the robot arm, was ultimately impractical and would have consumed too much time. Fortunately, I recognized my tendency to pursue the "best" solution rather than opting for more straightforward alternatives, such as purchasing metal parts online. This awareness has made me more cautious about such tendencies and will guide me in future projects. There are, however, situations where the opposite might be true. I would rather refine my ingenuity and manage it effectively, rather than having to develop this skill from scratch when it is critically needed.

I also realized that my communication skills, particularly for asynchronous and formal media such as emails, needed improvement. I found that I was not particularly adept in this area—ranging from writing verbose paragraphs to procrastinating on responses, and occasionally forgetting to include essential recipients in the CC field. Throughout this process, I worked on refining these practical skills. Despite my strengths in verbal communication, I made a concerted effort to improve my written communication to ensure effectiveness in all forms.

I believe that my User Experience Study is more advanced compared to my previous research during my bachelor thesis, incorporating a broader range of aspects recognized by the evaluation and validation community. However, I acknowledge that this is not my strongest area, and there may be some irregularities or issues with my approach. Specifically, the comparison between the baseline and the innovative aspects of the study appears to lean more towards a comparative analysis rather than a validation and feasibility study. Although this was not my initial intention, I understand why this shift might be noticeable to readers. Additionally, while my strong suit is not in the area of statistics, the study did not delve deeply into this aspect. Nonetheless, it is worth noting that enhancing the statistical analysis could further elevate the quality of the research.

To highlight some positive aspects of my work, I thoroughly enjoyed the iterative process involved in designing and fabricating the end-effector module. I approached problems with the mindset that they were initially unsolvable, but I quickly found solutions by exploring three main avenues: digging deeper into the issue, consulting experts with more experience, or reevaluating the problem from a different perspective. Sometimes, circumventing the problem entirely turned out to be a valid solution. Additionally, while we could never be entirely sure that solving one problem wouldn't lead to a larger, more complex issue, I appreciated the challenge. I also found it rewarding to act as an instructor for my own user experience study. Although it was challenging to remain impartial given my involvement in the creation, I took care to avoid influencing participants during their evaluation. I focused on making the experience as comfortable as possible and approached feedback with a open attitude. After all, it is through feedback that a decent solution can evolve into a great one.

This reflection on my experiences reveals an interesting aspect of my thought process and working style. I am thorough and tend to explore problems in depth, often uncovering multiple issues that need resolution. This detailed approach is evident in the length of my thesis and various sections. I acknowledge that I can be somewhat verbose when explaining problems, which sometimes leads to overly detailed descriptions. While I strive to be brief, accurate, and clear, I occasionally struggle with maintaining brevity without sacrificing comprehensibility.

Bibliography

[Adr+22] A. Adriaensen et al. "Teaming with industrial cobots: A socio-technical perspective on safety analysis". In: *Human Factors and Ergonomics in Manufacturing & Service Industries* 32.2 (2022), pp. 173–198. DOI: https://doi.org/10.1002/hfm.20939. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/hfm.20939. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/hfm.20939.

[Aga+23] Maria D. Agakhanova et al. "Experience with augmented reality technology in the surgical treatment of a patient with encapsulated metal foreign bodies of lower extremities". In: *Bulletin of the Russian Military Medical Academy* 25.2 (2023), pp. 261–268. ISSN: 1682-7392. URL: https://journals.eco-vector.com/1682-7392/article/view/321172.

[Age] European Space Agency. *Spacewalk training*. URL: https://www.esa.int/Science_Exploration/Human_and_Robotic_Exploration/Astronauts/Spacewalk_training.

[Age23] European Space Agency. *Underwater VR for Astronaut Training*. Dec. 2023. URL: https://nebula.esa.int/sites/default/files/neb_study/3202/C4000138140FP.pdf.

[AL11] Andreas Aristidou and Joan Lasenby. "FABRIK: A fast, iterative solver for the Inverse Kinematics problem". In: *Graph. Models* 73.5 (Sept. 2011), pp. 243–260. ISSN: 1524-0703. DOI: 10.1016/j.gmod.2011.05.003. URL: http://dx.doi.org/10.1016/j.gmod.2011.05.003.

[Appa] Apple. *Apple Vision Pro*. URL: https://www.apple.com/apple-vision-pro/.

[Appb] Apple. *ARKIT 6 - Augmented Reality - Apple Developer*. URL: https://developer.apple.com/augmented-reality/arkit/.

[Ard] Arduino. *Braccio++ kit*. URL: https://braccio.arduino.cc/.

[Azm+16] Mahdi Azmandian et al. "Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: Association for Computing Machinery, 2016, pp. 1968–1979. ISBN: 9781450333627. DOI: 10.1145/2858036.2858226. URL: https://doi.org/10.1145/2858036.2858226.

[Bar+23] Andres Barrio et al. "X-ARM: a Novel Arm Exoskeleton combined with Extended Realities to train Future Astronauts". In: Oct. 2023.

[BB95] Steven S. Beauchemin and John L. Barron. "The computation of optical flow". In: *ACM computing surveys (CSUR)* 27.3 (1995), pp. 433–466.

[BCF19] Barbara Rita Barricelli, Elena Casiraghi, and Daniela Fogli. "A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications". In: *IEEE Access* 7 (2019), pp. 167653–167671. DOI: 10.1109/ACCESS.2019.2953499.

[Bfe23] Bferrell. *How do VELCRO® Brand Fasteners Work? — VELCRO® Brand Blog*. Sept. 2023. URL: https://www.velcro.com/news-and-blog/2023/02/how-do-velcro-brand-fasteners-work/.

[BGB] BGB. *What is a Slip Ring and How Do They Work? — BGB*. URL: https://www.bgbinnovation.com/knowledge/news-and-articles/what-is-a-slip-ring#:~:text=Slip%20rings%2C%20which%20are%20an,'or%20vice%20versa')..

[BRH18] Teo Babic, Harald Reiterer, and Michael Haller. “Pocket6: A 6DoF Controller Based On A Simple Smartphone Application”. In: *Proceedings of the 2018 ACM Symposium on Spatial User Interaction*. SUI ’18. Berlin, Germany: Association for Computing Machinery, 2018, pp. 2–10. ISBN: 9781450357081. DOI: 10.1145/3267782.3267785. URL: https://doi.org/10.1145/3267782.3267785.

[Bro96] John Brooke. “SUS – a quick and dirty usability scale”. In: Jan. 1996, pp. 189–194.

[Bus04] Samuel Buss. “Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods”. In: *IEEE Transactions in Robotics and Automation* 17 (May 2004).

[Car23] Rebekah Carter. *Varjo XR-4 Series Review: Next-Level Mixed reality headsets*. Dec. 2023. URL: https://www.xrtoday.com/reviews/varjo-xr-4-series-review-next-level-mixed-reality-headsets/.

[Che+14] Lung-Pan Cheng et al. “Haptic turk: a motion platform based on people”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: Association for Computing Machinery, 2014, pp. 3463–3472. ISBN: 9781450324731. DOI: 10.1145/2556288.2557101. URL: https://doi.org/10.1145/2556288.2557101.

[Che+20] Lin Chen et al. “Hand Gesture Recognition Using Compact CNN via Surface Electromyography Signals”. In: *Sensors* 20.3 (2020). ISSN: 1424-8220. DOI: 10.3390/s20030672. URL: https://www.mdpi.com/1424-8220/20/3/672.

[Cho+20] Steven A. Cholewiak et al. “A perceptual eyebox for near-eye displays”. In: *Opt. Express* 28.25 (Dec. 2020), pp. 38008–38028. DOI: 10.1364/OE.408404. URL: https://opg.optica.org/oe/abstract.cfm?URI=oe-28-25-38008.

[CI20] Andreas Charalambous and Androniki Ioannou. “Virtual Reality and Augmeneted Reality for Managing Symptoms”. In: *Developing and Utilizing Digital Technology in Healthcare for Assessment and Monitoring*. Ed. by Andreas Charalambous. Cham: Springer International Publishing, 2020, pp. 85–104. ISBN: 978-3-030-60697-8. DOI: 10.1007/978-3-030-60697-8_7. URL: https://doi.org/10.1007/978-3-030-60697-8_7.

[CM18] Mai Lee Chang and Jessica J Marquez. *Human-automation allocations for current robotic space operations: space station remote manipulator system*. Tech. rep. NASA’s Human Research Program, Oct. 2018. URL: https://ntrs.nasa.gov/api/citations/20200003149/downloads/20200003149.pdf.

[CM22] Michael F. Cohen and Javier Asensio Meroño. “’Guilty bystanders’: VR gaming with audience participation via smartphone”. In: *International journal of entertainment technology and management* 1.4 (Jan. 2022), p. 339. DOI: 10.1504/ijenttm.2022.129638. URL: https://doi.org/10.1504/ijenttm.2022.129638.

[CMB17] Lung-Pan Cheng, Sebastian Marwecki, and Patrick Baudisch. “Mutual Human Actuation”. In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. UIST ’17. Québec City, QC, Canada: Association for Computing Machinery, 2017, pp. 797–805. ISBN: 9781450349819. DOI: 10.1145/3126594.3126667. URL: https://doi.org/10.1145/3126594.3126667.

[con24a] Wikipedia contributors. *Extravehicular activity*. Mar. 2024. URL: https://en.wikipedia.org/wiki/Extravehicular_activity.

[con24b] Wikipedia contributors. *Extravehicular Mobility unit*. Mar. 2024. URL: https://en.wikipedia.org/wiki/Extravehicular_Mobility_Unit.

[con24c] Wikipedia contributors. “Retina Display”. In: Mar. 2024. URL: https://en.wikipedia.org/wiki/Retina_display.

[con24d] Wikipedia contributors. “Simultaneous localization and mapping”. In: Mar. 2024. URL: https://en.wikipedia.org/wiki/Simultaneous_localization_and_mapping.

[con24e] Wikipedia contributors. “Tribology”. In: Mar. 2024. URL: https://en.wikipedia.org/wiki/Tribology.

[Das05] E. Julius Dasch. *A Dictionary of Space Exploration*. 2nd ed. Oxford University Press, USA, Jan. 2005. DOI: 10.1093/acref/9780192806314.001.0001. URL:

https://www.oxfordreference.com/display/10.1093/acref/9780192806314.001.0001/acref-9780192806314.

[Dea+17] Matthew Deans et al. "MINERVA: User-centered Science Operations Software Capability for future human Exploration". In: IEEE, Mar. 2017. DOI: 10.1109/aero.2017.7943609. URL: https://doi.org/10.1109/aero.2017.7943609.

[Deu+21] Bram van Deurzen et al. "HapticPanel: An Open System to Render Haptic Interfaces in Virtual Reality for Manufacturing Industry". In: *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology*. VRST '21. Osaka, Japan: Association for Computing Machinery, 2021. ISBN: 9781450390927. DOI: 10.1145/3489849.3489901. URL: https://doi.org/10.1145/3489849.3489901.

[Dis] Center of Disease Control (CDC). *Anthropometry — NIOSH — CDC*. URL: https://www.cdc.gov/niosh/topics/anthropometry/default.html.

[Döl+23] Nina Döllinger et al. "Are Embodied Avatars Harmful to our Self-Experience? The Impact of Virtual Embodiment on Body Awareness". In: (2023). DOI: 10.1145/3544548.3580918.

[EB22] M Eswaran and M V A Raju Bahubalendruni. "Challenges and opportunities on AR/VR technologies for manufacturing systems in the context of industry 4.0: A state of the art review". In: *Journal of Manufacturing Systems* 65 (2022), pp. 260–278. ISSN: 0278-6125. DOI: https://doi.org/10.1016/j.jmsy.2022.09.016. URL: https://www.sciencedirect.com/science/article/pii/S0278612522001650.

[Fat] Father.io. *Father.IO - Real Life gaming*. URL: https://father.io/.

[FBK15] Denis Fortun, Patrick Bouthemy, and Charles Kervrann. "Optical flow modeling and computation: A survey". In: *Computer Vision and Image Understanding* 134 (2015), pp. 1–21.

[FFS22] Mehrad Faridan, Marcus Friedel, and Ryo Suzuki. "UltraBots: Large-Area Mid-Air Haptics for VR with Robotically Actuated Ultrasound Transducers". In: *Adjunct Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*. UIST '22. ACM, Oct. 2022. DOI: 10.1145/3526114.3561350. URL: http://dx.doi.org/10.1145/3526114.3561350.

[Gao+23] Kyle Gao et al. *NeRF: Neural Radiance Field in 3D Vision, A Comprehensive Review*. 2023. arXiv: 2210.00379 [cs.CV].

[Gar23a] Angelica D. Garcia. *DST Technology Lab - NASA*. Sept. 2023. URL: https://www.nasa.gov/general/dst-technology-lab/.

[Gar23b] Angelica D. Garcia. *Systems Engineering Simulator - NASA*. Sept. 2023. URL: https://www.nasa.gov/general/systems-engineering-simulator/#:~:text=The%20Systems%20Engineering%20Simulator%20(SES,systems%20and%20other%20advanced%20concepts..

[Geo+23] Oliver George et al. "Augmented Reality in Medical Education: A Mixed Methods Feasibility Study". In: *Cureus* 15 (2023). DOI: 10.7759/cureus.36927.

[GN] BSG Web Group and NASA. *Search — 3D Resources*. URL: https://nasa3d.arc.nasa.gov/search/tools.

[Gol+20] Michael Goldhammer et al. "Intentions of Vulnerable Road Users—Detection and Forecasting by Means of Machine Learning". In: *IEEE Transactions on Intelligent Transportation Systems* 21.7 (2020), pp. 3035–3045. DOI: 10.1109/TITS.2019.2923319.

[Gon+21] Guilherme Gonçalves et al. "Systematic Review on Realism Research Methodologies on Immersive Virtual, Augmented and Mixed Realities". In: *IEEE Access* 9 (2021), pp. 89150–89161. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3089946.

[Gon+22] Eric J Gonzalez et al. "A Model Predictive Control Approach for Reach Redirection in Virtual Reality". In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI '22. New Orleans, LA, USA: Association for Computing Machinery, 2022. ISBN: 9781450391573. DOI: 10.1145/3491102.3501907. URL: https://doi.org/10.1145/3491102.3501907.

[Gooa] Google. *Build new augmented reality experiences that seamlessly blend the digital and physical worlds — ARCore — Google for Developers*. URL: https://developers.google.com/ar.

[Goob] Google. *Google Cardboard – Google VR*. URL: https://arvr.google.com/cardboard/.

[Gooc] Google. *WorldSense*. URL: https://developers.google.com/vr/discover/worldsense.

[GP18] Mar Gonzalez-Franco and Tabitha C. Peck. "Avatar Embodiment. Towards a Standardized Questionnaire". In: *Frontiers in Robotics and AI* 5 (2018). ISSN: 2296-9144. DOI: 10.3389/frobt.2018.00074. URL: https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2018.00074.

[Gro16] The Khronos® Group. *OpenXR - High-performance access to AR and VR —collectively known as XR— platforms and devices*. Dec. 2016. URL: https://www.khronos.org/api/index_2017/openxr.

[Gut10] Daniel A. Guttentag. "Virtual reality: Applications and implications for tourism". In: *Tourism Management* 31.5 (2010), pp. 637–651. ISSN: 0261-5177. DOI: https://doi.org/10.1016/j.tourman.2009.07.003. URL: https://www.sciencedirect.com/science/article/pii/S0261517709001332.

[Ham+22] Cecilia Hammar Wijkmark et al. "Journal of Applied Technical and Educational Sciences jATES Introducing Virtual Reality for Firefighter Skills Training Opinions from Sweden and Brazil". In: 12 (Oct. 2022), pp. 1–24. DOI: 10.24368/jates317.

[Haq+15] Mazhar Ul Haq et al. "Inverse kinematic analysis of three link mechanism for fin actuation of fish like micro device". In: *Journal of Biomimetics, Biomaterials and Biomedical Engineering* 24 (2015), pp. 77–81.

[Har+23] David Harris et al. "Exploring the role of virtual reality in military decision training". In: *Frontiers in Virtual Reality* 4 (Mar. 2023). DOI: 10.3389/frvir.2023.1165030.

[HC89] Joy Hirsch and Christine A. Curcio. "The spatial resolution capacity of human foveal retina". In: *Vision Research* 29.9 (1989), pp. 1095–1101. ISSN: 0042-6989. DOI: https://doi.org/10.1016/0042-6989(89)90058-8. URL: https://www.sciencedirect.com/science/article/pii/0042698989900588.

[Heo+18] Seongkook Heo et al. "Thor's Hammer: An Ungrounded Force Feedback Device Utilizing Propeller-Induced Propulsive Force". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–11. ISBN: 9781450356206. DOI: 10.1145/3173574.3174099. URL: https://doi.org/10.1145/3173574.3174099.

[HS23] Huggingface and Tyler Sehon. *blog/gaussian-splatting.md at main · huggingface/blog*. Dec. 2023. URL: https://github.com/huggingface/blog/blob/main/gaussian-splatting.md.

[Hu+21] Ruiqin Hu et al. "Research on force sensing and zero gravity motion simulation technology of industrial Robot". In: *Research Square (Research Square)* (Aug. 2021). DOI: 10.21203/rs.3.rs-762971/v1. URL: https://doi.org/10.21203/rs.3.rs-762971/v1.

[IKE17] IKEA. *Launch of new IKEA Place app – IKEA Global*. Sept. 2017. URL: https://www.ikea.com/global/en/newsroom/innovation/ikea-launches-ikea-place-a-new-app-that-allows-people-to-virtually-place-furniture-in-their-home-170912/.

[Jak23] Jakob. "The Ultimate AR Lens Guide and its implications - VRX by VR Expert". In: (May 2023). URL: https://vrx.vr-expert.com/the-ultimate-ar-lens-guide-and-its-implications/.

[JJ00] William Jones and Mark J. Jansen. *Space Tribology*. Dec. 2000. DOI: 10.1201/9780849377877.ch31. URL: https://doi.org/10.1201/9780849377877.ch31.

[JSN06] Nané Jordan, Joseph H. Saleh, and Dava Newman. "The extravehicular mobility unit: A review of environment, requirements, and design changes in the US spacesuit". In: *Acta astronautica* 59.12 (Dec. 2006), pp. 1135–1145. DOI: 10.1016/j.

actaastro.2006.04.014. URL: https://doi.org/10.1016/j.actaastro.2006.04.014.

[Kai] Kai-Morich. *GitHub - kai-morich/SimpleUsbTerminal: Android terminal app for devices with a serial / UART interface connected with a USB-to-serial-converter.* URL: https://github.com/kai-morich/SimpleUsbTerminal.

[Kei+23] Julian Keil et al. "Measuring Physiological Responses to Visualizations of Urban Planning Scenarios in Immersive Virtual Reality". In: 73 (May 2023). DOI: 10.1007/s42489-023-00137-7.

[Keo18] Jon Keon. "Creating Android JAR and AAR plugins for Unity - Karman Interactive - Medium". In: (May 2018). URL: https://medium.com/karmanltd/creating-android-jar-and-aar-plugins-for-unity-c293bb5258c9.

[Ker+23] Bernhard Kerbl et al. "3D Gaussian Splatting for Real-Time Radiance Field Rendering". In: *ACM Transactions on Graphics* 42.4 (July 2023). URL: https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

[KK17] Pavol Kaleja and Mária Kozlovská. "Virtual Reality as Innovative Approach to the Interior Designing". In: *Selected Scientific Papers - Journal of Civil Engineering* 12 (Jan. 2017). DOI: 10.1515/sspjce-2017-0011.

[Kni19] Joe Knipp. *What is neutral buoyancy and how to achieve perfect neutral buoyancy while diving.* July 2019. URL: https://scubaguru.org/nj-scuba-articles/what-is-neutral-buoyancy-and-how-to-achieve-perfect-neutral-buoyancy-while-diving/.

[Lea] Magic Leap. *Magic Leap 2.* URL: https://www.magicleap.com/magic-leap-2.

[Li+20] Guan Li et al. "Standing-Posture Recognition in Human–Robot Collaboration Based on Deep Learning and the Dempster–Shafer Evidence Theory". In: *Sensors* 20.4 (2020). ISSN: 1424-8220. DOI: 10.3390/s20041158. URL: https://www.mdpi.com/1424-8220/20/4/1158.

[Liu+19] Lee-Kai Liu et al. "Sensorless Control with Friction and Human Intention Estimation of Exoskeleton Robot for Upper-limb Rehabilitation". In: *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO).* 2019, pp. 290–296. DOI: 10.1109/ROBIO49542.2019.8961760.

[LP17] Kevin Lynch and Frank C. Park. *Modern robotics: mechanics, planning, and control.* May 2017. URL: https://openlibrary.org/books/OL28638262M/Modern_Robotics.

[mag] magicplan. *magicplan — Construction & Floor Plan App For Contractors.* URL: https://www.magicplan.app/.

[Mar+21] Humberto Marín-Vega et al. "ZeusAR: a process and an architecture to automate the development of augmented reality serious games". In: *Multimedia Tools and Applications* (2021), pp. 1–35. DOI: 10.1007/S11042-021-11695-1.

[Mar23] Kelli Mars. *Building on a mission: Astronaut training Facilities - NASA.* Sept. 2023. URL: https://www.nasa.gov/history/building-on-a-mission-astronaut-training-facilities/.

[McM24] Malcolm McMillan. *Meta Quest 2 review.* Mar. 2024. URL: https://www.tomsguide.com/reviews/oculus-quest-2-review.

[Met] Virtual Method. *OBI Physics for Unity - Big Picture.* URL: https://obi.virtualmethodstudio.com/manual/6.3/index.html.

[Met17] Virtual Method. *ObI Rope — Physics — Unity Asset Store.* Sept. 2017. URL: https://assetstore.unity.com/packages/tools/physics/obi-rope-55579.

[MG10] Sandra K. Moore and Matthew A. Gast. "21st Century extravehicular activities: Synergizing past and present training methods for future spacewalking success". In: *Acta astronautica* 67.7-8 (Oct. 2010), pp. 739–752. DOI: 10.1016/j.actaastro.2010.06.016. URL: https://doi.org/10.1016/j.actaastro.2010.06.016.

[Mic] Microsoft. *Microsoft HoloLens — Mixed Reality Technology for business.* URL: https://www.microsoft.com/en-us/hololens.

[MKH19] Dae-Hoon Moon, Donghan Kim, and Young-Dae Hong. "Development of a Single Leg Knee Exoskeleton and Sensing Knee Center of Rotation Change for Intention

Detection". In: *Sensors* 19.18 (2019). ISSN: 1424-8220. DOI: 10.3390/s19183960. URL: https://www.mdpi.com/1424-8220/19/18/3960.

[MNL99] David M Mount, Nathan S Netanyahu, and Jacqueline Le Moigne. "Efficient algorithms for robust feature matching". In: *Pattern recognition* 32.1 (1999), pp. 17–38.

[Mor19] Kai Morich. *Serial USB Terminal - Apps op Google Play*. 2019. URL: https://play.google.com/store/apps/details?id=de.kai_morich.serial_usb_terminal&hl=nl&pli=1.

[MS22] Denis Manolescu and Emanuele Secco. "Design of a 3-DOF Robotic Arm and implementation of D-H Forward Kinematics". In: Sept. 2022.

[NASa] NASA. *Astronaut tests SAFER Backpack - NASA*. URL: https://www.nasa.gov/image-article/astronaut-tests-safer-backpack/.

[NASb] NASA. *Engineering DOUG Graphics for Exploration (EDGE)(MSC-24663-1) — NASA Software Catalog*. URL: https://software.nasa.gov/software/MSC-24663-1.

[NASc] NASA. *Extravehicular Activities Art Collection*. URL: https://www.nasa.gov/johnson/exhibits/extravehicular-activities/.

[NASd] NASA. *Virtual Reality Training Lab - NASA*. URL: https://www.nasa.gov/virtual-reality-lab-doug/.

[NAS+10] NASA et al. *Human Integration Design Handbook (HIDH)*. Revision 1. Washington DC, us: National Aeronautics and Space Administration, Jan. 2010. URL: https://www.nasa.gov/feature/human-integration-design-handbook/.

[NAS07] NASA. *Fastenings: Velcro*. 2007. URL: https://www.nasa.gov/history/alsj/WOTM/WOTM-Velcro.html#:~:text=Velcro%20was%20used%20in%20many,for%20easy%20viewing%20or%20access..

[NAS20] NASA. *NASA FaCtSFS-2020-4-540-GSFC*. Tech. rep. 2020. URL: https://smd-cms.nasa.gov/wp-content/uploads/2023/04/hst-tools-fact-sheet.pdf.

[NAS95] NASA. *MAN-SYSTEMS INTEGRATION STANDARDS*. Revision B. Vol. I. July 1995. URL: https://msis.jsc.nasa.gov/sections/section14.htm.

[Ngu+20] Jilian Nguyen et al. "Screen door effect reduction using mechanical shifting for virtual reality displays". In: *Optical Architectures for Displays and Sensing in Augmented, Virtual, and Mixed Reality (AR, VR, MR)*. Ed. by Bernard C. Kress and Christophe Peroz. Vol. 11310. International Society for Optics and Photonics. SPIE, 2020, 113100P. DOI: 10.1117/12.2544479. URL: https://doi.org/10.1117/12.2544479.

[Nie+22] Jingxin Nie et al. "Wide color gamut, low non-visual effect, and their stabilities related to luminous properties in four-primary-color display". In: *Optics & Laser Technology* 156 (2022), p. 108565. ISSN: 0030-3992. DOI: https://doi.org/10.1016/j.optlastec.2022.108565. URL: https://www.sciencedirect.com/science/article/pii/S0030399222007149.

[Pad+] Eddie Paddock et al. *Overview of NASA EVA Virtual Reality training, past, present and future*. Tech. rep. URL: https://nvite.jsc.nasa.gov/presentations/b3/D2_3_Virtual_Reality_Panel_Paddock.pdf.

[Pan+21] Xuran Pan et al. "3D Object Detection With Pointformer". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 7463–7472.

[PCg23] PCgamer. *Pico 4 VR headset*. May 2023. URL: https://www.pcgamer.com/pico-4-ve-headset-review/.

[PCM21a] PCMag. *HTC Vive Pro 2 Review*. Aug. 2021. URL: https://www.pcmag.com/reviews/htc-vive-pro-2.

[PCM21b] PCMag. *Valve Index VR Kit review*. July 2021. URL: https://www.pcmag.com/reviews/valve-index-vr-kit.

[PCM23] PCMag. *Meta Quest 3 Review*. Oct. 2023. URL: https://www.pcmag.com/reviews/meta-quest-3.

[Pix24] 512 Pixels. *Apple Vision Pro Review: It's a carousel.* Feb. 2024. URL: https://512pixels.net/2024/02/apple-vision-pro-review/.

[Pol+23] Elizaveta Polishchuk et al. "The Theoretical Background of Virtual Reality and Its Implications for the Tourism Industry". In: *Sustainability* 15.13 (2023). ISSN: 2071-1050. DOI: 10.3390/su151310534. URL: https://www.mdpi.com/2071-1050/15/13/10534.

[Pur23] Kaustubh Purohit. "Enhancing Aviation Safety through Haptic Glove-Enabled Virtual Reality Simulations: An Economical and Scalable Approach for Pilot Training". In: (June 2023). DOI: 10.32920/23330219.v1. URL: https://rshare.library.torontomu.ca/articles/thesis/Enhancing_Aviation_Safety_through_Haptic_Glove-Enabled_Virtual_Reality_Simulations_An_Economical_and_Scalable_Approach_for_Pilot_Training/23330219.

[Rav+22] Ole Viktor Ravna et al. "Supporting Peer-Learning with Augmented Reality in Neuroscience and Medical Education". In: *Smart Education and e-Learning - Smart Pedagogy.* Ed. by Vladimir L. Uskov, Robert J. Howlett, and Lakhmi C. Jain. Singapore: Springer Nature Singapore, 2022, pp. 299–308. ISBN: 978-981-19-3112-3.

[RD85] Richard M. Ryan and Edward L. Deci. *Intrinsic Motivation and Self-Determination in Human Behavior*. New York: Plenum Press, 1985. ISBN: 978-0306413570.

[RQ05] Agency For Healthcare Research and Quality. *NASA Task Load Index — Digital Healthcare Research.* 2005. URL: https://digital.ahrq.gov/health-it-tools-and-resources/evaluation-resources/workflow-assessment-health-it-toolkit/all-workflow-tools/nasa-task-load-index.

[Sam21] Samsung. *Gear VR with Controller, Virtual Reality support — Samsung Care US.* Mar. 2021. URL: https://www.samsung.com/us/support/mobile/virtual-reality/gear-vr/gear-vr-with-controller/.

[Sar+20] Paul-Edouard Sarlin et al. "Superglue: Learning feature matching with graph neural networks". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2020, pp. 4938–4947.

[Sel+20] Elmedin Selmanović et al. "Improving Accessibility to Intangible Cultural Heritage Preservation Using Virtual Reality". In: *J. Comput. Cult. Herit.* 13.2 (May 2020). ISSN: 1556-4673. DOI: 10.1145/3377143. URL: https://doi.org/10.1145/3377143.

[SFR01] Thomas Schubert, Frank Friedmann, and Holger Regenbrecht. "The Experience of Presence: Factor Analytic Insights". In: *Presence* 10 (June 2001), pp. 266–281. DOI: 10.1162/105474601300343603.

[Sin+19] Christian Sinnott et al. *Underwater Virtual Reality System for Neutral Buoyancy Training: Development and Evaluation.* Nov. 2019. DOI: 10.1145/3359996.3364272. URL: https://doi.org/10.1145/3359996.3364272.

[Sla04] Mel Slater. "How Colorful Was Your Day? Why Questionnaires Cannot Assess Presence in Virtual Environments". In: *Presence* 13.4 (Aug. 2004), pp. 484–493. ISSN: 1054-7460. DOI: 10.1162/1054746041944849.

[Sol+23] Divyarajsinh Solanki et al. "AUGMENTED AND VIRTUAL REALITIES: THE FUTURE OF BUILDING DESIGN AND VISUALIZATION". In: *Civil and Environmental Engineering Reports* 33 (June 2023). DOI: 10.59440/ceer-2023-0002.

[SSO22] Tomáš Stejskal, Jozef Svetlík, and Štefan Ondočko. "Mapping Robot Singularities through the Monte Carlo Method". In: *Applied Sciences* 12.16 (2022). ISSN: 2076-3417. DOI: 10.3390/app12168330. URL: https://www.mdpi.com/2076-3417/12/16/8330.

[Stu] Android Studio. *Create an Android library.* URL: https://developer.android.com/studio/projects/android-library.html#groovy.

[Sun+21] Jiaming Sun et al. "LoFTR: Detector-free local feature matching with transformers". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2021, pp. 8922–8931.

[Suz+21] Ryo Suzuki et al. "HapticBots: Distributed Encountered-type Haptics for VR with Multiple Shape-changing Mobile Robots". In: *The 34th Annual ACM Symposium on User Interface Software and Technology*. UIST '21. Virtual Event, USA: Association for Computing Machinery, 2021, pp. 1269–1281. ISBN: 9781450386357. DOI: 10.1145/3472749.3474821. URL: https://doi.org/10.1145/3472749.3474821.

[Swa+95] Patrick L. Swaim Jr. et al. *The Charlotte vM Intra-Vehicular Robot*. Tech. rep. N95-23703. 1995. URL: https://ntrs.nasa.gov/search.jsp?R=19950017283.

[Sys20] Rotary Systems. *FAQs ≫ Rotary Systems, Inc.* Dec. 2020. URL: https://rotarysystems.com/support/faqs/.

[Tec] Unity Technologies. *Unity - Manual: Physics*. URL: https://docs.unity3d.com/Manual/class-PhysicsManager.html.

[Tut22] Valem Tutorials. *How to make Physics hands in VR - PART 1 - Unity VR Tutorial*. Jan. 2022. URL: https://www.youtube.com/watch?v=VG8hLKyTiJQ.

[Unia] Unity. *Unity - Manual AAR plug-ins and Android Libraries*. URL: https://docs.unity3d.com/2021.2/Documentation/Manual/AndroidAARPlugins.html.

[Unib] Unity. *Unity - Manual JAR plug-ins*. URL: https://docs.unity3d.com/2021.2/Documentation/Manual/AndroidJARPlugins.html#javawithhelper.

[Uso+00] Martin Usoh et al. "Using Presence Questionnaires in Reality". In: *Presence* 9.5 (Oct. 2000), pp. 497–503. ISSN: 1054-7460. DOI: 10.1162/105474600566989.

[Voi21] Voitanium. *Extending Unity Player Activity and calling Unity C# Methods from the Unity Android Plugin*. June 2021. URL: https://www.youtube.com/watch?v=R-cYild8ZYs.

[Vor+04] P Vorderer et al. "MEC spatial presence questionnaire (MEC-SPQ): Short documentation and instructions for application. Report to the European Community, Project Presence: MEC". In: *MEC (IST-2001-37661)* (2004).

[Wel93] Chris Welman. *INVERSE KINEMATICS AND GEOMETRIC CONSTRAINTS FOR ARTICULATED FIGURE MANIPULATION*. Jan. 1993. URL: https://ir.lib.sfu.ca/item/5706.

[Wen+18] Andreas Wendemuth et al. "Intention-Based Anticipatory Interactive Systems". In: *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 2018, pp. 2583–2588. DOI: 10.1109/SMC.2018.00442.

[Wij+08] Maarten Wijntjes et al. "Haptic Perception of Real and Virtual Curvature". In: vol. 5024. June 2008, pp. 361–366. ISBN: 978-3-540-69056-6. DOI: 10.1007/978-3-540-69057-3_46.

[Wil21] Daniel Wilches. *Ardity: Arduino + Unity communication made easy — Integration — Unity Asset Store*. July 2021. URL: https://assetstore.unity.com/packages/tools/integration/ardity-arduino-unity-communication-made-easy-123819.

[Woj23] Wojciech. *Nits vs Lumens vs Luminance: Understanding Display Brightness Metrics*. Aug. 2023. URL: https://riverdi.com/blog/nits-vs-lumens-vs-luminance-understanding-display-brightness-metrics.

[Wri19] Michael Eisenstein Science Writer. *Digital cinema faces a bright future*. Dec. 2019. URL: https://www.photonics.com/Articles/Digital_Cinema_Faces_a_Bright_Future/a65257.

[WS98] Bob G. Witmer and Michael J. Singer. "Measuring Presence in Virtual Environments: A Presence Questionnaire". In: *Presence* 7.3 (June 1998), pp. 225–240. ISSN: 1054-7460. DOI: 10.1162/105474698565686.

[XRE] XREAL. *XREAL*. URL: https://www.xreal.com/.

[ZD23] Yifei Zhang and Thomas Doyle. "Integrating intention-based systems in human-robot interaction: a scoping review of sensors, algorithms, and trust". In: *Frontiers in Robotics and AI* 10 (2023). ISSN: 2296-9144. DOI: 10.3389/frobt.2023.1233328. URL: https://www.frontiersin.org/articles/10.3389/frobt.2023.1233328.

[Zha+22] Xiaoming Zhao et al. "Alike: Accurate and lightweight keypoint detection and descriptor extraction". In: *IEEE Transactions on Multimedia* (2022).

[ZHL22] Isiah Zaplana, Hugo Hadfield, and Joan Lasenby. “Singularities of Serial Robots: Identification and Distance Computation Using Geometric Algebra”. In: *Mathematics* 10.12 (2022). ISSN: 2227-7390. DOI: 10.3390/math10122068. URL: https://www.mdpi.com/2227-7390/10/12/2068.