A Study of the User Experience of Redirected Walking in Small Tracking Spaces

LINDA KRUEGER

A THESIS SUBMITTED FOR THE DEGREE OF
Doctor of Philosophy

DEPARTMENT OF COMPUTER SCIENCE

MAYNOOTH UNIVERSITY

September 5, 2025

Head of Department: Dr. Aidan Mooney
Supervisor: Dr. Ralf BIERIG and
Dr. Charles MARKHAM

Contents

List of Figures

List of Tables

Abstract

In Virtual Reality, real walking is a natural and intuitive form of navigation. With real walking the user physically walks to move in a virtual environment. However, with real walking the size of the physical space needs to match the virtual space. This is rarely feasible, especially for large virtual worlds. Redirected Walking seeks to solve this problem by decoupling the mapping between the virtual trajectory of the user and their physical movement. There are two broad categories of redirected walking. Gain manipulation changes how the user's viewpoint is moved. For example, the user might turn at a different rate in the virtual environment than they are physically turning. This is called rotation gain. Environment manipulation changes the virtual environment itself - for example an expanding room.

This work aims to reduce the amount of physical space required for redirected walking while balancing the impact of redirected walking on user experience. The first part of this work looks at the performance of existing generalised redirected walking methods in a small tracking space. It was found that these existing methods have limited use in such small tracking spaces at levels that are comfortable for users. Next we look at the turning accuracy of users with larger rotation gain. We found, for example, at higher rotation gain levels users turned more than expected and the visual detail of the virtual environment impacted their turning accuracy.

Finally, based on these findings, a new method of virtual rotation is created: Segment Addition. With Segment Addition the virtual environment expands and contracts around the user so they must turn further or shorter to reach their goal. It is an example of an environmental manipulation method. A user study using Segment Addition found the environment natural and comfortable for users even when large changes were made to the virtual environment. This demonstrates Segment Addition as a promising new technique for redirecting users in small tracking spaces.

Declaration of Authorship

I, Linda KRUEGER, declare that this thesis titled, "A Study of the User Experience of Redirected Walking in Small Tracking Spaces" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

This thesis has been prepared in accordance with the PhD regulations of Maynooth University and is subject to copyright. For more information see PhD Regulations (December 2022).

Signed: Linda Krueger

Date: 02/07/2025

"Its continuing mission: to explore strange new worlds; to seek out new life and new civilizations; to boldly go where no one has gone before!"

Star Trek: The Next Generation

Acknowledgements

I would like to thank my supervisor, Prof. Ralf Bierig for giving me the opportunity to pursue this research. For being there through each step of the process from the beginning all the way to this thesis. Thank you for the helpful feedback on the (many!) drafts I have submitted.

My co-supervisor, Prof. Charles Markham, for the helpful suggestions and constructive conversations on human behaviour and experiment design. Thank you for lending an ear and offering advice on how to move forward whenever I got stuck.

The Computer Science department at Maynooth University for funding this research and providing the space for work and collaboration.

Prof. David Malone and Dr Rafael de Andrade Moral for their expert knowledge on data analysis and help with statistical modelling.

My fellow PhD students and demonstrators, especially Azeema, Ben, Mark and Rui, for walking this path alongside me. Thank you for the lunch time chats and hang outs, the advice, commiseration and shared excitement over the possibilities of future technology.

My friends Rebecca and Robin for their encouragement and listening to my rambling. Lydia, Maria and Finn, thank you for the support and grammar corrections of the chapters I dropped into your laps. Without you, this work would be far less readable!

My parents, Anja and Tilo, for providing a secure base to come back to. Thank you for the warm meals, advice and encouragement. Finally, to my partner Oleg, you have been my rock throughout this journey. Thank you for being there when I needed to vent, for putting up with my stress and for always believing in me.

For Eli - the rest is yours

Chapter 1

Introduction

Humans have always sought to share experiences - from story telling around a fire and paintings on cave walls, to the written word, to theatre, to modern television and video games. New technology opens new ways to share experiences and expands what it is possible to communicate. Virtual reality offers a deep sense of immersion to the user, a sense that the surrounding virtual environment is as real as they are. Virtual reality brings us to places we would never be able to visit ourselves (Beck, Rainoldi, and Egger, 2019), lets us inhabit different bodies (Kilteni, Groten, and Slater, 2012) and experience moments from another's perspective (Christofi and Michael-Grigoriou, 2017). In short, virtual reality creates new possibilities for how we communicate and share. In virtual reality it becomes possible to recreate our wildest dreams.

To foster this deep sense of immersion, users need to be able to explore and interact with the virtual environment in ways that feel natural. Real walking has the user physically walk through a real-world tracking space as they navigate in a virtual environment. Real walking is the most natural form of navigation in virtual reality (Homami, Quigley, and Barrera Machuca, 2025) as it facilitates presence (Slater, 2009; Langbehn, Lubos, and Steinicke, 2018b). The body-based cues from vestibular and proprioceptive senses used when walking aid in navigation and search-related tasks (Ruddle and Lessels, 2009). Many applications could benefit from the use of real walking to create immersive experiences for users.

For example, Star Wars: Droid Repair Bay[1] is an immersive experience where the user repairs robots from a large room in a space station (see Figures 1.1). In it's current

[1]Such as in Star Wars: Droid Repair Bay -
https://store.steampowered.com/app/726910/Star_Wars_Droid_Repair_Bay/

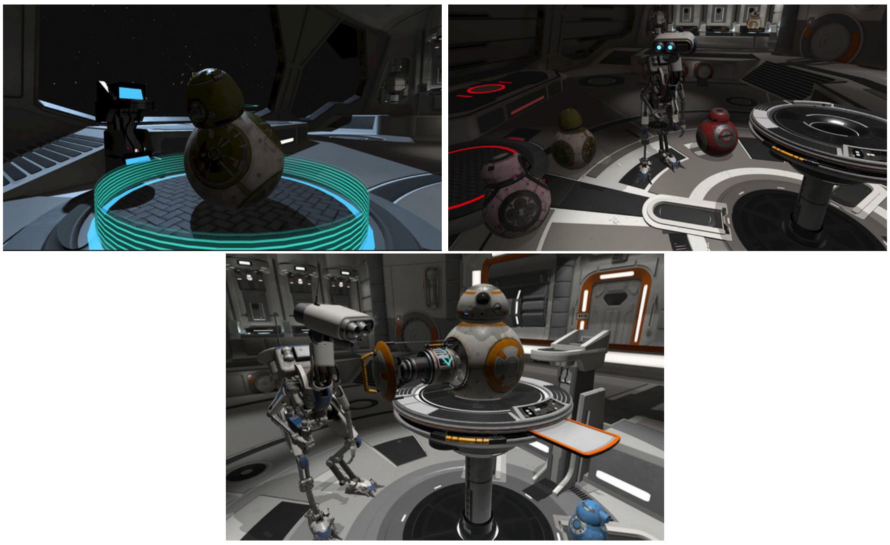


FIGURE 1.1: Screenshots taken from the official steam page of the 'Star Wars: Droid Repair Bay' video game. The user repairs robots ('droids') on a virtual space station with different interactive areas.

version, the user operates virtual machinery to move the robot close to them to repair it. After the robot is repaired, the little robot rolls around the room, interacting with different aspects of the environment. However, with the use of a large enough physical space to move in, the user could theoretically move around the virtual room in the space station. In this case, the user could physically move the small robot from one space to another within the room by guiding it with their own walking. They could chase the robot around the room or bring it to different interactive parts of the room. This could provide a more immersive experience to the user, as they feel like they can freely and easily move around in this interactive virtual room to complete the narrative goal of repairing robots. Many other applications to benefit from similar real walking capabilities - a virtual dodge ball match where the user can freely move around the court, a dinosaur museum tour where the user walks between exhibits at their leisure and can easily look at the relative size of the exhibits from all angles or a fire escape training exercise where the user needs to navigate through a virtual environment full of winding corridors and smoke.

However, with all these application examples, while virtual environments can be expansive, the real-world space the user can walk in is limited. Especially consumer grade hardware often uses limited tracking space. Many new users experience virtual reality in their own homes where the size of their room limits the available tracking space. Redirected walking seeks to alleviate this limitation by letting the user walk in a virtual environment that is larger than the physical tracking space available (Razzaque, Kohn, and Whitton, 2005). Redirected walking changes the way a user walks in a virtual environment compared to their movements in the tracking space. This allows the user to remain safely within the tracking space while they freely explore the virtual environment. There are multiple approaches to *redirected walking* that fit into two broad categories:

- **Viewport Manipulation** changes how the user moves in the virtual environment compared to the tracking space. The user might turn at a faster rate in the virtual environment than their physical turning speed. Or they might walk twice as far with every step in the virtual space compared to the tracking space.

- **Environment Manipulation** changes the virtual environment around the user to keep them inside the tracking space. Rooms might be bigger when the user is inside them but shrink when they have left. Doors might change walls so that the virtual room behind the door is within the tracking space. A twisting corridor might turn the user back the way they came while presenting them with a new virtual room.

Each redirected walking method comes with its own set of benefits and drawbacks. This thesis looks at the problem of effective redirected walking in small tracking spaces specifically. To this end, we consider the following questions:

- How effective are existing, generalised redirected walking methods in small tracking spaces?

- Can rotation gain be adapted into redirected walking algorithms more effectively for small tracking spaces?

- Are there alternatives to rotation gain that can make better use of redirected walking in small tracking spaces?

1.1 Contributions

This thesis studies the user experience of redirected walking and redirected walking algorithms with particular attention given to small tracking spaces in relation to the three research questions developed above. This thesis provides multiple contributions to expand the field of redirected walking, including a new redirected walking method based on the findings of multiple user studies.

First, the current state of the field of redirected walking is summarised in the literature review in Chapter 2. The different methods and approaches to redirected walking are detailed, including the strengths and weaknesses of each. An emphasis is placed on how real users respond to the differing redirection techniques in small tracking spaces, both in terms of user experience and task completion.

The first contribution is a user study that explores research question 1 - the **user experience of generalised reactive redirected walking algorithms in small tracking spaces** where users are free to roam within the virtual environment (see Chapter 3). It was found that in such small spaces, users had similar task performance with redirected walking algorithms compared to a baseline without consistent redirection. Additionally, users preferred simple algorithms that redirected them less to the more complex algorithms. Based on these results, guidelines were developed for designers implementing redirected walking algorithms in small tracking spaces.

The second contribution relates to research question 2 above, the **user response to rotation gain under different environment conditions** (see Chapter 4). Rotation gain is a redirected walking technique that changes how far the user turns in the virtual environment compared to the tracking space. The turning accuracy of 38 participants was measured in a virtual environment with a single visual cue and another with many visual cues at different levels of rotation gain. Users were more strongly affected by rotation gain in the virtual environment with multiple visual cues compared to the single visual cue environment. While the direction of the turn had no impact on user response, both the level of gain and the amount the user was asked to turn did have an affect on user response. These results increase our knowledge of the relative impact of different amounts of rotation gain and the factors that should be considered when implementing such gain in virtual environments, especially when used with predictive redirected walking algorithms where rotation accuracy is important to the efficiency of the algorithm.

The third contribution is the introduction of **Segment Addition - a new redirected walking method** (see Chapter 5). This method relates to research question 3 and provides a new, alternative approach to strong rotation gain for small tracking spaces. Segment Addition adds or removes sections of the environment (dubbed 'slices') to the virtual environment when the user rotates. The slices are added outside the user's field of view so that they have to turn further to reach their goal in the virtual space. There is a direct mapping of the user's movements between the virtual space and the physical tracking space - as they turn more in the virtual space they also turn more in the physical space. The results of our initial user study shows the benefits of the new method. Despite participants becoming aware that they seemed to be turning further inside the virtual environment, they felt the turn to be natural and pleasant. Additionally, simulator sickness remained low even though participants turned repeatedly inside the virtual environment with significant increases and decreases to the turn amount. The results of the initial study suggest Segment Addition is a promising new method for redirected walking.

Finally Chapter 6 summarises the work presented in this thesis and outlines potential avenues for future research.

1.2 Artifacts

Papers

- L. Krueger, C. Markham, and R. Bierig, "Exploring User Turning Perception and Comfort of the Segment Addition Redirected Walking Technique," 3rd International ACM Conference of the Greek SIGCHI Chapter (ACM CHIGreece 2025), Syros Greece, Association for Computing Machinery, 2025, (accepted)
- L. Krueger, C. Markham, and R. Bierig, "Investigation of Redirection Algorithms in Small Tracking Spaces," 30th ACM Symposium on Virtual Reality Software

and Technology (VRST ’24), Trier Germany, Association for Computing Machinery, 2024, Article 14, 1–9,

https://doi.org/10.1145/3641825.368774

- L. Krueger, C. Markham and R. Bierig, "Comparison of Two Novel Environmental Manipulation Methods for Rotating VR Users," 10th International Conference on Virtual Reality (ICVR), Bournemouth, United Kingdom, Institute of Electrical and Electronic Engineers, 2024, 362-368,

  https://doi.org/10.1109/ICVR62393.2024.10868877

- L. Krueger, C. Markham and R. Bierig, "An Experimental System to Measure Accuracy of Rotation Gain Under Different Conditions in Virtual Reality Systems," 2023 34th Irish Signals and Systems Conference (ISSC), Dublin, Ireland, Institute of Electrical and Electronic Engineers, 2023, 1-6,

  https://doi.org/10.1109/ISSC59246.2023.10162127

Systems

- Turning Accuracy Measuring System:

  https://github.com/chionic/VR-Turning-Accuracy-Environment

- Segment Addition implementation for User Study:

  https://github.com/chionic/Segment-Addition-Demo

Presentations

- L. Krueger, "Segment Addition: User Response to the Virtual Reality Redirected Walking Technique", XR Showcase 2025, D-REAL, SFI Centre for Research Training in Digitally Enhanced Reality, Dublin Ireland

- L. Krueger, R. De Andrade Moral, C. Markham and R. Bierig, "Rotation Accuracy in Virtual Reality Applications with Different Rotation Gain Levels", 43rd Conference on Applied Statistics in Ireland (CASI 2023), Killarney, Ireland, Poster Presentation, https://casi.ie/2023/p5/

Chapter 2

Literature Review

Virtual Reality (VR) technology continues to improve since first entering the consumer market a decade ago. In the past two years, multiple companies have announced new Head-Mounted Displays (HMD) for using XR systems, such as Apple's Vision Pro headset[1], Meta's Quest 3 and Quest 3S[2], and the Vive XR Elite launched by HTC Vive[3]. In this thesis the previous generation of HMDs have been used, such as the Meta quest 2 and the HTC Vive Pro Eye. VR has found application in healthcare (Iqbal et al., 2024), education (Lampropoulos and Kinshuk, 2024), training (Abich et al., 2021), tourism (Lurdes Calisto and Sarkar, 2024) and architecture (Safikhani et al., 2022), among others.

One of the most common tasks performed in VR is navigation through a virtual environment. There are many methods to navigate in VR (Al Zayer, MacNeilage, and Folmer, 2020; Martinez, Wu, and McMahan, 2022). Initially, controller-based navigation was adapted from video games for use with VR. With controller-based navigation the user holds a controller in their hand and presses buttons or uses a joystick to control their movement while physically staying in one spot, either standing or sitting. While requiring very little space, controller techniques often cause simulator sickness in users as they stay still in reality while feeling as though they are moving in the virtual space.

[1]See Apple Vision Pro announcement, reported by Apple: https://www.apple.com/newsroom/2023/06/introducing-apple-vision-pro/ (access, 20th July 2023
[2]See https://www.meta.com/ie/quest/quest-3s/
[3]See https://www.vive.com/us/product/vive-xr-elite/overview/

Al Zayer, MacNeilage, and Folmer, 2020 considers different methods for navigation in VR. Steering-based techniques have the user constantly moving forward in the virtual space. The user can change their trajectory by moving a part of their body - for example they might move in the direction that their head is rotated or where their gaze is. In contrast, with selection-based techniques the user stays in one spot in the virtual environment until they choose the exact location they wish to move to. When they choose this next location, they are instantly transported there through the use of teleportation or a route is planned for them and the system automatically moves the user along the planned route without further input. Teleportation allows for quick movement through the virtual world at the cost of users having a weaker sense of the layout of the environment. Manipulation-based methods present the user with a smaller representation of the virtual environment, avatars or objects that they can use to zoom in and out of a landscape (see Danyluk et al., 2021 for an overview). These methods provide users quick access to large virtual environments while still allowing them to get a sense of the layout of the virtual environment.

In contrast to the virtual methods described above, where the user moves virtually with limited physical motion, walking-based techniques aim to emulate real walking. Real walking has the user physically walk through a real-world tracking space as they navigate in a virtual environment. It is one of the most natural navigation methods available for VR (Homami, Quigley, and Barrera Machuca, 2025). It improves the user's sense of presence compared to other common locomotion techniques such as walking-in-place (Slater, 2009; Usoh et al., 1999; Homami, Quigley, and Barrera Machuca, 2025) or joystick control (Langbehn, Lubos, and Steinicke, 2018b). Additionally, the body-based cues from vestibular and proprioceptive senses used when walking help with navigation and search-related tasks in VR (Ruddle and Lessels, 2009). Users often prefer natural walking to other navigation methods (Mayor, Raya, and Sanchez, 2021; Homami, Quigley, and Barrera Machuca, 2025). With these advantages many applications could benefit from the use of real walking.

However, while virtual environments can be expansive, the real-world space the user can walk in is limited. With Partial Gait techniques (Al Zayer, MacNeilage, and

Folmer, 2020) users make similar movements to walking while staying in place. This is achieved by either having the user walking-in-place or with the use of treadmills. With walking-in-place gestures are used to move in the virtual environment while the user stays in the same place. For example, the user might be marching on the spot in the physical space while they move forward in the virtual environment. Other gestures for walking-in-place techniques include wiping, tapping, knee-bending and single-stepping (see Al Zayer, MacNeilage, and Folmer, 2020). These methods approximate real walking and are generally considered more natural than controller-based methods of navigation. However, they are seen as less natural than real walking while also requiring some physical exertion for the user (Usoh et al., 1999). Additionally, some walking-in-place methods cause positional drift over time where the user moves from their initial position over time. This drift needs to be accounted for in longer VR experiences in small spaces.

Gait-negation uses equipment such as treadmills or specific flooring and shoes to allow the user to move forward while keeping them in place. Treadmills and other specific locomotion hardware for VR, while commercially available[4], require dedicated space and are still expensive, thus they are only feasible for enthusiasts, research labs, or specialised businesses, such as VR arcades. Consumer grade hardware uses limited tracking space, with many new users experiencing virtual reality in their own homes in bounded rooms.

Finally, redirected walking seeks to alleviate the spatial limitation of real walking by letting the user walk in a virtual environment that is larger than the physical tracking space available (Razzaque, Kohn, and Whitton, 2005). Redirected walking changes the way a user walks in a virtual environment compared to their movements in the tracking space. The two major approaches to *redirected walking* are:

- **Gain manipulation** changes the position or rotation of the viewport of the user in the virtual space. This decouples the physical and virtual movements of the user - the user might move twice as far virtually with every step or rotate less in the virtual world than their physical rotation.

[4]E.g., Omni One by Virtuix: https://omni.virtuix.com/

- **Environment Manipulation** changes the environment around the user. Either the virtual environment is altered to fit inside the physical tracking space before the user ever enters the virtual world or hidden parts of the virtual environment are altered while the user is in VR. For example, a room might be bigger while the user is inside it and then shrink when the user moves into a different room.

The redirected walking methods explored in this review all lead back to one of these two fundamental techniques. Figure 2.1 shows an overview of the different types of redirected walking, moving from general approaches to specific techniques. Environment manipulation methods include static approaches that change the virtual environment before the user enters VR either by mapping the larger space onto the smaller tracking space using spatial compression (see 2.3.1) or by dividing the space into cells the size of the tracking space (see 2.3.1). It also includes dynamic approaches that change the space while the user is inside the virtual environment (see 2.3.2). Gain Manipulation methods include many different types of gain, these gains can be combined into redirected walking algorithms that aim to optimise how the gains are used (see Section 2.1). There are three different approaches to redirected walking algorithms - reactive algorithms consider only the current position of the user, predictive algorithms aim to forecast where the user will move next and machine learning algorithms train on large datasets to optimise gain (see Section 2.2).

Sometimes different techniques are combined to allow users to walk in even smaller tracked spaces (Suma et al., 2012). It is possible to tailor a virtual environment to a very small tracking space, such as a 2m $\times$ 2m or 3.5m $\times$ 3.5m (as used in Chapter 3) if the layout of the tracking space is known ahead of time and the paths inside the virtual environment are heavily constrained (Eklund, 2022; Ropelato, Menozzi, and Huang, 2022; Li and Fan, 2023). However, these techniques cannot be generalised to unknown virtual environments and tracking spaces. Both hardware such as eye tracking and software design techniques (for example, distractors) can be used to further enhance redirected walking techniques.

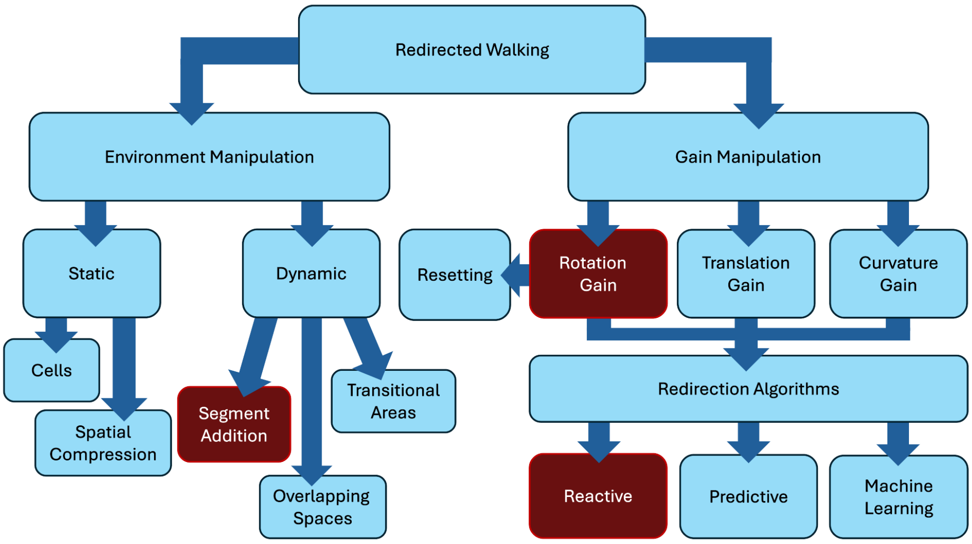


FIGURE 2.1: Overview of redirected walking showing key research areas

2.1 Gain Manipulation

With gain manipulation, the user walks along a different path in the tracking space than the one they perceive they are walking in the virtual environment (Razzaque, Kohn, and Whitton, 2005). This is achieved by deliberately moving the viewport camera in the virtual environment out of sync with the user's movement in the physical tracking space. The user unconsciously changes course to stay on the expected trajectory in the virtual environment. These changes to how the viewport camera moves are referred to as **gain**. There are multiple types of gain, which are used to optimise different situations (see Liu, Ren, and Gan, 2024 for an overview). Among these, three common types of gain are (Nilsson et al., 2018):

- **Translation Gain**: changes the distance a user moves in the virtual environment compared to the real world. Users tend to underestimate virtual distances (Steinicke et al., 2008; Bruder et al., 2012; Renner, Velichkovsky, and Helmert, 2013; Waldow, Fuhrmann, and Grunvogel, 2018). This is beneficial for translation gain, where with no translation gain present users believe they are walking more slowly in the virtual world than the physical space. Adding a small amount of translation gain makes users feel like they are being tracked more accurately. It has also been shown that as translation gain increases, users tend to walk more slowly (Nguyen, Cervellati, and Kunz, 2017). Translation gain can also be combined with other gains for additional benefit (Grechkin et al., 2016). The left most image in 2.2 shows an example of translation gain. With a ratio of 1:1, the distance the user walks in the real world is translated directly to the distance the user walks in the virtual environment. With a gain of 2:1, the user travels twice the distance in the virtual environment as the real world. The equation 2.1 describes how the viewport is altered to add translation gain.

$$G_T = D_V / D_R \tag{2.1}$$

where G_T is the translation gain, D_V is the virtual displacement and D_R is the real world displacement.

- **Rotation Gain**: changes the amount a user rotates in the virtual environment compared to the real world. This is measured by how much the user turns virtually compared to the real world. The second image in 2.2 shows an example of rotation gain. For example, a scenario where the user turning 60° in the real world would lead them to turn 90° in the virtual environment would have a gain of 1.5. Meanwhile, the user turning 180° in the real world and 90° virtually would be 0.5. See equation 2.2.

$$G_R = \theta_V / \theta_R \tag{2.2}$$

where G_R is the rotational gain, θ_V is the virtual rotation and θ_R is the real world rotation.

- **Curvature Gain**: a user walks along a curve in the real world as they perceive themselves to be walking along a straight path in the virtual environment. The third image in 2.2 shows an example of curvature gain. The radius of the circle that the curve could be placed on is used to describe curvature gain. A smaller radius value corresponds to a larger curvature gain as the curve of the path is more extreme. Bending Gain is a special case of curvature gain (Rietzler et al.,

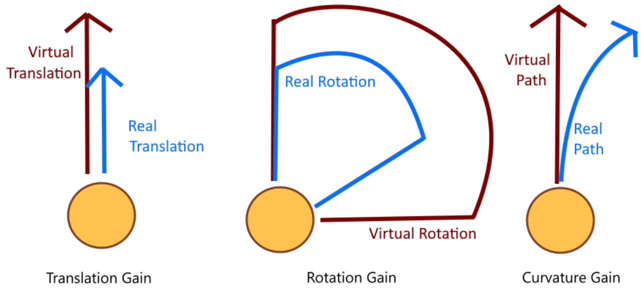


FIGURE 2.2: The diagram shows the three common types of gain. The red arrows represent the user's virtual movements and the blue arrows their real movements. The user is represented by the yellow circle.

2019) where a user walks along a curved path in the virtual environment, while walking along a more extreme curved path in the physical tracking space.

2.1.1 Subtle Redirection

Suma et al., 2012 divides walking gain strategies into categories depending on whether the gain is subtle (the user is unaware of the gain manipulation) or overt (the user notices the changes in how they turn). Subtle redirection benefits from improved spatial memory (Hodgson, Bachmann, and Waller, 2011; Langbehn, Lubos, and Steinicke, 2018a), higher immersion and less impact on task performance (Bruder, Lubos, and Steinicke, 2015; Mostajeran et al., 2024) than overt redirection. Subtle gains are the building blocks of redirected walking algorithms.

Detection Thresholds

Detection thresholds are the point at which most users become aware of the gain being added. Steinicke et al., 2008; Steinicke et al., 2010 first sought to find the detection thresholds for translation, rotation and curvature gain. Since then, multiple studies have sought to define the detection threshold for each type of gain under varying conditions (see Li, Steinicke, and Wang, 2022; Liu, Ren, and Gan, 2024 for detailed lists). Curvature gain detection thresholds are the most sensitive to the exact experiment conditions (Steinicke et al., 2010; Bruder et al., 2012; Grechkin et al., 2016; Green, 1993).

However, all detection thresholds vary significantly based on a variety of factors, for example:

- There is strong variance between individual users (Hutton et al., 2018; Nguyen et al., 2020a): Users who rely more heavily on visual cues when navigating tend to have higher detection thresholds (Rothacher et al., 2018). Additionally, users who primarily use their surroundings to judge the amount they are turning rather than internal body cues, such as proprioceptive and vestibular senses, have higher curvature gain detection thresholds (Jing et al., 2024). This has led to the creation of methods to tailor the gain level to each individual by adding a gain calibration phase before the users enters the target virtual environment (Hutton et al., 2018; Avola et al., 2023).

- Users experience level with VR: Experienced users detecting gain earlier than inexperienced users (Robb, Kohm, and Porter, 2022; Congdon and Steed, 2019).

- The cognitive load of the user: A high cognitive load increases the detection threshold (Nguyen et al., 2020a; Cools and Simeone, 2019; Mostajeran et al., 2024). Interactive tasks such as placing objects or reading text (Lee et al., 2024) or a physically exhausting task (Coelho, Steinicke, and Langbehn, 2022) can increase the rotation gain threshold by acting as a distraction from the gain.

- The speed the user is walking or turning at: Users are less sensitive to gain at slower speeds (Neth et al., 2012; Brument et al., 2021).

- The properties of the virtual environment: for example the brightness (Waldow, Fuhrmann, and Grunvogel, 2018), the height of the horizon (Kim et al., 2022) and the size of the environment (Kim et al., 2022; Gan et al., 2024). The number of visual cues users could orient themselves with sometimes changed detection thresholds (Kruse, Langbehn, and Steinicke, 2018; Paludan et al., 2016) and in other cases did not impact thresholds (Paludan et al., 2016; Gan et al., 2024). The level of realism and type of environment (urban or natural) did not impact thresholds for curvature gain (Mostajeran et al., 2024).

- The amount the user is moving at once: More gain can be added with smaller head turns than large head turns without the user noticing (Bruder et al., 2009).
- Habituation can increase curvature detection thresholds: When users had habituated to curvature gain below the detection threshold, higher gain in the same direction could be added without their notice (Bölling et al., 2019; Sakono et al., 2021; Lee and Kwon, 2023), while gain in the opposite direction became more noticeable (Bölling et al., 2019).
- The user's embodiment in VR: A first-person perspective increases immersion while making users more aware of curvature gain (Nguyen et al., 2020b). Similarly, if users feel they have greater agency in movement, it lowers the curvature gain detection threshold (Nguyen et al., 2020b). Adding a virtual avatar to the user made them feel as if they could better judge the gain, but had no impact on gain thresholds (Reimer et al., 2020). Having the user wear tracking devices on their feet that are then used to synchronise an animation of their virtual foot movement made users more aware of gain (Kruse, Langbehn, and Steinicke, 2018).
- The user's field of view (FOV): A larger FOV (110° compared to 40°) made rotation gain less noticeable (Williams and Peck, 2019).
- Using senses other than sight to reinforce the gain: Providing haptic feedback by using physical props increases immersion but reduces the detection threshold (Hoshikawa et al., 2022). Spatialised sound lowers curvature gain detection thresholds (Meyer, Nogalski, and Fohl, 2016) but has shown little impact on rotation gain detection thresholds (Nilsson et al., 2016). Adding vestibular noise so users rely less on their vestibular sense increases curvature gain detection thresholds (Matsumoto et al., 2021).
- A gradual increase in gain: A gradual increase has been shown to increase detection thresholds under some conditions (Congdon and Steed, 2019; Bao et al., 2024) but not others (Zhang and Kuhl, 2013).

- Gender: gender impacted rotation gain detection thresholds in Williams and Peck, 2019 but not in Gan et al., 2024.

With the variability in gain thresholds across different scenarios, different measures have also been created. The threshold of limited immersion set the point at which rotation gains were high enough that they broke a user's sense of presence (Schmitz et al., 2018). A user's immersion breaks quite late after the rotation gain becomes noticeable in cases where the virtual turn was larger than the physical turn, with a typical rotation gain immersion threshold of 1.85. Some participants remained immersed in the scenario even when the rotation gain was increased to 2.0. Similarly, Rietzler et al., 2019 prioritised user comfort while the users were aware of the gain. These higher thresholds bring down the physical space requirements for redirected walking. Curvature gains above the detection threshold have an impact on spatial and verbal memory task performance (Bruder, Lubos, and Steinicke, 2015); however it is unclear how these higher rates of gain effect other task performance, for example, such as tasks involving searching, executive function or attention to detail.

Chapter 4 of this thesis follows from this work, looking at how different levels of rotation gain impact users in both a complex and minimal virtual environment at levels at and above the detection threshold. Chapter 4 includes a low visual cue environment that contains a blue void with a yellow ball in front of the user to orient them in the space, similar to the empty space with just a horizon line and a stop marker in Kruse, Langbehn, and Steinicke, 2018.

2.1.2 Overt Redirection

It is possible to tailor a virtual environment to a specific small space requirement combining different redirection techniques (Langbehn and Steinicke, 2019; Serubugo et al., 2017). However, there is no easy way to generalise these principles to all virtual environments played in a room-scale tracking space. As a result, while subtle redirection is preferred when large tracking spaces are available, overt redirection can allow the user to walk safely even inside very small tracking spaces. Overt Redirection uses the different gains mentioned in Section 2.1.1 at levels above the detection thresholds. The downsides of these techniques include adding to cognitive load (Nguyen et al., 2020a) and reducing immersion (Schmitz et al., 2018).

Immersion can remain high if the gains added are only slightly above the detection thresholds such as below the threshold of limited immersion (Schmitz et al., 2018). Another example is using all gain types together at levels slightly above the detection threshold to allow users to walk in a small tracking space (2.43m $\times$ 2.43m) (Bozgeyikli et al., 2019). Twice the curvature gain detection threshold can be used without significantly increasing simulator sickness, and 70% of users will accept a curvature gain of four times the detection threshold, despite increases in simulator sickness (Rietzler et al., 2019). Chapter 4 of this work implements rotation gain beyond the detection threshold (introduced by Steinicke et al., 2010) and the threshold of limited immersion (Schmitz et al., 2018).

Larger translation gain can be used to allow users to walk large virtual distances. When users walked 3 times as far virtually as physically, a high sense of presence was maintained but there was also an increase in simulator sickness (Selzer, Larrea, and Castro, 2022). Seven League Boots (Interrante, Ries, and Anderson, 2007) scale the user only in one direction and work well with translation gain up to 3:1. Williams et al., 2006 shows that users can orient themselves within a virtual environment even with translation gain as large as 10:1. Different methods of large translation gain are most effective depending on the type of task the user is performing (Abtahi et al., 2019). Users prefer giant scaling (where the user becomes bigger compared to the environment) compared to seven-League boots or daddy long-legs scaling (the users legs are longer but their upper body remains the same size) for medium sized virtual environments (Zhao, Lindeman, and Piumsomboon, 2025).

Resetting

Even when using subtle redirection, occasionally an overt method will be necessary. With the gain mentioned in Section 2.1.1, sometimes the user will move to the edge of the tracking space available to them and must be quickly turned to stay within the boundaries of the space. Resetting is an overt form of redirection that gives the user explicit instructions on how to turn. This technique is used alongside other redirection techniques as a failsafe when a user is close to a boundary. Resetting caused less motion sickness than adding constant gains in a tracking space spanning 6m x 6m (Gao et al., 2022).

Williams et al., 2007 introduced the concept of resetting with three techniques – the freeze backup which freezes the viewport while the user steps backwards into the centre of the tracking space, the freeze turn that has the user rotate 180°while the viewport is frozen, and the 2:1 turn which has the user turn 180°while turning the viewport 360°. The 2:1 turn is seen as the best of these three techniques, while freeze backup leads to the least amount of error in how much the user turns.

Other techniques were introduced later, such as; Reset to Centre which aims to turn the user back to the centre of the tracking space, Modified Reset to Centre (MR2C) (Thomas and Rosenberg, 2019a) for spaces with internal obstacles, Reset to Gradient (R2G) which turns the user to face in the direction of the most negative gradient from their current position and SFR2G which computes multiple steps ahead to see if the direction found by R2G actually leads the user away from obstacles in the space.

ARC Reset (Williams, Bera, and Manocha, 2021a) aims to rotate the user to face away from the physical obstacle they are currently facing while aligning the virtual and physical obstacles as closely as possible. When virtual and physical obstacles are closely aligned, the user will naturally avoid physical obstacles while avoiding the virtual obstacles they can see. Lee et al., 2024 found ARC Reset useful in moving the user away from tight and narrow spaces.

Recently, resetting strategies have aimed to improve the user experience. The resetting strategy introduced in Xu et al., 2022 resets the user so that they can walk the longest distance until the next reset while moving in a straight line towards a point of interest to them (their goal). Virtually the user turns 360°, while in the physical space they turn towards the optimum direction for their chosen path. This strategy also places resets far away from points of interest the user can interact with to minimise the disruption to the users. Lee, Kim, and Lee, 2025 identified the shortcomings of resetting user interfaces in notifying users of when to reset, how far to turn and in which direction to turn. They created a new user interface for resetting that uses both auditory and visual cues, as well as a gauge to show how far the user has left to turn.

An algorithm for defining the best angle of reset for a given physical environment was introduced in Zhang, Chen, and Zollmann, 2022. The resetting system works alongside other redirection and resetting algorithms and optimises which way the user is facing in the physical tracking space after the reset. One-Step Out-Of-Place Resetting (Zhang et al., 2023) steers the user to a nearby location and direction that is likely to have long walking paths until another reset is required. This method not only turns the user, but changes their location in the physical tracking space while freezing the virtual viewport. Li and Fan, 2024 creates an energy map of the virtual space and the user's current trajectory to predict where the user will navigate to the next. They use this information to choose the ideal direction for the user to face after the reset to minimise the number of resets and the distance between resets. Chapter 3 of this work uses two different resetting techniques as part of its redirected walking algorithms.

2.2 Redirected Walking Algorithms

With redirected walking algorithms, different types of gain are combined to steer the user away from the edge of the tracking space and physical obstacles. These algorithms often implement gains below the detection threshold alongside a resetting technique to steer the user around the space.

Li, Steinicke, and Wang, 2022 divide redirected walking algorithms into categories based on the approach used. Reactive algorithms only analyse the user's current position in the real tracking space and optimise gains based on this information. Reactive algorithms are better able to take advantage of unexpected user movement to add additional redirection (Azmandian et al., 2022a). Predictive algorithms try to guess where the user will move next based on where they are in the virtual environment and their future trajectory. Learning-based techniques use machine learning algorithms to optimise redirection. These three types of redirected walking algorithm are described in more depth in the next three subsections.

2.2.1 Reactive Algorithms

Reactive Algorithms generally consider only the current position of the user and the environment, rather than trying to predict future movement. The first generalised redirected walking algorithms (introduced by Razzaque, Kohn, and Whitton, 2005) included Steer-to-Center (S2C), which is often used as a benchmarking algorithm in later works. S2C guides the user back to the centre of the tracking area. S2C works best in an unconstrained virtual environment, followed by steer to orbit (S2O), which guides the user along a circular path along the edge of the tracking space (Hodgson and Bachmann, 2013). In constrained virtual environments with a limited number of long straight paths S2O works best (Hodgson, Bachmann, and Thrash, 2014). Users were more likely to feel simulator sickness symptoms in virtual environments with high optical flow in large parts of their vision (Hodgson, Bachmann, and Thrash, 2014). Azmandian et al., 2015 compared S2C and S2O and defined the minimum viable tracking space for the two algorithms to be 6m x 6m.

The first reactive algorithms assumed the user had a square or rectangular tracking space with no obstacles inside it to move in. More recent reactive algorithms account for irregular shaped tracking spaces that might contain internal obstacles, as would often be found in users' homes. For example, the Artificial Potential Field Algorithm (APF)(Bachmann et al., 2019; Thomas and Rosenberg, 2019a) tries to steer users towards empty areas of the tracking space. In an empty square environment, APF is very similar to the S2C algorithm. An APF adaptation for irregular shaped tracking spaces outperformed S2C and other major reactive algorithms (Messinger, Hodgson, and Bachmann, 2019). APF has been extended to work with multiple users (Dong et al., 2020) and for walking backwards or sideways (Dong et al., 2023). APF-S2T combined APF with a steering to target algorithm to improve the performance of APF in narrow, corridor-like environments (Chen et al., 2024).

Some generalised redirected walking algorithms aim to align virtual walls and boundaries with real world walls and boundaries (Goldfeather and Interrante, 2012). The Alignment-based Reactive Controller (ARC) aims to align physical and virtual obstacles within the tracking space (Williams, Bera, and Manocha, 2021a). When these obstacles are aligned the user naturally avoids the virtual obstacles and in doing so, avoids the physical obstacles as well. The controller introduced in Williams, Bera, and Manocha, 2021b aims to align the spaces so that the user is walking in the two most similar visibility polygons for the physical and virtual spaces using gains. A new method to match the physical and virtual spaces was created that aligns the walls, floors and key game objects of the spaces by splitting the virtual space into cells and adding separate levels of translation gain to the x and y direction of each cell (Kim and Woo, 2023).

In a simulation comparing two reactive controllers (APF and S2C) and two environmental alignment controllers (ARC and Visibility Polygons), alignment controllers performed better when there was more local similarity between the physical and virtual environments (Williams, Bera, and Manocha, 2021b). That is, when the virtual and physical obstacles in an area have similar shapes and sizes. The Environment Navigation Incompatibility (ENI) index looks at how easy it would be for a user to walk in a given virtual environment based on the physical environment available to them (Williams, Bera, and Manocha, 2022). ENI describes how locally similar the two environments are and then creates a map of areas that are easier and harder to navigate virtually.

A new redirection method that combines techniques from APF and alignment controllers out-performed previous APF and alignment-based controllers in simulations with large tracking spaces and internal obstacles (Wu et al., 2023). Similar to previous work, alignment controllers were found to work when there is local similarity between the physical and virtual environments while APF controllers work well when there are large, obstacle free regions.

In Chapter 3, the performance and user preference of two different reactive algorithms described in this section (S2C and ARC) are compared in a small tracking space of 3.5m x 3.5m.

2.2.2 Predictive Algorithms

Predictive Algorithms add gains based on the optimal place for the user to move to in the tracking space based on predictions of their future trajectory. There are two parts to the predictive algorithm - first the algorithm must map the potential paths a user might take, and then calculate the gain based on this knowledge.

A constraint on such path prediction algorithms is the computing power available to predict future paths the user may take within an acceptable time frame. The more constrained the space in which the user can walk, the easier it will be for predictive algorithms. Fully Optimized Redirected Walking for Constrained Environment (FORCE) (Zmuda et al., 2013) uses path prediction on a limited number of natural paths the user can take in the virtual environment to optimise gains. MPCR (Nescher, Huang, and Kunz, 2014) sees redirection as an optimal control problem where different redirection techniques are dynamically switched between based on the best outcome using path prediction. COPPER (Azmandian et al., 2022a) switches between different redirection strategies (e.g. simple gain, S2C, MPCR, FORCE etc) based on on what is the best strategy to implement for a given input state. For wide-open virtual spaces with few set paths, a short-term (3m) planning prediction algorithm (Hirt et al., 2022a) can be used, as long-term predictions are difficult due to the number of possible trajectories.

Methods of efficiently mapping potential paths that can be combined with the existing predictive algorithms have also been implemented. Automatic path prediction using a set of navigation meshes for a specific environment can compute an average of 20m of path in 7.5ms (Azmandian et al., 2016a). Another system generates skeleton graphs for virtual environments with automatically or manually generated waypoints (Zank and Kunz, 2017). In this system additional waypoints can also be added by the designer and the system pre-computes likely trajectories. A different approach uses Inverse Kinematics to plan a path from the goal to the start position of a virtual environment while accounting for resetting and translation gain (Thomas, Yong, and Rosenberg, 2022). In Chapter 4 of this work, the turning accuracy of users under different rotation gain levels is measured.

2.2.3 Learning Based Algorithms

Approaches from the field of machine learning are being used to improve the performance of redirected walking algorithms and create new ones. A Long Term Short Term Memory network has been used to predict the future path of users in a maze environment (Cho, Lee, and Lee, 2018). Similarly, SRC (Lee et al., 2024) dynamically switches between S2C, SRL, TAPF and ARC depending on which controller suits the scenario best using Long Term Short Term memory network. In simulation SRC outperformed any individual algorithm used in its make-up.

Steer to Optimal Target (S2OT) (Lee, Cho, and Lee, 2019; Lee et al., 2020) uses double deep Q-Learning to predict the optimal steering target in the physical tracking space. Another redirection controller used reinforcement learning to choose which gain levels to apply to the user (Strauss et al., 2020). Similarly, reinforcement learning was used to train a controller that reduced resets compared to a basic heuristic controller in both a simulation and user study (Chang et al., 2021). While previous controllers used primarily the previous physical movements of users for prediction, a recurrent neural network took the virtual trajectory of the user and the features of the virtual environment into account to improve movement prediction (Lemic, Struye, and Famaey, 2022).

With the rise in learning based algorithms and predictive algorithms, understanding how users react to different gain levels becomes more important as this has a direct impact on how users are able to navigate the environment and how well prediction algorithms work. In Chapter 4, the accuracy of user rotation at different levels of rotation gain is analysed.

2.3 Environment Manipulation

Environment Manipulation changes the geometry around the user, to create impossible spaces that could not exist in our physical world as their geometry overlaps or changes over time. There are two approaches to this kind of geometry:

- **Static geometry** takes the user's tracking area into account when creating the virtual environment before the user enters the space.

- **Dynamic geometry** changes around the user while the application is running.

The novel Segment Addition technique described in Chapter 5 falls into the category of dynamic geometry. The Segment Addition technique fills a gap in the literature for an environment manipulation approach that can be used in virtual environments that contain wide open spaces with few occluding objects. It takes the idea of splitting up the virtual environment found in the static virtual cells approach. However, instead of dividing the environment based on the size of the tracking spaces, it adds slices to the environment dynamically by leveraging change blindness. To the author's knowledge, no other technique using segmentation similar to Segment Addition exists and it provides a novel approach to redirected walking.

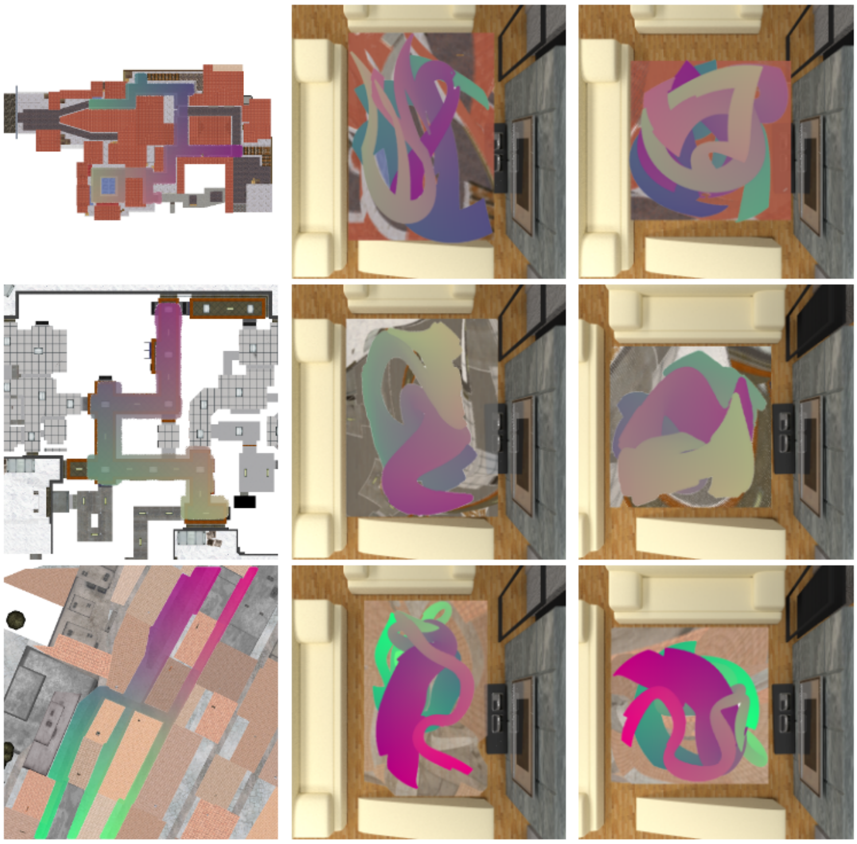
2.3.1 Static Geometry

Static Geometry uses a map of the tracking space to create a virtual environment before the user even enters VR. It includes two approaches. Firstly, **Spatial Compression** which maps a larger virtual environment onto a smaller tracking area using mapping techniques. Secondly, **Virtual Cells** which divide the virtual environment up based on the size and shape of the tracking space.

Spatial Compression

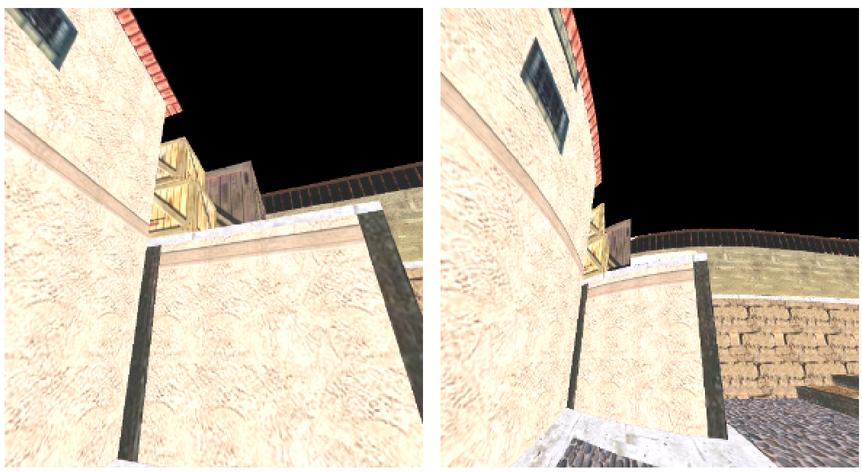
Spatial Compression uses planar mapping approaches to create a map from the physical tracking space and the virtual environment (see Figure 2.3). Spatial Compression causes less simulator sickness than gain manipulation techniques, which improves user experience (Gao et al., 2022). However, it is more sensitive to the layout of both the physical space and virtual environment. In wide, open outdoor scenes, distortion becomes more noticeable. Similarly, in very cluttered environments with no excess space that can be bent distortion becomes more noticeable.

Sun, Wei, and Kaufman, 2016 introduced planar mapping which created a version of the virtual environment where every point in the environment mapped onto



(a) *virtual scene* 200 × 200 **(b)** *real scene* 60 × 100 **(c)** *real scene* 70 × 70

(A) Planar Mapping comparing Virtual and Physical Environments - larger virtual environments (a) mapped onto two smaller physical tracking spaces (b,c).



(a) *virtual camera view* **(b)** *real camera view*

(B) A view of the virtual environment - (a) without any distortion caused by planar mapping. (b) what the user sees after the planar mapping is complete.

FIGURE 2.3: An example of a planar mapping approach that compresses a larger virtual space onto a smaller physical space. The figures are examples taken from the original paper on Planar Mapping Sun, Wei, and Kaufman, 2016)

a smaller physical tracking space. The map was locally injective, to stop objects from overlapping and allow users to navigate comfortably. Smooth Assembly Mapping (SAM) divides the virtual environment into sections and aims to minimise isometric distortions (Dong et al., 2017). SAM allowed the mapping of larger virtual environments to smaller physical tracking spaces.

Another spatial compression technique minimises distortions around areas of interest where the user focuses most of their attention (Cao et al., 2020). It comes at the cost of slightly higher distortions (compared to SAM) in areas of low interest. The space requirements of a virtual scene can be reduced up to 40% if it is moderately occluded without visual distortions for the user (Dong et al., 2021). DreamWalk, a dynamic mapping technique (Xiong et al., 2024) outperforms SAM in wide open spaces at the cost of being more GPU intensive as it computes the mapping at run time.

Virtual Cells

Virtual Cells have often been overlooked as a part of redirected walking despite being a technique that allows users to walk naturally inside a virtual environment that is larger than the tracking space set aside. This could be due to how the virtual environment is altered before the user ever steps inside it. Virtual Cells divide the virtual environment up into cells the size and shape of the tracking space. Within a cell, the user can use real walking. Moving between cells has the user move virtually while staying in the same place physically (see Figure 2.4). The biggest challenges with this technique are moving the user naturally from one cell to the next and generalising the method across different spaces.

Often, a transition metaphor is used to move users between cells. The most effective type of transition between cells will depend on what transition metaphor feels the most natural in the virtual environment. For example, a rotating bookshelf (Yu et al., 2017) or elevator (Marwecki and Baudisch, 2018), where the user turns with the rotating bookshelf or heads back out into a room on a different floor of the virtual environment with the elevator. The user can then walk around the next room normally, as they are once again facing an empty tracking area. For outdoor cells, sometimes a

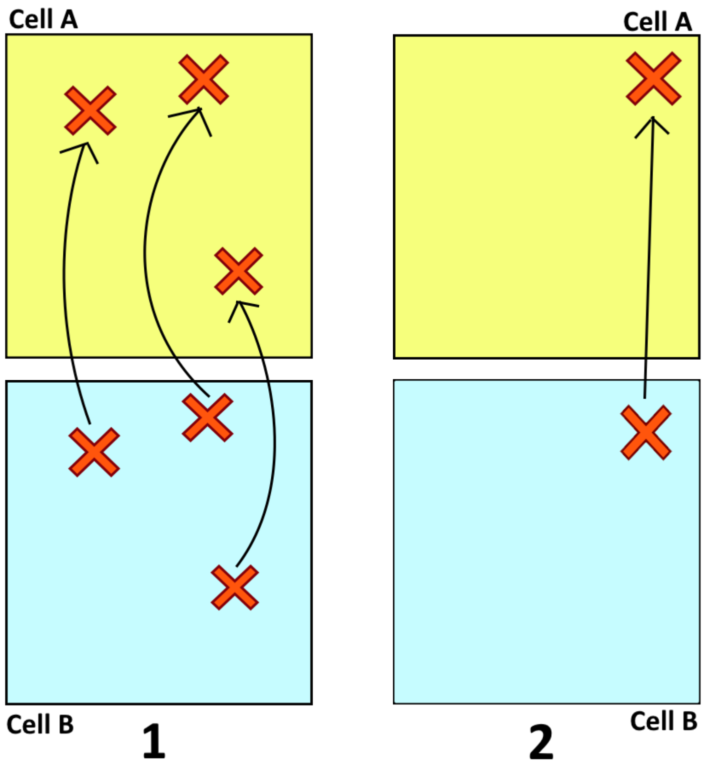


FIGURE 2.4: Moving from one cell to another - In some cells users can move directly from a spot in the current cell to the same spot in another cell (1). Other cells have a designated spot where the user can move from one cell to the next (2).

bird (Marwecki and Baudisch, 2018) is used - the user points to another cell and a bird flies them to the next cell. The bird deposits the user in the position in the new cell that corresponds to their position inside the tracking area (see 2.4). This way, the user can explore within the new cell freely without the need for gains. Portals can also be used to move from one cell to the next (Han, Moere, and Simeone, 2022; Rebelo, Ferreira, and Nóbrega, 2025), with portals that expand the virtual environment preferred by users to portals that spin the environment around the user. Another approach subdivides cells into four pieces, with different pieces being replaced when the user walks past key points in the virtual environment (Serubugo et al., 2017), allowing the cells to be moved between imperceptibly at the cost of adding additional constraints to the layout of the virtual environment. G-RFP (Kwon et al., 2022) creates an infinite virtual space to fit irregular tracking spaces using fixed reset points to move from one cell to another.

Static geometry can take advantage of small and irregular tracking spaces as the virtual environment is designed around it. However, this focus on the tracking space can make it difficult to generalise the virtual environment to different tracking areas. Scenograph (Marwecki and Baudisch, 2018) and similar programs can help to generalise static geometry approaches, as a single experience can work with multiple different tracking area layouts. 'virtualSpace' (Marwecki et al., 2018) is a platform that lets developers build virtual reality experiences that are tracking area agnostic. Another approach is to adapt a developer created virtual room layout to the space of the physical tracking area, keeping the relative layout of key objects and furniture as close to the original virtual room as possible (Sin et al., 2019). I-RFP is a cell based approach to tiling a convex physical space into a pre-existing virtual space (Kwon et al., 2022). A fixed reset point allows the user to move from one cell to the next in the virtual space. I-RFP had higher immersion, lower simulator sickness and users had a better sense of control over movement than the gain manipulation based redirected walking algorithm APF.

2.3.2 Dynamic Geometry

Dynamic Geometry changes the layout of the virtual environment while the application is running. An experiment with 20 participants compared different redirected walking techniques and found change blindness to be the best approach for medium and small environments (Gao et al., 2022). It had the lowest simulator sickness scores, fastest walking speeds and users found it most similar to natural walking.

Change Blindness

Change blindness is the tendency for people to remain unaware of changes in their environment that occur when they are not looking at them (Simons and Ambinder, 2005; Simons and Rensink, 2005). Dynamic geometry often relies on change blindness. In Suma et al., 2011 while the user was completing a task on a computer screen in a virtual office environment, the door to the office and the corridor behind it switched walls. Out of 77 participants, only one noticed the change. Additionally, the cognitive maps of users remained intact as if the door had never switched walls. The Segment Addition technique described in Chapter 5 relies on change blindness.

With change blindness, changes are more likely to remain undetected if the following conditions are met (Martin et al., 2023):

- **Far away**: The change happens further away from the user in space. Vasser, Kängsepp, and Aru, 2015 also found that users were more likely to detect changes in the foreground rather than the background.

- **Simple**: The object that changed was not complex.

- **Outside FOV**: The change happens outside of the user's field of view (FOV). Even when the user's only task is to detect changes in their environment, they will notice changes outside of their field of view in a novel environment about 40% of the time.

- **Novel Environment**: The environment is new to the user – across two separate experiments looking at change blindness in virtual reality environments, there was a habituation effect where users became better at detecting changes in a

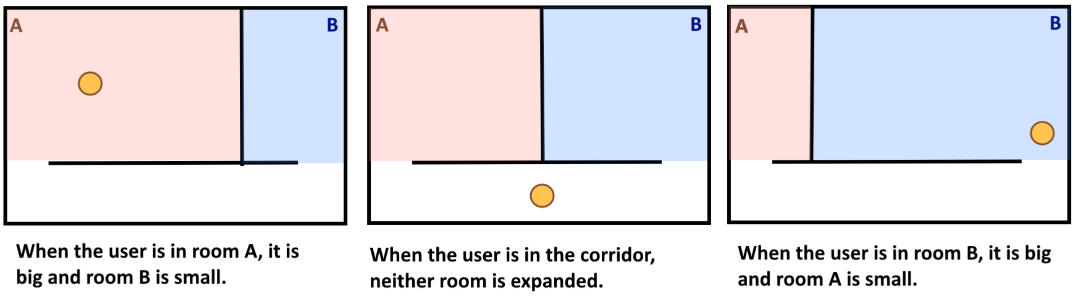


FIGURE 2.5: A top down view of an overlapping space with a corridor and two rooms (A, B). Depending on where the user (yellow circle) is in the environment, the size of the room changes.

scene in trials near the end of the experiment compared to trials near the beginning.

- **Multiple Changes**: even if multiple changes occur in a scene, the user is unlikely to notice more than 4 of them with an average detection rate of 2.68 changes, regardless of the actual number of changes that took place.

Boundaries and passages are used to obscure sections of the virtual environment to leverage change blindness. Boundaries are environmental features that delineate one area from another – the walls of a room, a garden fence, a hedge. The user must move to a specific spot or area, a passage through the boundary, to go through to a different area – a door, a gate, a gap in the shrubbery. Dynamic geometry changes the placement of these boundaries and passages depending on where the user is in the environment. The inside wall of a room in the virtual environment might expand to increase the size of the room while the user is in it. Similarly, passages between spaces might change position, like in Suma et al., 2011, so that what is beyond them is optimally placed inside the tracking space available to the user.

Overlapping Spaces

Overlapping spaces (Suma et al., 2012) play with the boundary walls between rooms. Two or more rooms are connected by a corridor and from the corridor it looks like these two rooms are placed beside each other in space. However, inside each room is bigger than would be possible in reality as the inside wall of the room overlaps where the next room would be (see 2.5). Small virtual rooms can overlap up to 56% and large virtual rooms can overlap 31% without users who are looking for the overlap noticing it at a rate better than chance. Users noticed room overlap of 40% or more when aware of the manipulation and without any extra tasks assigned to them, whereas when they had to dodge around a security camera in a corridor between two rooms, overlap up to 68% remained unnoticed (Ciumedean et al., 2020). These findings suggest that large overlap in room space is less likely to be noticed when users are given additional tasks to complete or are distracted.

The perception of width in an overlapping room while the user was only considering one room was accurate. In contrast when asked to compare the relative sizes of two overlapping rooms in the same space, the ratio between their size remained intact while the exact size of each was underestimated (Robb and Barwulor, 2021). This aligns with previous findings from (Suma et al., 2012) that even when users are made aware of overlapping rooms, they point to objects hidden from view in the corridor as if the rooms did not overlap. Han, Vande Moere, and Simeone, 2023 introduced a technique of creating a series of small virtual rooms on the fly in a virtual environment, based on the tracking space the user has available to them. This technique overtly creates the next and previous sub-rooms based on where the user is in the tracking space and their interest level in what is in the created rooms. The user can see previous rooms they have visited through various open doorways and windows. Rooms adjacent to the current room show up at ground level while previous rooms rise or lower as if on different floor levels to the user.

Transitional Areas

Transitional areas are spaces in the virtual environment whose purpose is to funnel users from one area to another. While rooms and other boundary enclosed spaces might contain interaction elements for the user to complete tasks, transitional areas do not contain such elements. Since the only purpose of transitional areas is to get the user from point A to point B they can be seen as a type of extended passage between two boundaries. This allows transitional areas to be very flexible in terms of geometry (and makes them ideal for adding redirected walking – see Langbehn and Steinicke, 2019 corridor bending gains). The segments added in the Segment Addition technique described in Chapter 5 could be seen as a type of a transitional area.

A classic example of a transitional area in an indoor environment is the corridor – a long plain hallway leading off into multiple rooms. A method to automatically generate twisting corridors between overlapping rooms was introduced by Vasylevska et al., 2013. In an initial user study, five users who were not told about the dynamic changes beforehand, described the corridors as 'maze-like' and did not notice the rooms overlapped. Long corridors with multiple turns were more effective at blinding the user to overlapping rooms compared to short, straight corridors or u-shaped corridors (Vasylevska and Kaufmann, 2015). With a c-shaped corridor between the two rooms an overlap of 60% remained unnoticed (Vasylevska and Kaufmann, 2015). A preliminary study with 7 participants suggests an overlap of up to 33.8% might still be effective when the corridors and/or overlapping rooms have no walls between (Ciumedean et al., 2021).

Previous dynamic geometry approaches require a space with many boundaries and passages to be useful. This makes dynamic geometry a good method in indoor environments or high boundary outdoor environments such as a city with narrow streets and garden mazes, but less useful in large open spaces such as football fields or grassy plains. This gap in the literature is addressed with a new spatial manipulation technique, Segment Addition, described in Chapter 4. Segment Addition introduces segmentation, where slices of the environment are dynamically added or removed outside the user's FOV, to Environment Manipulation approaches.

Spatial Contraction

Spatial Contraction is an overt environmental manipulation technique (Xu et al., 2024) where the space in the direction the user is walking is contracted so that they have to cross a shorter distance to reach their destination. As the speed of the user increases, the level of contraction also increases, allowing them to cover greater distances virtually while only taking a few steps. Unlike translation gain, this technique keeps a 1:1

mapping between the user's virtual and physical movements, so that there is no mismatch between the movements. It can be used alongside Translation Gain to further improve its performance, and in a study with 21 participants, there were no significant increases in simulator sickness scores compared to real walking with the Spatial Contraction method. Similarly to Spatial Contraction, Segment Addition dynamically alters the virtual environment around the user as they move, although the mechanism for how this is achieved differs (see Chapter 5 for details).

2.4 Simulator Sickness

Simulator sickness (also known as VR sickness or cybersickness) is a form of motion sickness that some users experience while inside computer-generated simulations. This section reviews simulator sickness specifically in the context of the user experience of redirected walking. For a general overview of simulator sickness and VR see Jerald, 2015 Chapter 12 and 17.

While the exact cause of simulator sickness is still unknown, one popular theory is that sensory conflict can cause simulator sickness. When someone moves, multiple senses are used to help them orient themselves in the environment. The eyes use visual cues and the relative position of objects to understand which way and at what speed the person is moving. The vestibular system, where our sense of balance comes from, keeps track of our position relative to gravity. Proprioception keeps track of our body in space. Normally when moving, all three of these senses align and there is no conflicting input between them. However, when moving in a vehicle our visual sense might tell us that we are moving when our proprioception and vestibular system do not notice much of a change. This can lead to motion sickness.

Similarly, in virtual reality, latency might change the input of the visual system to suggest we are moving in a different way than we are physically moving. Gain manipulation (see Section 2.1) techniques for redirected walking purposely introduce sensory conflict to redirect the user inside a virtual environment. This can lead to more simulator sickness, especially for new users or users who are inside the virtual environment for a long time. Chang, Kim, and Yoo, 2020 reviewed the literature on simulator sickness in virtual reality. They concluded hardware factors such as latency and display type, content factors, such as task duration and controllability, and human factors, such as demographics and individual differences, can all contribute to simulator sickness.

In relation to redirected walking, many studies in the field consider simulator sickness as an important metric to decide if a method or algorithm is applicable. Generally, such studies uses the Simulator Sickness Questionnaire (SSQ) (Kennedy et al., 1993) to measure sickness before and after users have experienced the virtual environment. Initially, an SSQ total score above 20 was assumed to be due to a bad simulator (Kennedy et al., 1993). However, later work (Gemert et al., 2024; Bimberg, Weissker, and Kulik, 2020) questioned this threshold for simulator sickness, as the original test was conducted on a set of military pilots. For the field of redirected walking, Gemert et al., 2024 suggests thresholds of None (< 5), Low (5–15), Medium (15–30), and High (> 30) when comparing different locomotion techniques to other works in the field.

While redirected walking has lower simulator sickness scores than some other forms of locomotion (such as joystick control, Mayor, Raya, and Sanchez, 2021), gain manipulation (especially rotation and curvature gain) has been shown to increase simulator sickness (Gemert et al., 2024). Higher gain thresholds (users who are less likely to notice gain) are correlated with higher simulator sickness scores (Williams and Peck, 2019). This could be due to being exposed to higher gains in general than specific to those users with a higher gain threshold. Curvature gains below the detection threshold increased simulator sickness after both 50 and 200 trials (Lee and Kwon, 2023). Gao et al., 2022 compared five redirected walking techniques in the amount of simulator sickness they produced. Gains cause the most simulator sickness, followed by resetting and spatial compression. Overlapping architecture caused less simulator sickness while Change Blindness techniques had the lowest simulator sickness scores.

With the negative impacts of simulator sickness, studies have tried to find ways to catch signs of users feeling simulator sickness early and varying virtual reality content accordingly. Williams et al., 2025 found that both postural sway and gaze velocity increased with higher rotation gain levels and the longer users spent inside the virtual environment. Varying the level of curvature and bending gain applied over time reduces discomfort for curvature gain by 16% and bending gain by 9% (Sakono et al., 2021). The Long-Term Short-Term machine learning model introduced by Wang et al., 2022 used eye-tracking data and virtual character movement to predict when users were feeling simulator sickness symptoms in Virtual Reality. The model was more accurate for users who experienced a lot of simulator sickness. Kundu et al., 2023 used an explainable machine learning model and trained it on to detect simulator sickness. The weights within the model can help show which factors contribute the most to increased simulator sickness.

In their paper Liao et al., 2022 introduce an equation that measures accumulated simulator sickness in relation to the SSQ, magnitude of optical flow, redirected walking gain and time spent in the virtual environment. Based on the results, they create a redirected walking controller that accounts for simulator sickness.

Simulator sickness scores are used to measure the user experience in both the study comparing redirected walking algorithms in small tracking spaces (Chapter 3) and the study on Segment Addition (Chapter 5).

2.5 Thesis Concepts

This section briefly describes the key concepts from the literature used in each of the upcoming chapters.

In the next Chapter (3) a user study on reactive redirected walking algorithms in small tracking spaces (see Section 2.2.1) is reported on. Azmandian et al., 2015 and Bozgeyikli et al., 2019 both considered the effectiveness of redirected walking algorithms in small tracking spaces. The redirected walking algorithms use gain manipulation (see Section 2.1) and resetting algorithms (see Section 2.1.2). The redirected walking toolkit (Azmandian et al., 2016b) was used to implement two of the algorithms. The third algorithm was built within the toolkit system based on the paper that first implemented the algorithm (see Williams, Bera, and Manocha, 2021a). The simulator sickness (see section 2.4) of participants was measured at the end of the study.

In Chapter 4 the turning accuracy of participants with rotation gain (see Section 2.1) at and above the detection threshold (see Section 2.1.1 and Section 2.1.2) is measured. The detection threshold (Steinicke et al., 2010) and threshold of limited immersion (Schmitz et al., 2018) are used to guide the level of rotation gain used in the study. Two different virtual environments with different levels of visual cues are introduced within the study based on the findings of Paludan et al., 2016 and Bayramova et al., 2021. The results of this study should be considered when implementing predictive redirected walking algorithms (see Section 2.2.2).

In Chapter 5 a new redirected walking technique (see section 2) is introduced - Segment Addition. Segment Addition fills a gap in the literature. It makes use of change blindness (2.3.2) to dynamically alter (2.3.2) wide, open virtual environments. This expands the type of virtual environments where Environment Manipulation techniques can be used (2.3). It does so by dynamically adding and removing segments, called "slices" to the virtual environment outside the user's FOV. The concept of segmenting the environment is similar to cell division approaches (2.3.1), however the application of dynamically adding segments and how they are added into the environment is novel. The author is unaware of any other segmentation approach for dynamic Environmental Manipulation. A user study investigates the experience of users with Segment Addition, including user comfort (see section 2.4). A future outlook is presented that includes potential modifications to the technique using virtual cells (see section 2.3.1) and gain manipulation (see section 2.1).

Chapter 3

User Experience of Redirected Walking Algorithms in a Small Tracking Space

Redirected walking brings an immersive way to explore virtual reality worlds to users who have limited physical space in which they can safely navigate. Many consumers have very limited tracking space when they use virtual reality applications in their own homes, as the size of their room limits the tracking space. This chapter contributes to the under-explored area of the user experience of redirected walking. It describes how users respond to reactive redirected walking algorithms in small tracking spaces. A user study with 36 participants was conducted that compared three approaches to redirected walking in different virtual environments with a tracking space of only 3.5m × 3.5m.

The results describe objective performance metrics and users' preferences and experiences of the algorithms. The simplest of the three algorithms, Reset Only, had a similar performance to the Steer-to-Center algorithm and was also preferred by users. The Alignment Based Redirection Controller in contrast had fewer breaks in presence caused by resets but users also had worse task performance. Based on these results, recommendations are proposed for designers using redirection. Suggestions include implementing a predictable resetting system and choosing algorithms focused on resetting or environmental manipulation in small tracking spaces.

3.1 Introduction

Real walking, where the user walks in the real world as they walk in the virtual environment, can increase presence and immersion (Hodgson and Bachmann, 2013). For a user to walk in a virtual environment, physical tracking space must be set aside for them to walk safely while immersed in the virtual environment. However, tracking space is limited by the size of the room, especially when using virtual reality (VR) at home.

Redirected walking algorithms subtly redirect the user to remain in the tracking space while they are walking in a larger virtual environment. This is achieved by adding 'gain' to the user's movements. For example the user might rotate a little more or walk further with every step in the virtual environment compared to their physical movement. Section 2.1 details the different types of gain.

There are many different redirected walking algorithms (see Section 2.4 for an overview). These algorithms usually use automated simulations for an initial performance evaluation, often followed by a small user study. Many of these automated simulations model users who walk in straight lines and stop to turn in place. A user study that constrained real users to walking paths similar to these simulations (Azmandian et al., 2022b) found that redirected walking algorithms had similar performance in the user study to automated simulations. However, this method does not account for how users naturally behave in virtual environments. Real users walk along curved trajectories, move in non-forward directions or accelerate (Hirt et al., 2022b). These behaviours are not accounted for in automated simulations. Simulation alone cannot adequately describe user response to a redirected walking algorithm (Hirt et al., 2022b). Common performance metrics for redirected walking, such as number of resets and distance between resets, differ between user studies and automated simulations.

User experience is an under-explored area in VR locomotion (Martinez, Wu, and McMahan, 2022). Very few studies focus on the response of real users to redirected walking algorithms in small tracking spaces. It was suggested to use spaces that are 6m × 6m or larger for redirected walking (Azmandian et al., 2015). However, later work found redirected walking to be viable in a space of about 2.44m × 2.44m with higher gain levels (Bozgeyikli et al., 2019).

The main hypothesis of this chapter is that gain may be used to allow users to walk in larger virtual spaces, even when confined to small tracking spaces. This chapter describes a user study as a contribution to this under-explored area, focusing on the user experience of redirection with three different redirected walking algorithms. The user study compares the performance of the three algorithms to each other in three different virtual environments with a small physical tracking space. Additionally, users' preferences among the algorithms and user experience (based on usability and simulator sickness metrics) are measured.

3.2 Implementation

Different redirected walking algorithms are best suited for different scenarios. Predictive algorithms work well when the virtual space is mapped out and has structured paths for the user to follow (Zmuda et al., 2013; Nescher, Huang, and Kunz, 2014; Qi, Liu, and Cui, 2010; Fan et al., 2023). Machine-learning approaches can be leveraged with enough time and processing power (Lee, Cho, and Lee, 2019; Lee et al., 2024; Lemic, Struye, and Famaey, 2022). Eye-tracking can be used to predict a user's future direction (Jeon et al., 2024) or to add additional gains (Nguyen and Kunz, 2018); however, eyetracking is not present in most consumer grade VR headsets. Reactive algorithms are the most flexible (Razzaque, Kohn, and Whitton, 2005; Bachmann et al., 2019; Thomas and Rosenberg, 2019b; Williams, Bera, and Manocha, 2021a). Reactive algorithms use only the current position of the user and make fewer assumptions about the space they are in.

For users experiencing VR at home, the following assumptions are considered as the most likely set of circumstances:

- They are using a consumer-grade headset without eyetracking - such as the Meta Quest 2, HTC Vive or Meta Quest 3S.

- Tracking space is limited by room size, with the shape of the tracking space varying depending on the layout of the room.
- Computing power is either limited by the power of the headset or their computer which is not specialised for VR. As a result the redirected walking algorithm should run smoothly at a high frame rate with limited processing power, while also accounting for the processing cost of running the rest of the VR application.

This study focuses on reactive algorithms, as they are the most applicable when processing power and knowledge of the space is limited. Three different virtual environments were chosen to emulate various use cases for redirection in virtual reality applications. The user study was conducted with 36 participants that compared three different approaches to redirected walking algorithms in a small tracking space of 3.5m × 3.5m with an emphasis on user experience.

3.2.1 System

The Unity Game Engine was chosen as the platform to create the experimental system as it was a platform commonly used for redirected walking algorithms and an open source implementation of one of the algorithms (S2C) and resetters (2:1 Turn) was available for the platform as part of a Redirected Walking Toolkit (Azmandian et al., 2016b). The experimental system was created using the Unity Game Engine version 2019.4.22f1 and ran on a laptop with the Windows 11 Pro operating system. The laptop contained an Intel i7-11800H 2.30GHz CPU and a NVIDIA GeForce RTX 3070 graphics card with 32GB RAM.

The HTC Vive Pro Eye headset was connected to the PC via a USB-C display port connection. The headset provides a stereoscopic view with a resolution of 1440 × 1600 pixels per eye, a refresh rate of 90Hz, and a field of view of around 110°. The system could also be run using a HTC Vive headset, as the system does not use the eye-tracking features of the Pro Eye headset. Two base stations were used to create a physical tracking space of 3.5m $\times$ 3.5m. Users were given two HTC Vive controllers to enable them to interact with the waypoints in the virtual environment.

First, the existing redirected walking toolkit was installed and tested for use with the HTC Vive headset. The system originally focused on simulation-based studies of redirected walking systems and thus some adaptation was necessary for use with the headset. After the existing algorithms had been tested, two new redirected walking controllers were implemented into the system to account for the redirected walking algorithms not already present within the toolkit (RO and ARC). A new resetting algorithm (ARC reset) was also added. The details of these implementations can be found in Section 3.2.2 below. The new controllers were tested first in simulation and then with the HTC Vive Headset to ensure they worked as intended. Once all the controllers had been implemented, three sample environments (described in 3.2.4) were added to the system. Finally, logging features which output csv files were added to keep track of how the user moved within the virtual environments, as well as track key metrics such as the number of waypoints collected and how many times the resetting controller was triggered.

3.2.2 Redirected Walking Algorithms

Three redirected walking algorithms were chosen to best compare the different reactive approaches to redirected walking. Two of the algorithms (S2C and RO) were adapted from the redirected walking toolkit (Azmandian et al., 2016b) to our application. For the third algorithm, ARC, a new redirection controller was added based on the instructions in Williams, Bera, and Manocha, 2021a. All the redirected walking algorithms used the detection thresholds as the maximum level of gain (Steinicke et al., 2010). A reset was triggered if a user was 0.5m away from an obstacle.

Reset Only (RO)

The Reset Only (RO) condition did not use any gain at all but simply translated the users' movements directly into the virtual space. An empty controller which did not apply any redirection was added to the redirected walking toolkit for this condition. The 2:1 turn reset was used to stop the user when they were about to bump into an obstacle. Overt redirection techniques, such as resetting (see section 2.1.2 for more detail), are commonly used in redirected walking algorithms when more subtle forms of redirection have failed. The 2:1 turn reset was adapted from the redirected walking toolkit (Azmandian et al., 2016b).

Steer-to-Center (S2C)

Steer-to-Center (S2C) aims to steer the user back to the centre of the physical tracking space and only tracks the current position and direction of the user. When the user is close to the centre of the space, no redirection is applied. As they move away from the centre area more redirection is added up until the detection threshold (Steinicke et al., 2010) is reached.

S2C (Razzaque, Kohn, and Whitton, 2005) was one of the first redirected walking algorithms and is commonly used as a baseline comparison to test how well other algorithms perform, thus it was chosen as a good example of a reactive algorithm. Additionally, it allows the findings of this chapter to be more easily compared to other research in the field.

The S2C algorithm implementation used was the one that came with the redirected walking toolkit (Azmandian et al., 2016b), the only changes were to add log file additions and integrating the algorithm into our experiment system. The S2C algorithm used the 2:1 turn reset described in more detail in 3.2.3.

Alignment-Based Redirection Controller (ARC)

Unlike most other reactive redirected walking algorithms which only track the physical environment of the user, the ARC controller makes use of both the physical and virtual environment. The aim of ARC is to align virtual obstacles with physical obstacles in the environment. In this way the user will naturally avoid the physical obstacles by avoiding the virtual obstacles they can see in the environment. ARC additionally has the potential to create haptic feedback when the real and virtual obstacles are almost aligned within the space.

A new controller was created based on the adapted redirection controller from the redirected walking toolkit (Azmandian et al., 2016b). The controller was a variation on

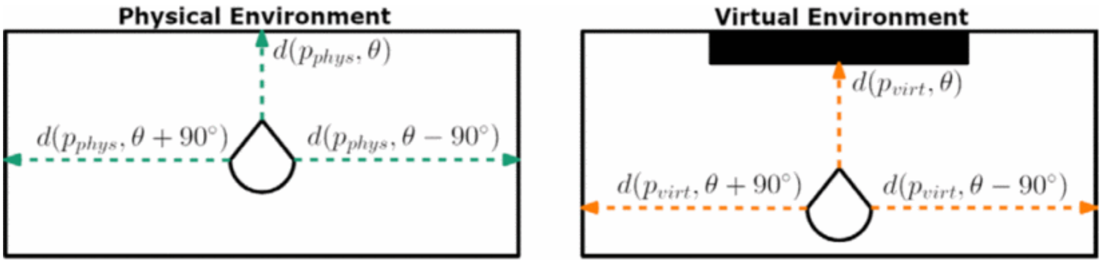


FIGURE 3.1: An example of how ARC uses raycasting to calculate gain, as originally presented in Williams, Bera, and Manocha, 2021a. ARC sends out three rays around the user in both the virtual space (VE) and a virtual representation of the physical space (PE).

the ARC system outlined in Williams, Bera, and Manocha, 2021a. When the physical and virtual obstacles are aligned, ARC does not redirect the user at all. When they are not aligned, ARC uses different kinds of gains to move the two environments to be aligned more closely.

ARC uses raycasting, sending out virtual straight-line beams that hit off objects, to detect the distance of objects from the user (see Figure 3.1). Six rays are sent out around the user, three in the virtual environment and three in the virtual representation of the physical space. One ray is sent out directly in front of the user and one to either side of them. Each ray measures how far away the nearest obstacle is from the user in that direction in either the physical or virtual environment. Once measured the following gains are added (if applicable):

- *Translation Gain*: If the user is walking at a speed of more than 0.2 m/s then translation gain is added to the user. For translation gain, only the rays facing forward are analysed. The distance from the physical obstacle is divided by the distance from the virtual obstacle to get a ratio between the two. A value greater than 1 suggests the virtual obstacle is closer, whereas a value less than 1 indicates the physical obstacle is closer to the user. If the ratio falls within the range of the translation gain (when the gain is centred around 1) then the resulting number is used as the translation gain applied, if the result is outside of the detection threshold the closest number within the threshold is chosen as the translation gain.

- *Rotation Gain*: If the user has rotated more than the threshold for rotation in the last frame, then rotation gain is added to the user. While the original ARC algorithm considered only the alignment of the prior and current frame, our implementation uses the average alignment of the last five frames. If the current alignment is worse, than the average alignment of the last five frames, then the virtual rotation is slowed down and the minimal rotation gain level is used. In contrast, if the user rotation is improving the alignment, the rotation is sped up until it reaches the detection threshold. The average alignment stops the rotation gain from ending abruptly if the alignment gets worse briefly before improving again in the next few frames of rotation. Additionally, the rotation gain has a smoothing function, that linearly interpolates to the maximum gain level as users are less likely to notice the gain this way (Congdon and Steed, 2019).

- *Curvature Gain*: If the user is walking at a speed of more than 0.2 m/s, curvature gain is also added to the user. Translation gain and curvature gain can be applied together without changing the detection threshold (Grechkin et al., 2016). First, the left and right rays are checked to see if the obstacles are more misaligned on the left or on the right of the user. The side with more misalignment is then chosen. If the physical obstacle is closer to the user than the virtual obstacle then the curvature gain turns the user towards that obstacle, otherwise it turns the user away from the obstacle.

A scaling factor is used to decide the magnitude of the gain applied using the formulae 3.1, 3.2:

$$curveGainRotation = \frac{360}{\pi \times 2} \times \frac{Position_{\Delta}}{CurvatureRadius} \tag{3.1}$$

$$scalingFactor = min(curveGainRotation, maxCurveGain \times Time_{\Delta}) \tag{3.2}$$

where $Position_{\Delta}$ is the magnitude of the change in position from the last frame to the current frame, *CurvatureRadius* is the smallest radius of a circle which the curvature gain steers the user along (in this case 7.5m as in Williams, Bera, and Manocha, 2021a) and *maxCurveGain* is the maximum number of the degrees the user can be rotated per second based on the curvature gain (set to 15 degrees).

Finally, a smoothing factor is added so that the curvature gain does not suddenly change but slowly ramps up. This variation allows a greater degree of curvature gain to be applied than in the original ARC algorithm.

3.2.3 Resetting

Resetting as described in 2.1.2 is used when the redirected walking algorithm cannot stop a user from avoiding a physical obstacle in the tracking space by itself. In this case, the subtle forms of redirection are dropped, and instead, the user is given explicit instructions to turn in the virtual environment when they are about to bump into a physical obstacle. The reset uses rotation gain to turn the user away from the physical obstacle while turning them 360° in the virtual environment. Multiple resetting methods exist which have different drawbacks and advantages (Williams et al., 2007; Thomas and Rosenberg, 2019a). The following resetting algorithms were implemented in the experiment.

2:1 Turn

The 2:1 turn activates when the user is close to bumping into a physical obstacle. Text appears in front of the user telling them to turn 360°. The user turns twice as fast in the virtual environment compared to the tracking space. After the turn the user has turned 180° away from the physical obstacle while in the initial facing direction in the virtual environment. The 2:1 turn, is the best of the three resetting algorithms introduced (Williams et al., 2007). It is simple in execution and thus can be used with many different redirected walking algorithms. I implemented the 2:1 turn for the Reset Only and Steer-to-Center conditions. The redirected walking toolkit contains a 2:1 turn reset, this base script was modified to allow the actual user to turn in the environment set up for the experiment.

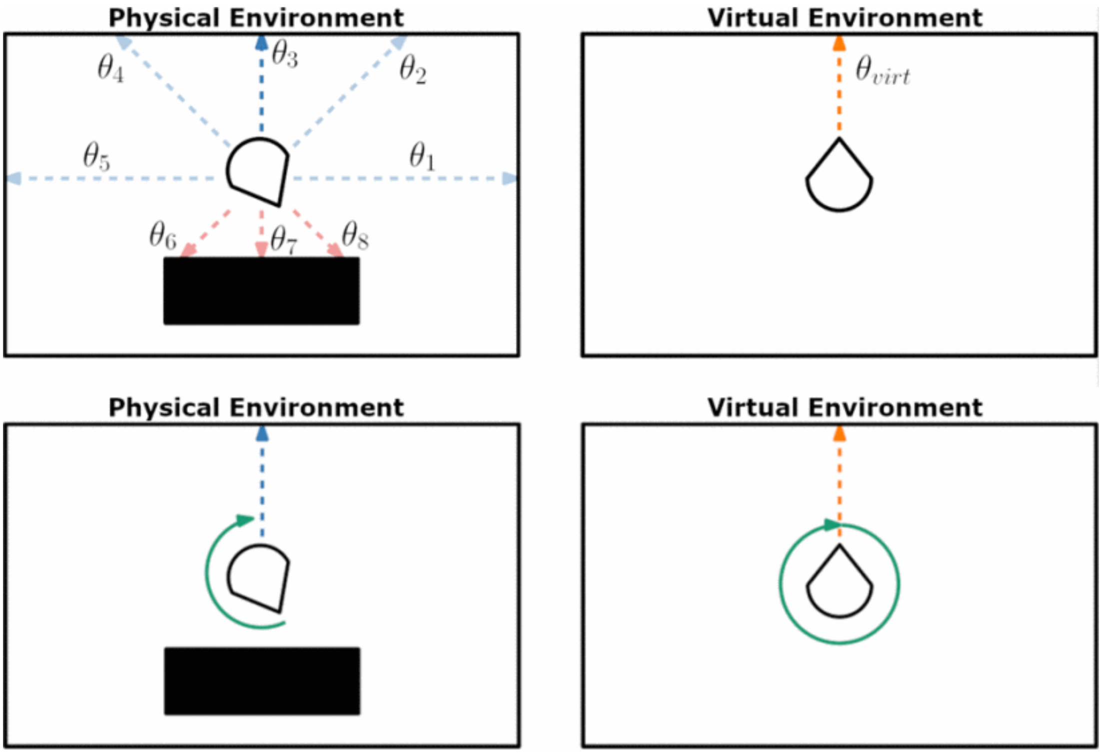


FIGURE 3.2: An example of how ARC uses raycasting to calculate the optimal reset direction, as originally presented in Williams, Bera, and Manocha, 2021a. Note that less rays are shown in the image to reduce visual clutter.

ARC Reset

The ARC algorithm has a unique resetting algorithm compared to S2C and RO. As well as redirecting the user away from the obstacle, ARC seeks to align virtual obstacles with physical obstacles. ARC Reset also aims to align the two spaces. The process of how this is done is described in detail in Williams, Bera, and Manocha, 2021a.

In ARC Reset, the user turns 360° virtually. Before the turn, a ray is sent out in front of the user to measure the virtual obstacles nearby (see Figure 3.2). In the virtual representation of the tracking space 20 rays are sent out as well, to find the boundaries and physical obstacles around the user. The algorithm seeks to align the user so that there are no obstacles directly in front of them, while the physical and virtual obstacles align more closely. Once the ray that best achieves these goals is calculated, the direction the user should turn is shown.

In our implementation, an arrow directed the user to turn clockwise or counter-clockwise, as shown in Figure 3.3a. The virtual rotation for the reset is always 360°. The physical rotation is the size of the larger of the two rotation angles (Clockwise or Counter-Clockwise) that the user needs to turn to face the ideal direction calculated from the rays. The minimum the user can physically turn is 180° as the larger of the two angles is always chosen to direct the user to turn along. This means the user turns at a gain level between 1 and 2, which is either less than or equal to the amount of a 2:1 turn. Additionally, the reset is far less likely to get stuck in a corner as ARC uses both virtual and physical obstacles to calculate the ideal direction after the turn.

However, since the real and virtual environments seek to be aligned based on how much the user turned, the algorithm works best when the direction the user was facing when the reset was initiated is the direction, they were planning to continue walking in. If the user was about to turn in a different direction, this algorithm can cause another reset to happen as the user might turn back in the direction of the obstacles.

3.2.4 Virtual Environments

The virtual environments in the experiment were based on the environments introduced in Williams, Bera, and Manocha, 2021a. The environments provide a good sample of virtual environments of different complexities. These environments are used to compare the redirected walking algorithms across different environments. The three environments were:

- Small: a small environment with a single central obstacle in the shape of a cube. This environment, while still larger than the tracking space, was closest to the tracking space in size. It maps well onto virtual environments where the user has a base area they spend most of their time in - a set of small escape rooms[1] or a training area[2] for example.
- Big: a large environment containing multiple irregular obstacles that the player had to navigate around. The virtual environment was many times bigger than the tracking area. The area gives an example of a natural virtual environment that is much larger than the tracking space with obstacles with no specific pattern such as an outdoor area[3][4].
- Six Square: a medium environment with six square obstacles placed in two rows inside the environment. This scenario best represents ordered virtual environments, such as a museum[5] or a maze of corridors and rooms inside a building[6].

Each user went through each of these three environments once, starting in the Small environment, followed by the Big environment and finally the Six Square environment. The users were split into three groups with each group having a unique

[1]Such as in Escape Simulator - store.steampowered.com/app/1435790/Escape_Simulator/, last accessed: September 5, 2025

[2]Such as in Job Simulator - https://store.steampowered.com/app/448280/Job_Simulator/, last accessed: September 5, 2025

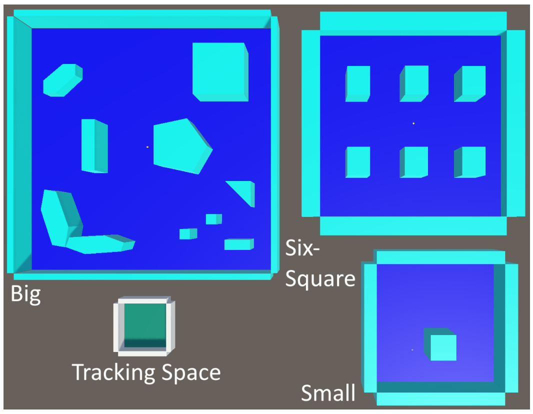
[3]Such as in Gorilla Tag - https://store.steampowered.com/app/1533390/Gorilla_Tag/, last accessed: September 5, 2025

[4]Such as in Paper Beast - https://store.steampowered.com/app/1232570/Paper_Beast/, last accessed: September 5, 2025

[5]Such as in Smithsonian American Art Museum Beyond the Walls - https://store.steampowered.com/app/1087320/Smithsonian_American_Art_Museum_Beyond_The_Walls/, last accessed: September 5, 2025

[6]Such as in Superhot - https://store.steampowered.com/app/617830/SUPERHOT_VR/, last accessed: September 5, 2025

(A) Starting an ARC Reset - a prompt appears in front of the user with an arrow for the turn direction. At the bottom of the screenshot are experimenter-only view windows showing the alignment of the virtual and physical space.



(B) The three virtual environments and a virtual representation of the 3.5m × 3.5m tracking space (included for relative size reference).

FIGURE 3.3: The Virtual Environments

TABLE 3.1: Experiment Groups

Experiment Group	Small	Big	Six Square	Group Size
A	RO	S2C	ARC	13
B	ARC	RO	S2C	11
C	S2C	ARC	RO	12

environment-algorithm combination. Table 3.1 shows the three groups and which algorithm they had for each environment. More participants went through Experiment Group A than B and C due to a recording error with the experiment system for two participants. Non-parametric tests were chosen so that the difference in group size would not impact the analysis.

3.3 User Study

3.3.1 Participants

Ethical approval for the study was received from the ethics board of Maynooth University and the study was advertised around campus. Within the single experiment session, each participant completed two tasks, a rotation task (see Chapter 4) and the navigation task described in this study. Thirty eight participants were recruited, however due to technical difficulties two participants were unable to complete the navigation task. Thirty six participants completed the navigation task - 20 Male, 14 Female, 1 Non-binary and 1 preferred not to identify, with an age range of 18-54 (mean age 24).

36.4% of participants had no VR experience, another 30.3% had a little VR experience, 21.2% had some VR experience and 12.1% had extensive VR experience.

3.3.2 Procedure

1. The participant entered the user study room, signed a consent form and was given instructions on how to put on the VR headset and use the controllers.

2. The participant completed a rotation task. This task was completed prior to this user study, see Chapter 4 for details.

3. The participant took a break of at least 2 minutes.

4. The navigation task was started in the Small environment, when the participant was ready. In the navigation task, the participant walked through the presented virtual environment for 2 minutes collecting orbs. The first orb appeared directly in front of them and every subsequent orb was between 2 and 6 meters away in the virtual environment in a preset order in each environment. A text prompt appeared when the user was about to leave the tracking area which started the reset procedure to keep the user within the tracking space. See Figure 3.4 for an example of what a participant saw inside the environment.

5. The participant was offered a short break.

6. The participant completed the Big environment, with the same task as step 4.

7. The participant was offered a short break.

8. The participant completed the Six Square environment, with the same task as step 4.

9. The participant completed a questionnaire containing a System Usability Scale (SUS) (Brooke, 1996), a simulator sickness questionnaire (SSQ) (Kennedy et al., 1993) and a question asking them to rank which of the environments they preferred. A free text box for additional comments was offered as well. Appendix A and B show examples of these questionnaires.

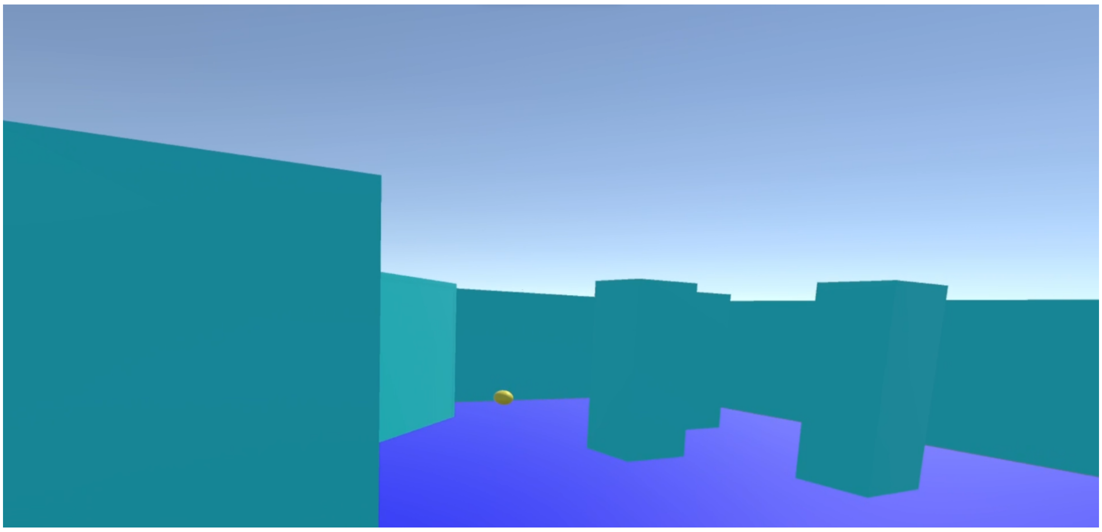


FIGURE 3.4: The participants collected orbs in the virtual environment. Here a yellow orb is shown that the user is walking towards.

Each of the three virtual environments in the user study used a different redirected walking algorithm. Participants were assigned a group in order of appearance – the first participant was in group A, the second in group B, the third in group C with the environment-algorithm combinations described in 3.2.1. On average, the navigation task and questionnaires took about 15 minutes to complete.

3.4 Results

In this section, research questions are presented for how the participants will respond to the different environments. First user preferences and comments are analysed, which adds to the area of user experience with redirection. Then the simulator sickness questionnaire and environment metrics are described as these are commonly measured when evaluating redirected walking algorithms and can be compared with the existing literature.

Research Questions

Based on existing literature and the focus of the user study, the following research questions were introduced:

- **RQ1 - Which redirected walking algorithm did users prefer and why?** Initially, it was hypothesised that users might prefer the algorithm with the fewest resets. Resets interrupt user navigation and tasks which might disrupt users. Additionally, the strong redirection required for resetting can lead to breaks in presence where the user no longer feels immersed in the virtual environment.
- **RQ2 - Which of the redirected walking algorithms had the best performance in the small tracking space?** As S2C uses more redirection to keep users inside the tracking space, it was expected that S2C would have fewer resets and thus enable users to walk larger distances and collect more waypoints than RO. Similarly, ARC might perform better than RO as ARC has more strategies to keep the user inside the tracking space. It might have fewer resets due to the increased redirection and thus users should travel further, and collect more waypoints.
- **RQ3 - Did the layout of the virtual environment impact the performance of the redirected walking algorithms?** The three different virtual environments had different layouts and sizes. Each virtual environment also had a different number and complexity of obstacles. This could have impacted the efficiency of the redirected walking algorithms.
- **RQ4 - How did the use of redirected walking algorithms impact the user experience of navigating through virtual environments?** Each of the redirected walking algorithms used has a different strategy for redirecting users. This could impact how users perceived the virtual environments they were in. Additionally, the user of strong gain for prolonged periods can lead to simulator sickness which could impact user's experience in VR.

3.4.1 Participant Preference

This section discusses which of the three algorithms (RO, S2C and ARC) users preferred in relation to RQ1 (see section 3.4). Participants were given a questionnaire to fill out at the end of the user study. It included a question on which of the environments they preferred. Participants ranked their favourite of the three environments

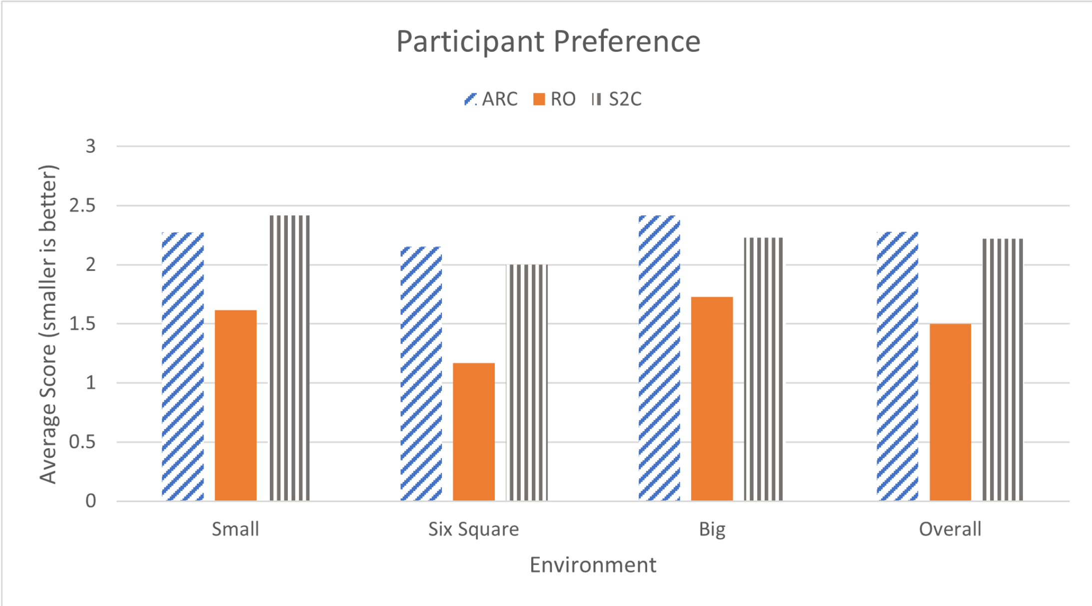


FIGURE 3.5: Participant preference for each algorithm in the different environments as well as across the different environments. Lower scores suggest participants preferred the algorithm.

as 1, their second favourite as 2 and their least favourite as 3. Participants were unaware of the different algorithms assigned to the environments depending on their group. Thus, their environment preference score was used to indicate both algorithm and environment preference. The scores each participant gave each environment were added up to see if there was a general preference in either algorithm or environment. A lower score suggests that participants in general preferred that algorithm or environment. Both the S2C algorithm and the ARC algorithm received similar overall scores of preference from the participants with total summed scores of 80 ($\overline{x}$ = 2.22) and 82 ($\overline{x}$ = 2.28) respectively. In contrast, the RO algorithm received a score of 54 ($\overline{x}$ = 1.44). Figure 3.5 shows the mean score of each algorithm broken down by each environment condition as well as the overall score. Wilcoxon Signed-Rank Tests found a statistically significant difference in participant preference. Participants preferred the RO condition to the S2C algorithm (T = 144, $p < 0.01$) and the ARC algorithm (T = 153, $p < 0.01$) for all environments.

Of the three environments, it appears participants preferred the Six Square environment, giving it an average score of 1.8. Both in the Small environment with one central obstacle and the Big environment with multiple, irregular obstacles received an average score of 2.1. However, the order of environments was not counterbalanced (for practical reasons), this finding may be circumstantial as a result.

3.4.2 Algorithm Performance

This section compares the performance of the three different algorithms (RO, S2C and ARC) to each other and to themselves across the three different virtual environments (Small, Big and Six Square). This relates to RQ2 and 3 (see Section 3.4). The metrics for comparing the performance of the algorithms is described. This is followed by an initial visual analysis of the data. Based on the visual analysis, statistical test are run to find any significant differences between the metrics.

Environment Metrics

The following metrics were recorded for each participant in the study:

- **Number of Waypoints Collected**: participants were asked to collect orbs around the environment. Although the aim of orb collection was simply to give the participant a task that made them move around the environment, the number of orbs collected can indicate how well users are able to complete navigation tasks with the different algorithms.
- **Number of Resets**: how many times a reset was activated during redirection. Since resetting is an overt technique that can cause a user to become aware of the redirection, the aim is to reset as little as possible. This is a common metric used in redirected walking algorithm studies (Hodgson and Bachmann, 2013; Azmandian et al., 2015; Thomas and Rosenberg, 2019a; Bachmann et al., 2019; Williams, Bera, and Manocha, 2021a; Fan et al., 2023).
- **Average Distance between Resets**: this measures on average how far the user moved before they were reset. Measured in meters.
- **Average Time between Resets**: how long on average did it take for the next reset to trigger after the previous one. This metric alongside distance between

TABLE 3.2: Mean Algorithm Efficiency based on Numeric Metrics - the environment-algorithm columns give the combination of the environment-algorithm the participants were in. The mean Waypoints, Resets, Distance Resets (DR), Time Resets (TR) and Distance Travelled (DT) are the metrics described in section 3.4.2.

Environment	Algorithm	Waypoints	Resets	DR (m)	TR (s)	DT (m)
Small	ARC	3.82	8	4.55	22.06	39.67
Small	RO	7.08	11	3.76	15.45	42
Small	S2C	8	11.75	3.78	16.54	45.78
Big	ARC	3.17	10.83	3.92	15.54	43.91
Big	RO	4.55	14.27	4.10	11.42	54.92
Big	S2C	3.69	13.23	3.68	12.25	49.55
Six Square	ARC	6.62	12	3.36	12.58	42.02
Six Square	RO	10.5	13.17	3.28	11.27	44.78
Six Square	S2C	10.82	12.27	3.71	11.53	47.92

resets can indicate different walking speeds - the user may cover little distance between resets but walk slower or vice versa. Measured in seconds.

- **Distance Travelled**: How far each participant travelled in each environment. A larger number suggests the participant walked faster or faced less disruptions from resets. This would suggest participants found the environment easier to navigate. Measured in meters.

Table 3.2 shows the mean score of each metric for every environment-algorithm combination.

Visual Analysis

Box plots were created to get an initial sense of the spread of the data for each of the five metrics. Figure 3.6 shows a breakdown of the relative performance of each metric for each algorithm in each environment. Each box plot shows the median (line inside the box plot), the mean (the X inside the plot) and the interquartile range of one of the three algorithms. The blue diagonally striped box plots relate to ARC, the orange box plots relate to RO and the grey vertical striped box plots relate to S2C. On the X-axis the environment conditions (Small, Six Square and Big) are described. The Y-axis shows shows the score of the metric. The title of the plot describes which of the metrics the box-plot refers to.

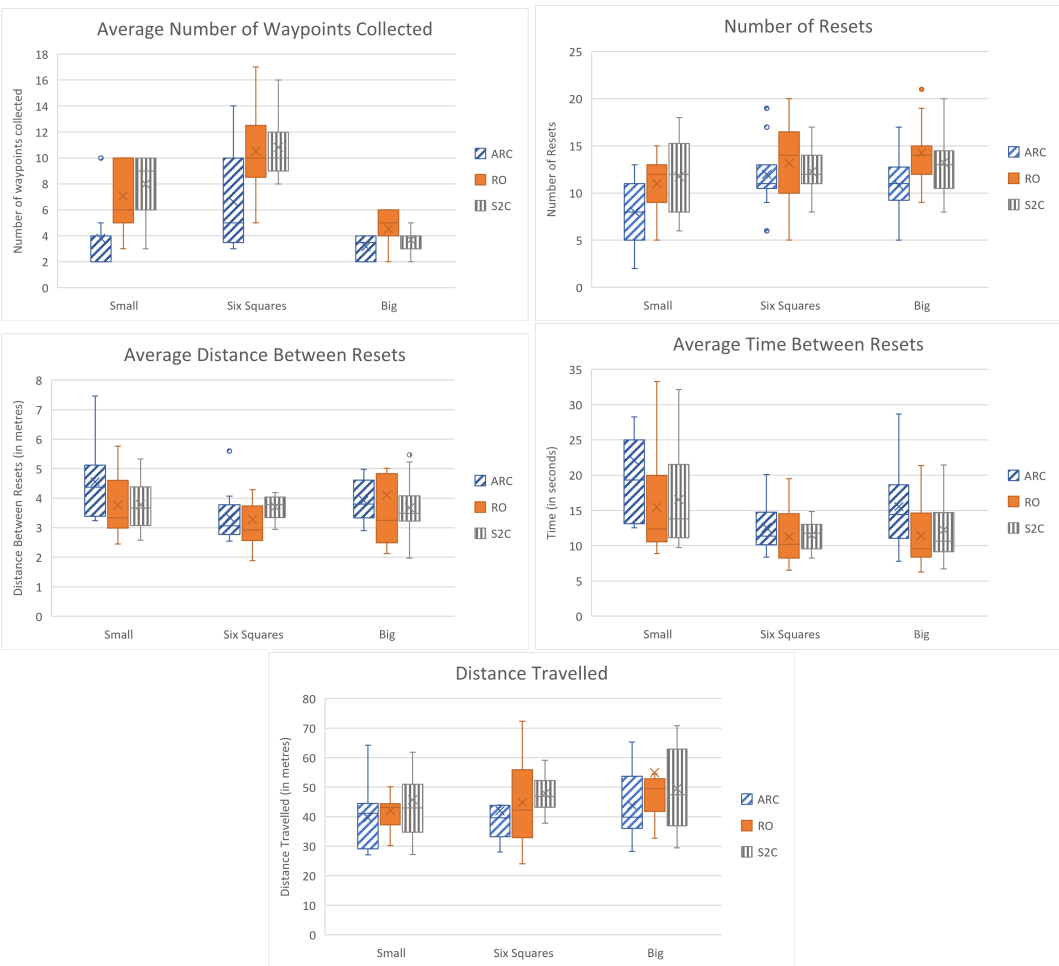


FIGURE 3.6: Relative performance of algorithms in the different environments for each metric.

From the plots, the Average Number of Waypoints collected shows that in the Big environment users collected fewer waypoints than in Six Square or (for RO and S2C) Small environment. The ARC algorithm in the Small environment also led users to collect fewer waypoints. In general, users collected less waypoints in ARC than the other two algorithms. However, the Average Number of Resets for ARC is lower than for S2C and RO, especially in the Small environment.

In the Average Time Between Resets, the Small environment had the longest time between resets and ARC had longer times on average than S2C or RO which aligns with the fewer resets found when using the ARC algorithm.

The Average Distance Travelled in all three environments shows significant variation between participants. However the average distance travelled was roughly similar when comparing the three algorithms and environments together.

Statistical Analysis

Based on the differences found in Figure 3.6, statistical analyses were performed. The metrics were compared in two ways in relation to RQ2 and RQ3 (see Section 3.4:

- **By Algorithm**: the performance of the three algorithms (Reset Only, S2C, ARC) was compared within a single environment, this relates back to RQ2.
- **By Environment**: the performance of a single algorithm was compared to itself across the three different environment conditions (Big, Small, Six Square). This relates to RQ3.

Due to the small sample size in each group and the difference in group size (11, 12 or 13), non-parametric tests were chosen for the statistical analysis. The Kruskal-Wallis test is a non-parametric equivalent of the ANOVA. It's used to find differences between two or more groups. To determine which of the groups are different, the Mann-Whitney U test was used as a post-hoc test. The Mann-Whitney U test compares two samples to each other with the null hypothesis that they are the same. Rejecting the null hypothesis suggests there is a difference between the samples.

The Kruskal-Wallis tests were conducted for each of the algorithms and environment comparisons mentioned above at the 0.05 significance level. If the Kruskal-Wallis test found a difference between the groups, three Mann-Whitney U tests were run to find which groups were different from one another. A Bonferroni correction was used with the results reported at a 0.0167 significance level as significant.

Table 3.3 shows the significant results of the Mann-Whitney U follow-up tests comparing algorithms [7]. Table 3.4 shows the significant results of the environment comparison. The metric column describes which metric (see section 3.4.2) the difference was found in. The algorithm and environment columns give the algorithms and environments that were compared. The p column gives the significance level of the *p-value*, U is the test statistic and r is the effect size. The *p-value* describes whether two groups are statistically different from one another, while the effect size describes how big the difference between the two groups is. An effect size of 0.1 to 0.3 is considered small, 0.3 to 0.5 moderate, and larger than 0.5 large (Cohen, 1988).

Environment Comparison

Based on the statistical tests summarised in Table 3.3 the following conclusions were drawn about the different virtual environments in relation to RQ3 (the impact of the virtual environment layout on performance):

- **Participants walked similar distances in all the environments**: There were no significant differences between the Average Distance participants walked either between environments (ARC: $H(2) = 0.7851$, $p > 0.05$, S2C: $H(2) = 0.8263$, $p > 0.05$, RO: $H(2)= 2.1579$, $p > 0.05$) or between algorithms (Small: $H(2) = 1.2644$, $p > 0.05$, Big: $H(2) = 1.2292$, $p > 0.05$, Six Square: $H(2)= 3.6777$, $p > 0.05$). This suggests participants were equally comfortable walking under all the experiment conditions. As a result, the average Distance Travelled was not evaluated further.

- **Users had a more difficult time finding waypoints in the Big environment than the Six Square environment**: Table 3.3 shows there were significantly fewer waypoints collected in the Big environment than the Six Square environment

[7] Due to the large number of items tested, only statistically significant results are shown in Table 3.3 and Table 3.4

across all three algorithms. There was also a significant difference in the number of waypoints collected between the Six Square and Small environments with the RO algorithm and between the Big and Small environments with the S2C algorithm. This difference is likely due to the Big environment containing waypoints that were hidden from view by environmental obstacles in some locations.

- **The layout of the virtual environment had a significant effect on resetting with the ARC algorithm**: With the ARC algorithm significant differences were found in the Number of Resets and the Distance between Resets (see Table 3.3) between the Six Square and Small environments. ARC's performance changed depending on the environment.

TABLE 3.3: Mann-Whitney U Test Significant Results comparing Environments, statistically significant results are highlighted as * = p<0.0167, **= p<0.005, *** = p<0.001

Metric	Algo	Env 1	Env 2	p	U	r
Waypoints	RO	Big	Six Square	***	5	0.78
Waypoints	S2C	Big	Six Square	***	0	-0.84
Waypoints	ARC	Big	Six Square	*	30	0.52
Waypoints	RO	Six Square	Small	*	30	-0.52
Waypoints	S2C	Big	Small	***	10	-0.73
Resets	ARC	Six Square	Small	*	28	0.51
Distance between Resets	ARC	Six Square	Small	*	25	-0.54

TABLE 3.4: Mann-Whitney U Test Significant Results comparing Algorithms statistically significant results are highlighted as * = p<0.0167, **= p<0.005, *** = p<0.001

Metric	Algo 1	Algo 2	Env	p	U	r
Waypoints	ARC	RO	Small	**	20	0.6
Waypoints	ARC	RO	Six Square	*	32.5	-0.49
Waypoints	ARC	S2C	Small	**	14.5	0.65
Waypoints	ARC	S2C	Six Square	**	26.5	-0.53

Algorithm Comparison

In relation to the performance of the redirected algorithms in a small tracking space (RQ2), statistical tests were run to find any differences in performance. The Mann-Whitney U Tests did not find any significant differences between the RO and S2C algorithms. In contrast, ARC had fewer resets than S2C and RO. A Kruskal-Wallis test showed a significant difference in the number of resets between the three algorithms in the Small ($H(2) = 6.0047$, $p < 0.05$) and Big ($H(2) = 6.1876$, $p < 0.05$) environments. Figure 3.6 'Number of Resets' shows that ARC had fewer resets on average than the RO or S2C algorithm in the Small and Big environments. However, follow-up Mann-Whitney U tests were not significant at the 0.0167 confidence level.

Despite causing less resets, participants also collected fewer waypoints in ARC compared to S2C and RO. The results of the Kruskal-Wallis test show significant differences between the number of waypoints collected in the three environments between the algorithms (Small: $H(2) = 13.0642$, $p < 0.01$, Big: $H(2) = 0.0424$, $p < 0.05$, Six Square: $H(2)= 8.9189$, $p < 0.05$). Additionally, Table 3.4 shows that significantly fewer waypoints were found in ARC compared to S2C and RO in the Small and Six Square environments.

Based on the results of the Mann-Whitney U Tests a rating system was devised to compare the performance of the algorithms to each other. The results of the Mann-Whitney U Test were ranked, with the best algorithm receiving a score of 3, the second best a 2 and the worst a 1. If no significant difference was found in the Mann-Whitney U test between two conditions, the mean of the two scores was taken and used for both. For example, there was a significant difference found in the number of Waypoints collected between ARC and Reset Only and also ARC and S2C in the Small environment. However there was no significant difference between Reset Only and S2C under the same conditions. Since ARC had significantly fewer waypoints than the other two algorithms, it received a rank of 1 (the worst). S2C and Reset Only then received a rank of 2 and 3. However, since there was no significant difference between the two, the rank of 2 and 3 was averaged to 2.5. Both S2C and Reset Only then received the rank of 2.5.

Table 3.5 shows how well each algorithm performed in each environment for each metric. No significant differences were found between the Total Distance or the Distance between Resets metrics, and they were omitted from the graph as a result. Table 3.5 indicates that ARC performed better in the Small environment than RO and S2C

TABLE 3.5: Comparative Performance of the Algorithms Across Environments - a larger total suggests better performance.

	Small			Big			Six Square		
Metric	RO	S2C	ARC	RO	S2C	ARC	RO	S2C	ARC
Waypoints	2.5	2.5	1	2.5	2	1.5	2.5	2.5	1
Time Resets	1.5	1.5	3	2	2	2	2	2	2
Resets	1.5	1.5	3	1.5	2	2.5	2	2	2
Total	5.5	5.5	7	6	6	6	6.5	6.5	5

- participants had the fewest resets and the longest time between resets. Similarly, in the Big environment participants were reset less with ARC. However, participants also collected fewer waypoints with the ARC algorithm than S2C and RO across all three environments.

3.4.3 User Experience

The following section describes the user experience of the virtual environments and algorithms during the user study. This relates back to RQ3 (see section 3.4). Two questionnaires and an open question box were given to participants to help understand their experience.

Usability

Usability is a measure of how easy and intuitive a system is to use. Low usability suggests user found it difficult to use a system. This could lead users to have a worse experience with the system as they struggle to complete tasks in it. Conversely, high usability suggests an easy to use system that brings a better user experience.

In this user study, the participant completed a System Usability Scale (SUS) questionnaire (Brooke, 1996) after they had completed all three virtual environments. The SUS measures how easy a system is to use. It makes ten statements about different aspects of the system, which users can agree or disagree with on a scale from 1 (strongly disagree) to 5 (strongly agree). A complete copy of the SUS used in this user study can be found in Appendix B. The mean SUS score was 74 with a standard deviation of 14.59. This suggests participants found the system to have above average usability.

Simulator Sickness Levels

Simulator Sickness can cause users to feel disoriented or sick during or shortly after experiencing VR. This can have a strong impact on user experience with a system. Users are less likely to want to use a system that makes them feel sick and it can also impact task performance. The Simulator Sickness Questionnaire (SSQ) (Kennedy et al., 1993) is commonly used to measure participant simulator sickness after completing a task in Virtual Reality.

Participants completed the SSQ at the end of the user study, after they had completed all the tasks in VR. Figure 3.7 shows the SSQ scores for each of the three subscales and the total score of the SSQ. The mean total score after the experiment was 15.58. For the field of redirected walking, Gemert et al., 2024 suggests thresholds of None (< 5), Low (5–15), Medium (15–30), and High (> 30) when comparing different locomotion techniques to other works in the field. The mean score of 15.58 is on the low side of the medium bracket. Compared to other studies that use rotation and curvature gain, this is a lower score than would be predicted (20.61 ± 1.48 using Gemert et al., 2024). This could be due to the short duration of time participants spent in each of the virtual environments. Participants spent less than 10 minutes total across the three virtual environments presented in this task.

The disorientation subscale showed the highest level of symptoms (mean = 29.77), this is in line with other virtual reality studies which report higher levels of disorientation. Nine participants showed no simulator sickness symptoms and a further fifteen marked between 1 and 5 points across all the symptoms in the questionnaire. This suggests that most participants experienced either none or mild simulator sickness during the study.

Participant Comments

In addition to the SUS and SSQ, direct comments from participants about the system also uncovered aspects of the user experience. These responses give additional insight into the system and provide examples of why participants preferred the RO algorithm to S2C and ARC. A comment box was left for participants at the end to give these

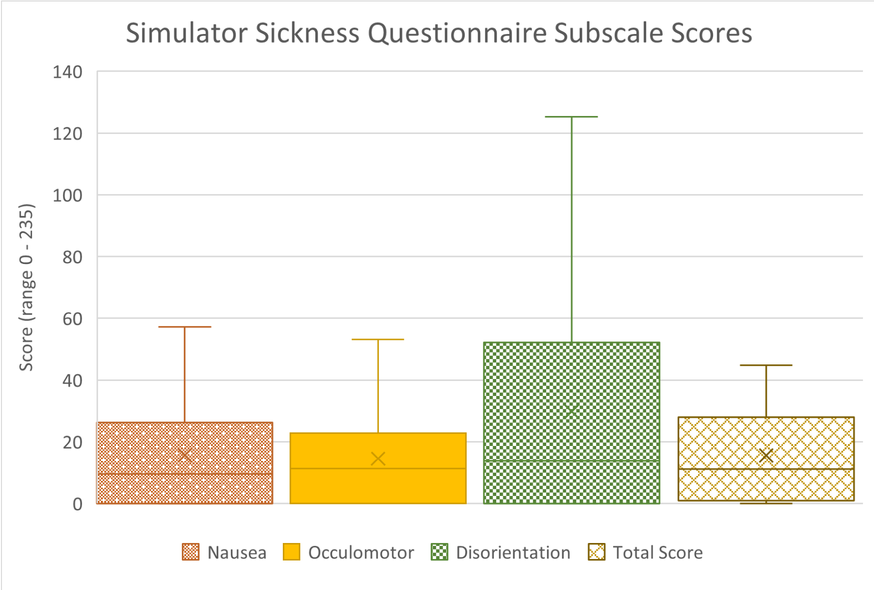


FIGURE 3.7: Simulator Sickness Questionnaire (SSQ) SubscaleResults - X is the mean, the line in each box the median. The SSQ score is just above the beginning of the medium score bracket for redirected walking (15-30, Gemert et al., 2024).

additional comments. Appendix C contains a list of all the comments participants left. Among the comments there were a few running themes which are summarised in this section:

- **Navigation in the virtual environments:** Some participants commented more specifically on the environments presented. A participant was worried about bumping into walls as they no longer felt oriented within the tracking space but only the virtual space, “There was a strong sense I could walk into a wall as I did not know where in the real room I was”. Some participants noticed some of the effects the gain had on the environment, commenting “movement felt mostly consistent, but position of environment...seemed to drift” and "lurch or motion blur that made orientation feel off". Another participant commented “I enjoyed the last level (S2C) the most as it challenged my perception”.

- **ARC Reset causes confusion:** A participant mentioned preferring the 2:1 turn compared to ARC Reset due to the direction arrows causing confusion. “I felt more comfortable with the second environment. There was no directional arrows but I had the right information to move. The third environment has more information about direction but it sometimes showed a different direction than where the ball was. I was confused in the 3rd game then.” The goal of ARC Reset is to turn the participant 360° virtually while turning them the larger of two directions in the real space (hence the arrows). However if a participant turns in a different direction to the arrow they will end up facing a different direction than the ball by the time the turn finishes. This suggests that simpler resetting mechanisms might be better even if they use higher gain levels to achieve the reset so as not to confuse participants.

- **Hardware constraints negatively impacted user experience:** Multiple participants commented on the wire of the headset getting in the way of the experience, especially when turning – “handling the wire during the game was a bit difficult”, “Felt afraid to move due to the wire as the system didn’t estimate that”. The wire was held up off the ground by the experimenter but it could tangle if the system asked the participant to spin multiple times in short succession. Some modern headsets do not use a wire to anchor the headset to a PC and the industry has come up with solution to place wires overhead to avoid such issues. Future experiments should keep these solutions in mind, especially when asking the participant to turn often. Additionally a participant mentioned they wanted “more space to move” suggesting that a larger tracking area might be preferred.

- **Simulator Sickness:** Two users commented on a "dizzy sensation" or being "a little dizzy" in the third virtual environment. This supports the idea that the comparatively low simulator sickness found in this user study could be partially due to the short amount of time users spent in VR. A longer study where users spend more time in each environment is needed to draw any more conclusive evidence.

3.4.4 Discussion

From the previous results and analysis the following conclusions were drawn:

- **Participants preferred the RO algorithm**: In relation to RQ1, participants were asked for their preference of the three algorithms. The RO algorithm was preferred in all three environments by participants. S2C and ARC received similar preference scores, as discussed in Section 3.4.1. The environment preference did not correlate with the number of resets as was initially hypothesised. Instead, participants preferred the algorithm which had a greater number of resets but did not otherwise use gain. RO only applied gain during resets as compared to the constant gain used in S2C and ARC. Additionally, some participants experienced mild simulator sickness symptoms during the study which might have impacted their preference.

- **S2C had similar performance to RO, while ARC caused fewer resets**: The performance of the three algorithms relative to each other was considered in RQ2. Across all environment conditions, S2C had a similar performance to RO with no significant differences found between the two algorithms. This suggests that in small tracking spaces at gain levels below the detection threshold, S2C is of limited use. ARC had fewer resets than S2C and RO in the Small and Big environments. Despite this, participants collected fewer waypoints with the ARC algorithm in the Small and Six Square environments than the other two algorithms. This is a mixed result, suggesting that ARC has different advantages and disadvantages when used in small tracking spaces compared to RO and S2C.

- **The layout of the virtual environment had a significant effect on resetting with the ARC algorithm**: RQ3 compared the performance of each algorithm across the different environments. All three algorithms differed in the number of waypoints users collected across the environments. There was a difference in the number of waypoints collected across all algorithms in the Big environment compared to the Six Square environment. This is most likely due to the waypoints being easier for users to find and collect in some environments than others.

S2C and RO had similar performance in terms of the other metrics across all three environments. In contrast, ARC's performance changed depending on the environment. Significant differences were found within the ARC algorithm in the Number of Resets and the Distance between Resets (see Table 3.3) between the Six Square and Small environments. This is in line with previous studies (Williams, Bera, and Manocha, 2021a; Williams, Bera, and Manocha, 2022), where the performance of ARC is dependent on the local similarity between the tracking space and the virtual environment it is trying to align.

- **User experience was impacted by hardware and software constraints**: RQ4 considered the user experience of participants using the three redirected walking algorithms within the user study. Participants gave the system an above average usability and had below average simulator sickness for this type of system. Despite this, user comments revealed that by the third environment some users felt dizzy. This suggests the lower simulator sickness could be due to the short duration of the study.

  Hardware constraints impacted user experience, such as the wire on the VR headset hindering movement and the limited tracking space. Some participants also commented on the navigation experience. They mentioned feeling disoriented in the tracking space as they were unsure of where they were compared to the virtual environment. Additionally some participants described the results of the gain used for redirection as being disorientating, describing it as 'motion blur', 'drift' and 'jitter'.

3.4.5 Limitations

A limitation of this study was the short time participants experienced each of the redirected walking algorithms. Participants spent two minutes in each of the virtual environments covering an average distance of 45m. A longer period of time may have shown further differences between the algorithms. More research is needed to find the full effects of redirected walking algorithms have on participants. Additionally, while this user study had a larger participant count (36) than most studies using redirected walking algorithms, a greater number of participants would provide more robust results.

While the environments had different layouts and numbers of obstacles, all three shared a purple and blue colour scheme, flat lighting and low-poly visuals. The similarity of the environments minimised potential differences that were not directly related to the number of obstacles present for navigation. However, this limits the study's applicability in more realistic environments. A more realistic virtual environment with multiple light sources might aid users' navigation and depth perception compared to the flat edges and lighting of the environments in this experiment. The additional environmental detail in more complex scenes would give the user more reference points to help with navigation. Future work could explore users' response to redirected walking algorithms in realistic virtual environments and how it compares to simplistic virtual environments. The impact of the number of visual cues in a virtual environment on user perception of rotation gain is discussed further in Chapter 4.

3.5 Recommendations for Developers

This user study contributes to the under-represented field of redirected walking in small tracking spaces with emphasis placed on the user experience. The results of this user study can help developers create redirected walking applications to better suit real users. The following recommendations are offered to developers of Virtual Reality applications targeting consumer-grade headsets for personal use:

- **Use Resetting or Environment Manipulation** - In the small tracking space of 3.5m $\times$ 3.5m the RO algorithm performed similarly to S2C while being preferred by participants. This reinforces the finding that in small tracking spaces, reactive redirected walking algorithms have limited applicability (Williams, Bera, and Manocha, 2021a). In such tracking spaces, developers could use alternative redirection techniques such as environmental manipulation (Dong et al., 2021, Langbehn and Steinicke, 2019, Suma et al., 2011, Krueger, Markham, and Bierig,

2024) or choose an algorithm that focuses on resets (such as RO or Point of Interest aware Redirected Walking (Xu et al., 2022)).

- **Use Predictable Resets** - ARC Reset confused participants as it changed the amount and direction participants had to turn physically. Even though participants had fewer resets using the ARC algorithm, participants collected fewer waypoints than in S2C and RO, which both used the 2:1 turn reset.
- **Ensure Local Similarity with ARC** - In the Small environment, which held a single obstacle and best matched the physical tracking space, ARC performed better than S2C and RO in terms of the number of resets. This reinforces a similar finding in Williams, Bera, and Manocha, 2021b that the local similarity of the physical and virtual space impacts ARC performance. It is suggested to use the ENI measurement tool to find the similarity (Williams, Bera, and Manocha, 2021b). This limits the applicability of ARC for VR experiences where the tracking space dimensions are not known in advance.
- **Ensure User Comfort** - Participant comments describe cumbersome aspects of using the VR headset such as the wire attached to the PC. Some users were concerned about bumping into obstacles as they felt disorientated in the tracking space, despite resetting keeping them inside it. These elements, which do not appear in simulation studies, should be accounted for when implementing redirected walking algorithms for live users.

3.6 Summary

From the findings of the literature review, the research question of "How effective are existing, generalised redirected walking methods in small tracking spaces?" was introduced. This chapter considers this question with a user study comparing three different redirected walking algorithms in a small tracking space of 3.5m × 3.5m. The first algorithm, Reset Only (RO), reset the user when they were near the edge of the tracking space, but otherwise mapped physical movements directly onto virtual movements. Steer-to-Center (S2C) aimed to redirect the user towards the centre of the tracking space. Alignment-based Redirection Controller (ARC) aligned real and virtual obstacles, so the user could avoid real obstacles in the tracking space as they avoided the virtual obstacles.

Participants preferred the RO algorithm, despite RO causing more resets than ARC. Participant comments suggest this could be due to the predictable 2:1 reset in RO. Compared to S2C, RO had similar performance while redirecting participants less. ARC had fewer resets than RO and S2C but participants also collected fewer waypoints within the task. Additionally, ARC's performance varied depending on the local similarity between the physical and virtual environments.

Participants rated the usability of the system as above average and simulator sickness in this study was lower than in other redirected walking studies that used rotation and curvature gain. This could be due to the short duration of the study, especially since two participants reported they felt dizzy in the final environment. Participants also made note of cumbersome aspects of the hardware (such as the wire) and software (such as resetting), suggesting these aspects impact user experience.

Redirected walking gain below the detection threshold was shown to have minimal benefit in small tracking spaces based on this user study. Recommendations are offered to developers - consider user comfort with VR and redirection when building applications. Users preferred the simple, overt redirection of resetting. Resetting and environment manipulation show promise for use with small tracking spaces.

Based on the results of this study, the limitations of existing generalised redirected walking algorithms in small tracking spaces were highlighted. The next two studies, described in Chapter 4 and 5, consider different methods to help overcome this limitation. In the next Chapter, rotation gain above the detection threshold is considered in terms of its impact on user turning accuracy. Other environmental factors are also looked at, especially in relation to predicting future user trajectories.

Chapter 4

User Study of Rotation Accuracy Under Different Rotation Gain Conditions

This chapter analyses turning accuracy under different levels of rotation gain and environment conditions. It expands our knowledge of how rotation gain impacts users and proposes a model of user response to gain. A user study with 38 participants was run that compared the accuracy of user turns under varying conditions. Users were asked to turn three different amounts in both directions in two different virtual environments at four different levels of gain. The complex virtual environment provided plenty of visual cues for the user to help orient themselves in the environment. The minimal virtual environment contained a single visual cue to help users assess the gain level.

The results of the user study show the direction of the turn had no impact on turn behaviour. In contrast, the virtual environment, the gain level and the turn amount all had a statistically significant impact on rotation accuracy. Participants were strongly affected by gain in the complex environment, while only being affected by higher gain levels in the minimal environment. Participants were grouped by how they responded to the gain to help future developers create prediction models for how users will move in VR under varying gain conditions.

4.1 Introduction

Walking is how we naturally move through physical spaces in our day to day lives, avoiding obstacles and turning to change the direction we travel in. In virtual reality (VR), real walking is considered the most natural navigation technique as the user moves through the virtual environment by walking as they would in the physical space (Slater, 2009; Usoh et al., 1999; Langbehn, Lubos, and Steinicke, 2018b). Real walking aids in navigation and search-related tasks in VR (Ruddle and Lessels, 2009) and users often prefer it to other navigation methods (Mayor, Raya, and Sanchez, 2021).

However, the physical tracking space a user can safely walk in while immersed in VR is often limited. Redirected walking uses perceptual manipulations to redirect the user around a virtual environment that is larger than the physical space available to them. This extends how long a user can walk in a virtual environment without moving outside the tracking space, increasing immersion.

In the physical world, users have multiple sensory modalities they use to keep track of their position in space and how far they have rotated. The vestibular sense approximates the angular velocity and linear acceleration, while the proprioceptive sense tells the user where their body is in space. Vision allows the user to pinpoint their position relative to the environment based on visual cues in their surroundings (for more information refer to Chapter 8 in Jerald, 2015).

When these sensory modalities all contribute to the user's awareness of the turn, they can more accurately judge the amount they have turned compared to when a modality is missing (Bayramova et al., 2021). For example, without a visual sense of the space, the user will slowly drift out of sync in terms of how much they think they have turned compared to how much they actually have turned (Howard et al., 1986). However, the weight users put on different sensory modalities is different among users. In general, users rely less on podokinetic (legs and feet proprioception) cues than optokinetic (visual) cues in judging how far they have turned. In contrast, how much users were influenced by optokinetic cues rather than vestibular cues varied from person to person (Jürgens and Becker, 2011).

Rotation gain (Razzaque, Kohn, and Whitton, 2005) takes advantage of this human limitation. Rotation gain presents a visual representation of a turn that is faster or slower in the virtual environment. The user sees the virtual environment through a viewport. This viewport is manipulated to rotate at a different speed in the virtual environment than the user is physically rotating in the tracking space. However, this also introduces incongruence between users' different senses - they have a mismatch between how much their vestibular and visual systems perceive they have turned. Below a certain threshold, called the detection threshold, the user is unaware of the difference between the amount they perceive they have turned in the virtual environment compared to how much they have actually turned in the physical space (Steinicke et al., 2010).

Redirected walking algorithms combine rotation gain and other types of gain to redirect users inside a small tracking space while they explore a larger virtual environment. The first redirected walking algorithms were reactive, responding only to the user's current position in the tracking space (Razzaque, Kohn, and Whitton, 2005). More complex predictive algorithms instead try to tailor the gain level based on the predicted future movements and paths of the user (see Li, Steinicke, and Wang, 2022 for an overview). For example, Fully Optimized Redirected Walking for Constrained Environment (FORCE) uses path prediction on a limited number of natural paths the user can take in the virtual environment to optimise gains (Zmuda et al., 2013). An alternative approach is to see redirection as an optimal control problem such as with Model Predictive Control Redirection (MPCR), where different redirection techniques are dynamically switched between based on the best outcome using path prediction (Nescher, Huang, and Kunz, 2014). Additionally, Machine learning algorithms, have been used to try and predict the user's future position (for example Cho, Lee, and Lee, 2018; Lemic, Struye, and Famaey, 2022).

With predictive algorithms, it should be understood how users behave at different levels of gain to make more accurate predictions of their future trajectory. If higher levels of rotation gain can be used without decreasing rotation accuracy, then these predictive algorithms could be used effectively in smaller tracking spaces. However, many of these algorithms have primarily been simulated (in some cases have only been simulated, such as Lemic, Struye, and Famaey, 2022), followed by a proof of concept user study. Most of the simulation tests do not adequately account for how real users turn in tracking spaces. The predictions of these algorithms rely on the user accurately turning the amount they are expected to turn. However, the actual behaviour of users in virtual reality deviates from these simulations (Hirt et al., 2022b).

This chapter describes a user study that measures user's rotation accuracy at varying rotation gain levels when turning 45°, 90° and 180° in both Clockwise and Counter-clockwise direction in environments with different levels of visual cues. The following research questions are considered:

- RQ1: Did the different factors within the experiment have an impact on turn accuracy?
  - RQ1.1 **direction** of the turn (Clockwise or Counter-Clockwise)
  - RQ1.2 **turn amount** (45,90,180 degrees)
  - RQ1.3 **virtual environment** users were presented with (minimal vs complex)
  - RQ1.4 **gain level** (1,1.245,1.49,1.98) - see Table 4.1.
- RQ2: Were there interaction effects between the factors that combined to impact turn accuracy?
- RQ3: How did turning in VR compare to turning in the real world without a headset (eyes open or eyes closed)?
- RQ4: Did users become aware of the induced gain level, and if so which factors had an influence?

User models are also presented to help predict how users will turn under different rotation conditions in the user study.

4.2 Measuring Rotation Gain

The rotation gain level is measured as the decimal ratio of the virtual compared to the physical turn amount, as shown in Equation 4.1. A larger difference between the virtual and physical rotation speeds is a stronger level of gain. At a gain level of 1, the size of the physical and virtual rotation are the same. A number greater than 1 has the user turning more in the virtual environment. With a number less than 1 they turn less in the virtual environment than in the physical tracking space. For example, a scenario where the user is turning 60° in the tracking space while turning 90° in the virtual environment would have a gain of 1.5. A user turning 180° in the tracking space leading to a 90° virtual rotation within VR would be a 0.5 gain.

$$G_R = \theta_V / (\theta_R) \tag{4.1}$$

where G_R is the rotational gain, θ_V is the virtual rotation, and θ_R is the physical rotation.

Below a certain threshold, the added gain in the virtual environment is not perceived by the user – they believe they are walking the same way in the virtual and the physical space, even though they are being subtly redirected with every step (Razzaque, Kohn, and Whitton, 2005) . The point at which the user becomes aware of the manipulation is called the *detection threshold*. Initially, it was found that users could be turned physically about 49% more or 20% less than virtually using rotation gain before they noticed the change (Steinicke et al., 2010). Multiple studies (e.g . Grechkin et al., 2016; Langbehn et al., 2017; Hutton et al., 2018; Coelho, Steinicke, and Langbehn, 2022; Brument et al., 2021) have aimed to identify the detection threshold of the user under differing conditions. The exact threshold depends on both the user and the task. Individual rotation gain thresholds vary strongly between users (Hutton et al., 2018; Nguyen et al., 2020a) and even between the same user as they become more experienced with VR (Robb, Kohm, and Porter, 2022).

With the variability in gain thresholds across scenarios, different measures have

TABLE 4.1: Physical Equivalent of 90° Virtual Turn at Different Levels of Rotation Gain. The levels are taken from Steinicke et al., 2010 and Schmitz et al., 2018.

Gain Level	Physical w/ 90° Virtual	Type of Threshold
0.67	134.3	Detection Threshold
1.24	72.6	Detection Threshold
1.85	48.65	Threshold of Limited Immersion
2	45	Highest level Tested

also been introduced. The *threshold of limited immersion* is the point at which the rotation gain level breaks the user's sense of presence (as introduced by Schmitz et al., 2018). The threshold of limited immersion was found to be 1.85. However, the responses of the participants varied in the user study, and some still felt immersed at a gain level of 2, the highest level tested in the user study. Table 4.1 shows the different threshold levels and the physical rotation equivalent for a 90° virtual rotation.

These higher thresholds bring down the physical space requirements for redirected walking. It is unclear how these higher rates of gain affect other task performance. However, curvature gain above the detection threshold can have a negative impact on spatial and verbal memory task performance (Bruder, Lubos, and Steinicke, 2015). The user study on rotation accuracy, which is described in this chapter, used rotation gain levels between 1 and 2. At these levels, users are likely to become aware of the gain with some remaining immersed in the environment at all gain levels. The goal of this study is to find the impact on user turning accuracy of these increased gain levels. This will help future developers in choosing the ideal gain level, especially with predictive algorithms.

4.2.1 Visual Cues

Visual cues are objects in the environment that help the user orient themselves. A complex environment has lots of visual cues, while a sparse or minimal environment has few visual cues. For example, a cluttered room or busy street contains many visual cues, while an empty white room offers limited visual cues. Fewer visual cues in the environment induce higher translation gain detection thresholds while decreasing immersion (Kruse, Langbehn, and Steinicke, 2018). In contrast, in some large virtual spaces where users find it more difficult to measure distance, visual cues in the environment could distract users so that they are less aware of translation gain (Kim et al., 2022).

Paludan et al., 2016 compared rotation gain detection thresholds in three different environments. An environment with no visual cues, an environment with four objects placed in a circle around the user and an environment with sixteen objects placed around the user. They did not find a statistically significant difference between the four and sixteen object condition. However, users were unaware of the gain in the no visual cue condition. The rotation accuracy of users while seated in a turning chair when they turned 90° and 180° with varying levels of visual cues in both virtual reality and a physical space was compared (see Bayramova et al., 2021). It was found that removing visual cues of the user's body, the corners of the room and relative landmarks within the space except for floating circles with no sense of depth, led users to turn less accurately in the virtual environment when compared to a virtual environment where these cues were present. In contrast, removing similar cues in the physical space did not have a significant difference on users' turning accuracy. They hypothesise that the optic flow in the VR condition would be disrupted more by such changes than an equivalent real world condition and thus have a greater impact on turning accuracy.

Based on these findings, two virtual environments were created - a complex environment with many visual cues and a minimal environment with a single visual cue, that falls between the no and four object condition (from Paludan et al., 2016) and few depth cues (like in Bayramova et al., 2021). This allows for comparison of the impact of different environment conditions on user rotation accuracy.

4.3 User Study Setup

This section contains a description of the virtual reality system that was created for the user study, in relation to the key variables chosen and other factors. The system itself is described followed by the procedure of the user study, the participant demographics and data collection methods. Ethical approval for the user study was obtained from the Maynooth University Ethics board.

4.3.1 Independent Variables

Rotation Direction

There were two directions in which participants were instructed to turn - Clockwise (CW) and Counter-clockwise (CCW). Turning repeatedly in the same direction can cause dizziness, thus the system instructed users to turn in different directions. Additionally, this would show if users were more accurate in turning in one direction or another.

Gain Level

In this user study, three gain levels are introduced alongside a control. Each of the three gain levels is twice the previous level of gain and is based on the values of the detection thresholds found by Steinicke et al., 2010 and the threshold of limited immersion introduced by Schmitz et al., 2018:

- **1** - This is a scenario with no gain that acts as a control and provides a baseline measurement for user turning accuracy inside a virtual environment.
- **1.245** - The lowest gain level is 1.245, just above the detection threshold found in Steinicke et al., 2010. The 1.245 condition measures user accuracy at a typical level of gain used in redirected walking algorithms. It is just above the detection threshold to keep it at half the level of the next gain level.
- **1.49** - double the rate of gain compared to the previous level. 1.49 is above the detection threshold but below the threshold of limited immersion. 1.49 aims to find the impact of detection on user rotation accuracy. It also provides an intermediate step to map any change in accuracy between the 1.245 and 1.98 gain.
- **1.98** - a gain level far above the detection threshold and slightly above the threshold of limited immersion. It considers how user accuracy is impacted by potential breaks in presence.

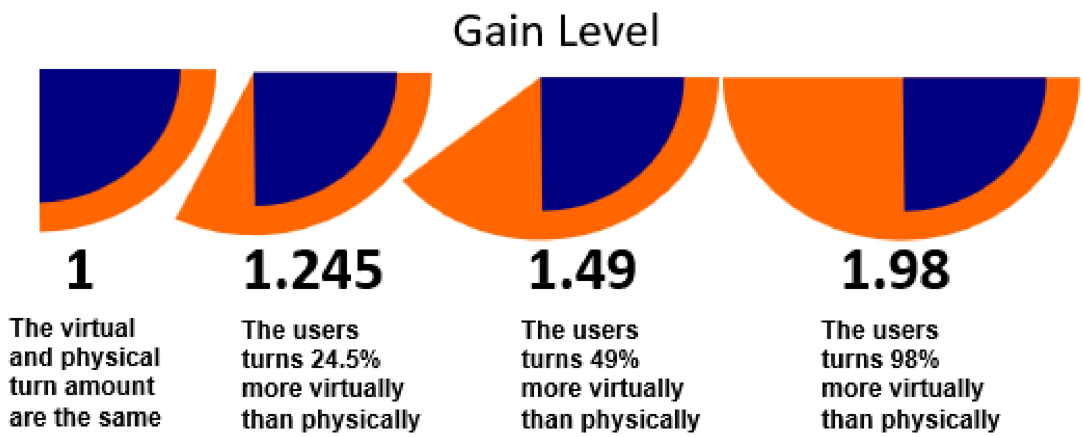


FIGURE 4.1: Gain Levels showing how users turn more in the virtual environment than the physical tracking space at different levels of gain.

TABLE 4.2: Physical Turn Amount Equivalents of Accurate Virtual Turns at Different Gain Levels

Turn Amount	Control 1	Gain 1.245	1.49	1.98
45°	45°	36.1°	30.2°	22.7°
90°	90°	72.3°	60.4°	45.45°
180°	180°	144.6°	120.8°	90.9°

Turn Amount

Three different turn amounts aim to show if greater or smaller turns change the accuracy of the rotation. The three turn amounts were chosen to cover a range of turns from small to large while also being easy to explain to participants. 180° can be easily explained as a half turn to participants, similarly 90° is a quarter turn and 45° is half of a half turn. Table 4.2 below shows the physical turn equivalents of turning 45°, 90° and 180° in the virtual environment under the different gain conditions. Figure 4.1 shows a visual representation of these different gain levels when the user turns 90° in the physical space.

Environment Conditions

Four different environment conditions are presented - two virtual environments using an HMD and two physical conditions without a headset as controls.

The user study system has two environments – a *minimal Environment* and a *complex Environment* as shown in Figure 4.2:

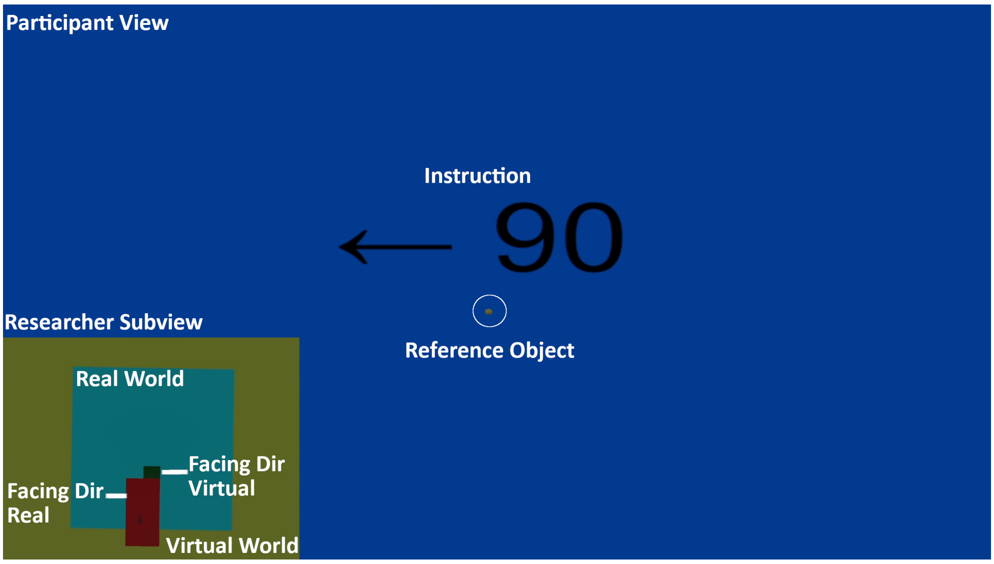
- The *minimal Environment (fig. 1 (a))* consists of a blank blue space with a single yellow ball floating a short distance away from the user, emulating a scenario with minimal visual input in the form of just one visual cue. At least one visual cue is needed for users to recognise gain (Paludan et al., 2016).

- The *complex environment (fig. 1 (b))* is a more visually detailed environment of an office conference room. The user is standing in the middle of the room. Unlike the minimal Environment, this environment offers a high visual density with multiple reference points that can help orient the user. The complex environment provided lots of visual cues and emulates a realistic environment with plenty of visual feedback when turning.

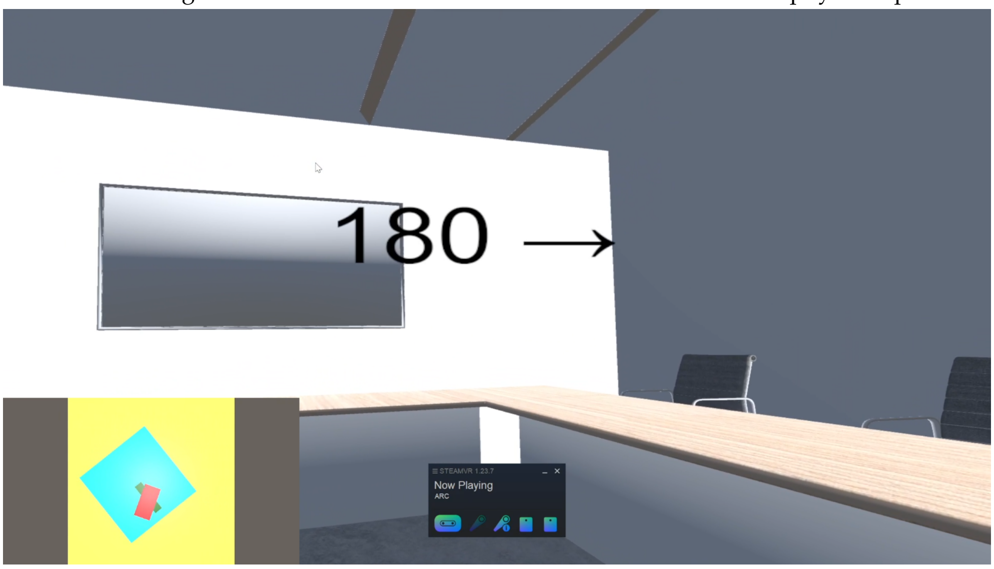
In each virtual environment, the participant is tasked with turning the amount and direction shown on the screen in front of them (like in Figure 4.2). They press the trigger button on the Vive Controller when they start the rotation. This makes the instruction in front of them disappear to allow the participant to turn. They then press the button again when they complete the turn and the next instruction appears.

The two physical conditions ask the participant to turn in the physical space at 45°, 90° and 180° in both Clockwise and Counter-clockwise directions. First, users complete all the turns with their eyes open, and then again with their eyes closed. An audio recording tells participants which way to turn in a random order. Participants hold an HTC Vive controller steady in front of their chest close to their body. Each time they start and complete a turn, they press the trigger button on the controller. This provides a baseline rotation accuracy for the user with the controller rotation being used as a proxy for the amount the user has turned.

It was hypothesised that the accuracy of rotation in the complex environment with a gain of 1 would be similar to turning in the physical space with eyes open. The minimal environment provided a single visual cue for the user and was meant to emulate the way a user turned in the physical space with their eyes closed while still providing enough visual feedback to make the user aware of the gain. However, on completion of the experiment, there was no statistically significant difference between the accuracy of the eyes open and eyes closed control conditions.



(A) *minimal environment* with annotations (white text) showing the *Participant View*, the *Reference Object*, and an *Instruction* for a participant. The *Researcher Subview* (lower left) shows the relative facing direction of the user in the virtual environment and the physical space.



(B) *complex environment* shows an office environment with many possible visual reference points.

FIGURE 4.2: (A) Tracking space aligned with the virtual environment.
(B) Tracking space is no longer aligned with virtual environment.

TABLE 4.3: Environment Presentation Order

# Participants	Condition 1	Condition 2
10	Increasing Complex	Decreasing Minimal
9	Decreasing Complex	Increasing Minimal
10	Increasing Minimal	Decreasing Complex
9	Decreasing Minimal	Increasing Complex

4.3.2 Presentation Order

Gain could be presented in either increasing (from 1 to 1.98) or decreasing (from 1.98 to 1) order within both the minimal and complex virtual environment conditions. Each participant was presented with two of these conditions, one increasing and one decreasing, as shown in Table 4.3. The different presentation orders were chosen to help eliminate any bias caused by habituation. Habituation can lead participants to become less aware of the gain over time as they become used to it.

4.3.3 Equipment

The experimental system was created and run using the Unity Game Engine version 2019.4.22f1 and run on a laptop with the windows 11 pro operating system. The laptop contained an intel i7-11800H 2.30GHz CPU and a NVIDIA GeForce RTX 3070 graphics card with 32GB RAM. The HTC Vive Pro Eye headset[1] was connected to the PC via a USB-C display port connection. The headset provides a stereoscopic view with a resolution of 1440 x 1600 pixels per eye, a refresh rate of 90Hz, and a field of view of 110°. Two base stations were used to create a physical tracking space. Participants were given two HTC Vive controllers to hold.

4.3.4 Implementation

The system was built based on the factors previously specified (see Section 4.3). Of the three systems described in this thesis, it was the easiest to implement as the platform only required rotation gain for the user study. Similar to the system in Chapter 3, the system was built in Unity using the SteamVR plug-in for VR support. The minimal

[1]The system could also be run using an HTC Vive headset, as the system does not use the eyetracking features of the Pro Eye headset.

virtual environment was built using basic assets available in Unity. For the Complex scene, a pre-built virtual office environment was found online[2]. The different turn amounts and directions were scripted in the system, and presented in a random order for each level of gain to remain unpredictable.

In addition to these two environments which required the user to wear a VR headset, an additional virtual environment was created which was empty except for a VR system to track a HTC Vive Controller when run and to output the change in position of the controller to a csv log file. This was used to track how much users turned in the Eyes Open and Eyes Closed conditions in the user study. Voice clips were also recorded by the experiment describing which way a participant should turn and the amount. These were used for the Eyes Open and Eyes Closed conditions, as the participant could not see any virtual visual cues describing which direction they should turn in. Similarly to the virtual cues, the audio cues were played in a random order with a new audio cue being played after the participant had completed the previous turn.

4.3.5 Data Collection

While the user inside the virtual environment only sees what is in the *Participant View*, the researcher can see the additional *Researcher Subview* on their computer screen while observing the user study. The views show the environment (minimal or complex) and user's available physical tracking space as it is currently defined in the laboratory setup. This tracking space is shown in cyan. The facing direction of the user is shown by two strips - the green strip shows the way the user is facing in the virtual environment, and the red strip shows the direction the user is facing in the physical space. Figure 4.2 (a) shows a scenario where the virtual and physical environment are aligned, and Figure 4.2 (b) depicts a situation where rotation gain has been applied and thus the physical and the virtual environments are no longer aligned.

This view is hidden from participants as it might be confusing or introduce bias if they rely on the view to judge their accuracy rather than the virtual environment.

[2]See the demo version of the experimental system on Github for a full list of credits and full code - https://github.com/chionic/VR-Turning-Accuracy-Environment

Researchers can see in real time the relative position of the user in both the virtual space and the physical amount the user has turned. This allows the researcher to intuitively see the effect of the gain as the user rotates. This improves the researchers understanding of the user response to the gain during the user study. The researcher can record their screen while the user study is running for additional visual data on how the user turns for later analysis.

In addition to researcher views during the user study, for each participant log files were created. The log records the position and facing direction of the user in both the physical tracking space and virtual environment. When the participant presses a button to start and complete a turn, this is logged in the system. The amount the user turned in the virtual environment and physical tracking space is calculated based on the number of degrees turned compared to the number of degrees they were asked to rotate to get a percentage accuracy of their rotation. Additionally, how much the user rotates every frame in the environment is measured. This data can then be used to graph how smoothly the user rotates and whether they "overshot" before turning back to complete the rotation.

While both the amount the user turns in the physical tracking space and the virtual environment are recorded, the two numbers are in ratio to one another – if you know the value of one you can calculate the value of the other. To get the virtual rotation from the physical rotation multiply the rotation by the gain factor so that:

$$\theta_V = \theta_R * G_R \tag{4.2}$$

where G_R is the rotational gain, θ_V is the virtual rotation, and θ_R is the physical rotation. The same can be done in reverse if the virtual rotation is known but the physical rotation is not:

$$\theta_R = \theta_V / G_R \tag{4.3}$$

Since the two numbers are related in this way, the calculations have been based off the physical rotations, although similar results could be found were the virtual rotation chosen instead.

In the Eyes Open and Eyes Closed conditions where the user turns without an HMD, the position and facing direction of the controller is tracked instead. Whenever the participant presses a button, a record is made of the amount their facing direction had changed since the last button press.

In addition to the quantitative logging of turns, the awareness the user had of the gain was measured. It was expected that users would be aware of the gain as it was above the detection threshold. However, in the experiments that originally found these thresholds, the participants were not naïve to the gain. Users were asked a series of questions to better understand their experience in the form of a questionnaire after each virtual environment. An example of the questionnaire can be found in Appendix G and the results are summarised in Section 4.4.6.

4.3.6 Data Extraction

As part of the initial data analysis, the amount that participants turned under the various conditions was extracted from the log files. Generally the log files automatically calculate the size of the user turns. However, occasionally the system would miss a turn from the participant, either due to missing button presses or the system not recording the value.

In these cases, first a manual calculation of the change would be attempted. The log file would note the start of a turn when the user pressed the button and also the end of a turn on the next press. Once the place where the turn that was missing was found, the facing direction of the user on the next line of the log file was taken. A second facing direction was taken on the line before the user completed the turn. By finding the difference in facing direction between the first frame and the last frame, the size of the rotation could be calculated. This would slightly underestimate the amount the user turned as the beginning and final frames of the turn were not included.

If this method did not return a value or returned a very small value, the data record would be left blank. Similarly if the turn was incredibly small (less than 10°) the turn was considered a mis-log and removed from the dataset. Out of 2280 total turn observations, 98 were missing leaving 4.3% of data records blank.

4.3.7 Procedure

1. Experiment System is setup.

2. The experiment is explained to the participant and consent forms are signed. Appendix G shows a copy of the Information sheet and Consent Form.

3. The participant puts on the HMD and familiarises themselves with the controls.

4. The first environment starts with instructions shown to the participant as seen in Figure 4.2. The environment presented first can be any of the four conditions described in Table 4.3.

    (a) Participant completes 3 practice turns.

    (b) Participant completes 6 recorded turns at the initial gain level presented in a random order (45, 90,180 in both CW and CCW directions) to keep the next turn amount unpredictable.

    (c) The next gain level starts and the user repeats the turns in the previous step.

5. Once all the gain levels are complete, the participant removes the HMD and fills out a questionnaire. There are seven questions but only two are relevant (3 and 5) to the user study as they relate to gain perception. The others are decoy questions, similar to other user studies (Nie, Adhanom, and Rosenberg, 2023; Peck, Fuchs, and Whitton, 2009; Suma et al., 2011). Appendix G shows an example of the questionnaire.

6. The participant puts on the HMD and the second environment begins, the second environment in the same row as the first in Table 4.3. The participant repeats steps 4 and 5 in this second environment.

7. The participant turns in the physical space with their **eyes open** without the HMD (45, 90, 180 in both CW and CCW directions).

8. The participant turns in the physical space with their **eyes closed** without the HMD (45, 90, 180 in both CW and CCW directions).

Participants completed a second task after this turning accuracy task in the same experiment session. A detailed breakdown of the second task can be found in Chapter 3.3.2. The turning accuracy task was finished first for all participants for consistency and thus all parts of the turning accuracy task were completed before the second task was introduced to participants.

4.3.8 Participants

After ethical approval was received from the Maynooth University Ethics board, the study was advertised on campus. The user study was conducted with 38 participants[3], 20 Male, 16 Female, 1 Non-binary and 1 preferred not to identify, with an age range of 18-54 (mean age 24). 34% of participants had no VR experience, another 34% had a little VR experience, 20% had some VR experience and 11% had extensive VR experience.

4.4 Results

This section describes the results generated from the dataset collected from the user study. The aim of this study is to aid developers in predicting user behaviour in response to rotation gain. For each of the research questions the data analysis is presented. After the data is analysed, a model of the different types of response to rotation gain is introduced.

4.4.1 Comparison of CW and CCW Rotation

A comparison of the clockwise (CW) and counter-clockwise (CCW) turns of the participants, under each condition was run. This was done using the Kolmogorov-Smirnov method. This method compared the distribution of the CW turns to CCW turns. While the method is often used to compare a distribution to the normal distribution, in this case it was used to compare the distributions of the CW and CCW conditions.

[3]Data from one participant was removed due to an early cancellation for motion sickness. Motion sickness has been covered as an unlikely but possible side-effect in our university-approved ethics procedure

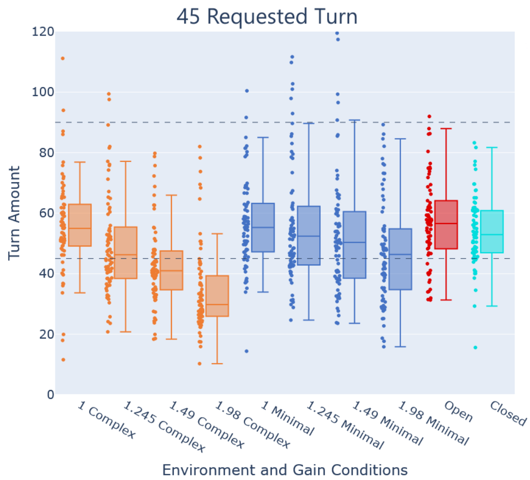
Table 4.4 shows the analysis of the data. The condition column lists turns of the same size and in the same environment, CW and CCW. The statistic column describes the test statistic D and the *p-value* for each gain level. D represents the maximum vertical distance between the two distribution functions of the two samples. The *p-value* measures how likely it is that the two distributions being tested are the same. A *p-value* of 0.05 or below is considered a significant result which means the two distributions are different. No *p-value* in Table 4.4 is below 0.05. The lowest value, 0.197, is for a minimal turn of 45°.

TABLE 4.4: Kolmogorov-Smirnov test results comparing CW and CCW experiment conditions

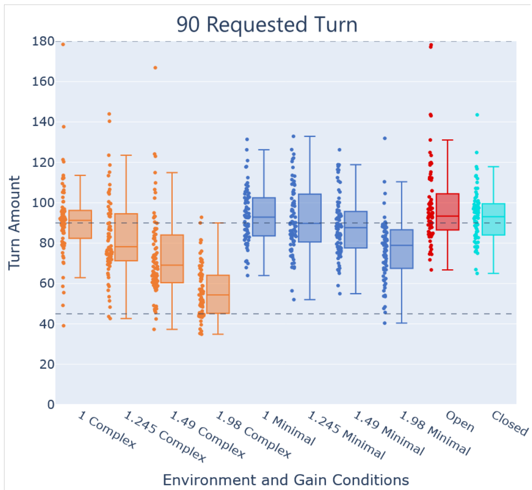
Condition	Statistic	Gain			
		1	1.245	1.49	1.98
45 Minimal	D	0.242	0.132	0.184	0.142
	p-value	0.197	0.903	0.545	0.793
90 Minimal	D	0.114	0.081	0.237	0.158
	p-value	0.979	0.999	0.239	0.738
180 Minimal	D	0.216	0.105	0.184	0.237
	p-value	0.357	0.987	0.545	0.239
45 Complex	D	0.1313	0.119	0.177	0.132
	p-value	0.844	0.907	0.534	0.903
90 Complex	D	0.173	0.116	0.105	0.168
	p-value	0.572	0.935	0.987	0.599
180 Complex	D	0.176	0.072	0.158	0.158
	p-value	0.525	0.999	0.738	0.738

No statistically significant differences were found (see Table 4.4 for details). This means that the direction of the turn made no difference to the turning accuracy of the participants. This result agrees with similar studies on rotation gain (Nguyen and Kunz, 2018; Williams and Peck, 2019; Brument et al., 2021). Additionally, turning repeatedly in one direction leads to dizziness which can be reduced by changing the direction of the turn frequently.

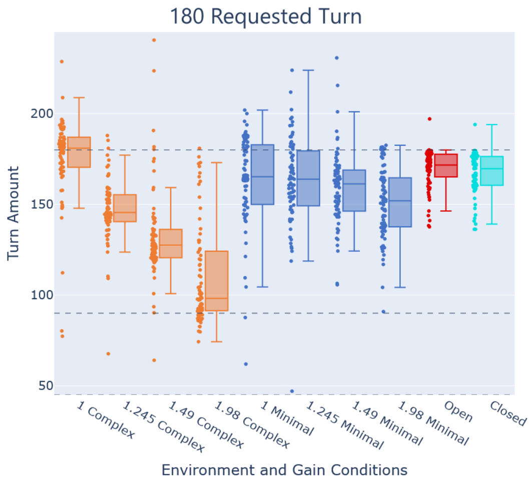
Since there was no statistically significance difference between the CW and CCW conditions, they were pooled in later analysis as their distributions were similar.



(A) A box plot of how much users turned under the different conditions when requested to turn 45°.



(B) A box plot of how much users turned under the different conditions when requested to turn 90°.



(C) A box plot of how much users turned under the different conditions when requested to turn 180°.

FIGURE 4.3: Box Plots of Actual Turns - A dot shows a participant's actual turn response in the physical space. The box plots show the average of the responses. The complex environment (orange), the minimal environment (blue), eyes open (red) and eyes closed (cyan). Each plot has a different y-axis scale, based on the requested turn.

4.4.2 Box-plot Analysis of User Response

The mean turn amount and standard error in terms of degrees under each of the different conditions was calculated (see Appendix D). The three Figures 4.3a, 4.3b and 4.3c capture all the data that was collected. They provide an overview of the response. Since the requested turns differ, the data has been separated into three figures based on each requested turn. The scale of the y-axis is different in each graph. Three horizontal lines at the 180°, 90° and 45° have been added to highlight the relative scales of the graph.

Each individual turn in the real tracking space is represented by a singular dot in the Figures 4.3a, 4.3b and 4.3c. The box plots show the median amount participants turned. The X-axis shows the environment and gain conditions of the turn. The Y-axis represents the number of degrees turned. Each environment has a different colour. Blue for the minimal virtual environment, orange for the complex virtual environment, red for the eyes open condition and cyan for the eyes closed condition. Note that each of the box-plots has a different Y-axis scale based on the how much participants were requested to turn.

In general, as the gain increased, participants turned more than expected in the virtual environment, turning larger amounts virtually than the requested turn amount. Participants turned more in the minimal environment than the complex environment at higher gain levels suggesting that increased visual cues lead gain to be more effective. For example, in Figure 4.3c, in the 180° condition at 1.98 gain users turned a median of about 152° (mean: 151°) in the minimal environment which is close to the 180° that would be expected if users had no response to gain. In contrast, in the complex Environment there was a median of 98° (mean: 110°) suggesting users responded to the gain. This is analysed further in the next section.

4.4.3 Results of User Response to Gain in different conditions

The percentage difference between the expected rotation and the actual rotation that users turned is shown in Table 4.5. For Table 4.5 a 0% value would mean the user responded to the gain expected. A negative value means the user turned less than expected. A positive value means the user turned more than expected. This is based on the results described in Appendix D. Formula 4.4 was used to evaluate the percentage difference:

$$\frac{ActualTurn - (RequestedTurn/Gain)}{RequestedTurn/Gain} \times 100 \tag{4.4}$$

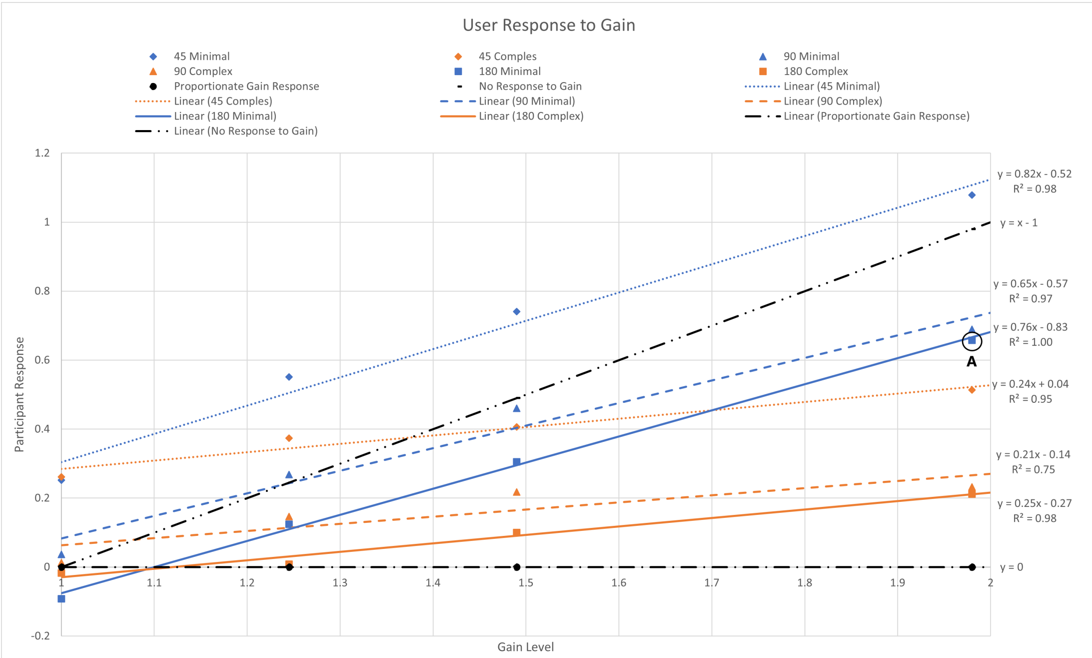


FIGURE 4.4: Comparing Gain Set to Real Turn Response - The graph compares how responsive users were to gain ("Proportionate Gain Response") to the actual amount users turned relative to the expected gain (see equation). The lines show the general trend of the user response based on the data points indicated in the graph.

Figure 4.4 compares the expected rotation in the physical space under the different gain conditions to the actual rotation. If the participants rotated accurately in the virtual environment, then the line would have a slope of 0. This is represented by the 'Proportionate Response to Gain' line on the graph. In contrast, if the gain had no impact on participants, their responses would align with the 'No Response to Gain' line. For example, looking at point A on the graph, which represents the minimal 180 condition at 1.98 gain, it would be expected for users to turn 90.9° in the physical space if they responded perfectly to the gain. The actual mean amount participants' turned was 150.69° (see Appendix D and Figure 4.3c), the participants overshot how much they were expected to turn by 65.8% (see Table 4.5 and Figure 4.4). In this example, the users turned more than instructed in the virtual environment.

Influence of Gain on Turning Accuracy

This section discusses the influence of gain on user turning accuracy in relation to Research Question 1.2 described further in section 4.1. Figure 4.4 compares the expected rotation to the actual rotation. In general, users 'overshot' how much they turned in the virtual environment at higher gain levels. When no gain was present, the users actual rotation was close to the expected rotation in the complex virtual environment. This was seen with both the quarter-turn of 90° (1.3%) and the half-turn of 180° (-1.8%). In contrast at the highest gain level of 1.98, users turned about 218° (21.2%) when asked to turn 180° and 110° (23.2%) when asked to turn 90°. This is an overshoot of about 20% at the 1.98 gain level.

A similar trend can be seen in the 45° condition where there is a 25.3% difference between no gain (1.0, 26.1%) and maximum gain (1.98, 51.4%). The larger error in the 45° could be due to participants finding it more difficult to judge how far to turn. Unlike the easier reference points of a half-turn for 180° and a quarter-turn for the 90° conditions. Appendix D shows the mean average amount users turned under each condition, while Table 4.5 shows how users started to overshoot more as gain increased. This is also illustrated in Figure 4.4 where the distance from the actual turn to the 'Proportionate Gain Response' increases with the gain level. These results show that gain impacted the accuracy of turning for users. The results show a linear correlation between the increase of gain and the average deviation.

Influence of Environment Complexity on Turning Accuracy

This section discusses how the number of visual cues in the environment affects user turning accuracy (see research question 1.3 in section 4.1). In the user study, participants were presented with two different virtual environments. The complex environment had numerous visual cues and was modelled after an office. The minimal

TABLE 4.5: Percentage difference of actual turn in the tracking space compared to the expected turn accounting for gain.

Condition		**Real**			
Environment	Turn	Gain			
		1	1.245	1.49	1.98
Minimal	180	-9.1%	12.4%	30.5%	65.8%
Complex	180	-1.8%	0.9%	10%	21.2%
Minimal	90	3.6%	26.9%	46.1%	69%
Complex	90	1.3%	14.7%	21.7%	23.2%
Minimal	45	23.5%	55.1%	74.1%	108%
Complex	45	26.1%	37.4%	40.7%	51.4%

environment contained only a single visual cue in the form of a yellow ball (see Figure 4.2).

Gain influenced participants more strongly in the complex environment (orange lines) than the minimal environment (blue lines), as shown in Figure 4.4. For the complex environment the slopes of the lines in the graph are around 0.22. This is closer to the ideal response of a slope of 0 than in the minimal environment. In the minimal environment the slopes range from 0.65 to 0.83. This suggests users were impacted less by the gain in the minimal environment than in the complex environment. This suggests, the weight of different sensory modalities might be influenced by the number of visual cues in the environment. This is discussed further in Section 4.4.8.

Influence of Requested Turn Amount on Turning Accuracy

In this section the requested turn (45°, 90°, 180°) is compared to the actual turning accuracy of the participants (see research question 1.4 in section 4.1). When no gain was present, participants turned quite accurately when requested to make a 90° quarter-turn (complex: -1.3%, minimal: -3.6%) and 180° half-turn (complex: 1.8%, minimal: 9.1%) (see Table 4.5).

In contrast, in all the environment conditions with no gain, the mean amount users turned was closer to 55° than 45°. This was similar across all four environments - in the Eyes Open (mean=56.82, standard error=1.65), Eyes Closed (mean=53.7, standard error=1.34), minimal (mean= 56.34, standard error=1.71) and complex (mean=56.76, standard error=2.14) environments (see Appendix D). This trend can also be seen in

Figure 4.3a. Similarly, participants turned more in Figure 4.4 when asked to turn 45° with no gain. In both the minimal and complex environments, users turned at a rate significantly above the 0 line. This suggests they turned too much. In the minimal environment, users turned 23.5% more and in the complex environment they turned 26.1% more (see Table 4.5).

This could be due to users being unused to turning 45° as compared to a 180° half-turn or a 90° quarter-turn. Alternatively, users might find accurately judging smaller turns more difficult. Further studies with other requested turn amounts (such as 60°, 120° or 15°) are needed to fully understand this result.

4.4.4 Statistical Tests

Observing the trends in the boxplot Figures 4.3a, 4.3b, 4.3c and relative turning accuracy Figure 4.4 suggests that different environment, requested turn, and gain conditions affect turning accuracy. This is discussed in Sections 4.4.2 and 4.4.3. Based on the trends found, statistical tests were run to find any statistically significant differences between the conditions.

The normality for each condition, with pooled CW and CCW turns (see section 4.4.1), was tested using the Shapiro-Wilk normality test (see Table 4.6). The W test statistic in the Shapiro-Wilk normality test measures how close the condition is to a normal distribution. A W value close to 1 (e.g. 0.97) suggests the data is approximately normally distributed. In contrast, a lower W value (e.g. 0.613) suggests the dataset is far from a normal distribution. The *p-value* describes how confident the test is that the set is normally distributed. A *p-value* less than 0.05 suggests the data is *not* normally distributed.

A few of the subsets in Table 4.6 were normally distributed. Such as in the minimal environment when requested to turn 90°. However, most subsets were not normally distributed. Non-parametric tests were selected to account for the data not following a normal distribution. Additionally, since each participant rotated multiple times in each environment condition, the data was not independent. The non-parametric

Friedman test of differences among repeated measures was chosen to account for this.

TABLE 4.6: Shapiro-Wilk Normality test results for each of the virtual environment, requested turn amount and gain conditions. A *p-value* less than 0.05 suggests the data is not normal.

Condition	Statistic	Gain			
		1	1.245	1.49	1.98
45 Minimal	W	0.97	0.9	0.901	0.962
	p-value	0.091	0.00002	0.00002	0.027
90 Minimal	W	0.986	0.986	0.974	0.961
	p-value	0.613	0.615	0.116	0.019
180 Minimal	W	0.896	0.933	0.887	0.974
	p-value	0.00002	0.0006	0.000006	0.129
45 Complex	W	0.881	0.933	0.931	0.85
	p-value	0.000005	0.0006	0.0005	0.0000003
90 Complex	W	0.890	0.965	0.901	0.954
	p-value	0.00001	0.038	0.00002	0.008
180 Complex	W	0.802	0.805	0.816	0.843
	p-value	0.00000001	0.00000002	0.00000002	0.0000002

The Friedman test was run to determine if there were statistically significant differences in the average physical rotation in any of the four different environments. The mean of the combined CW and CCW turns under each condition was used in the Friedman test. Table 4.7 shows the results of the Friedman test. The turn amount column describes how much participants were requested to turn. The gain columns describe the gain level present in the two virtual environments. Each gain column includes the Q test statistic for the Friedman test. The significance level of the result is described by the number of * beside the Q statistic. One or more * suggests that there was a statistically significant difference between at least two of the four environments (minimal, complex, eyes open and eyes closed) under the conditions of that row and column.

Statistically significant differences were found at all gain levels above 1. If the Friedman test found a statistically significant difference, pairwise Wilcoxon tests were run using the Bonferroni adjustment method for multiple comparisons. The complete results of the Wilcox tests can be found in Appendix F. A summary of the statistically significant. results of the Wilcoxon tests are shown in Table 4.8.

TABLE 4.7: Friedman Rank Sum Test Comparing Environments - The test has 3 degrees of freedom. The * indicate significance levels. * < .05, ** < .01, *** < .001

Turn Amount	Gain 1	Gain 1.245	Gain 1.49	Gain 1.98
45 χ^2	1.87	13.87**	24.74***	43.66***
90 χ^2	6.87	17.63***	33.1***	76.43***
180 χ^2	11.02*	35.26***	53.78***	75.31***

TABLE 4.8: Significant differences found using Wilcoxon tests between the four environment conditions - minimal and complex while wearing a headset, and eyes open and eyes closed when the user was turning without a headset.

Turn Amount	Gain Level	Environment	Minimal	Complex	Open	Closed
45	1.245	Complex	-	X	*	*
	1.49	Complex	*	X	***	***
	1.98	Complex	***	X	***	***
	1.98	Minimal	X	***	*	*
90	1.245	Complex	-	X	**	**
	1.49	Complex	***	X	***	***
	1.98	Complex	***	X	***	***
	1.49	Minimal	X	***	*	-
	1.98	Minimal	X	***	***	***
180	1	Complex	*	X	**	**
	1.245	Minimal	X	**	-	-
	1.49	Minimal	X	***	**	*
	1.98	Minimal	X	***	***	***

Comparing rotation in the virtual environments to the physical space

The section contains an analysis of the turning accuracy of participants in the virtual environments compared to the physical space concerning research question 3 (see section 4.1). Figures 4.3a, 4.3b and 4.3c show the general trends of the minimal and complex virtual environment conditions compared to each other. The figures also compare the virtual environments to the eyes open and eyes closed conditions, where the user was not wearing an HMD. In general, as the gain increases the difference between the amount participants turned in the virtual environments compared to the eyes open and eyes closed condition increases. The complex environment holds a larger rate of change than the minimal environment at the same gain levels. Tables 4.7 and 4.8 show the statistically significant differences between the four conditions. The test results show a similar trend to Figures 4.3a, 4.3b and 4.3c.

No statistically significant differences were found between the eyes open and eyes closed conditions for any of the requested turns. This suggests that users had roughly equal accuracy when turning with their eyes open and eyes closed when not wearing a VR headset. This could be due to the relatively small turns requested of participants (45°, 90° and 180°). Due to this similarity, statistically significant differences were found between the virtual environments and the eyes open condition at similar rates to the eyes closed condition.

The 45° condition only showed a difference between the minimal environment and the eyes open and eyes closed condition at the highest gain level of 1.98. At 90° and 180° this difference was statistically significant at both the 1.49 and 1.98 gain levels. This suggests that rotation gain only influences users in an environment with minimal visual cues when the gain is well above the level of the detection threshold. If the user is asked to turn a larger amount, the gain is more likely to impact them.

In contrast, with the complex environment all gain levels above 1 were statistically significant when compared to the eyes open and eyes closed conditions, in both the 45° and 90°. This suggests that in a virtual environment with lots of visual cues, gain causes users to turn less much earlier than in a condition with only a singular visual cue. The 180° turn shows a similar trend in relation to Figure 4.3c. However, due to there being a statistically significant difference between the eyes open and eyes closed condition in comparison to the complex environment condition when no gain was added, the results of the Friedman and Wilcoxon tests are difficult to interpret for this scenario.

To summarise, at higher gain levels there was a significant difference in how participants turned in the two virtual environments compared to the Eyes Open and Eyes Closed condition. This trend was stronger in the complex environment with lots of visual cues than in the minimal environment with a singular visual cue. Additionally, when asked to turn 180° in the complex environment without gain, participants turned a statistically significant different amount compared to the eyes open and eyes closed condition. The implications of these results are discussed further in Section 4.4.8.

4.4.5 Linear Mixed Models

Interaction effects show how different variables influence each other. Sometimes, the effect of one variable on its own might not be significant. However, a second variable might influence the value of the first variable. Thus when combined with other variables there might be a statistically significant effect. In relation to research question 2 (see section 4.1) this section analyses the impact of interaction effects on turning accuracy.

In this user study, each participant provided multiple data points as they turned at different gain levels, turn amounts, directions and virtual environments. Thus the data was not independent. Therefore, the pre-conditions for an ANOVA were not met. Instead, Linear Mixed Models (LMMs) were chosen as they do not assume independence of the data. The random effects in the LMM account for the non-independence of the dataset. LMMs were used to analyse potential interaction effects between the different variables (turn direction, requested turn, gain level and environment).

LMMs are ideal for handling correlated data (such as data collected from the same person under different conditions) and hierarchical data (such as when there are multiple effects both within a group and between groups being studied) (Gałecki and

Burzykowski, 2013). LMMs model fixed effects, which come from variables implemented within the study, such as the gain level and requested turn. These variables predict the outcome of the model. The model helps us understand the average effect of these fixed variables on the overall turning accuracy of users. Additionally, LMMs use random effects to model variations between groups that are not accounted for by the fixed effects.

The LMM considered four independent variables as shown in equation 4.5. ε is often used in LMMs to describe the residual errors not accounted for directly by the model, that is variation that the other variables do no account for.

$$\begin{aligned} D\_Turn = R\_Turn + Gain + Env + Direction + (R\_Turn * Gain) + (R\_Turn * Env) \\ +(Gain * Env) + (R\_Turn * Gain * Env) + (Participant) + \varepsilon \end{aligned} \tag{4.5}$$

Where:

- D_Turn : difference between the actual amount the participant turned and the requested turn (actual turn – requested turn)
- R_Turn : the requested turn as a factor
- Gain : the level of gain added
- Env : the environment the participant was in
- Direction : which direction the participant was asked to turn (clockwise/counter-clockwise)
- Participant : the participant's ID as a factor
- ε : residual errors - variation in the data not accounted for by the other variables.

Participant and ε were regarded as random effects.

The model did not have any four way interaction effects. Next, smaller models were run to find any three- and two- way interaction effects. This was done by comparing the accuracy of the smaller models to the larger model. The results of these models are detailed in Appendix E.

There was a three-way interaction effect between the environment, the gain level and the requested turn amount. Two-way interaction effects were found between the requested turn and the gain, the requested turn and the environment and the gain and the environment. Appendix E shows the significance of the different interaction effects.

To get a better understanding of the different two way interaction effects, one factor of the three way interaction effect was held constant with each of the possible values for that factor. The model was then run again to see if the two way interaction effect under that condition was statistically significant. It was found that the requested turn and gain level interaction was only statistically significant in the complex environment while all other two-way interactions were always statistically significant.

Influence of Interaction Effects on turning accuracy

In relation to research question 2 (see Section 4.1), the interaction effects between the different variables was analysed using LMMs. The direction participants turned (CW or CCW) did not have an effect or any interaction effects within the model. This is similar to the test results from the Friedman and Wilcoxon tests.

The gain level, the environment, and the requested turn amount had a three way interaction effect. This suggests that all three factors influence the turning accuracy of users when combined. Two-way interaction effects were found between the requested turn and the gain, the requested turn and the environment, and the gain and the environment. That is, all the combinations of two out of three variables in the three way interaction effects also had significant effect on each other. Additionally, it was found that the requested turn and gain level interaction was only statistically significant in the complex environment. All other two-way interactions were statistically significant. This supports the idea that gain has a stronger influence on the turning accuracy of users in the complex environment than in the minimal environment at the same requested turn amount.

The number of interaction effects found in the model suggests that the turning accuracy of users is impacted by multiple different variables and how those variables interact with each other. Developers should consider their system as a whole when predicting how much users will turn as different aspects of the system are likely to influence each other.

4.4.6 Questionnaire Responses

As well as the turning accuracy of participants, the user study also analysed participants' awareness of the gain in the virtual environments and their experience of turning. When users are unaware of rotation gain it can improve task performance (Bruder, Lubos, and Steinicke, 2015; Mostajeran et al., 2024). However the level of gain that can be added to a system before users become aware of it is limited.

A questionnaire was conducted after each virtual environment condition (see Appendix G). Participants were asked two questions relating to the gain level ("The world is moving?" and "You're turning more/less than you thought?") in the questionnaire. These two questions indicate whether a participant was aware of the gain inside the virtual environment. A positive response to at least one of the two questions suggests the participant noticed the gain inside the environment.

Participants were less likely to become aware of the added gain in the minimal Environment if it was presented to them first. When presented with the minimal environment first 42% of participants responded yes to one or both of the questions. However, 68% of participants responded yes to in the minimal environment when it was presented second. In contrast, with the complex environment, similar numbers (68% vs 63%) of participants responded 'yes' to one or both of the questions regardless of if it was presented first or second. This supports the idea that fewer visual cues make gain less perceptible to the user. This effect might be mitigated by previous experience with gain, like in the complex environment.

The majority of participants recognised the gain in either one (37%) or both (42%) environments. However, a subset of participants (21%) did not respond 'yes' in either of the environments. This suggests that these participants remained unaware of the gain. This is surprising since the higher levels of gain were well above the previously established detection threshold. It was expected that all participants would become aware of the gain.

In terms of the user experience of gain, five participants (13%) made explicit comments about the gain. One user described how they "spun faster virtually for what felt like the first 2-3 turns and then normalised" after experiencing the office descending condition. Another participant explained after the office environment that "the world spun at a faster rate than I did". Similarly, a participant noted that "I turned faster in VR than in real life". One participant described the room as "spinning" and another described it as "tilting". This provides anecdotal evidence that some users perceive their self-motion as changing in the virtual environment while others perceive the virtual environment as changing around them.

Additionally, some participants commented on their experience of turning in the virtual environments. Three participants (8%) mentioned they felt like they had turned inaccurately in the virtual environment, and one participant mentioned how they lost track of where they were in the physical world. One participant thought the researcher had moved 90° in the room when they had remained in the same spot throughout.

User experience of Rotation Gain

This section discusses the questionnaire responses of participants in relation to their awareness of the rotation gain, as described by Research Question 4 (see Section 4.1). When asked about the gain, participants who started in the minimal environment made less comments about the gain. This suggests that gain was less perceptible to participants with only a single visual cue within the environment. However, gain was more perceptible in the minimal environment after the participant had already been in the complex environment. This suggests that the participants awareness of gain in the complex environment made it easier for them to recognise the gain in the minimal environment despite the lack of visual cues.

Not all participants were aware of the gain. Based on the questionnaire responses, 21% remained unaware of the gain across both environment conditions. This is surprising as the gain presented was well above the detection threshold at the levels of

1.49 and 1.98. This could be explained by the participants remaining naïve to the goals of the user study and therefore not actively looking for changes in the virtual environment or the way they were moving. This is in contrast to previous studies where users were made aware of the gain at the beginning of the study. Alternatively, participants may have become aware of the gain but did not perceive it as relating to the questions being asked. Additional user studies are needed to draw stronger conclusions.

4.4.7 Modelling User Response

After analysing the data in consideration of the research questions (see Section 4.1), how to model and predict users' response was considered. First, a graph was created for each individual participant comparing their response to each gain level in the minimal and complex environment. In this graph, the amount the participant turned in the real world was compared to the requested turn (45°, 90° and 180°).

$$GainResponseLevel = Abs(RequestedTurn)/Abs(ActualTurn) \tag{4.6}$$

The result of the equation shows the rate at which the participant responded to the gain (GainResponseLevel) - if they turned less than requested, the participant was likely influenced by the gain. If they turned the requested amount, they did not respond to any gain. For example, if a participant was asked to turn 180° and they actually turned 93° then the formula would be:

$$180/93 = 1.94$$

In this case the participant turned significantly less than 180° and the amount they actually turned suggests they were responding to a gain level of 1.94. If their *ActualTurn* had been 180° then they would have responded at a gain level of 1 (no added gain). The gain response level of each turn of a participant was calculated using Equation 4.6. The turns were then split by the actual gain level that was applied to the participant - 1, 1.245, 1.49 and 1.98. For example, in the calculation above, with the actual gain level of 1.98 and a requested turn of 180°, the participant responded to the

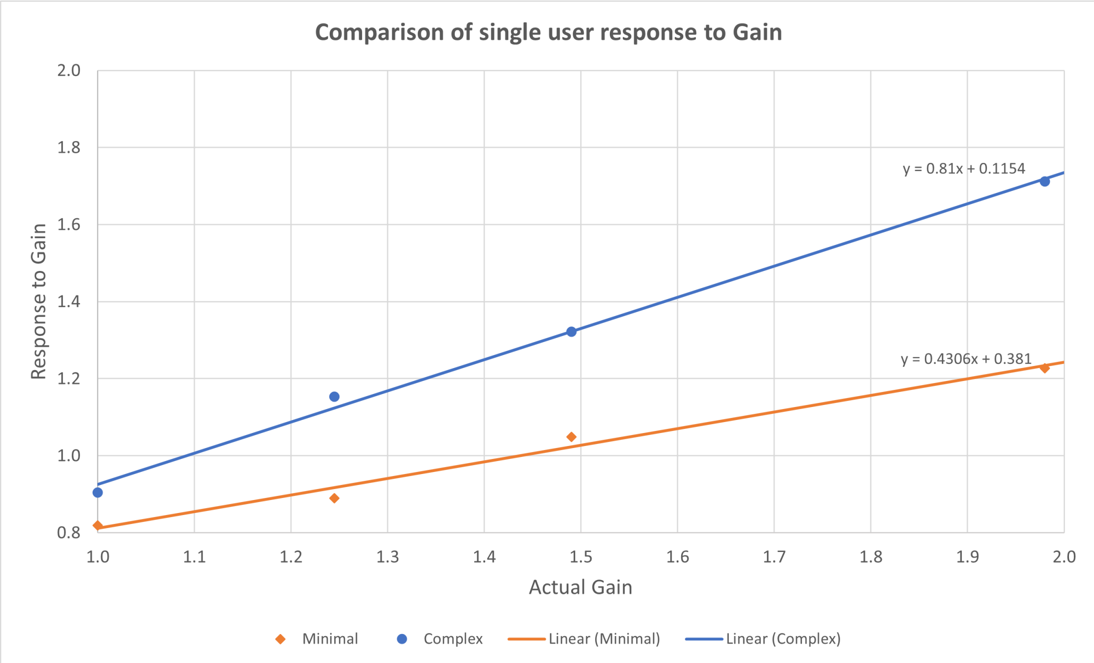


FIGURE 4.5: Comparison of a single participant's response in the minimal (orange) and complex (blue) environments to changes in gain. The x-axis shows the level of gain added to the virtual environment. The y-axis shows the level of response the participant had to the gain.

gain at a rate of 1.94. This is close to the 1.98 gain level, suggesting that the participant was strongly influenced by the gain for this turn. For each level of gain, which included six individual turns, the score was averaged to find a final score for the participant response at that gain level. This was done once for the complex environment and once for the minimal environment.

Each point was graphed and a trend line using the four points was drawn to estimate how the participant would respond to different gain levels between 1 (no added gain) and 2 (the participant is turning twice as fast virtually as in the tracking space). See Figure 4.5 for an example graph from one particular participant. The graph shows the average accuracy of the participant turn at every level of gain. The slope of the line in the graph shows the overall accuracy of the response to the gain of the participant for each environment. If the participant responded to the gain at the exact rate of the actual gain introduced the line would have a slope of 1. If they were not affected by the gain at all, the line would have a slope of 0. In the example Figure 4.5 the participant had a response of 0.81 to the gain in the complex environment and a response of

0.43 to the gain in the minimal environment. This suggests they were highly impacted by the gain in the complex environment and less so in the minimal environment.

The graphs for each participant were compared to find any trends in user behaviour. Trends specific to certain requested turn amounts among the participants were also considered. For example, when requested to turn 45° participants often turned more than 45°. The mean of all participant responses for each condition was taken, excluding missing data records. CW and CCW were merged as there was not a statistically significant difference between the measurements (see section 4.4.1). Appendix D tabulates the mean of participants' turns as well as the standard error for each condition.

4.4.8 K-means

In addition, K-means clustering was used to identify possible groups of user behaviour (see Figure 4.7). Two dimensions were used based on earlier analysis to identify behaviour patterns of participants across the different conditions[4]. One dimension was the gain response slope of the complex environment and the other dimension was the gain response slope of the minimal environment. Each dot in Figure 4.7 is a single participant's response. The numbers in Figure 4.7 refer to the order in which participants were put into the plot.

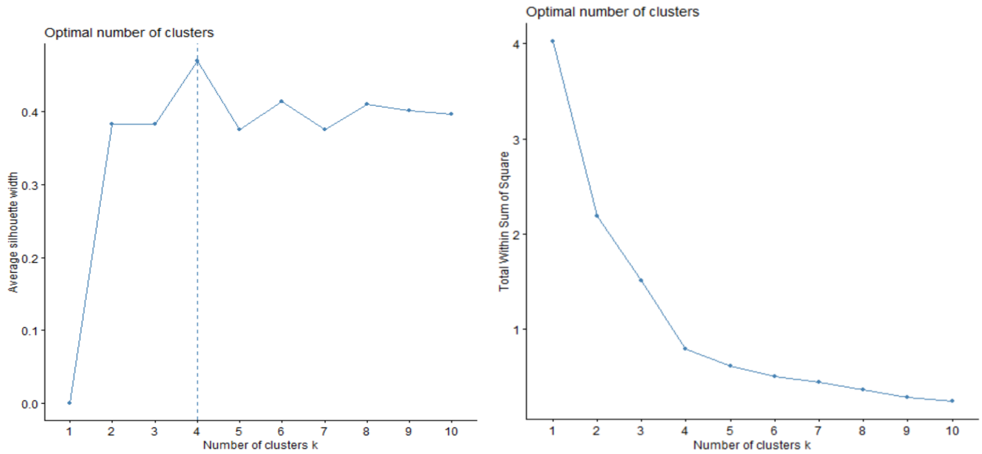
The silhouette and elbow method were used to find the optimal number of clusters for the data. Both methods confirmed that 4 was the ideal number of clusters as show in Figure 4.6a and 4.6b.

For each group cluster in the k-means graph (Figure 4.7), a new graph was created. The graph shows the average response of the participants in that group to gain. These graphs show four different profiles to explain participant behaviour (see Figure 4.8).

Low Response Group (6 Participants, 16%)

The participants in this group showed little or no response to gain under both environmental conditions. This group turned similarly under all gain conditions and both

[4]One participant was an extreme outlier with a cluster that formed with only their response. This participant's data was removed from the K-means dataset since it was so far outside the other results.



(A) Silhouette method identifying the optimal number of clusters (B) Elbow method identifying the optimal number of clusters

FIGURE 4.6: Finding the Optimal Number of Clusters for K-means

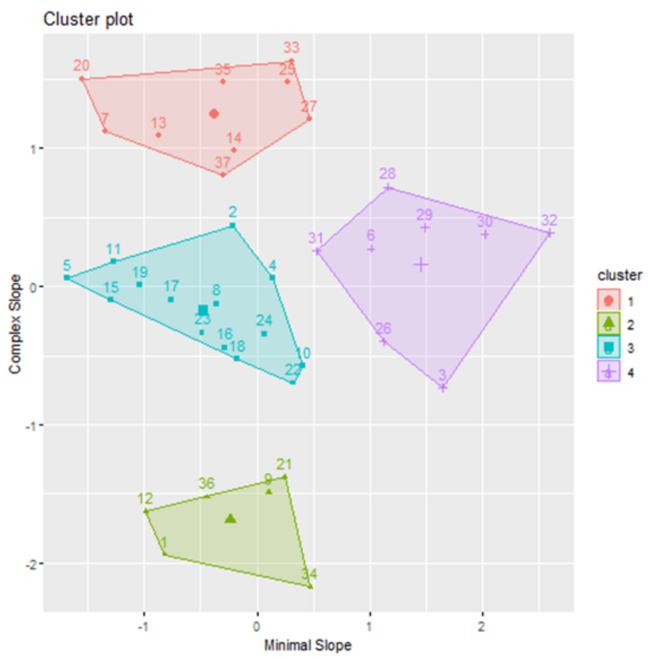
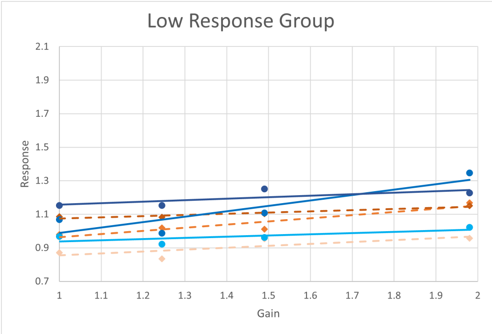
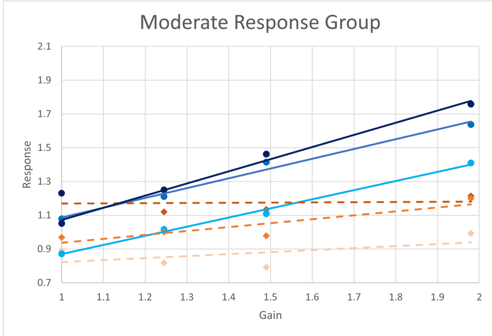


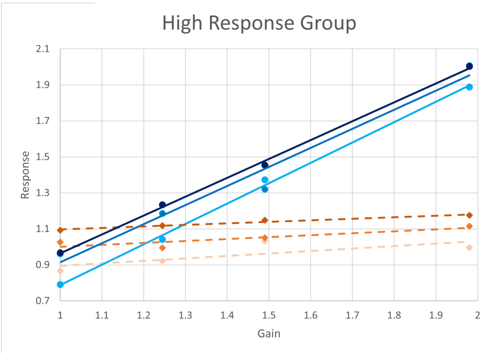
FIGURE 4.7: K-Means Clustering - four clusters were identified. The Low Response Group with 6 participants (16%, green), the Moderate Response Group with 14 participants (37%, blue), the High Response Group with 9 participants (24%, red) and the All Response Group with 8 participants (21%, purple).



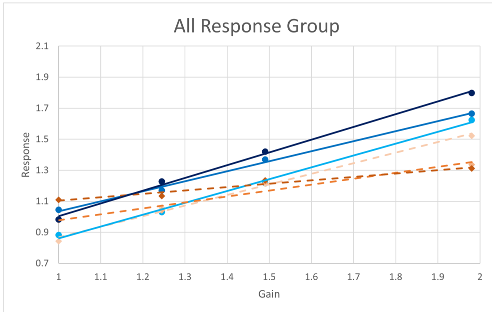
(A) The Low Response Group showed a similar low response to gain in both the minimal environment (orange diamonds) and complex environment (blue circles)



(B) Moderate Response Group showed a moderate response to gain in complex environment (blue circles), while showing little response to gain in the minimal environment (orange diamonds)



(C) The High Response Group showed a high response to gain in complex environment (blue circles), while showing little response to gain in the minimal environment (orange diamonds)



(D) The All Response Group showed a moderate response to gain in both the complex environment (blue circles) and in the minimal environment (orange diamonds)

FIGURE 4.8: User Response Groups to Gain in the Minimal and Complex Environment

environments. They treated both the complex and minimal scenario as if no gain was present in either. The slopes of the complex and minimal environment lines were similar. The graph (Figure 4.8a) shows the mean slope of the six participants in this group when asked to turn 45°, 90° and 180° in the minimal (orange dashed) and complex (blue solid) environments.

In the 45° and 180° complex condition, the slope of the response was 0.07 and 0.09 respectively. The 90° complex condition was the only line that showed any response with a slope of 0.32. The minimal environment slopes were 0.11 for 45°, 0.19 for 90° and 0.07 for 180°. Similarly to the 90° turn in the complex environment, the 90° minimal environment had a slightly bigger slope than the other two conditions.

Moderate Response Group (14 participants, 37%)

The second group was moderately affected by gain in the complex environment while remaining largely unaffected by gain in the minimal environment. This was the largest group, containing 14 participants. Despite being affected by gain in the complex environment, their response was still less than the level of gain added. This means they turned more in the virtual environment than they were requested to at higher gain levels.

In the complex environment, participants were more affected by the gains as the amount they were asked to turn increased. The slope of the 45° complex condition is 0.54, for the 90° requested turn it is 0.58 and for the 180° requested turn the line has a slope of 0.72. In the minimal environment, when requested to turn 45° the slope of the response was 0.12. For 90° it was 0.23 and for 180 it was 0.01.

High Response Group (9 participants, 24%)

Similar to the Moderate Response Group, the participants of the High Response Group responded strongly to the gain in the complex environment (see Figure 4.8c). However, unlike the Moderate Response Group, their response to the gain mapped almost perfectly onto the gain being presented in the complex environment. The response to the gain in the minimal environment was close to 0. This suggests that the gain had limited effect in the minimal environment for this group.

When asked to turn 45° in the complex environment, the slope of the response was 1.13, for the 90° requested turn it was 1.06 and for the 180° requested turn it was 1.05. All three of these responses are slightly larger than the perfect response to the increasing gain where the slope of the line would be 1. In contrast, for the minimal environment the slope of the lines are 0.14 when asked to turn 45°, 0.11 when asked to turn 90° and 0.08 when asked to turn 180°. This suggests that participants in this group were not impacted by the gain in the minimal environment while being strongly impacted by gain in the complex environment.

All Response Group (8 participants, 21%)

The All Response Group responded to the gain in both the minimal environment and the complex environment. Despite this, the All Response Group was still slightly more affected by gain in the complex environment.

Figure 4.8d shows that when asked to turn 45, 90 and 180° in the complex environment, the slope of the response were 0.76, 0.65 and 0.82 respectively. In the minimal environment when asked to turn 45°, 90° and 180°, the lines had slopes of 0.68, 0.38 and 0.22. This suggests that even in an environment with very few reference points, the All Response Group will respond to gain.

Discussion

Based on the analysis of the results and the user model, suggestions for developers and VR environment designers when modelling user turning behaviour in VR were created:

- **Users have similar accuracy turning Clockwise (CW) and Counter-clockwise (CCW)** - No statistically significant differences were found between how much users turned in the CW and CCW conditions. This confirms the findings of other studies on rotation gain (Nguyen and Kunz, 2018; Williams and Peck, 2019; Brument et al., 2021). Combined, this is strong evidence that designers do not need to account for the turning direction of the user when creating virtual environments as there is no difference in how users respond to the turn in the CW and CCW directions.

- **Users will rotate about 55° if asked to rotate 45°** - When asked to rotate 45°, users tended to rotate about 10° extra when no gain was present, both in a virtual environment and a physical space (see Table D.1 for details). This could be because users find it difficult to judge the size of a 45° turn. A half turn maps onto 180°, a quarter turn 90°. However, users are rarely asked to turn 45° which might have led users to overestimate the size of a 45° turn. Alternatively, users might be less accurate at judging small turns.

- **The number of visual cues in the environment impacts user gain perception** - Figure 4.4 compares the amount users rotated at different gain levels in both the complex and minimal environments. Users responded more strongly to the gain in the complex environment than the minimal. Wilcoxon tests found a statistically significant difference between the control conditions and complex condition at lower levels of gain (1.245 and 1.49). In contrast, in the minimal environment a statistically significant difference was only found at higher gain levels (some 1.49 and most 1.98), suggesting users were far less impacted by the gain.

  This suggests that the number of visual cues in the environment has a direct impact on how much user's are influenced by the rotation gain. As a result user's turn a different amount depending on the number of visual cues. This might be due to the weight different sensory modalities are given in each scenario. Fewer visual cues might cause users to rely more on their other senses - such as their vestibular system and proprioception. As gain does not directly influence these other senses, users may turn close to the requested turn as if no gain was present. When there are many visual cues present users might rely more on their eyesight to see how far they have turned. Thus users may be more strongly impacted by gain in environments with lots of visual cues.

  This is a similar effect to translation gain, where fewer visual cues increased translation gain thresholds (Kruse, Langbehn, and Steinicke, 2018). However, there is a limit to the improvement of gain response based on the number of visual cues. Users were impacted by rotation gain at roughly equal levels regardless if 4 or 16 objects were placed around them in a circle (Paludan et al., 2016). How many visual cues need to be distributed around a virtual environment for gain to be highly effective remains an open research question.

- **Users are likely to overturn at higher gain levels** - The amount users rotated in the virtual environment deviated strongly as gain level increased. In the complex environment users rotated about 180° or 90° when no gain (1) was present. Participants overshot the expected turn by about 20% in the highest gain condition (1.98) for both 180° and 90°. Each gain level between 1 and 1.98 led to slightly higher mean virtual rotation. A similar trend can be seen in the 45° condition where there is a 11.34° difference between no gain (1) and maximum gain (1.98). This result is especially important to consider for systems using higher gain levels in small tracking spaces. Compared to a prediction model that assumes near perfect user accuracy, actual users are likely to turn further than predicted. This has a direct impact on the future trajectory of the user in the tracking space and needs to be accounted for when building prediction models.

- **Users tend to rotate less in the physical space as the gain increases** - this is in line with similar studies in the field (Steinicke et al., 2010; Grechkin et al., 2016; Langbehn et al., 2017). Friedman and Wilcoxon statistical tests found statistically significant differences between users turning without gain in the real space compared to turning in the complex virtual environment with gain in the 45° and 90° condition. A similar trend can be seen in the minimal environment at stronger gain levels. A statistically significant change in turn amount was found in the minimal environment when users turned 180° with a gain level of 1.49 and 1.98.

- **Interaction Effects** - Within the user study, there was a combined effect of different independent variables that influenced turning accuracy more than just the individual variables could in isolation. There were interaction effects between the type of environment (complex or minimal), the gain level (1.245, 1.49 and 1.98) and the amount users were asked to turn (45°, 90° and 180°). Developers should consider how different aspects of their system could impact each other and the users.

- **Visual Cues Affect Gain Perception** - the number of visual cues affected the rate at which participants became aware of the gain. Participants who were presented with the minimal environment first became aware of the gain at a rate of 42%. In contrast, participants who were presented with the complex environment first, noticed the gain in the minimal environment at a rate of 68%. This suggests that while participants found it difficult to perceive gain with few visual cues, they were more sensitive to gain after they had already recognised it in the complex environment.

- **Modelling User Response** - The varied response of users to rotation gain has been noted in papers looking at the detection thresholds of rotation gain (Engel et al., 2008; Steinicke et al., 2010; Schmitz et al., 2018). These results suggest that not only the users perception threshold should be accounted for but also their response to the level of gain and how much they are naturally impacted by it. This study identified four different user groups, each of whom had a different response to gain when analysed with k-means clustering:

  - Low Response Group: 16% of participants did not respond to the gain and simply turned as if no gain was present.
  - Moderate Response Group: 37% of participants were moderately affected by the gain when there were enough reference points in the environment. This could be due to the multiple reference points giving participants a lot of visual feedback about how much they were turning. They still underestimated how much they had turned in the virtual environment at higher gain levels but did turn less overall as the gain increased.
  - High Response Group: a further 24% of participants were highly affected by the gain when there were plenty of reference points in the environment. This group turned very accurately within the virtual environment, adapting easily to the different gain levels.
  - All Response Group: 21% of participants were affected by the gain in both the visually complex environment AND in the environment with only one reference point to use to judge the rotation amount. This group had similar scores in the complex environment to the Moderate Response Group while responding to the gain in the minimal environment at an average rate of 1.43.

4.5 Summary

A user study was conducted measuring the accuracy of user rotation in VR at different gain levels. Participants turned in a complex virtual environment with many visual cues to help with orientation. They also turned in a minimal virtual environment which had a single visual cue. 38 participants were asked to turn 45°, 90° and 180° both CW and CCW at four different levels of gain in each of the environments. They also turned in the physical tracking space without a headset with their eyes open and closed.

The direction of the turn had no impact on turn behaviour. In contrast, the virtual environment, the gain level and the turn amount all had an impact on rotation accuracy. When asked to turn 45° with no rotation gain, participants turned closer to 55°. Participants were more strongly affected by gain in the complex environment compared to the minimal environment where they only responded to high levels of gain. At 1.98 gain in the complex environment users turned about 20% more virtually than they did when no gain was present. This suggests users thought they had turned less in the virtual environment than they actually had turned with the gain present.

There was a wide variation in how users responded to gain, similar to other research in the field (Engel et al., 2008; Steinicke et al., 2010; Schmitz et al., 2018). Participants were divided into four groups based on their style of response – 16% of users ignored the gain, 37% mildly underestimated the gain in the complex environment while not being effected by the gain in the minimal environment, a further 24% turned accurately in the complex environment at all gain levels while not being effected by gain in the minimal environment and finally 21% of users responded to the gain in both the minimal and complex environments.

Future research directions include studying how this different response to gain might impact user experience and accuracy when navigating a virtual environment rather than just rotating in place, finding the number of reference points needed for users to be impacted by the gain in a virtual environment and expanding the experimental system to also measure translation and curvature gain response.

Chapter 5

Segment Addition: A Novel Method for Redirected Walking in Virtual Environments

This chapter describes an implementation and exploratory user study of a novel redirection technique, *Segment Addition*. *Segment Addition* takes advantage of change blindness to add and remove slices in the virtual environment as the user turns. The additional slices cause the user to turn more to reach their destination, which allows manipulating the amount the user turns. The algorithm for *Segment Addition* is detailed and a user study implementing the *Segment Addition* method is analysed that shows the benefits of *Segment Addition* compared to other redirection techniques. In the study, users found the virtual environments natural and comfortable to turn in, even with significant additions made to the environment. Simulator sickness scores remained lower than rotation gain based redirection techniques. Segment Addition can be used in wide open virtual environments with few obstacles, unlike previous methods that rely on change blindness. The potential of the method in expanding where environment manipulation techniques can be used, as an alternative to rotation gain and as a method of changing the user's direction quickly and comfortably is discussed.

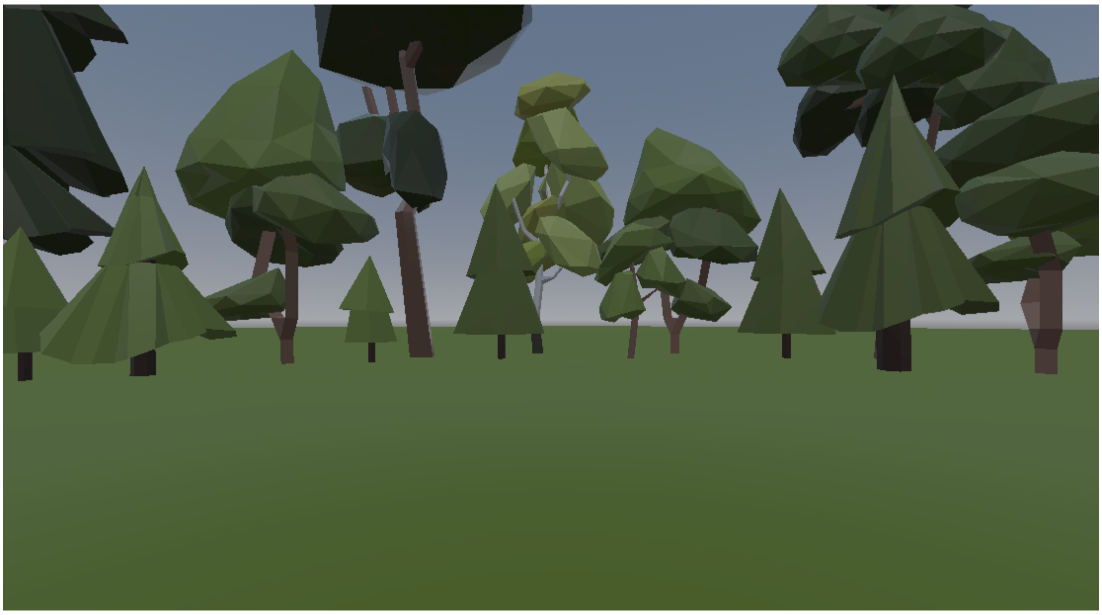


FIGURE 5.1: A Partial View of the Virtual Forest Environment

5.1 Introduction

This chapter describes a novel method of Environment Manipulation, *Segment Addition* which uses concepts from the field of redirected walking. There are two fundamental approaches to redirected walking: **Gain Manipulation** decouples the mapping between user movement in the tracking space from movement in the virtual environment (Razzaque, Kohn, and Whitton, 2005) and **Environment Manipulation** changes the virtual environment around the user to keep them within the tracking space.

Environment Manipulation changes the virtual environment and often take advantage of change blindness (Simons and Ambinder, 2005; Simons and Rensink, 2005), where changes to a person's surroundings remain unnoticed when they are looking elsewhere (see Section 2.3 for more details). Elements of the environment such as doors and walls are moved when the user is not looking, creating spaces that would be impossible in reality (Suma et al., 2011; Vasylevska et al., 2013). Most environmental manipulation techniques use boundaries such as walls and corridors (Vasylevska and Kaufmann, 2015) to hide the environmental changes from view (Suma et al., 2012; Robb and Barwulor, 2021; Ciumedean et al., 2020). An experiment comparing five different redirected walking strategies found change blindness was the best method for medium and small virtual environments (Gao et al., 2022). It had the lowest simulator sickness scores, the fastest walking speed, and users found it the most similar to natural walking. Despite the improved user experience, environment manipulation methods are limited by the kind of VEs they can be used in, as they need multiple virtual boundaries to take advantage of change blindness.

Gain Manipulation decouples the mapping between user movement in the tracking space from movement in the virtual environment (Razzaque, Kohn, and Whitton, 2005). This is achieved by deliberately moving the viewport camera in the virtual environment out of sync with user movement in the tracking space. The user unconsciously changes course to stay on their expected trajectory in the VE. There are many types of gain (see Section 2.1 for an overview), such as, rotation gain that changes the speed of the virtual rotation - the user turns faster or slower in the virtual environment than the tracking space. Below a certain level of gain, called the detection threshold, the user remains unaware of the change (see Li, Steinicke, and Wang, 2022 for a summary of detection thresholds). When the user is unaware of the gain, they have better cognitive task performance (Bruder, Lubos, and Steinicke, 2015). However occasionally gains above the threshold are needed to stop the user from moving beyond the edge of the tracking space. This can impact the user experience by causing breaks in presence (Schmitz et al., 2018) and increase simulator sickness (Schmitz et al., 2018; Hildebrandt et al., 2018).

Resetting uses rotation gains above the detection threshold to quickly change the user's facing direction in the tracking space. In small tracking spaces, resetting is common and can be disruptive to users (Peck, Fuchs, and Whitton, 2012). Gains significantly above the threshold also contribute to simulator sickness (Rietzler et al., 2019). See Section 2.1.2 for more details.

Segment Addition adds or subtracts pieces (called slices) to and from the environment outside the user's field of view (FOV). The addition of slices allows the system to control how much a user turns to reach their goal in both the tracking space and the virtual environment. In the physical space, the direction the user is facing after the turn is optimised to keep them within the tracking space. This technique can be used in wide open virtual spaces, even when few obstacles are present.

For comparison, an additional baseline technique was developed - *Moving Goalposts*. In *Moving Goalposts*, everything except the goal stays in the same location. The goal is moved around the user in a circular arc. When the user reaches the goal, the other potential goals are moved so that the relative position of all the goals remains the same after the turn. This technique is only suitable for environments without unique landmarks, as the user is more likely to notice the relative change in position of the goals to the static landmarks.

In addition to detecting the changes caused by redirected walking, user experience within the virtual environment can help determine how comfortable users are with a redirection technique. In the user study presented in this chapter, three factors of user experience are considered - naturalness, usability and simulator sickness.

Naturalness describes how intuitive a virtual experience feels to the user. Walking and turning are intuitive actions to navigate a virtual environment as they are how we navigate in the real world. One of the core benefits of redirected walking compared to other navigation techniques, such as teleportation or joystick control, is its perceived naturalness (Rietzler et al., 2019).

Usability describes the ease with which participants were able to apply the system, usually in the context of a task or outcome. Usability is often measured using the System Usability Scale (SUS) introduced by Brooke, 1996. However, it would have been challenging to display all 10 SUS questions after each turn in this user study. It would have overloaded participants' attention with entering responses and would have led to a less immersive experience. Instead, usability was condensed to a single question similar to Rietzler et al., 2019 and described in Section 5.5.

Simulator Sickness describes how users might feel motion sick while immersed in a VE. The exact cause of simulator sickness is unknown, however, the Sensory Mismatch Theory is often suggested as a possible explanation (Reason, 1978; Rebenitsch and Owen, 2016). It suggests that a conflict between sensory inputs causes users to feel simulator sickness. In a virtual environment with gain manipulation, the eyes will perceive a different speed of rotation than the vestibular system. Feelings of nausea or other simulator sickness symptoms can result from this sensory conflict. The Simulator Sickness Questionnaire introduced by Kennedy et al., 1993 is a common measure of simulator sickness that was also used in this study.

A user study was conducted to investigate the benefits and limitations of *Segment Addition* in comparison to the *Moving Goalposts* method. Our primary objectives were based on user perception and comfort when using the *Segment Addition* technique. These objectives are described in more depth in Section 5.5.

5.2 Method

5.2.1 Graphical Overview

Tables 5.1 and 5.2 show a graphical overview of *Segment Addition*. The technique has eight steps, with some potential variations. The left hand column displays a top down view of the virtual environment which illustrates what is happening all around the disk. The right hand side column displays a first person view of what the user sees in the stitch disk environment. Note that the ground textures for the hidden slices have been highlighted in bright orange for illustration purposes, usually they would be the same texture as the rest of the grounds so that the additional slices remain subtle. The added trees on the additional slices are unchanged and keep the colours the user will see. Below each image set is a text explanation of the process. At the start of this text is a number labelling the step which is referenced in Section 5.2 which explains the algorithm used in the method.

Segment Addition can similarly remove slices when the user is meant to turn less. In this case, instead of the hidden slices being added, default slices are removed outside the user's FOV. Later slices are moved over to compensate for the gaps. Once the turn is complete, the default slices that were hidden are added back in.

There is an edge case where the user has hidden slices in their FOV at the end of the turn, as seen in Figure 5.2. In this case, the hidden slices within the user's FOV stay visible. When all the slices have been shuffled over after a turn, the default slice

TABLE 5.1: Graphical Overview of Segment Addition Steps 1-4

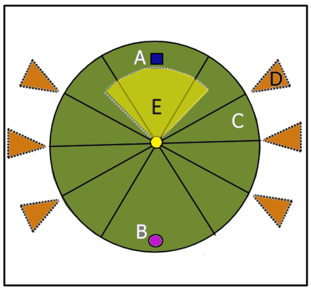
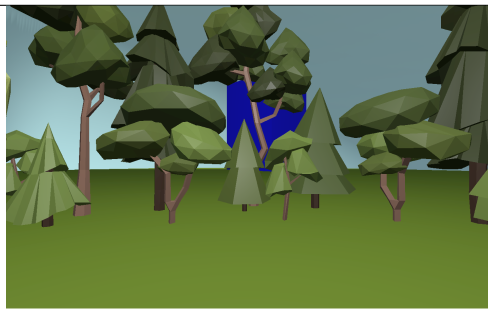
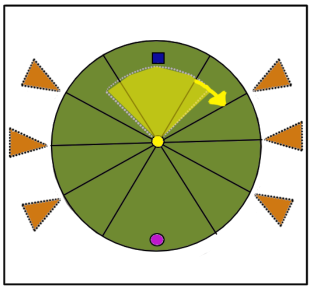
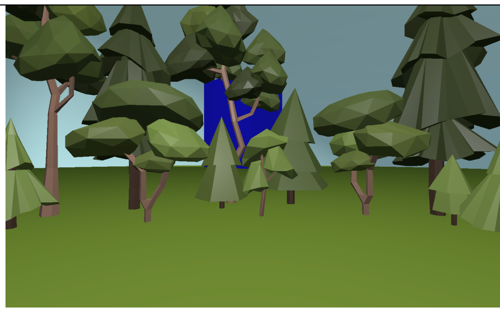
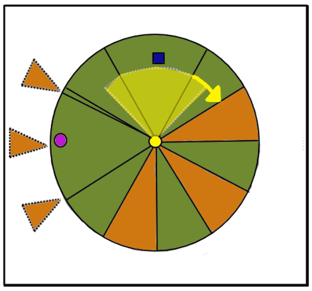
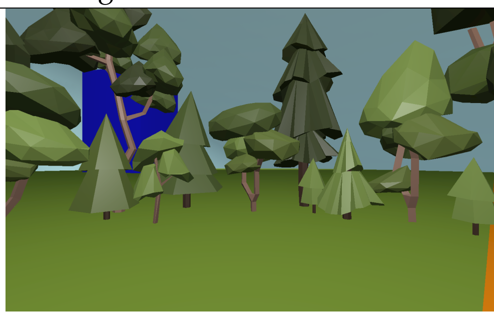
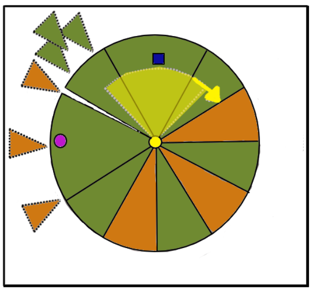
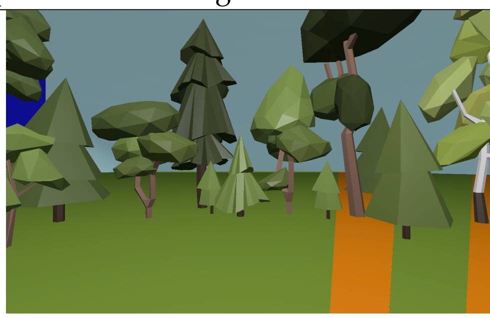
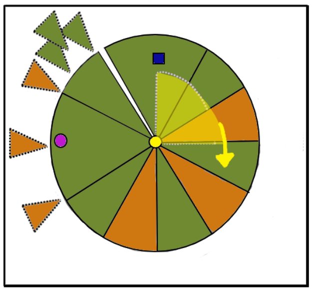
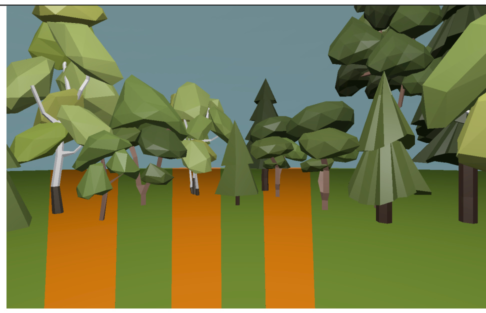
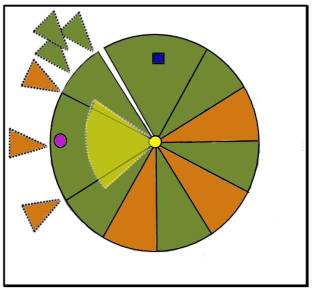
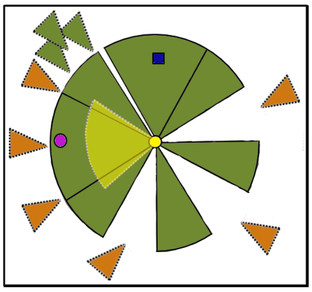
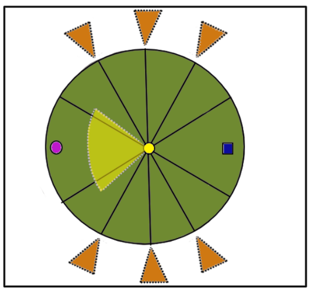
<table>
<tr><th>Top-Down View</th><th>First Person View</th></tr>
<tr><td>
</td><td></td></tr>
<tr><td colspan="2">1. A top down view of the Segment Addition environment: A. The currently facing slice. B. The goal slice the user wants to turn towards (pink circle). C. Default slices – these are visible by default. D. Hidden slices – these are invisible by default. The user cannot see them until they are added to the environment. E. The current Field of View (FOV) of the user. The user is standing at the centre of the disk and is currently directly facing slice A. Adjacent slices are also visible within their FOV.</td></tr>
<tr><td></td><td></td></tr>
<tr><td colspan="2">2. The user starts to turn. The system collects data on the slices currently within the user's FOV, the direction of the turn and the goal of the turn.</td></tr>
<tr><td></td><td></td></tr>
<tr><td colspan="2">3. Outside the user's FOV, hidden slices are added between the default slices. The default slices are moved over to compensate until the goal slice is reached.</td></tr>
<tr><td></td><td></td></tr>
<tr><td colspan="2">4. The slices between the goal slice and the start of the user's FOV are rotated around the centre point where the user is, starting from where the last additional slice was placed. If a slice overlaps with the user's FOV the slice is hidden.</td></tr>
</table>

TABLE 5.2: Graphical Overview of Segment Addition Steps 5-8

<table>
<tr><th>Top-Down View</th><th>First Person View</th></tr>
<tr><td></td><td></td></tr>
<tr><td colspan="2">5. The user turns until the last slice in their FOV is the original facing slice, which was in the centre of their FOV at the beginning of the turn. These slices, which are no longer in the user's FOV are hidden. To replace them, the slices between the goal slice and the original facing slice are shown until one of the slices overlaps the original facing slice.</td></tr>
<tr><td></td><td></td></tr>
<tr><td colspan="2">6. The user finishes the turn by stopping their rotation while looking at the goal.</td></tr>
<tr><td></td><td>
</td></tr>
<tr><td colspan="2">7. The hidden slices are made invisible once more.</td></tr>
<tr><td></td><td>
</td></tr>
<tr><td colspan="2">8. The default slices that were hidden during the turn, become visible and are placed after the last shown slice after the goal. All the slices after are turned to compensate for the new configuration of the environment.</td></tr>
</table>

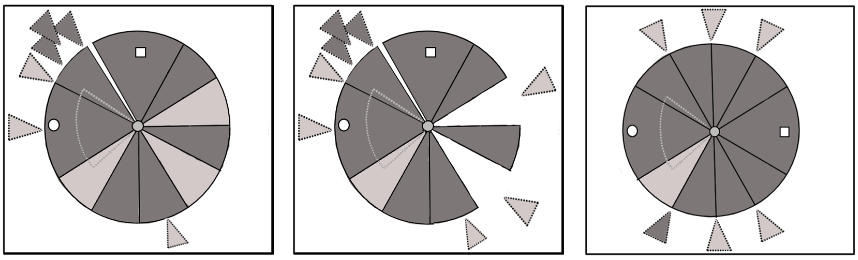


FIGURE 5.2: Overlapping hidden slice - when a hidden slices overlaps the user's FOV when they complete the turn, the slice remains unhidden. Instead, a default slice is hidden.

becomes invisible if it overlaps the visible hidden slice. After the next turn, the hidden slice is unlikely to be in the user FOV again and only the default slices will be shown after the turn. The system thus corrects for this edge case in later turns.

5.2.2 The Segment Addition Algorithm

This section describes the technical implementation of the *Segment Addition* technique and the potential trade-offs a developer can make in the implementation of the system. High-level pseudo-code is provided to aid in understanding the algorithm. For a more in depth look at the implementation, see the proof-of-concept system that was implemented for the user study, available online on Github[1]. The algorithm has been broken down into multiple steps, with each step running after the previous step of the algorithm has finished:

1. Once the user starts turning, the number of slices that need to be added or removed is calculated. This is based on the ideal facing direction of the user in the physical tracking space after the turn is complete. This algorithm section covers steps 1 and 2 in Table 5.1.

2. The chosen number of slices are added, until the calculated number is reached. Any slices left between the changed slices and the goal slice are shifted round to create a continuous arc. This section covers step 3 and 4 of Table 5.1.

[1]A sample implementation of the system can be found at https://github.com/chionic/Segment-Addition-Demo

3. Once the user turns far enough that slices that were inside their FOV are no longer visible, these slices are hidden. In their place, slices after the goal slice are rotated around to their final position. This section correlates to step 5 in Table 5.2.

4. Finally, once the user is facing the goal, the slices that were added to the turn or hidden during it are brought back to their initial state and shifted so that the relative positions of the goals remain the same. This describes Steps 6, 7 and 8 in Table 5.2.

Once all the steps of the algorithm are complete, the algorithm repeats from the first step. The algorithm sections below describe these steps in more detail. First, the variables are defined and tracked. The algorithm runs on an update loop, where every frame or update (here defined by time t) the VirtPos and PhysPos position vectors are tracked and updated.

$VirtPos \leftarrow$ position vector of the user in the virtual environment.
$PhysPos \leftarrow$ position vector of the user in the tracking space.
$Visible \leftarrow$ the set of slices currently visible in the user's FOV.
$Hidden \leftarrow$ the set of hidden slices.
$SliceList \leftarrow$ a complete list of all the slices.
$isTurn \leftarrow$ TRUE if the user is currently turning, otherwise FALSE.
$dir \leftarrow$ the direction of the turn (clockwise or counterclockwise).
$goalDir \leftarrow$ the direction of the goal relative to the player.
$startSlice \leftarrow$ the slice the user is directly facing at the beginning of the turn.
$firstVis \leftarrow$ the first slice visible to the user at the beginning of the turn.
$currDir \leftarrow$ the direction of the current slice the user is facing relative to the player.
$sliceSize \leftarrow$ the size (in terms of its arc) of a slice.
$t \leftarrow$ the current time in the application.

Algorithm 1 Gets the number of slices to add or remove

if $(delta(VirtPos(t), VirtPos(t-1)) \geq turnThreshold)$ **then**
  $OptimalDir \leftarrow findOptimal(PhysPos(t), dir)$
  $numSlices \leftarrow selectSlices(VirtPos(t), OptimalDir, goalDir)$
  $isTurning \leftarrow TRUE$
end if

Algorithm 1 runs once every time step t until a turn is started. It encompasses step 1 and step 2 of Table 5.1, deciding how many slices to add.

$turnThreshold$ describes the point at which the user is reported to be turning. The $turnThreshold$ accounts for minor variations in user movement that do not indicate the start of a turn. It is measured as the change in yaw rotation between the previous position vector of the user and the current one. If the change is above the threshold, a turn is started.

$findOptimal$ calculates the best facing direction in the physical space after the user has completed the turn. The ideal heuristic is the path that allows the user to avoid any future collisions with physical boundaries for as long as possible. However, this would require knowledge of the user's future path to calculate effectively, adding a predictive element to the system. Instead, the assumption was made that the user will continue forward in the direction they are facing after they complete the rotation. In this case, the user should ideally face in the direction where the physical obstacles are the furthest away from them on a straight-line path. One method to approximate this is to send out rays around the user, measuring the distance to when the ray hits an obstacle. The longest distance is chosen as the best direction for the user to face after turning, a similar method can be found in (Williams, Bera, and Manocha, 2021a).

$selectSlices$ calculates the number of slices to be added or subtracted for the user to face this optimal direction. First, the rotation required is calculated for the user to turn from their current facing direction to face the goal. Then, the difference is found between this turn and the ideal facing direction in the tracking space $OptimalDir$. Finally, the number of slices that need to be added or removed is calculated. This is based on the difference in turn amount divided by the size of an individual slice. See Algorithm 2 for a pseudocode description of this process.

Algorithm 2 Adding a slice

$naturalRotation \leftarrow abs(goalDir - currDir)$
$addAmount \leftarrow naturalRotation - OptimalDir$
$numSlices \leftarrow Math.Floor(addAmount/sliceSize)$

Once the number of slices is calculated, Algorithm 1 is not run again until the turn is complete. Instead, the procedure to change the number of slices begins with Algorithm 3.

Algorithm 3 Adds and Removes Slices

$firstOut \leftarrow outsideFOV(VirtPos(t), dir)$
for $slice < numSlices$ **do**
  $lastSlice \leftarrow addRemove(firstOut, slice, Hidden, SliceList)$
  $currSlice \leftarrow shiftSlice(lastSlice, SliceList)$
end for
$currSlice \leftarrow shiftSlices(lastSlice, Visible, SliceList)$

In 3, *outsideFOV* finds the first slice outside the user's FOV in the direction they are turning. This is the point where slices can start to be added and removed. This completes step 2 in Table 5.1.

addRemove adds or removes the next slice. If adding, it takes a slice from *Hidden* and places it next to $firstOut$. If removing, it takes the next slice after $firstOut$ and makes it invisible.

$shiftSlice$ tracks the position of the *lastSlice*, the previously added or removed slice. It moves the slice after *lastSlice* to start making an arc, until the ideal number *numSlices* of additional slices has been added or removed. This completes step 3 in Table 5.1.

$shiftSlices$ takes the position of *lastSlice*, and moves all the slices after *lastSlice* to make an arc until they overlap with the currently visible slices *Visible*. Any further slices are hidden from view rather than shifted around. This completes step 4 in Table 5.1.

Once the slices have been shifted, the next section of the algorithm is run. This section checks for when previously visible slices the user was facing are outside the user's FOV. As the slices are no longer in the user's FOV, they are removed and the slices after the goal slice are shifted, as shown in Step 5 of Table 5.2 and described by Algorithm 4.

Algorithm 4 Shift Slices after Goal Slice

$Visible \leftarrow findVisible()$
if $isVisible(startSlice, Visible) == FALSE$ **then**
  $hideExtra(firstVis, startSlice, SliceList)$
  $currSlice \leftarrow shiftSlices(currSlice, Visible, SliceList)$
end if

$findVisible$ returns a list of all the slices that are currently visible in the user's FOV.

$isVisible$ checks if a given slice is currently in the $Visible$ slice list, and so can be seen by the player.

$hideExtra$ hides the input slice(s), as shown in Step 5 of Table 5.2. The slices between $firstVis$ (the first slice visible to the user at the beginning of the turn) and $startSlice$ (the slice the user was directly facing at the beginning of the turn) are hidden.

$shiftSlices$ then shuffles the slices after the goal slice to fill the space of the hidden slices.

The final step of the algorithm, Algorithm 5, begins when the user is facing the goal slice and has completed the turn. It is equivalent to steps 6, 7 and 8 in Table 5.2. Once complete the virtual environment and slices are rotated so that the relative position of all the objects in the environment is the same as before the turn.

Algorithm 5 Shift slices to retain relative position of the goals

if $facing(goalDir, VirtPos(t))$ **then**
  $Visible \leftarrow findVisible()$
  $hideExtra(Hidden, SliceList)$
  $shiftSlices(currSlice, Visible, SliceList)$
  $isTurning \leftarrow FALSE$
end if

$facing$ compares the facing direction of the user (based on their virtual position) and the direction of the goal to the user. If the user is within 5° either side of the direction of the $goalDir$, the user is seen as facing the goal.

Once the user is facing the goal and the turn is complete with all the slices back in their correct positions (and hidden if they were originally in $Hidden$), the relative position of the virtual obstacles and points of interest are the same as before the turn. The relative position of the virtual obstacles to the tracking space has been re-aligned so that the user is in a more optimal position to continue walking without having to be reset (see step 8 in Table 5.2 for an example of the updated virtual environment). After this, the turn is finished and the algorithm returns to its initial state of waiting for a turn to begin as described in Algorithm 1.

In the algorithm above, additional slices are added or subtracted between existing slices. This approach adds multiple small slices between existing ones. It allows more

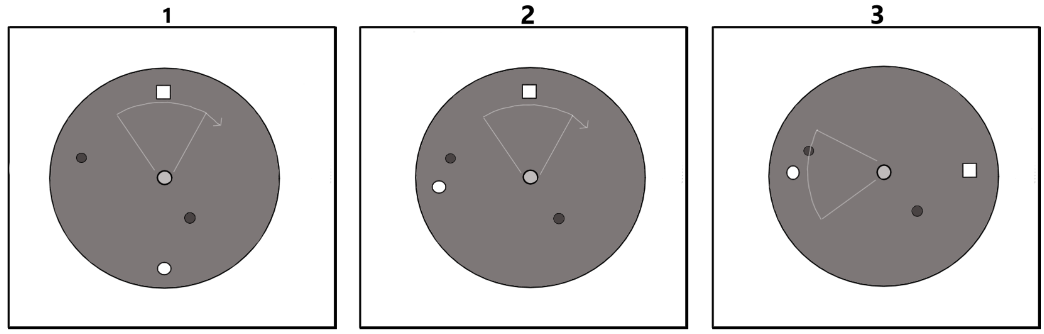


FIGURE 5.3: Steps in the Moving Goalposts Technique - 1. The user starts a turn. 2. The goal the user is turning towards is moved away in a circular arc. 3. When the user completes the turn, the goal they were initially facing is moved to keep its relative position to the other goal.

familiar areas of the environment to remain among the added or removed section, making the difference less noticeable. If the user is likely to be very aware of a slice, perhaps because it contains a unique landmark or feature, this slice should ideally remain as slices are removed or alternatively be part of the interleaved slices. An alternative approach is to add a singular, large additional area to the environment while the user turns. However, this might make the user more aware of the environmental change.

The size of the slices is a trade-off between the flexibility of the system and the computation cost of the algorithm. The system can place additional slices more easily if there are many small slices to place them between. It also gives a more accurate measurement of where the user's FOV ends, creating a larger area to add slices to. However, as the algorithm runs in real time, adding or removing too many slices in a single step can cause the system to slow down. Large slices require less processing power as they are treated as a single unit by the algorithm while covering a larger area. Breaking up each step of the algorithm, so only a couple of the slices are changed in one time step (i.e. one frame) alleviates this issue.

5.2.3 Moving Goalposts

Moving Goalposts changes the position of the goal, while keeping the rest of the environment the same. Figure 5.3 describes the process of the Moving Goalposts technique. When the user starts a turn, the goal they are turning towards is moved closer or further away along a circular arc. This causes the user to change how much they turn compared to the original distance between their direction and the goal. Once they are facing the goal, any other goals in the environment are re-positioned. The relative position of the goals remains the same after the turn, but the rest of the environment is offset.

In an empty environment with no other obstacles, this method would be indistinguishable from the *Segment Addition* method. However, as other objects are added, the user will likely notice the position of the goals moving over time relative to the rest of the objects in the environment. The user is more likely to notice this environmental manipulation technique when there are memorable, unique objects in the environment.

5.3 Implementation and Configuration

The initial concept for Segment Addition was first drafted in a series of pen and paper drawings, as a process sequences similar to the one shown in Tables 5.1 and 5.2. On paper, which could be folded and unfolded, the steps could be physically recreated. Once the concept was clear, a software system implementing the idea was considered.

The Godot 4.1.1 game engine was chosen as it was an open-source game engine with plug-ins such as OpenXR which could be used to build cross-platform VR applications. Based on the user comments of the previous experiment (described in Section 5.7) where participants found the large wire on the headset cumbersome with frequent turning, the Meta Quest 2 headset was chosen as the initial VR platform. Since the core of Segment Addition requires the user to turn the Quest 2 headset is ideal as it is a stand alone system with inside out tracking and without any wires. Additionally, unlike the turning accuracy experiment described in Chapter 4, the exact amount the user turned would not need to be measured in Segment Addition. The current facing direction of the user would be the primary method for determining when a turn was complete.

Via the android debug bridge (ADB) available in Godot and a Meta developer account, it was possible to have the Segment Addition application run natively on the

Quest 2 HMD as an Android application. A few scenes were built in Godot with the simplest version of the system - an empty plain of one colour where slices could easily be added or removed. After this an, initial virtual environment for a proof-of-concept user study was considered. A forest scene was chosen as it provided both a wide-open space and some objects for orientation within the environment. The initial slices for the system were created using Blender and a tree asset pack was used to create a forest scene[2].

The system adapted the Segment Addition algorithm to suit the requirements of this user study:

- Instead of calculating the optimal turn, the turn was chosen from a list of different addition and subtraction levels. Each level was presented to the user twice in random order, and no addition was presented four times. This was to better fit with the goals of the user study as it provided a predictable range of turn amounts.

- A pilot study with 3 participants was run to ensure the method and system worked. The *turnThreshold* was chosen based on the results of the pilot study. If the threshold is too fine, then small head motions of the user are seen as beginning a turn even if the user has no intention of turning. In contrast, if the threshold is too coarse, then slow rotations are not considered rotations by the system.

- Adjacent slices were removed when the number of slices was decreased. The first slice outside the user's FOV was hidden, followed by subsequent slices until the total number of slice removals had been reached. As there were no specific slices that held areas of interest other than the goal slice, there was no reason to keep in-between slices from being hidden. In contrast, additional slices were interleaved with existing slices as shown in Table 5.2.

[2]For further programming details, see the proof-of-concept project up on Github: https://github.com/chionic/Segment-Addition-Demo

The Moving Goalposts environment was a similar forest clearing with two goals as the *Segment Addition* environment. However, rather than the environment expanding and contracting with each turn, instead the goals were moved in a circular arc as described in Section 5.2.3. The system was piloted with three users to fine-tune and calibrate the point at which users' head rotation was indicative of the beginning head-turn.

5.4 User Study

5.4.1 System and Setup

Ethical approval for this user study was received from our university ethics board and the study was advertised on campus. Our exclusion criteria were:

- being under 18 or over 65 years of age.
- having photosensitive epilepsy.
- experiencing severe motion sickness when travelling.

This user study presented two VEs, each containing a forest clearing with trees, and that used either the *Segment Addition* or the *Moving Goalposts* technique. Users were asked to turn between two goals represented by a large purple cylinder and a blue square in the distance. In the beginning, these two goals were 180° apart. The user begins facing one of the goal slices (Fig. 5.4a). As they turn, slices are added or removed from the environment, outside their FOV. The user only sees the changed trees (Fig. 5.4b). The user completes the turn by facing the other goal and a menu appears with questions about the turn (Fig. 5.4c). The environments, while graphically similar, used the two different algorithms *Segment Addition* and *Moving Goalposts*. A log file was created each time the virtual environment was launched, which tracked how the user turned, their responses to the menu questions, and the state of the virtual environment.

The environments, while graphically similar, used the two different algorithms described in Section 5.2. In the *Segment Addition* environment, additional slices with

(A) The user's view facing the blue goal (B) user's view while turning through the environment (C) a menu appears when the user faces the pink goal

FIGURE 5.4: Snapshots of the User's View of the Environment.

their own trees were placed between existing slices of the environment. The relative position of the trees surrounding the goal remained the same. In the *Moving Goalposts* environment, the trees stayed in the same position while the goals moved in an arc around the user in steps of 11.2°.

At 0°, no addition was made to the 180° turn, while at 11.2° either a single slice was added to the turn (*Segment Addition*) or the goal was moved further away in an arc of 11.2° (*Moving Goalposts*). At -11.2° the goal was moved closer by 11.2° or a single slice between the two goals was removed. The maximum number of added or removed slices was six (equivalent to 67.2°) within the experiment. This led to turns between 112.8° and 247.2° in steps of 11.2°. Table 5.3 shows a list of the number of degrees added to expand and contract the environment.

Users went through each turn amount (-67.2° to 67.2° in steps of 11.2°) twice in each environment, except for the 0° condition which was presented four times. The default and hidden slices have an arc 11.2° while the goal slices have an arc of 56.0°. The system would also work with all slices having an equal arc. However, it is more efficient to have wider slices around the goal since the user will be able to see them at the start and end of the turn and the goals are fixed.

In total users turned 28 times in each environment. In *Segment Addition* four extra 0° turns were added to the end of the turn sequence if the user had not triggered the 67.2° turn four times yet. This was done to increase the odds of the user triggering the condition as they had to be facing the goal slice while the slice beside it was outside

TABLE 5.3: Environment Contraction and Expansion - List of the Number of Degrees that could be added with each turn

Environment Contracted						
-67.2°	-56°	-44.8°	-33.6°	-22.4°	-11.2°	0°

Environment Expanded					
11.2°	22.4°	33.6°	44.8°	56°	67.2°

TABLE 5.4: Participant Age

Age	Percentage
18-25	48%
26-35	28%
36-45	3%
46-55	14%
56-65	3%
Prefer not to say	3%

TABLE 5.5: Participant Previous VR Experience

VR Experience	Percentage
None	17%
Used VR once or twice	34%
Used VR a few times	31%
Regularly use/own headset	17%

their FOV.

5.4.2 Participants

In total 30 participants took part in the user study, however one participant was only recorded for the first environment due to a logging error. Thus, data from 29 participants was used in the analysis. Tables 5.4 and 5.5 show the age distribution and previous VR experience of the participants.

5.4.3 Procedure

At the start of the study, participants received an information sheet and signed a consent form. They completed a simulator sickness questionnaire (SSQ) as a pre-test and then were fitted with the Meta Quest 2 headset.

The first of the two environments was started. Each virtual environment included a grassy plain with trees and two large goals in the distance, one pink and one blue (see Figure 5.4a and 5.4c). The participant repeatedly turned between the two goals. They could turn in either direction (clockwise or counter-clockwise) but were instructed to occasionally change the direction of their turn, to mitigate dizziness.

After each turn, a menu appeared asking the participant to rate their experience of the turn, as shown in Figure 5.4c). First, the participant decided whether the turn felt bigger, smaller, or the same as a half turn (180°). Then the participant rated their agreement with the following statements on a 7-point Likert scale:

1. The turn felt natural.
2. The turn was pleasant.
3. I could imagine using this rotating technique to move inside virtual worlds.

These questions are similar to those asked by (Rietzler et al., 2019). After every fifth turn, user comfort within the environment was measured on a scale of 1 (uncomfortable) to 10 (very comfortable). If the participant selected 1 or 2 on the scale, the study was stopped early.

Once the participant finished the first set of turns, they removed the headset and completed another SSQ. A short break of at least two minutes was taken before the user completed the second environment. Finally, the participant took off the headset and completed an SSQ and a general questionnaire on their experiences.

Of the 29 participants in the study, 14 saw *Segment Addition* first, followed by *Moving Goalposts*. The other 15 participants saw *Moving Goalposts* followed by *Segment Addition*. Non-parametric tests were chosen when comparing *Segment Addition* and *Moving Goalposts*, to remove any potential confounding effects from this minor difference in group size.

5.5 Results

This section describes the user experience and response to *Segment Addition*. Four research questions were considered in relation to the user study:

- RQ1. Do users notice the environmental changes induced by the *Segment Addition* and *Moving Goalposts* techniques?
- RQ2. How many slices can be added or subtracted from the environment using *Segment Addition* before the user notices the change?
- RQ3. Are the users comfortable with the expansion of the environment with the *Segment Addition* technique?

TABLE 5.6: Participant Response to different turn amounts - 'Turn Amount' is the degrees altered from a 180° turn. 'Perfect Response' is the 'accurate' response - Smaller with degree subtraction and Bigger with degree addition. '% Accuracy SA' and '% Accuracy MG' is the percentage of users in the Segment Addition and Moving Goalposts environments respectively who matched 'Perfect Response'.

	Environment Contracted						
Turn Amount	-67.2°	-56°	-44.8°	-33.6°	-22.4°	-11.2°	0°
Perfect Response	Smaller						Same
% Accuracy SA	80%	67.4%	77.1%	62.5%	50%	31.3%	50%
% Accuracy MG	97.9%	85.4%	81.3%	58.3%	41.7%	25%	51%

	Environment Expanded					
Turn Amount	11.2°	22.4°	33.6°	44.8°	56°	67.2°
Perfect Response	Bigger					
% Accuracy SA	37.5%	68.1%	79.2%	66.7%	72.9%	68.4%
% Accuracy MG	45.8%	60.4%	81.3%	81.3%	81.3%	85.4%

- RQ4. Do *Moving Goalposts* or *Segment Addition* contribute to simulator sickness and if so, to what degree?

The first two research questions aim to highlight the user's perception of the turn while the latter two focus on user comfort within the virtual environment. These questions were examined as an initial evaluation of the practical use and personal comfort of the *Segment Addition* method.

5.5.1 Turn Perception

After each turn was completed by the participant, a menu inside the virtual environment asked, "Did the turn feel bigger or smaller than a half turn (180 degrees)?". Participants could select "Bigger", "Smaller" or "The Same" as options. This Turn Perception question was chosen to investigate RQ1 and RQ2 that relate to user turn perception. For the Turn Perception question, in total 24 participant responses were analysed, as five participants responses had to be excluded due to a recording error.

The "Perfect Response" row in Table 5.6 models how users would be expected to respond to the Turn Perception question if they could perfectly identify exactly how much they turned. This model, therefore, represents an idealised perceptual response for a theoretical user that would not react to the intervention. That is used here as a baseline to measure the effectiveness of the two techniques. The next two rows in

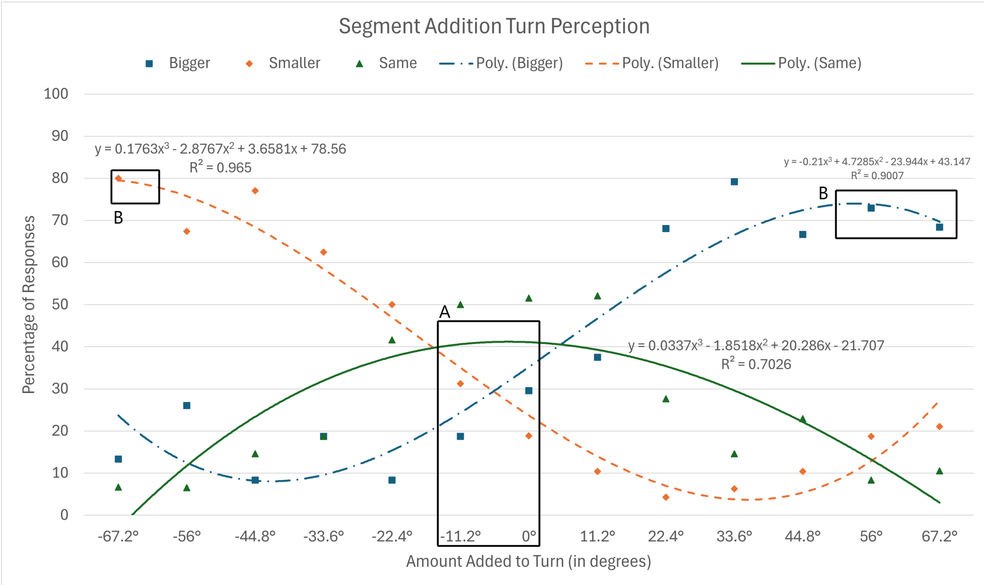


FIGURE 5.5: The proportion of responses of 'Bigger', 'Smaller' or 'The Same' at each level of added arc to the turn in the Segment Addition Environment

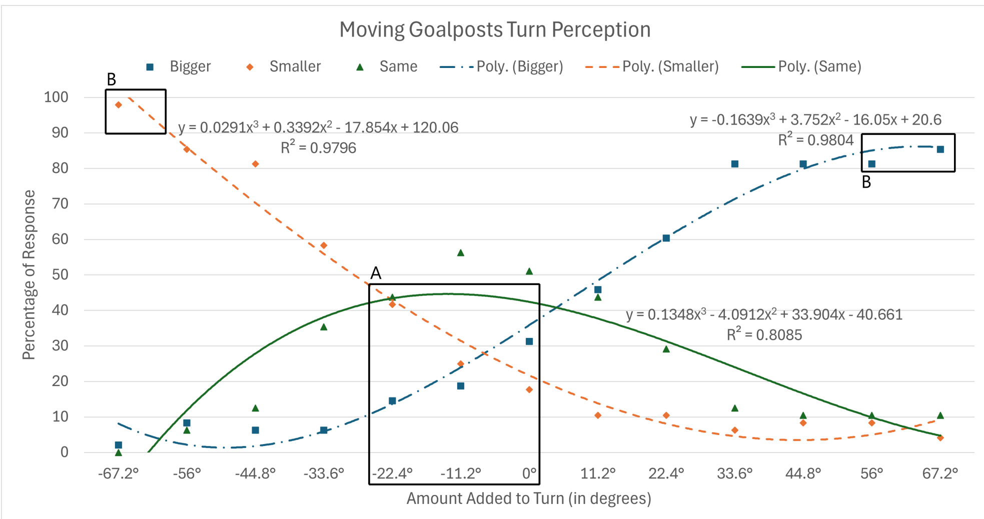


FIGURE 5.6: The proportion of responses of 'Bigger', 'Smaller' or 'The Same' at each level of added arc to the turn in the Moving Goalposts Environment

the Table 5.6 show the percentage of participants who correctly identified the turn in the Segment Addition (% Accuracy SA) and the Moving Goalposts (% Accuracy MG) conditions. If *Segment Addition* did not change how users perceived the environment, a response similar to that modelled in the 'perfect response' column of Table 5.6 would be expected - where users would all reply 'smaller' when parts of the environment had been removed, 'bigger' when the environment expanded and 'the same' when nothing was changed in the environment. Unlike the Perfect Response row, participants did indeed respond to both techniques and they did so non-uniformly.

Based on this response, descriptive user models were created for turn perception (Figs. 5.5 and 5.6) using polynomial equations to describe the general trend of participants' responses to the change in their turn amount. The R^2 value in the models describes how accurate the model is compared to the observed response of participants, with the model being more accurate the closer it is to 1. From these models, the following conclusions are drawn:

- **Small changes remain unnoticed**: When only one segment (± 11.2°) was added or subtracted in the environment, participants were more likely to describe the turn as the same as 180° than as bigger or smaller.

- **Underestimation of 180°**: With *Moving Goalposts*, participants were more likely to see a turn of 168.8° as the same as 180° (-11.2° and 0° conditions in box A in Fig. 5.6). Similarly, participants were more likely to describe a 180° turn as bigger than 180° than smaller (0° condition in box A in Figs.5.5 and 5.6). The point at which participants are equally likely to describe a turn as bigger or smaller than 180° is when the turn is slightly smaller than 180°.

- **Participants notice larger deviations**: as more of an arc was added to the turn, more participants stated the turn felt 'Bigger' than a half turn. Similarly, as the arc decreased, more participants stated the turn felt 'Smaller' than a half turn in both environments (Boxes B in Figs 5.5 and 5.6).

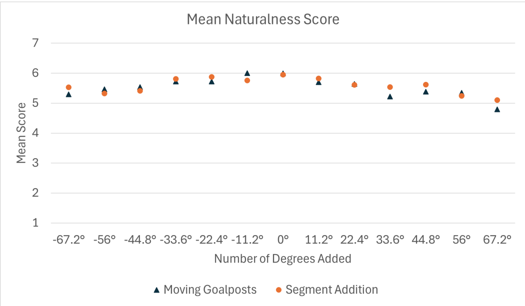
5.5.2 Naturalness and Usability

The study also analysed the user experience of the environmental changes in relation to RQ3, both before and after participants became aware of the change in turn amount. This experience was measured by the naturalness and usability of the system and the user's comfort with the system.

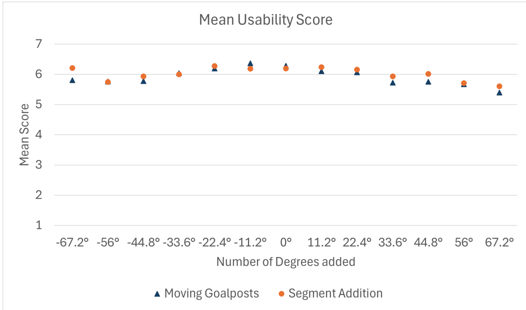
One of the core benefits of redirected walking compared to other navigation techniques, such as teleportation or joystick control, is its perceived naturalness by users (as described by Rietzler et al., 2019). Walking and turning are intuitive actions to navigate a virtual environment as they are how we navigate in the real world. Participants were asked to rate their agreement with the statement "The turn felt natural" on a scale of 1 (not natural at all) to 7 (very natural). Figure 5.7a shows the mean naturalness of each turn. Overall, the turns received a high rating with a mean score of 5.65 for *Segment Addition* and 5.56 for *Moving Goalposts*. As the turns deviated further from 180° the mean naturalness decreased slightly.

Usability describes the ease with which participants were able to use the system. Usability is often measured by using the System Usability Scale (SUS) introduced by Brooke, 1996. However, it would have been challenging to display all 10 SUS questions after each turn in the virtual environment. It would have overloaded participants attention with entering responses and would have led to a less immersive experience. Instead, a single question was asked, "I could imagine using this rotating technique to move inside virtual worlds" with a system usability scale from 1 (unusable) to 7 (very easy to use). The mean usability score remained high regardless of how much was added or subtracted to the turn, as shown in Figure 5.7b. The mean usability score across all turn amounts was 6.05 for *Segment Addition* and 5.94 for *Moving Goalposts*.

Figures 5.7a and 5.7a show the high overall scores given to both naturalness and usability as well as the slow decline in mean scores as the turn deviated further from a half turn. The turn that received the highest naturalness and usability scores varied from -22.4° to 0° between the *Segment Addition* (0° and -22.4°) and *Moving Goalposts* (-11.2° and-11.2°) conditions. Notably, in the *Moving Goalposts* condition users perceived the environment to be the most natural and useable when the amount they turned was



(A) The mean score between 1 and 7 participants gave to the question "The turn felt natural" at varying turn levels.



(B) The mean score participants gave to "I could imagine using this rotating technique to move inside virtual worlds."

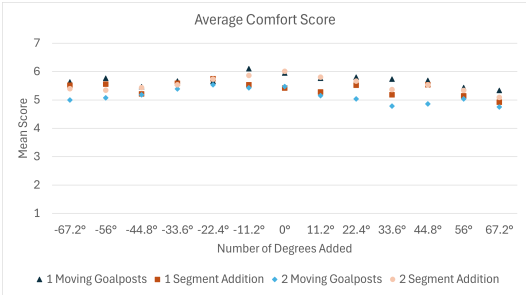
FIGURE 5.7: Naturalness and Usability scores

slightly less than 0°, at -11.2°. In contrast, participants consistently gave the lowest scores to the addition of 67.2°.

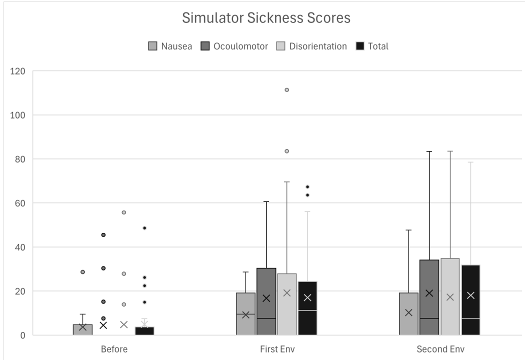
Non-parametric Wilcoxon tests found a statistically significant difference between the highest and lowest scores when comparing naturalness (Moving Goalposts $W = 24$ with approx normal distribution, $z = -4.4$, $p < .001$, Segment Addition $W = 107$, $p < .05$). These two results suggest that perceived naturalness decreases as more changes are made to the virtual environment. The rate of decrease is slow as even when 67.2° was added to a turn, the average naturalness score remained high. Similarly, for usability non-parametric Wilcoxon tests comparing the highest and lowest scores found a statistically significant difference (Moving Goalposts $W = 29.5$, $p < .05$, Segment Addition $W = 68.5$, $p < .05$). This is a similar trend to naturalness, where as the number of segments increases there is a slow decline in the mean usability scores. While there is a statistically significant difference between the highest and lowest scores, the high overall mean scores and slow rate of decrease (all mean scores remained above 4 on the 7-point Likert scale) suggest a large amount can be added or subtracted to a turn while retaining high naturalness and usability using the *Moving Goalposts* and *Segment Addition* techniques.

5.5.3 Simulator Sickness

Three measures of comfort were implemented in this user study in relation to RQ3 and RQ4. First, participants were asked how comfortable they felt after each turn in the



(A) Mean Comfort Scores participants gave for each turn by the amount added to the turn, the algorithm (Moving Goalposts or Segment Addition) and the order of presentation (1 — first environment, 2 — second environment)



(B) Simulator Sickness Questionnaire scores for before, after the first environment and after the second environment at the end of the user study.

FIGURE 5.8: Simulator Sickness and Comfort scores

virtual environment. Secondly, participants could end the user study early once every five turns by selecting a 1 or 2 on a 10-point scale asking about their comfort in the environment in relation to simulator sickness. Finally, the Simulator Sickness questionnaire was used once at the beginning of the study, once after the first environment and again at the end. The results of each of these measures are described below.

Comfort Scores per Turn

The mean comfort score for all turns was 5.5 on a possible 7-point Likert scale (1 = very uncomfortable, 7 = very comfortable). It was analysed if there were any significant differences between the scores based on how much was added/subtracted to the turn, if the turn was earlier or later in the turn order, or which algorithm was used. No statistically significant differences were found, with the mean score remaining similar for all these factors; see Figure 5.8a for exact values.

Sickness Scores during the Virtual Turns

Every five turns participants' were asked to rate their comfort with the environment on a scale from 1 (very uncomfortable) to 10 (very comfortable). If the user chose 1 or 2 on the scale, the study was ended early. Of 29 participants, three chose to end the experiment early after the 25th turn in the second environment. In two of these cases, the participants commented on feeling tired due to holding either the controllers or

TABLE 5.7: Simulator Sickness Mean, Standard Deviation and Standard Error across conditions

Condition	Mean	St Dev	St Error
Pre			
Nausea	3.62	7.82	1.45
Oculomotor	4.44	11.37	2.11
Disorientation	4.8	11.93	2.21
Total	4.90	11	2.04
First Env			
Nausea	9.21	10.35	1.92
Oculomotor	16.73	20	3.71
Disorientation	19.2	29.15	5.41
Total	17.02	19.83	3.68
Post			
Nausea	10.2	13.23	2.46
Oculomotor	19.08	23.94	4.45
Disorientation	17.28	24.86	4.62
Total	18.06	21.97	4.08

headset up for a prolonged period of time. One participant did not comment on why they finished early despite their previous comfort scores being high. In all three cases, participants experienced between 2 and 5 total simulator sickness points, suggesting they experienced slight simulator sickness by the end of the experiment.

The mean comfort score for both the *Segment Addition* and *Moving Goalposts* conditions was 8.5 (Median: 9). There were only minor variations in average score depending on if the user was in the first or second environment and the turn number (e.g. 5, 10, 25 etc) the participant was scoring. This indicates that most participants were very comfortable in both environments.

Simulator Sickness Questionnaire

The Simulator Sickness Questionnaire (SSQ) is a popular measure of participant comfort in virtual environments. The simulator sickness scores (including the disorientation, oculomotor, and nausea subscales) were calculated using the method described in Kennedy et al., 1993). Figure 5.8b shows the results of the SSQ, including the mean, median and interquartile range. Table 5.7 shows the standard deviation as suggested by Bimberg, Weissker, and Kulik, 2020.

Similarly to other SSQ results in the VR field, that SSQ data from this study did not follow a normal distribution, with some participants experiencing no simulator sickness symptoms at all while other participants experienced multiple symptoms. As the data was not normally distributed, non-parametric tests were used.

Whether the two different algorithms were significantly different from each other was analysed. This was done by combining all the *Segment Addition* condition results and all the Moving Goalposts results and then running a paired Wilcoxon test. No statistically significant differences were found between the two algorithms (V = 149.5, p .05). Additionally, there were no statistically significant differences between the SSQ scores of participants after the first environment using *Segment Addition* compared to after the second environment using *Segment Addition* (W = 124.5, p .05). Similarly, no statistically significant difference was found for the two Moving Goalpost scenarios (W = 86.5, p .05). This suggests that the algorithms used - *Segment Addition* and *Moving Goalposts* - contributed equally to any simulator sickness experienced by participants.

As there were no statistically significant differences between the algorithms, the analysis instead compared Simulator Sickness before, after the first environment and at the end of the user study. Friedman tests found no statistically significant differences in the disorientation (DoF = 2, Q = 4.34, $p > .05$) and nausea scales (DoF = 2, Q = 4.67, $p > .05$). However there was a difference in the Oculomotor (DoF = 2, Q = 9.64, $p < .001$) and Total scores (DoF = 2, Q = 13.4, $p < .001$). The follow-up paired Wilcoxon tests found a statistically significant difference between the questionnaire from the beginning and the one completed at the end of the user study (Oculomotor: V = 19.5, $p < .01$, Total: V = 28, $p < .01$), as well as between the before and after the first environment questionnaires (Oculomotor: V = 7, $p < .01$, Total: V = 18.5, $p < .001$).

This indicates that being inside the virtual environment increased simulator sickness. However, the time spent in VR during the first environment compared to the completion of both environments did not alter simulator sickness scores.

Comparing the mean total scores of 17.02 (after the first environment) and 18.06

(at the end of the experiment) to simulator sickness models for redirected walking introduced by Gemert et al., 2024, the simulator sickness scores in this work can be considered on the low side of the medium bracket (Low (5–15), Medium (15–30)). They are slightly higher than the model prediction of Gemert et al., 2024 for real walking studies (15.65 $\pm$ 0.98) and lower than the prediction for reorientation techniques such as curvature gain and rotation gain (20.61 $\pm$ 1.48). With rotation gain, stronger gain increases simulator sickness scores significantly and increases drop out rates among participants (Schmitz et al., 2018; Hildebrandt et al., 2018). In contrast, the simulator sickness scores with *Segment Addition* and *Moving Goalposts* stayed comparatively low even when a large number of additional slices were added to the turn.

5.5.4 Participant Comments

This section is a summary of the participants' responses to the final questionnaire and a comparison of comments with similar content. See Appendix I for a full list of responses.

User Preference

Participants had roughly equal preference for the two techniques (48% for *Segment Addition*, 38% for *Moving Goalposts*, 14% did not have a preference). This question relates to RQ1 - how user's perceived *Segment Addition* and *Moving Goalposts*. When asked to describe their preference qualitatively, most participants described their preferred environment as 'smoother' and 'more natural' or referred to the order of environment presentation as the reason for their preference.

The most common reason for the environment preference was the order of presentation. Some participants preferred the second environment because they had a better sense of the task ("was more used to the experiment and it was quicker as a result"). Others found the first environment more engaging ("first, because it was the begin. I had curiosity to be in the environment and do interaction").

Other participants described aspects of the virtual environment itself as influencing their preference. They described their preferred environment as 'smoother' or 'more natural'. Of these, half preferred *Segment Addition* and half *Moving Goalposts*. A few (10%) participants mentioned that turns felt more like 180° in the *Segment Addition* environment - "felt I turned exactly 180 degrees". Two (7%) users preferred how the *Segment Addition* environment changed, as it held their attention better than *Moving Goalposts* ("the second (Segment Addition) appeared to have changes in scenery on occasion that grabbed my attention").

This suggests participants found both *Segment Addition* and *Moving Goalposts* equally comfortable with the order of presentation being a common reason for preferring one over the other.

Environmental Changes

Participants were asked about anything unusual they noticed inside the virtual environments to gain further insight into RQ1 and RQ2. Almost no one could specifically describe how the environment changed with *Segment Addition*. Some participants accepted the environments without further comment (38%). Environmental details unrelated to the two different algorithms were described by 24% (e.g. "the size of the blue + the pink shapes changed", "Nothing out of the ordinary except the hands were of different shapes"). Environment details related to the trees ("A tree I hadn't noticed before seemed to appear") or to the turns ("the landmark moving relative to the environment") were commented on by 38% of participants. Only 10% of the participants commented directly on the Segment Addition technique when describing environmental details (e.g. "trees were replaced by different trees", and "appeared to have changes in scenery"). This suggests that while participants did notice the change in turn amount, most did not understand how the environment changed around them with *Segment Addition*.

Additional Participant Comments

At the end of the questionnaire was a blank textbox for additional comments, which 48% of participants completed. Comments on the VR experience in general being "fun" or "interesting" were left by 24% of participants. Discomfort later in the experiment due to the headset and virtual environment ("the weight of the vr headset was noticeable") was described by 14%. Finally, 7% of participants referred back to their experience of the tasks in the experiment providing comments such as "It's difficult to continuously turn on the spot without translating, especially when vision is obscured." and "tended to prefer larger or similar turns, disliked turns that felt smaller". Finally, one participant described the effects of the *Moving Goalposts* technique while perceiving that something else was going on in the *Segment Addition* environment, without being able to tell what exactly was changing.

5.6 Discussion

This section reflects on the similarities between Segment Addition and similar literature, as well as, the results of this user study and the potential future directions of *Segment Addition* as a technique for virtual environment adaptation.

Literature Comparison

There are several approaches that can be compared to Segment Addition within the field of redirected walking. We do not have scope to make a quantitative comparison between the approaches but provide a high-level commentary on the similarities here.

Environment Manipulation techniques, such as expanding rooms (Suma et al., 2012), alter the virtual environment at run time by making use of change blindness. This is similar to the approach used in Segment Addition which alters the environment outside the users' field of view. Expanding rooms, and similar methods, work well in highly occluded spaces such as indoor office environments or mazes. In contrast, Segment Addition excels in wide-open spaces with few occluding objects, as shown in the user study. Thus the use case for Segment Addition differs from existing Environment Manipulation techniques such as expanding rooms.

Virtual cell (Yu et al., 2017; Marwecki and Baudisch, 2018) approaches split a virtual environment up into cells the size of the tracking space before the user ever enters VR. The user must interact with a transition metaphor to move from one cell to the next and is deposited in the same 'spot' in the next cell as the previous cell. Segment Addition also splits the environment into pieces, 'slices', to aid in keeping the user inside the tracking space. Virtual cell approaches are limited by the transition metaphors available. For example, inside a building, a lift might deposit a user in the same location on a higher floor than the user entered it and thus is a very natural metaphor. In contrast, birds carrying the user from one cell to the next have been suggested as a transition metaphor out of doors. Segment Addition does not require any such metaphor, as the change happens outside the user's field of view at run time. The results of the user study show that Segment Addition was rated as having high naturalness. As a result, we believe Segment Addition can be more easily adapted to different types of virtual environments that the virtual cell approach.

Rotation gain (first introduced by Razzaque, Kohn, and Whitton, 2005), alters how much a user turns in the physical tracking space with the goal of keeping the user within the bounds of the space. Similarly, Segment Addition also optimises how much the user turns to keep them within the tracking space. Rotation gain leads to significant increase in simulator sickness symptoms when large alterations are made to the turn (Schmitz et al., 2018; Hildebrandt et al., 2018). In contrast, the results of the user study show that with Segment Addition users remained comfortable even when large changes were made to amount they turned. We hypothesise this could be due to the fact that, unlike rotation gain, Segment Addition does not introduce a sensory mismatch - the user turns the same amount in both the physical and virtual space. The amount they have to turn to reach their goal in the virtual environment is altered instead. After our experiments and user studies we believe that Segment Addition is better than approach than rotation gain when large changes need to be made to a turn as a result.

Segment Addition Applications

The following areas are described where *Segment Addition* could be used, based on the results of the user study described in Section 5.5, either in combination with or instead of alternative Redirection Techniques:

1. **Environment Manipulation for diverse virtual spaces**: Environmental Manipulation techniques such as impossible spaces (Ciumedean et al., 2021) and moving doors (Suma et al., 2011) have shown lower simulator sickness among users than other redirected walking techniques (Gao et al., 2022). However, these techniques require many boundaries such as walls to obscure their environmental changes.

   Segment Addition instead uses change blindness to alter the environment outside the user's FOV. In relation to RQ1 and RQ2, the model described in Figure 5.5 shows how users slowly become aware of the smaller and larger turn as the change in rotation amount deviates further from 180° (e.g. only 31.3% of user described a turn of -168.8° as "Smaller" than 180° compared to 80% who described a 112.8° turn as "Smaller" - see Tab 5.6 for more detail). Even once the change was noticed by users, they continued to find the *Segment Addition* method natural and easy to use (RQ3) with only a slight decline in average naturalness (max $\overline{x}$ = 5.95, min $\overline{x}$ = 5.1) and usability (max $\overline{x}$ = 6.28, min $\overline{x}$ = 5.6) scores. Additionally, while participants noticed the change in how much they turned, only 10% of participants described how the environment was altered to achieve this within the *Segment Addition* environment. This suggests *Segment Addition* can be used to make small adaptations to the environment without the user noticing. It can also be used to make larger adaptations to the environment that the user perceives while remaining both natural and comfortable.

   Segment Addition was designed for wide open virtual spaces with few points of interest such as the one in this user study. It can be used in outside spaces between enclosed spaces where other Environment Manipulation techniques are used. This makes it a natural addition to the set of Environmental Manipulation techniques. For example, the game "Tea for God"[3] uses impossible spaces and limited virtual locomotion to create unique indoor spaces for the player to move through. Incorporating *Segment Addition*, wide open outdoor spaces could be

[3] Tea for God, a virtual reality videogame that uses impossible spaces to facilitate redirected walking. Available on Steam: `https://store.steampowered.com/app/1764400/Tea_For_God/` Last Accessed: September 5, 2025

added to the system for the player to navigate as well. The user might enter into the virtual home of a villager that uses moving doorways and expanding rooms to direct the user around the space (Suma et al., 2011). Then when they exit the home into the village square, *Segment Addition* could direct the user to stay within the tracking space as they move between buildings. The user might look around to find their next destination, while *Segment Addition* could use the rotation to direct them towards an open area of the tracking space. As the user enters another building, this would provide an additional opportunity to reset the *Segment Addition* space while the user is not looking.

2. **A Rotation Gain alternative**: The number of slices added or removed to the turn in this user study was equivalent to rotation gain levels between 0.63 and 1.37, which are above the detection threshold described by Steinicke et al., 2010. Simulator Sickness remained similar in both *Segment Addition* and *Moving Goalposts*, with users giving a high comfort rating (RQ4) after each turn ($\overline{x}$ = 5.5/7) and a high mean comfort rating for the system ($\overline{x}$ = 8.5/10). The mean Simulator Sickness score (RQ4) after the first environment was 17.02 and after the second environment 18.06, with no statistically significant differences between *Segment Addition* and *Moving Goalposts*. These scores are lower than the predicted score for reorientation techniques such as rotation and curvature gain (20.61 $\pm$ 1.48) described by Gemert et al., 2024. The scores are slightly higher than their real walking predictions (15.65 $\pm$ 0.98). Thus, *Segment Addition* is a promising alternative techniques to rotation gain for those prone to Simulator Sickness. However, *Segment Addition* needs to be expanded to account for user translation before this potential can be fully realised. Ways to introduce translation gain to *Segment Addition* are discussed in Section 5.6.1.

3. **When Resetting**: In most resetting algorithms, the user is told to turn 360°in the virtual environment while their physical rotation is scaled so that they are facing away from the physical obstacle after the turn. While there are many resetting algorithms (see section 2.1.2 for details), most algorithms use rotation

gains far above the detection threshold or freeze the viewport until the rotation is complete.

Segment Addition could be used as an alternative technique to resetting in such cases. With *Segment Addition*, even when users were aware of the increase in rotation amount, they continued to find the movement natural and the virtual environment comfortable. The sensory mismatch theory of Simulator Sickness suggests that users feel sick when they receive conflicting input signals from their senses – such as their vestibular, proprioceptive and visual systems. Regular gain in virtual reality introduces this sensory mismatch whereas *Segment Addition* does not, leading to potentially lower simulator sickness for those susceptible to it.

A virtual environment could use gains at levels below the detection threshold but when the user resets instead of using rotation gain, *Segment Addition* could be substituted for users susceptible to simulator sickness. This means that the redirected walking method chosen for a virtual environment could accommodate user preferences and needs. It could even adapt automatically to user comfort requirements over time based on the simulator sickness symptoms of users. For example a system such as the one introduced by Wang et al., 2022 could be used to automatically measure user's simulator sickness levels in the environment and find patterns of discomfort for the user and then adapt methods accordingly. In this scenario, to account for the translation movement of the user between resets, the *Segment Addition* technique needs to be expanded as described in section 5.6.1 below.

4. **Comfortable environments that change drastically**: Participants continued to perceive the turn as natural and usable (RQ3), and comfort scores remained high (RQ4) throughout the different rotation levels. In total, 10% of participants mentioned environmental changes related to *Segment Addition* in their comments (RQ1, RQ2), suggesting that while users noticed the change in the size of the rotation, they did not comprehend how the change took place. This is a promising

result, as it suggests many more slices can be added without increasing simulator sickness or breaking the user's sense of naturalness. Unlike rotation gain methods of redirecting the user, results from this user study indicate that *Segment Addition* could be used to quickly change the user's direction by a large amount while the user remains comfortable.

5.6.1 Expanding Segment Addition

The user study described in this chapter is an initial investigation of the *Segment Addition* technique. Within the scope of this study, the aim was to identify how users responded to the technique to estimate its potential, especially in relation to perception and comfort. The results of this initial user study are promising and highlight areas for more detailed investigation.

The current *Segment Addition* method described does not account for user translation as participants rotated repeatedly in place. As a result, the centre of the rotation remained the same throughout the user study and slices could easily be added and removed. However, when the user walks through the environment, additional slices during a rotation are centred around the user which can be at any point in the space depending on where the user is currently located. There are multiple potential implementations, which should be considered in future studies.

- **Fixed Centre Points:** With this method, *Segment Addition* would only trigger when the user is close to a fixed centre point. These fixed centre points can be manually implemented by the developer and scattered throughout the environment. This would allow for the crafting of additional segments based around these fixed centre points.

- **Cells:** The space could be divided into cells, similar to the static environment manipulation techniques described in section 2.3.1. The centre of rotation for the additional slices in the environment would be the centre of the cell, instead of being centred around the user. This would make it easier to generate slices for an environment as the centre of rotation is known in advance when the slices are created. Unlike fixed centre points above, the developer does not manually need to choose ideal spots for the user to turn when creating the software. Instead the size of the cell dictates the centre of rotation.

- **Automating slice addition:** While the slices in this user study in section 5.4 were created by hand, procedural generation (see Cogo et al., 2024; Marwecki and Baudisch, 2018) could be used to create new slices to add to the environment based on a set of predefined rules. In such a scenario, first all the key features in the environment should be labelled as well as a set of rules established for where a new slice can be added. The slice itself could either be chosen from a preset list of slice types or procedurally generated (see Fukaya, Daylamani-Zad, and Agius, 2025 for an overview of procedurally generating objects). The procedural generation could use the textures and features of the surrounding slices as a starting point. For example, a slice with the same ground texture as the two slices on either side of it or adding an additional section to the outside of a building to make it longer and thus align with the building on either side of the new section.

  Machine Learning methods, such as the Segment Anything Model (SAM) (Zhang et al., 2024) could be implemented to find the ideal point at which to split the environment with minimal marking of environmental features by the developer. The SAM method could take a snapshot of what the user sees from their viewport in the virtual environment and then identify key features within the snapshot. It could suggest vertical lines to cut through the environment where there are no key features. Once this area of the environment is outside the field of view of the user, these vertical lines could then be used as a reference point for where to add new slices. Alternatively, segments without key features between two vertical lines could be removed to shorten the turn.

More studies with these potential solutions are needed to determine the efficacy of the extended *Segment Addition* method.

Additionally, the system currently assumes the user is turning toward a known goal. This will be the case in some VR applications, such as when a user is on a guided tour in a virtual museum or walking through a panoramic video (Li and Fan, 2023).

When the goal is known the current *Segment Addition* technique can be implemented. However, to further increase the number of use cases, *Segment Addition* could also be extended to situations where the user does not have a goal in mind or is otherwise free to roam in the virtual space.

In this case, the criteria for adding or removing slices from the environment would instead be regulated similarly to rotation gain — if the user is turning to face towards the centre of the space or towards a more optimal direction, slices would be added so that the user has to turn further in the desired direction. In contrast, if the user is turning in a direction with more physical obstacles, slices would be removed to decrease how much the user is turning. This adaptation of the method would require slices to be shuffled around the user as they turn rather than once they reach their goal. Additional slices could be removed as soon as they are outside the FOV of the user and the remaining slices would be shuffled to compensate for the gaps.

Slices can be easily hidden in a stationary environment, as the next turn starts in the same location as the previous one ended, additional slices can be hidden as soon as they are outside the FOV of the user during the next turn. However, when the user is moving there is a new challenge of how to remove additional slices when the fixed point the slices are turning around has changed between one turn and the next. One solution would be to keep the relative position of the key features in the environment similar and treat any slices without points of interest as interchangeable. In this case, additional slices would not need to be removed if they are inside the user's field of view.

5.6.2 Future Work

While this proof-of-concept implementation shows promising results, more studies are needed to better understand the benefits and limitations of this technique. The following areas are suggested for potential future studies:

- **Introducing different virtual environments**: The virtual environment in the present study contained two specified goals surrounded by a number of trees.

Future work should consider **other environment types where no goal is specified** to the user and there are **more unique features** in the environment, for example a city square or a farmyard. The environmental characteristics in the current study were static. The **addition of dynamic obstacles** such as car traffic or animals to the scene have the potential to be used to distract the user from changes in the environment (Peck, Fuchs, and Whitton, 2009; Williams and Peck, 2019).

- **User Experience under varied conditions**: In the current study, users spent about 15 minutes turning in each of the two environment conditions - *Moving Goalposts* and *Segment Addition*. Future work should analyse the **impact of longer periods spent** with the *Segment Addition* technique when the user is **performing other tasks**. Especially how the shifting environment impacts users' sense of direction and spatial orientation. The system could also be **adjusted to the user**, with the types and size of additional slices being altered depending on the comfort and preferences of individual users, similar to how rotation gain can be adapted (Hutton et al., 2018).

- **Combining Segment Addition with other techniques**: A direct comparison between *Segment Addition* and rotation gain in the same virtual environment could help show the benefits and drawbacks of each technique as the current experiment did not compare *Segment Addition* with a gain based method such as rotation gain. It is also theoretically possible to **combine both *Segment Addition* and rotation gain together** for additional user redirection as the two techniques work in very different ways. Future user studies should focus on how the use of rotation gain in conjunction with *Segment Addition* impacts the user's awareness of the redirection techniques. Additionally, user's orientation ability and their comfort should be investigated.

5.7 Summary

This chapter describes a new redirected walking technique - *Segment Addition* that was first published by Krueger, Markham, and Bierig, 2024. The *Segment Addition* method is described in depth with a presentation of the algorithm to aid future researchers in reproducing the method. An initial implementation of *Segment Addition* was created for this user study. It was compared with an additional method used as a baseline technique for comparison - *Moving Goalposts*. This user study was conducted with 29 participants and considered four key research questions outlined in Section 5.5.

Users noticed the deviation from a half turn as the number of added slices increased (RQ1), however few were able to describe how the environment changed between turns (RQ2). This could be due to a mixture of change blindness and user expectation - user's did not see the change occur, therefore were not aware of it. They also expected the geography of the environment to remain the same and, therefore, did not identify when it changed. Users found the turns remained natural, usable and comfortable even when many slices were added to the turn (RQ3). This suggests that even turns far above the detection threshold for rotation gain could be implemented using the *Segment Addition* technique, without causing a strong increase simulator sickness. *Segment Addition* causes less simulator sickness than similar reorientation techniques such as rotation gain (RQ4). *Segment Addition* aligns different sensory modalities - users see themselves turning the same amount they are actually turning in the physical space, unlike gain-based redirection techniques which introduce a sensory mismatch that is theorised to cause simulator sickness.

Segment Addition keeps users comfortable while drastic changes are made to the virtual environment. It is ideal for virtual environments that contain lots of wide open space such as fields, deserts or natural environments. It can be used alongside other redirected walking techniques or as an alternative to rotation gain for users prone to simulator sickness.

Future studies should focus on adding user translation into the system. Three possible methods for achieving translation are suggested - using fixed points (set points of environment expansion), cell-based division or procedural generation (automated environment segmentation and slice creation). Additionally, a method to expand *Segment Addition* to scenarios where the how much the user will turn is unknown at the beginning is detailed. In addition to these two expansions of *Segment Addition*, future work could also include studies into more varied VEs and combining *Segment Addition* with other redirection techniques such as rotation gain or other environment manipulation techniques.

Chapter 6

Conclusion

This work investigated how redirected walking methods can be effectively implemented in small tracking spaces, with a focus on providing a positive user experience. In relation to this problem, three research questions were considered:

- How effective are existing, generalised redirected walking methods in small tracking spaces?
- Can rotation gain be adapted into redirected walking algorithms more effectively for small tracking spaces?
- Are there alternatives to rotation gain that can make better use of redirected walking in small tracking spaces?

In relation to the first question, the literature review (see Chapter 2) outlined the benefits and drawbacks of existing redirected walking methods. Environment manipulation techniques are effective in small tracking spaces, however the layout of the virtual environment must include plenty of obstacles such as walls and corridors to be effective. In contrast, gain manipulation methods do not restrict the virtual environment. However, gain-based redirection algorithms have been generally tested in either larger tracking spaces or have seen very targetted use in small tracking spaces where the user is restricted to a predetermined linear path. In the first user study, the **user experience of redirected walking algorithms in a small tracking space** of 3.5m ×3.5m (see Chapter 3) was compared. The user study compared existing generalised redirected walking methods where the user could walk freely within the virtual environment and the size of the tracking space was not considered when developing the virtual environment. Generalised redirected walking algorithms were chosen as they could be run on consumer grade hardware without restricting the layout of the virtual environment. Three algorithms were compared - Reset Only (RO) reset users when they were at the edge of the tracking space, Steer-to-Center (S2C) incorporated gain to move the user towards the center of the tracking space and Alignment-Based Redirection Controller (ARC) implemented gain to improve the alignment of the physical space and the virtual environment so that users could avoid bumping into physical obstacles by avoiding virtual obstacles.

Participants preferred the algorithm that redirected them the least (RO), despite it causing more resets than ARC. The performance of S2C was very similar to RO. Additionally, ARC's performance varied depending on the local similarity between the physical and virtual environments. Participants also noted aspects of the hardware (such as the wire on the HMD and tracking space size) and software (the number of resets) having a negative impact on their user experience.

Redirected walking gain below the detection threshold was shown to have minimal benefit in small tracking spaces based on this user study. Developers should consider user comfort with VR and redirection when building applications. Developers could try implementing alternative redirection strategies to generalised redirected walking algorithms in small tracking spaces. For example, using environment manipulation techniques or relying more on resetting as users preferred the simple, overt redirection of resetting. Based on the conclusions of this user study in relation to research question 1 on the effectiveness of redirected walking algorithms in small tracking spaces, alternative methods were sought to improve the efficacy of redirected walking in small tracking spaces.

Based on the results of the first user study, the second research question was formulated which considered how rotation gain could be adapted for use in smaller tracking spaces. In the second **user study of rotation accuracy under different rotation gain conditions** (see chapter 4), participants turned in a complex virtual environment with many visual cues and in a minimal virtual environment with a single visual cue. The direction of the turn had no impact on turn behaviour. In contrast, the number of visual cues in the virtual environment, the gain level and the turn amount all impacted rotation accuracy. Participants only responded to higher levels of gain in the environment with a singular visual cue and were less likely to notice the gain if the singular visual cue environment was presented first. In contrast, participants were affected by gain at lower levels in the environment with lots of visual cues. At the highest gain level in the high cue environment users turned about 20% more virtually than they did when no gain was present.

There was a wide variation in how users responded to gain, similar to other studies in the field (Engel et al., 2008; Steinicke et al., 2010; Schmitz et al., 2018). Participants were divided into four groups based on their style of response – the low response group (16%) ignored the gain, the moderate response group (37%) mildly underestimated the gain in the high cue environment while not being effected by the gain in the single cue environment, the high response group (24%) turned accurately in the high cue environment at all gain levels while not being effected by gain in the single cue environment and the all response group (21%) were effected by the gain under both environment conditions. These four response groups should be considered in future predictive and learning based redirected walking algorithms. Especially when rotation gain levels above the detection threshold are used to increase the effectiveness of the predictive algorithm in small tracking spaces or during resetting. The impact that the number of visual cues in the environment have on user awareness of rotation gain should also be accounted for.

A potential future direction would be to expand the experimental system to measure translation and curvature gain. The impact on turning accuracy and user experience of the different response types to gain while navigating in a virtual environment could then be measured. An open research question is the number of reference points needed in a virtual environment for users to be impacted by the gain and how other environmental features (lighting, visual complexity) impact this perception.

While the results of the previous study show how rotation gain could be adapted to suit smaller tracking spaces, how much rotation gain can be added to a turn is still limited as strong rotation gain can lead to increased simulator sickness. Based on these results, the third and final research question considers alternatives to rotation gain. To this end, **Segment Addition, a novel method for redirected walking in virtual environments** was created (see chapter 5). With Segment Addition, the virtual environment around the user is expanded or contracted as they rotate so that they have to turn a different amount to reach their goal. This allows the system to manipulate the direction the user is facing in the tracking space after the turn. An initial proof-of-concept implementation was built and a user study with the system was conducted to find the efficacy of the method.

Participants in the user study noticed they were deviating from the expected turn as the size of the changes to the virtual environment increased. However, few were able to describe how the environment changed. The turns remained natural, usable and comfortable for participants even when many additions were made to the turn. This suggests that even turns far above the detection threshold for rotation gain could be implemented using the Segment Addition technique without causing a strong increase simulator sickness.

Based on these findings, potential use cases for Segment Addition are suggested. Segment Addition is ideal for wide open virtual environments such as fields, deserts or natural environments as it alters the environment outside the user's field of view. Segment Addition keeps users comfortable while drastic changes are made to the virtual environment. It can be used alongside other redirected walking techniques such as impossible spaces or as an alternative to rotation gain for users prone to simulator sickness.

As the results of the initial user study showed promising results for this new technique, there are many possible avenues for future research related to Segment Addition. In the initial implementation, users turned in place, as such future studies should focus on adding translation to the system. Three methods are suggested for how this could be achieved: fixed points, cell-based division or procedural generation. A method to expand Segment Addition to scenarios where users are turning an unknown amount is also described. Future work could also look at Segment Addition's performance in varied virtual environments and when it is combined with other redirection techniques.

6.0.1 Avenues for Future Work

This section briefly outlines different potential avenues for future work based on the results of the studies described in this thesis.

Movement Accuracy Expansion and Implementation: In Chapter 4 a system for measuring the accuracy of user rotation was designed and implemented. A user study based on this system found disparate user profiles in response to different levels of rotation gain under various environment conditions. While the results of this initial study can be used to more accurately predict user rotation behaviour, other aspects of user locomotion were not considered within the scope of the study.

This system could be expanded to include other types of gain (e.g. translation, curvature, bending) and additional environment conditions (e.g. different visual fidelity levels, lighting, outdoor scenes, scenes with moving obstacles) to create an in-depth test platform for Locomotion Behaviour User Studies. Features such as in-VR survey questions and cross-platform VR support could be used to create in-depth user behaviour profiles. The survey question results could be stored in the log files, which could help gain insight into user experience. Meanwhile, cross-platform VR support would allow for a direct comparison between user behaviour with different VR headsets.

With this expanded testing platform, a number of user studies into Locomotion Behaviour could be run. Once complete, the results of these studies could be integrated into an existing predictive redirected walking algorithm. Alternatively, a new predictive or machine learning algorithm could integrate the results of these user studies to more accurately predict the future locomotion behaviour of users.

Introducing translation into the Segment Addition technique: In Chapter 5 a new redirected walking technique was introduced - Segment Addition. Segment Addition expands the types of virtual spaces that Environment Manipulation techniques can be used in. It uses a novel technique of adding slices to the virtual environment to change the amount the user turns virtually, and thus physically. An initial proof-of-concept implementation of Segment Addition showed that users found the technique natural, usable, and comfortable, even when large changes were made to the virtual environment.

Segment Addition is a promising new technique, however it needs to be expanded upon to be integrated into diverse virtual environments. The system could be expanded to account for user translation, as the current proof-of-concept implementation of Segment Addition does not account for it. Three methods for how this could theoretically be achieved are outlined in 5.6.1. The implementation of Segment Addition within the system could also be optimised using threads to spread the calculation of slice movement evenly across multiple frames. This optimisation could also be used to increase the flexibility of the system by more easily integrating turns that do not have a distinct goal/stopping point.

Once these alterations have been made, a demo system of Segment Addition alongside other Environment Manipulation techniques could be built to show the full capabilities of Segment Addition. An in depth comparison between this demo system and existing redirected walking algorithms such as S2C or ARC could then be run to find the ideal use case for Segment Addition.

Expanding Segmentation Techniques for redirected walking: The segmentation of virtual environments involves breaking the virtual space into parts that can then be better adapted to fit the tracking space of a user. Static methods of segmentation are reviewed in Section 2.3.1 and a new technique that introduces dynamic segmentation, Segment Addition, is introduced in Chapter 5. Future work could consider where or how else dynamic segmentation can be used to improve redirected walking.

For example, where Segment Addition uses additional slices to change the amount the user turns, the slices themselves could also be dynamically altered. Perhaps the width of each slice is stretched by 20% to have the user turn further within the tracking space. To keep this change subtle, the features on the slice should ideally keep the same width and height proportions as before but be moved apart to account for the widened slice.

Alternatively, the virtual environment itself could be procedurally generated out of segments that redirected the user in a desired way. The user might walk from an open area into a small room to allow the outdoor space to reset easily while keeping the user inside the tracking space. Or the user might happen upon an area of interest which includes a new interaction possibility (eg an npc) that happens to lead them towards the centre (or other ideal location) in the tracking space. These new segmentation approaches could then be implemented alongside existing redirected walking techniques to leverage the most room out of every tracking space.

Appendix A

Simulator Sickness Questionnaire

The Simulator Sickness Questionnaire

Participant Number: _____ Pre-Test / After first environment / Post-Test

Are you motion sick now? Circle YES or NO

If you are sick, when did you first notice the symptoms? Time: _________ Date: ______

Circle how much each symptom below is affecting you now.

0 = "not at all" 1 = "mild" 2 = "moderate" 3 = "severe"

1. General discomfort	0	1	2	3
2. Fatigue	0	1	2	3
3. Headache	0	1	2	3
4. Eyestrain	0	1	2	3
5. Difficulty focusing	0	1	2	3
6. Increased salivation	0	1	2	3
7. Sweating	0	1	2	3
8. Nausea	0	1	2	3
9. Difficulty concentrating	0	1	2	3
10. Fullness of head	0	1	2	3
11. Blurred vision	0	1	2	3
12. Dizziness (eyes open)	0	1	2	3
13. Dizziness (eyes closed)	0	1	2	3
14. Vertigo*	0	1	2	3
15. Stomach awareness*	0	1	2	3
16. Burping	0	1	2	3

***Vertigo** is experienced as loss of orientation with respect to vertical upright, **Stomach Awareness** is usually used to indicate a feeling of discomfort that is just short of nausea.

FIGURE A.1: A copy of the Simulator Sickness Questionnaire participants in the Redirected Walking in Small Tracking Spaces (see Chapter 3) and Segment Addition (see Chapter 5) received to complete.

Appendix B

Redirected Walking in Small Tracking Spaces - Participant Questionnaire

Questionnaires

SUS Exit Interview on Alignment Based Redirection
Participant #:
Task:

	Strongly Disagree			Strongly Agree
1. I think that I would like to use such a VR game frequently	1	2	3	4	5
2. I found the VR games unnecessarily complex	1	2	3	4	5
3. I thought the VR game was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to play this VR game	1	2	3	4	5
5. I found the various functions in this VR game were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this VR game	1	2	3	4	5
7. I would imagine that most people would learn to use this VR game very quickly	1	2	3	4	5
8. I found the VR game very cumbersome to use	1	2	3	4	5
9. I felt very confident using the VR game	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this VR game	1	2	3	4	5

Rank the three game levels in order of preference:

Level 1 ____ Level 2 ____ Level 3 ____

Additional Comments:

FIGURE B.1: Participant Questionnaire on Usability and Environment Preference - a copy of the questionnaire participants filled out after completing the three virtual environments containing the ARC, S2C and Reset Only algorithms. The System Usability Scale (SUS) measures how easy users found the system to use.

Appendix C

Redirected Walking in Small Tracking Spaces - Participant Responses

TABLE C.1: Additional Comments from participants at the end of the user study

Participant Comments
open to trying more vr games etc, it was a fun experience
There was a strong sense I could walk into a wall as I did not know where in the real room I was more space to move movement felt mostly consistent, but position of environment and eg the controllers seemed to drift on occasion learned a lot about my ability to orient in environments Spinning feature was interesting, I could not tell I was walking in a different direction.
In level 3 the world did not seem to follow the movement of my head as smoothly as the other levels, it contributed to the dizzy sensation I was a little dizzy on the last one
there was input lag for turning in level 3. Any game requiring spinning in place needs to account for power lead. Felt afraid to move due to the wire as the system didn't estimate that I guess handling the wire during the game was a bit difficult I felt more comfortable with the second environment.
There was no directional arrows but I had right information to move. The third environment has more information about direction but it sometimes showed different direction than where the ball was. I was confused in the 3rd game then. learned a lot about my ability to orient in environments There was a small amount of jitter in the games. Also in some (especially the 2nd turning game in the office) had a little delay or lurch of motion blur that made orientation feel off the last level was stopped a few seconds early due to jitter if less glitches were to occur, I believe it could be a great game to train your reflexes, memory and decision making I enjoyed the last level (S2C) the most as it challenged my perception

Appendix D

Rotation Accuracy - Mean and Standard Error of turns under different rotation conditions

The table D.1 shows the average mean and standard error amount user turned under each of the different rotation conditions.

TABLE D.1: Mean and Standard Error of Rotation at different Gain Levels

Condition		**Real**			
Environment	Turn	Gain			
		1	1.245	1.49	1.98
Minimal	180	163.45±2.9	162.53±2.89	157.68±2.78	150.69±2.22
Complex	180	176.81± 2.94	145.82±2.48	132.85±2.94	110.15±3.18
Open	180	169.4±1.27	-	-	-
Closed	180	166.9±1.36	-	-	-
Minimal	90	93.27±1.57	91.7±1.93	88.22±1.73	76.8±1.73
Complex	90	91.13±2.16	82.9±2.27	73.53±2.45	56±1.47
Open	90	98.71±2.38	-	-	-
Closed	90	93.63±1.53	-	-	-
Minimal	45	56.34±1.71	56.07±2.26	52.57±2.24	47.24±1.98
Complex	45	56.76±2.14	49.66±1.91	42.48±1.51	34.41±1.64
Open	45	56.82±1.65	-	-	-
Closed	45	53.7±1.34	-	-	-
Condition		**Virtual**			
Environment	Turn	Gain			
		1	1.245	1.49	1.98
Minimal	180	163.86±3.6	202.33±3.6	235.6±3.81	298.77±4.4
Complex	180	176.81±2.94	180.41±3.69	197.96±4.38	218.14±6.34
Open	180	-	-	-	-
Closed	180	-	-	-	-
Minimal	90	93.27±1.57	114.15±2.41	130.83±2.52	151.04±3.31
Complex	90	91.13±2.16	103.2±2.82	109.3±3.54	111.24±2.96
Open	90	-	-	-	-
Closed	90	-	-	-	-
Minimal	45	56.34±1.71	69.79±2.82	78.29±3.34	93.49±3.91
Complex	45	56.76±2.14	61.83±2.37	63.31±2.25	68.1±3.23
Open	45	-	-	-	-
Closed	45	-	-	-	-

Appendix E

Rotation Accuracy - Interaction Factors Results using Linear Mixed Models

The table E.1 shows the statistical significance of the interaction effects between the different factors described in the turning accuracy user study. The factors and full model are described in section 4.4.5.

The table E.1 compares the accuracy of the full model to a set of smaller models that contain only the factors listed in the 'Effect' column. The 'Constant Factor' column describes when a single factor was held constant (ie only took one value from its set of possible values) to test if interaction effects existed under that specific factor condition. The 'npar' column describes the difference in the number of parameters between the full model and the smaller model being compared. The 'LRT' column describes the result of a likelihood ratio test between the two models. The 'Pr(Chi)' describes the test statistic of the Likelihood Ratio Test which follows a Chi distribution. The * show statistical significance with *= .05, **=.01, ***=.001.

TABLE E.1: Statistical Significance of Interaction Effects between the Factors in the User Study

Effect	Constant Factor	npar	LRT	Pr(Chi)	*
R_turn:Gain:Env		6	59.383	6.01E-11	***
R_turn:Gain		6	73.271	8.71E-14	***
R_turn:Env		2	22.165	1.54E-05	***
gain:Env		3	140.399	<2E-16	***
Direction		1	0.51	0.4768	
R_turn		2	768.84	<2E-16	***
Gain		3	419.67	<2E-16	***
Env		1	169.57	<2E-16	***
R_turn:Env	Gain 1	2	14.2995	7.85E-04	***
R_turn:Env	Gain 1.245	2	6.6083	0.03673	*
R_turn:Env	Gain 1.49	2	14.2741	0.0007951	***
R_turn:Env	Gain 1.98	2	52.43	4.12E-12	***
R_turn:Gain	Env Complex	6	128.313	<2E-16	***
R_turn:Gain	Env Minimal	6	4.6182	0.5936	
Gain:Env	R_turn 45	3	18.7725	0.0003047	***
Gain:Env	R_turn 90	3	29.9532	1.41E-06	***
Gain:Env	R_turn 180	3	101.67	<2E-16	***

Appendix F

Rotation Accuracy - Significant Wilcoxon Test Results

The table below shows the results of Wilcoxon tests using the Bonferroni correction to compare if two conditions were statistically significant after a positive Friedman test. Note that within the graph an 'X' marks a test condition as not being run either because it is being compared to itself (eg Complex/Complex) or a version of it was already run (eg Complex/Closed is empty because Closed/Complex was run).

TABLE F.1: A table presenting the results of Wilcoxon comparison tests when a Friedman test showed there were significant differences between some elements of the group.

Turn Amount	Gain	Environment	Minimal	Complex	Closed
45	1.245	Complex	0.363	X	X
45	1.245	Closed	1	0.037	X
45	1.245	Open	1	0.034	1
45	1.49	Complex	0.03549	X	X
45	1.49	Closed	1	0.00011	X
45	1.49	Open	0.54048	2.3e-05	1
45	1.98	Complex	0.00029	X	X
45	1.98	Closed	0.04047	8.4e-08	X
45	1.98	Open	0.01359	3.4e-08	1
90	1.245	Complex	0.0881	X	X
90	1.245	Closed	1	0.0067	X
90	1.245	Open	0.6496	0.0015	1
90	1.49	Complex	0.00012	X	X
90	1.49	Closed	0.15318	1.5e-06	X
90	1.49	Open	0.02906	5.3e-07	1
90	1.98	Complex	4.2e-10	X	X
90	1.98	Closed	5.4e-08	1.9e-12	X
90	1.98	Open	1.6e-08	1.8e-12	1
180	1	Complex	0.0355	X	X
180	1	Closed	1	0.0012	X
180	1	Open	1	0.0105	1
180	1.245	Complex	0.0016	X	X
180	1.245	Closed	1	1.5e-07	X
180	1.245	Open	0.5433	3.5e-08	1
180	1.49	Complex	6.7e-08	X	X
180	1.49	Closed	0.0217	7.3e-09	X
180	1.49	Open	0.0012	3.8e-09	1
180	1.98	Complex	9.0e-09	X	X
180	1.98	Closed	1.1e-05	4.5e-10	X
180	1.98	Open	1.5e-06	1.6e-10	1

Appendix G

Rotation Accuracy - Participant Questionnaire

Questionnaire between task 1 (b)

1. Did you notice of any of following:

a.	Additional objects appearing in the environment	Yes	No	Not sure
b.	Changes in how bright it is	Yes	No	Not sure
c.	The world moving	Yes	No	Not sure
d.	The ground tilting towards away/from you	Yes	No	Not sure
e.	Turning more/less than you thought	Yes	No	Not sure
f.	Objects disappearing that were once there	Yes	No	Not sure
g.	Objects changing colour	Yes	No	Not sure

2. Is there anything else you'd like to note:

FIGURE G.1: Participant Questionnaire on Gain Perception - a copy of the questionnaire participants filled out after completing each of the two virtual environment conditions, Minimal and Complex. Participants were asked the questions orally while the experimenter filled in the responses.

Appendix H

Segment Addition - Participant Questionnaire

General Comments on the Experiment System

Participant Number: ____

Age: 18-25   26-35   36-45   46-55   56-65

Previous VR Experience:
None
Have used VR once or twice
Have used VR a few times
Use VR regularly/own my own headset

Which of the two environments did you prefer?
The first   The second

Explain your preference:

Did you notice anything unusual in either of the virtual environments?

Is there anything you would like to add about your experience with the experiment system?

FIGURE H.1: Participant Questionnaire after the participant had completed the Segment Addition and Moving Goalposts environments - a copy of the questionnaire participants filled out after completing each of the two virtual environment conditions.

Appendix I

Segment Addition - Complete List of Comments

A complete list of comments from participants, sorted by the question they were responding to and roughly by topic. One participant's comments have been bolded, and another italicised - in each case this is to show it is the same participant across all three responses.

TABLE I.1: Segment Addition - Additional Comments

Pref	Order	Comment
-	-	good first experience with vr
-	-	It was interesting
SA	2	I enjoyed the experience of being a participant
MG	1	it was enjoyable and I never felt really uncomfortable
SA	1	it was fun
SA	1	nothing in particular other than that I like the idea of VR games and immersing yourself in an environment like that. Thanks for the opportunity.
SA	2	this was fun!
MG	2	I am prone to migraines and have noticed mild eye strain and headache after the second use.
SA	1	the weight of the vr headset was noticeable, there were some frame rate issue for a second and some pixel issues with the trees.
SA	2	**While I did not feel particularly ill after the experiment, I think I might have had a few more symptoms if it had gone on longer, especially based on previous VR experiences.**
MG	1	I enjoyed and interacted more comfortable when I started but later headset and many turns made it a bit of a hard thing to keep doing.
SA	2	*I paid particular attention to the trees in the second environment which I did not do in the first. I did not get the impression that the trees were moving however I could see the landmarks didn't move relative to the trees which confused me. Furthermore towards the end of the second experiment I believe the landmark shifted slightly in front of me and at the same time I thought I saw a gap in the trees but I am unclear of the details of what I saw.*
SA	2	It's difficult to continuously turn on the spot without translating, especially when vision is obscured
MG	2	tended to prefer larger or similar turns, disliked turns that felt smaller

TABLE I.2: Segment Addition - Preferred Environment and Why Part 1

Pref	Order	Comment
-	-	-
-	-	No Preference
-	-	didn't notice a difference from an environment point of view
-	-	No preference, thought they were both the same
MG	1	No real strong preference either way
MG	2	having never used VR before I felt more comfortable in the second one (Moving Goalposts). Better understanding of what was going on. The navigation felt more natural and comfortable in the second.
MG	2	more used to what I was doing and the environments
MG	2	it felt smoother but I may just have gotten used to it from the first environment
MG	2	was more used to the experiment and it was quicker as a result
MG	1	first (Moving Goalposts), because it was the begin. I had curiosity to be in the environment and do interaction. Once I started to feel heavy on the head and became used to the environment, it was more like the routine to do same or at least felt the same.
SA	1	because by the time we are doing the second (Moving Goalposts) the hands are tired from holding controllers.
SA	1	with the environment, both were the same to me but after standing and moving a while, it was a lot more difficult to maintain myself. Starting off fresh with the first (Segment Addition), there was a desire to explore a bit when turning.
SA	2	although the environments seemed the same, I wasn't wearing my glasses inside the headset. Not wearing my glasses made the experience more pleasant.
MG	1	I was more 'comfortable' the first time (Moving Goalposts), the second time was weirder, I felt that something was off.
MG	1	haven't noticed too much of a difference but there were occasional uncomfortable turns in both, more frequent in the 2nd (Segment Addition)
SA	1	look like more open, with vision field bigger
SA	1	felt better and more orientated, didn't end abruptly
SA	1	It (Segment Addition) seemed more responsive. When I turned the menu box would appear immediately. In the 2nd (Moving Goalposts) environment it took a few seconds to appear on some turns.
MG	2	it felt more natural
MG	1	turns felt less jarring
MG	2	the turning in the second environment (Moving Goalposts) was better
SA	2	The second (Segment Addition) felt more predictable for moving, maybe because it felt smoother, but difference was subtle.
SA	2	it (Segment Addition) felt slightly smoother, the headset was also less blurry
SA	2	it (Segment Addition) felt smoother to navigate

TABLE I.3: Segment Addition - Preferred Environment and Why Part 2

Pref	Order	Comment
SA	2	There were more times at the end it felt I turned exactly 180 degrees
SA	2	the turns seemed more true to 180 degrees on average
SA	1	I felt like I had to turn less. When I did turn, it was generally 180 degrees or less
SA	2	**I didn't notice major differences, but the second (Segment Addition) appeared to have changes in scenery on occasion that grabbed my attention**
SA	2	*The second environment (Segment Addition) was more interesting. I enjoyed the first (Moving Goalposts) but the repetition of it made it less interesting. After the first quarter/third I felt I understood what was happening. The second one (Segment Addition) on the other hand is not same.*

TABLE I.4: Segment Addition - Did you notice anything unusual in the virtual environments? Part 1

Pref	Order	Comment
-	-	No
-	-	no
-	-	no nothing
MG	2	no
MG	1	no
MG	2	no
MG	1	No
SA	1	no
SA	1	nope, just some blue box and pink cylinders
SA	2	Nope (maybe horizontal line as if a mirror - probs from lenses)
SA	1	-
-	-	menus didn't always appear
MG	2	a grid appeared briefly in the first environment (Segment Addition) note: this refers to the boundary grid shown on the meta quest when the user is near the edge of the tracking area)
MG	2	the size of the blue + the pink shapes changed.
MG	1	no, but maybe the questions box (note: menu) sometime look bigger, the second time (Segment Addition) the first time (Moving Goalposts) no.
SA	1	Nothing in particular apart from the trees and their vertices. While for the most part they seemed mono-coloured and shading, some trees had some patterns on a particular face.
SA	1	Nothing out of the ordinary except the hands were of different shapes. The first (Segment Addition) was a spherical/hockey shaped the second was cylinder like (Moving Goalposts)
SA	2	white outlines on trees in the second experiment (Segment Addition)

TABLE I.5: Segment Addition - Did you notice anything unusual in the virtual environments? Part 2

Pref	Order	Comment
MG	1	some turns it felt choppy, or as if the trees were slightly teleporting
MG	1	In the 2nd one (Segment Addition), some turns were more bigger than in the 1st one (Moving Goalposts). This is how I felt.
MG	2	1.when I first put the headset on in the second there seemed to be a glitch where the second/middle frame didn't move but the frames above and below did. This resolved after reset, 2. there were a few times when the block seemed further than I was expecting it to be and I felt a little surprised at the distance/turn/
SA	1	The shapes became curved in both on some turns
SA	2	In the first (Moving Goalposts), the forest was causing some disorientation
SA	2	The trees seemed to get bigger towards the end of the second environment (Segment Addition). A tree I hadn't noticed before seemed to appear.
SA	2	At one stage in environment two (Segment Addition), I noticed some of the trees were replaced by different trees
SA	2	I think the trees may have been turning differently to the blocks
SA	2	*In the first one (Moving Goalposts) the distance between the landmarks seemed to change slightly each time although it took a while for my brain to process this. In the second one (Segment Addition), I has a similar impression although to a lesser extent where I was much less sure the "landmark" was moving.*
SA	2	**In the second one (Segment Addition), the pattern of the trees seemed to change. If the first one (Moving Goalposts) did so, the second did it much more regularly**
MG	2	In the second environment (Moving Goalposts) I noticed what I thought was the landmark moving relative to the environment which I hadn't noticed in the first (Segment Addition).

Bibliography

Abich, Julian et al. (Jan. 2021). "A review of the evidence for training effectiveness with virtual reality technology". In: *Virtual Reality* 25 (4), 919–933. DOI: 10.1007/s10055-020-00498-8.

Abtahi, Parastoo et al. (May 2019). "I'm a giant: Walking in large virtual environments at high speed gains". In: *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery. ISBN: 9781450359702. DOI: 10.1145/3290605.3300752.

Al Zayer, Majed, Paul MacNeilage, and Eelke Folmer (2020). "Virtual Locomotion: A Survey". In: *IEEE Transactions on Visualization and Computer Graphics* 26.6, pp. 2315–2334. DOI: 10.1109/TVCG.2018.2887379.

Avola, D. et al. (2023). "A novel low cybersickness dynamic rotation gain enhancer based on spatial position and orientation in virtual environments". In: *Springer Virtual Reality* 27, 1009–1017. DOI: doi.org/10.1007/s10055-023-00865-1.

Azmandian, Mahdi et al. (2015). "Physical Space Requirements for Redirected Walking: How Size and Shape Affect Performance". In: *International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments, ICAT-EGVE 2015*, pp. 93–100. DOI: 10.2312/egve.20151315.

— (Apr. 2016a). "Automated path prediction for redirected walking using navigation meshes". In: *2016 IEEE Symposium on 3D User Interfaces, 3DUI 2016 - Proceedings*, pp. 63–66. ISBN: 9781509008421. DOI: 10.1109/3DUI.2016.7460032.

— (2016b). "The redirected walking toolkit: a unified development platform for exploring large virtual environments". In: *2016 IEEE 2nd Workshop on Everyday Virtual Reality (WEVR)*, pp. 9–14. DOI: 10.1109/WEVR.2016.7859537.

Azmandian, Mahdi et al. (May 2022a). "Adaptive Redirection: A Context-Aware Redirected Walking Meta-Strategy". In: *IEEE Transactions on Visualization and Computer Graphics* 28 (5), pp. 2277–2287. ISSN: 19410506. DOI: 10.1109/TVCG.2022.3150500.

Azmandian, Mahdi et al. (May 2022b). "Validating Simulation-Based Evaluation of Redirected Walking Systems". In: *IEEE Transactions on Visualization and Computer Graphics* 28 (5), pp. 2288–2298. ISSN: 19410506. DOI: 10.1109/TVCG.2022.3150466.

Bachmann, Eric R. et al. (May 2019). "Multi-User Redirected Walking and Resetting Using Artificial Potential Fields". In: *IEEE Transactions on Visualization and Computer Graphics* 25 (5), pp. 2022–2031. ISSN: 19410506. DOI: 10.1109/TVCG.2019.2898764.

Bao, Xiyu et al. (2024). "Dynamic Translation Gain in Redirected Walking: Towards Exploring Larger Virtual Spaces". In: *2024 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 473–474. DOI: 10.1109/ISMAR-Adjunct64951.2024.00137.

Bayramova, R. et al. (2021). "The role of vision and proprioception in self-motion encoding: An immersive virtual reality study." In: *Springer Attention, Perception and Psychophysics* 83, pp. 2865–2878. DOI: 10.3758/s13414-021-02344-8.

Beck, J, M. Rainoldi, and R. Egger (2019). "Virtual reality in tourism: a state-of-the-art review". In: *Tourism Review* 74, pp. 586–612. DOI: https://doi.org/10.1108/TR-03-2017-0049.

Bimberg, Pauline, Tim Weissker, and Alexander Kulik (2020). "On the Usage of the Simulator Sickness Questionnaire for Virtual Reality Research". In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 464–467. DOI: 10.1109/VRW50115.2020.00098.

Bozgeyikli, Evren et al. (Feb. 2019). "Locomotion in virtual reality for room scale tracked areas". In: *International Journal of Human Computer Studies* 122, pp. 38–49. ISSN: 10959300. DOI: 10.1016/j.ijhcs.2018.08.002.

Brooke, John (1996). "SUS: A 'Quick and Dirty' Usability Scale". In: *Usability Evaluation In Industry*. Vol. 189. London, UK: Taylor & Francis, 189–194. ISBN: 9780429157011.

Bruder, Gerd, Paul Lubos, and Frank Steinicke (2015). “Cognitive Resource Demands of Redirected Walking”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.4, pp. 539–544. DOI: 10.1109/TVCG.2015.2391864.

Bruder, Gerd et al. (2009). “Reorientation during body turns”. In: *Proceedings of the 15th Joint Virtual Reality Eurographics Conference on Virtual Environments*. JVRC’09. Lyon, France: Eurographics Association, 145–152. ISBN: 9783905674200.

Bruder, Gerd et al. (2012). “Redirecting Walking and Driving for Natural Navigation in Immersive Virtual Environments”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.4, pp. 538–545. DOI: 10.1109/TVCG.2012.55.

Brument, Hugo et al. (Nov. 2021). “Studying the Influence of Translational and Rotational Motion on the Perception of Rotation Gains in Virtual Environments”. In: *Proceedings - SUI 2021: ACM Spatial User Interaction 2021*. Association for Computing Machinery, Inc. ISBN: 9781450390910. DOI: 10.1145/3485279.3485282.

Bölling, Luke et al. (May 2019). “Shrinking Circles: Adaptation to Increased Curvature Gain in Redirected Walking”. In: *IEEE Transactions on Visualization and Computer Graphics* 25 (5), pp. 2032–2039. ISSN: 19410506. DOI: 10.1109/TVCG.2019.2899228.

Cao, Antong et al. (2020). “Feature Guided Path Redirection for VR Navigation”. In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 137–145. DOI: 10.1109/VR46266.2020.00032.

Chang, Eunhee, Hyun Taek Kim, and Byounghyun Yoo (2020). “Virtual Reality Sickness: A Review of Causes and Measurements”. In: *International Journal of Human-Computer Interaction*, pp. 1658–1682. ISSN: 15327590. DOI: 10.1080/10447318.2020.1778351.

Chang, Yuchen et al. (2021). “Redirection Controller Using Reinforcement Learning”. In: *IEEE Access* 9, pp. 145083–145097. ISSN: 21693536. DOI: 10.1109/ACCESS.2021.3118056.

Chen, Jun-Jie et al. (2024). “APF-S2T: Steering to Target Redirection Walking Based on Artificial Potential Fields”. In: *IEEE Transactions on Visualization and Computer Graphics* 30.5, pp. 2464–2473. DOI: 10.1109/TVCG.2024.3372052.

Cho, Yong-Hun, Dong-Yong Lee, and In-Kwon Lee (2018). "Path Prediction Using LSTM Network for Redirected Walking". In: *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 527–528. DOI: 10.1109/VR.2018.8446442.

Christofi, Maria and Despina Michael-Grigoriou (2017). "Virtual reality for inducing empathy and reducing prejudice towards stigmatized groups: A survey". In: *2017 23rd International Conference on Virtual System and Multimedia VSMM*, pp. 1–8. DOI: 10.1109/VSMM.2017.8346252.

Ciumedean, Claudiu et al. (Mar. 2021). "Impossible open spaces: Exploring the effects of occlusion on the noticeability of self-overlapping virtual environments". In: *Proceedings - 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops, VRW 2021*. Institute of Electrical and Electronics Engineers Inc., pp. 389–390. ISBN: 9780738113678. DOI: 10.1109/VRW52623.2021.00081.

Ciumedean, Claudiu-Bogdan et al. (2020). "Mission Impossible Spaces: Using Challenge-Based Distractors to Reduce Noticeability of Self-Overlapping Virtual Architecture". In: *Proceedings of the 2020 ACM Symposium on Spatial User Interaction*. SUI '20. Virtual Event, Canada: Association for Computing Machinery. ISBN: 9781450379434. DOI: 10.1145/3385959.3418453.

Coelho, Daniel Neves, Frank Steinicke, and Eike Langbehn (2022). "Design of Mentally and Physically Demanding Tasks as Distractors of Rotation Gains". In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 620–621. DOI: 10.1109/VRW55335.2022.00161.

Cogo, Emir et al. (2024). "A Survey of Procedural Modelling Methods for Layout Generation of Virtual Scenes". In: *Computer Graphics Forum* 43.1, e14989. DOI: 10.1111/cgf.14989.

Cohen, Jacob (1988). *Statistical Power Analysis for the Behavioral Sciences (2nd ed.)* Routledge. ISBN: 9780203771587. DOI: 10.4324/9780203771587.

Congdon, Ben J. and Anthony Steed (Nov. 2019). "Sensitivity to rate of change in gains applied by redirected walking". In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*. Association for Computing Machinery. ISBN: 9781450370011. DOI: 10.1145/3359996.3364277.

Cools, Robbe and Adalberto L. Simeone (Oct. 2019). "Investigating the effect of distractor interactivity for redirected walking in virtual reality". In: *Proceedings - SUI 2019: ACM Conference on Spatial User Interaction*. Association for Computing Machinery, Inc. ISBN: 9781450369756. DOI: 10.1145/3357251.3357580.

Danyluk, Kurtis et al. (2021). "A Design Space Exploration of Worlds in Miniature". In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. Yokohama, Japan: Association for Computing Machinery. ISBN: 9781450380966. DOI: 10.1145/3411764.3445098.

Dong, Tianyang et al. (2020). "Dynamic Artificial Potential Fields for Multi-User Redirected Walking". In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 146–154. DOI: 10.1109/VR46266.2020.00033.

Dong, Tianyang et al. (May 2023). "FREE-RDW: A Multi-user Redirected Walking Method for Supporting Non-forward Steps". In: *IEEE Transactions on Visualization and Computer Graphics*. ISSN: 19410506. DOI: 10.1109/TVCG.2023.3247107.

Dong, Zhi Chao et al. (Nov. 2017). "Smooth assembled mappings for large-scale real walking". In: *ACM Transactions on Graphics* 36 (6). ISSN: 15577368. DOI: 10.1145/3130800.3130893.

Dong, Zhi Chao et al. (Oct. 2021). "Tailored Reality: Perception-aware Scene Restructuring for Adaptive VR Navigation". In: *ACM Transactions on Graphics* 40 (5). ISSN: 15577368. DOI: 10.1145/3470847.

Eklund, Vendela (2022). *Maximizing the VR Play Space by Using Procedurally Generated Impossible Spaces : Research on VR Play Spaces and Their Impact on Game Development*. Dissertation Blekinge Institute of Technology Faculty of Computing Department of Computer Science.

Engel, David et al. (2008). "A Psychophysically Calibrated Controller for Navigating through Large Environments in a Limited Free-Walking Space". In: *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology*. VRST '08. Bordeaux, France: Association for Computing Machinery, 157–164. ISBN: 9781595939517. DOI: 10.1145/1450579.1450612.

Fan, Cheng-Wei et al. (2023). “Redirected Walking Based on Historical User Walking Data”. In: *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 53–62. DOI: 10.1109/VR55154.2023.00021.

Fukaya, Kaisei, Damon Daylamani-Zad, and Harry Agius (Jan. 2025). “Intelligent Generation of Graphical Game Assets: A Conceptual Framework and Systematic Review of the State of the Art”. In: *ACM Computing Surveys* 57.5, 1–38. ISSN: 1557-7341. DOI: 10.1145/3708499.

Gałecki, Andrzej and Tomasz Burzykowski (2013). “Linear Mixed-Effects Model”. In: *Linear Mixed-Effects Models Using R: A Step-by-Step Approach*. New York, NY: Springer New York, pp. 245–273. ISBN: 978-1-4614-3900-4. DOI: 10.1007/978-1-4614-3900-4_13.

Gan, Qi Wen et al. (2024). “Exploring the Impact of Visual Scene Characteristics and Adaptation Effects on Rotation Gain Perception in VR”. In: *Proceedings of the 30th ACM Symposium on Virtual Reality Software and Technology*. VRST ’24. Trier, Germany: Association for Computing Machinery. ISBN: 9798400705359. DOI: 10.1145/3641825.3687733.

Gao, Yuze et al. (2022). “Redirected Walking for Virtual Environments: Investigation and Evaluation”. In: *International Conference on Virtual Rehabilitation, ICVR*. Vol. 2022-May. Institute of Electrical and Electronics Engineers Inc., pp. 147–154. ISBN: 9781665479110. DOI: 10.1109/ICVR55215.2022.9847874.

Gemert, Thomas van et al. (2024). “Sicknificant Steps: A Systematic Review and Meta-analysis of VR Sickness in Walking-based Locomotion for Virtual Reality”. In: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. CHI ’24. Honolulu, HI, USA: Association for Computing Machinery. ISBN: 9798400703300. DOI: 10.1145/3613904.3641974.

Goldfeather, Jack and Victoria Interrante (2012). “Adaptive redirected walking in a virtual world”. In: *2012 IEEE VR Workshop on Perceptual Illusions in Virtual Environments, PIVE 2012*, pp. 17–20. ISBN: 9781467312172. DOI: 10.1109/PIVE.2012.6229795.

Grechkin, Timofey et al. (July 2016). “Revisiting detection thresholds for redirected walking: Combining translation and curvature gains”. In: *Proceedings of the ACM Symposium on Applied Perception, SAP 2016*. Association for Computing Machinery, Inc, pp. 113–120. ISBN: 9781450343831. DOI: 10.1145/2931002.2931018.

Green, David M. (1993). “A maximum-likelihood method for estimating thresholds in a yes-no task”. In: *Journal of the Acoustical Society of America* 93 (4), pp. 2096–2105. ISSN: NA. DOI: 10.1121/1.406696.

Han, Jihae, Andrew Vande Moere, and Adalberto L. Simeone (2022). “Foldable Spaces: An Overt Redirection Approach for Natural Walking in Virtual Reality”. In: *Proceedings - 2022 IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 167–175. ISBN: 9781665496179. DOI: 10.1109/VR51125.2022.00035.

Han, Jihae, Andrew Vande Moere, and Adalberto L. Simeone (2023). “Architectural Narrative VR: Towards Generatively Designing Natural Walkable Spaces”. In: *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. DIS ’23. Pittsburgh PA USA: Association for Computing Machinery, 523–536. ISBN: 9781450398930. DOI: 10.1145/3563657.3596008.

Hildebrandt, Julian et al. (2018). “Get Well Soon! Human Factors’ Influence on Cybersickness After Redirected Walking Exposure in Virtual Reality”. In: *Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation. VAMR 2018. Lecture Notes in Computer Science*. Cham: Springer International Publishing, pp. 82–101. ISBN: 978-3-319-91581-4.

Hirt, Christian et al. (2022a). “Short-term Path Prediction for Spontaneous Human Locomotion in Arbitrary Virtual Spaces”. In: *Proceedings - 2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct, ISMAR-Adjunct 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 554–559. ISBN: 9781665453653. DOI: 10.1109/ISMAR-Adjunct57072.2022.00116.

Hirt, Christian et al. (2022b). “The Chaotic Behavior of Redirection - Revisiting Simulations in Redirected Walking”. In: *Proceedings - 2022 IEEE Conference on Virtual*

Reality and 3D User Interfaces, VR 2022, pp. 524–533. DOI: 10.1109/VR51125.2022.00072.

Hodgson, Eric and Eric Bachmann (Apr. 2013). "Comparing four approaches to generalized redirected walking: Simulation and live user data". In: *IEEE Transactions on Visualization and Computer Graphics* 19 (4), pp. 634–643. ISSN: 10772626. DOI: 10.1109/TVCG.2013.28.

Hodgson, Eric, Eric Bachmann, and Tyler Thrash (2014). "Performance of redirected walking algorithms in a constrained virtual world". In: *IEEE Transactions on Visualization and Computer Graphics* 20 (4), pp. 579–587. ISSN: 10772626. DOI: 10.1109/TVCG.2014.34.

Hodgson, Eric, Eric Bachmann, and David Waller (Nov. 2011). "Redirected walking to explore virtual environments: Assessing the potential for spatial interference". In: *ACM Transactions on Applied Perception* 8 (4). ISSN: 15443558. DOI: 10.1145/2043603.2043604.

Homami, Helia, Adria Quigley, and Mayra Donaji Barrera Machuca (2025). "Omnidirectional VR Treadmills Walking Techniques: Comparing Walking-in-Place and Sliding vs Natural Walking". In: *2025 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 634–644. DOI: 10.1109/VR59515.2025.00086.

Hoshikawa, Yukai et al. (2022). "RedirectedDoors: Redirection while Opening Doors in Virtual Reality". In: *Proceedings - 2022 IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 464–473. ISBN: 9781665496179. DOI: 10.1109/VR51125.2022.00066.

Howard, I. P. et al. (1986). *The Vestibular System in Handbook of Perception and Human Performance (Vol. 1).* New York: Wiley- Interscience., pp. 107, 109, 137, 164.

Hutton, C. et al. (2018). "Individualized Calibration of Rotation Gain Thresholds for RedirectedWalking". In: *ICAT-EGVE 2018 - 28th International Conference on Artificial Reality and Telexistence and 23rd Eurographics Symposium on Virtual Environments*. The Eurographics Association, pp. 61–64. ISBN: 9783038680581. DOI: 10.2312/egve.20181315.

Interrante, Victoria, Brian Ries, and Lee Anderson (2007). "Seven League Boots: A New Metaphor for Augmented Locomotion through Moderately Large Scale Immersive Virtual Environments". In: *2007 IEEE Symposium on 3D User Interfaces*. DOI: 10.1109/3DUI.2007.340791.

Iqbal, Arham I. et al. (2024). "Immersive Technologies in Healthcare: An In-Depth Exploration of Virtual Reality and Augmented Reality in Enhancing Patient Care, Medical Education, and Training Paradigms". In: *Journal of Primary Care & Community Health* 15. PMID: 39439304. DOI: 10.1177/21501319241293311.

Jeon, Sang-Bin et al. (2024). "F-RDW: Redirected Walking With Forecasting Future Position". In: *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15. DOI: 10.1109/TVCG.2024.3376080.

Jerald, Jason (2015). *The VR Book: Human-Centered Design for Virtual Reality*. Association for Computing Machinery and Morgan & Claypool. ISBN: 9781970001129.

Jing, Rui et al. (2024). "The Role of the Field Dependence-Independence Construct on the Curvature Gain of Redirected Walking Technology in Virtual Reality". In: *Advances in Computer Graphics*. Ed. by Bin Sheng et al. Cham: Springer Nature Switzerland, pp. 364–375. ISBN: 978-3-031-50075-6.

Jürgens, R. and W. Becker (2011). "Human spatial orientation in non-stationary environments: relation between self-turning perception and detection of surround motion". In: *Springer Experential Brain Research* 215, 327–344. DOI: 10.1007/s00221-011-2900-z.

Kennedy, Robert S. et al. (1993). "Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness". In: *The International Journal of Aviation Psychology* 3.3, pp. 203–220. DOI: 10.1207/s15327108ijap0303_3.

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

Kim, Dooyoung and Woontack Woo (2023). "Edge-Centric Space Rescaling with Redirected Walking for Dissimilar Physical-Virtual Space Registration". In: *2023 IEEE*

International Symposium on Mixed and Augmented Reality (ISMAR), pp. 829–838. DOI: 10.1109/ISMAR59233.2023.00098.

Kim, Dooyoung et al. (2022). "Effects of Virtual Room Size and Objects on Relative Translation Gain Thresholds in Redirected Walking". In: *Proceedings - 2022 IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 379–388. ISBN: 9781665496179. DOI: 10.1109/VR51125.2022.00057.

Krueger, Linda, Charles Markham, and Ralf Bierig (2024). "Comparison of Two Novel Environmental Manipulation Methods for Rotating VR Users". In: *2024 10th International Conference on Virtual Reality (ICVR)*, pp. 362–368. DOI: 10.1109/ICVR62393.2024.10868877.

Kruse, Lucie, Eike Langbehn, and Frank Steinicke (2018). "I Can See on My Feet While Walking: Sensitivity to Translation Gains with Visible Feet". In: *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 305–312. DOI: 10.1109/VR.2018.8446216.

Kundu, Ripan Kumar et al. (Mar. 2023). "LiteVR: Interpretable and Lightweight Cybersickness Detection using Explainable AI". In: *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. IEEE, pp. 609–619. ISBN: 979-8-3503-4815-6. DOI: 10.1109/VR55154.2023.00076.

Kwon, Soon Uk et al. (2022). "Infinite Virtual Space Exploration Using Space Tiling and Perceivable Reset at Fixed Positions". In: *Proceedings - 2022 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 758–767. ISBN: 9781665453257. DOI: 10.1109/ISMAR55827.2022.00094.

Lampropoulos, Georgios and Kinshuk (June 2024). "Virtual reality and gamification in education: a systematic review". In: *Educational technology research and development* 72 (3), 1691–1785. DOI: 10.1007/s11423-024-10351-3.

Langbehn, Eike, Paul Lubos, and Frank Steinicke (Apr. 2018a). "Evaluation of locomotion techniques for room-scale VR: Joystick, teleportation, and redirected walking". In: *ACM International Conference Proceeding Series*. Association for Computing Machinery. ISBN: 9781450353816. DOI: 10.1145/3234253.3234291.

— (Aug. 2018b). "Redirected Spaces: Going beyond Borders". In: *25th IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2018 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 767–768. ISBN: 9781538633656. DOI: 10.1109/VR.2018.8446167.

Langbehn, Eike and Frank Steinicke (July 2019). "Space walk: A combination of subtle redirected walking techniques integrated with gameplay and narration". In: *ACM SIGGRAPH 2019 Emerging Technologies, SIGGRAPH 2019*. Association for Computing Machinery, Inc. ISBN: 9781450363082. DOI: 10.1145/3305367.3327976.

Langbehn, Eike et al. (Apr. 2017). "Bending the Curve: Sensitivity to Bending of Curved Paths and Application in Room-Scale VR". In: *IEEE Transactions on Visualization and Computer Graphics* 23 (4), pp. 1389–1398. ISSN: 19410506. DOI: 10.1109/TVCG.2017.2657220.

Lee, Chang Gyu and Ohung Kwon (2023). "Identification of the difference threshold for curvature gain of redirected walking". In: *Virtual Reality*. ISSN: 14349957. DOI: 10.1007/s10055-023-00763-6.

Lee, Dong-Yong, Yong-Hun Cho, and In-Kwon Lee (2019). "Real-time Optimal Planning for Redirected Walking Using Deep Q-Learning". In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 63–71. DOI: 10.1109/VR.2019.8798121.

Lee, Dong-Yong et al. (2020). "Optimal Planning for Redirected Walking Based on Reinforcement Learning in Multi-user Environment with Irregularly Shaped Physical Space". In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 155–163. DOI: 10.1109/VR46266.2020.00034.

Lee, Ho Jung, Hyunjeong Kim, and In-Kwon Lee (2025). "Multimodal Turn in Place: A Comparative Analysis of Visual and Auditory Reset UIs in Redirected Walking".

In: *IEEE Transactions on Visualization and Computer Graphics* 31.5, pp. 2622–2632. DOI: 10.1109/TVCG.2025.3549852.

Lee, Ho Jung et al. (2024). "Redirection Strategy Switching: Selective Redirection Controller for Dynamic Environment Adaptation". In: *IEEE Transactions on Visualization and Computer Graphics* 30.5, pp. 2474–2484. DOI: 10.1109/TVCG.2024.3372056.

Lemic, Filip, Jakob Struye, and Jeroen Famaey (2022). "Short-Term Trajectory Prediction for Full-Immersive Multiuser Virtual Reality with Redirected Walking". In: *2022 IEEE Global Communications Conference, GLOBECOM 2022 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 6139–6145. ISBN: 9781665435406. DOI: 10.1109/GLOBECOM48099.2022.10001349.

Li, Huiyu and Linwei Fan (2023). "A Segmented Redirection Mapping Method for Roadmaps of Large Constrained Virtual Environments". In: *IEEE Transactions on Visualization and Computer Graphics* 29.12, pp. 5308–5324. DOI: 10.1109/TVCG.2022.3207004.

— (2024). "Enabling Predictive Redirection Reset Based on Virtual-Real Spatial Probability Density Distributions". In: *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–17. DOI: 10.1109/TVCG.2024.3409734.

Li, Y J, F Steinicke, and M Wang (2022). "A comprehensive review of redirected walking techniques: Taxonomy, methods, and future directions". In: *JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY* 37 (3), pp. 561–583. DOI: 10.1007/s11390-022-2266-7.

Liao, Keh Yeun et al. (2022). "Redirected Walking with IRS-assisted Beamforming". In: *IEEE International Conference on Communications*. Vol. 2022-May. Institute of Electrical and Electronics Engineers Inc., pp. 3352–3357. ISBN: 9781538683477. DOI: 10.1109/ICC45855.2022.9838715.

Liu, JH., YF. Ren, and Q.W. et al. Gan (2024). "Overcoming Spatial Constraints in VR: A Survey of Redirected Walking Techniques." In: *J. Comput. Sci. Technol.* 39, pp. 841–870. DOI: doi.org/10.1007/s11390-024-4585-3.

Lurdes Calisto, Maria de and Soumodip Sarkar (2024). "A systematic review of virtual reality in tourism and hospitality: The known and the paths to follow". In:

International Journal of Hospitality Management 116, p. 103623. ISSN: 0278-4319. DOI: https://doi.org/10.1016/j.ijhm.2023.103623.

Martin, Daniel et al. (May 2023). "A Study of Change Blindness in Immersive Environments". In: *IEEE Transactions on Visualization and Computer Graphics*. ISSN: 19410506. DOI: 10.1109/TVCG.2023.3247102.

Martinez, Esteban Segarra, Annie S. Wu, and Ryan P. McMahan (2022). "Research Trends in Virtual Reality Locomotion Techniques". In: *Proceedings - 2022 IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 270–280. ISBN: 9781665496179. DOI: 10.1109/VR51125.2022.00046.

Marwecki, Sebastian and Patrick Baudisch (Oct. 2018). "Scenograph: Fitting real-walking VR experiences into various tracking volumes". In: *UIST 2018 - Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, Inc, pp. 511–520. ISBN: 9781450359481. DOI: 10.1145/3242587.3242648.

Marwecki, Sebastian et al. (Apr. 2018). "VirtualSpace - Overloading physical space with multiple virtual reality users". In: *Conference on Human Factors in Computing Systems - Proceedings*. Vol. 2018-April. Association for Computing Machinery. ISBN: 9781450356206. DOI: 10.1145/3173574.3173815.

Matsumoto, Keigo et al. (2021). "Redirected Walking using Noisy Galvanic Vestibular Stimulation". In: *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 498–507. DOI: 10.1109/ISMAR52148.2021.00067.

Mayor, Jesus, Laura Raya, and Alberto Sanchez (2021). "A Comparative Study of Virtual Reality Methods of Interaction and Locomotion Based on Presence, Cybersickness, and Usability". In: *IEEE Transactions on Emerging Topics in Computing* 9.3, pp. 1542–1553. DOI: 10.1109/TETC.2019.2915287.

Messinger, Justin, Eric Hodgson, and Eric R. Bachmann (2019). "Effects of Tracking Area Shape and Size on Artificial Potential Field Redirected Walking". In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 72–80. DOI: 10.1109/VR.2019.8797818.

Meyer, Florian, Malte Nogalski, and Wolfgang Fohl (2016). "Detection thresholds in audio-visual redirected walking". In: *Proc. Sound and Music Comp. Conf.(SMC)*. Vol. 16. 1, pp. 17–27. URL: https://core.ac.uk/download/pdf/144846422.pdf.

Mostajeran, Fariba et al. (2024). "Analyzing Cognitive Demands and Detection Thresholds for Redirected Walking in Immersive Forest and Urban Environments". In: *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 61–71. DOI: 10.1109/VR58804.2024.00030.

Nescher, Thomas, Ying-Yin Huang, and Andreas Kunz (2014). "Planning redirection techniques for optimal free walking experience using model predictive control". In: *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 111–118. DOI: 10.1109/3DUI.2014.6798851.

Neth, Christian T. et al. (2012). "Velocity-dependent dynamic curvature gain for redirected walking". In: *IEEE Transactions on Visualization and Computer Graphics* 18 (7), pp. 1041–1052. ISSN: 10772626. DOI: 10.1109/TVCG.2011.275.

Nguyen, Anh, Federico Cervellati, and Andreas Kunz (Nov. 2017). "Gain compensation in redirected walking". In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*. Association for Computing Machinery. ISBN: 9781450355483. DOI: 10.1145/3139131.3139167.

Nguyen, Anh and Andreas Kunz (Nov. 2018). "Discrete scene rotation during blinks and its effect on redirected walking algorithms". In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*. Association for Computing Machinery. ISBN: 9781450360869. DOI: 10.1145/3281505.3281515.

Nguyen, Anh et al. (Nov. 2020a). "Effect of Cognitive Load on Curvature Redirected Walking Thresholds". In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*. Association for Computing Machinery. ISBN: 9781450376198. DOI: 10.1145/3385956.3418950.

Nguyen, Anh et al. (Sept. 2020b). "Effect of Sense of Embodiment on Curvature Redirected Walking Thresholds". In: *Proceedings - SAP 2020: ACM Symposium on Applied Perception*. Association for Computing Machinery. ISBN: 9781450376181. DOI: 10.1145/3385955.3407932.

Nie, Tongyu, Isayas Berhe Adhanom, and Evan Suma Rosenberg (2023). "Like a Rolling Stone: Effects of Space Deformation During Linear Acceleration on Slope Perception and Cybersickness". In: *Proceedings - 2023 IEEE Conference Virtual Reality and 3D User Interfaces, VR 2023*. Institute of Electrical and Electronics Engineers Inc., pp. 658–668. ISBN: 9798350348156. DOI: 10.1109/VR55154.2023.00081.

Nilsson, Niels Christian et al. (July 2016). "Estimation of detection thresholds for audiovisual rotation gains". In: *Proceedings - IEEE Virtual Reality*. Vol. 2016-July. IEEE Computer Society, pp. 241–242. ISBN: 9781509008360. DOI: 10.1109/VR.2016.7504743.

Nilsson, Niels Christian et al. (Mar. 2018). "15 Years of Research on Redirected Walking in Immersive Virtual Environments". In: *IEEE Computer Graphics and Applications* 38 (2), pp. 44–56. ISSN: 02721716. DOI: 10.1109/MCG.2018.111125628.

Paludan, Anders et al. (July 2016). "Disguising rotational gain for redirected walking in virtual reality: Effect of visual density". In: *Proceedings - IEEE Virtual Reality*. Vol. 2016-July. IEEE Computer Society, pp. 259–260. ISBN: 9781509008360. DOI: 10.1109/VR.2016.7504752.

Peck, Tabitha C., Henry Fuchs, and Mary C. Whitton (May 2009). "Evaluation of reorientation techniques and distractors for walking in large virtual environments". In: *IEEE Transactions on Visualization and Computer Graphics* 15 (3), pp. 383–394. ISSN: 10772626. DOI: 10.1109/TVCG.2008.191.

— (2012). "The design and evaluation of a large-scale real-walking locomotion interface". In: *IEEE Transactions on Visualization and Computer Graphics* 18 (7), pp. 1053–1067. ISSN: 10772626. DOI: 10.1109/TVCG.2011.289.

Qi, Meng, Yunqiu Liu, and Jia Cui (Aug. 2010). "A mapping-based redirected walking algorithm for large-scale VR". In: *Springer Virtual Reality* 16 (27), 2745–2756. DOI: 10.1007/s10055-023-00841-9.

Razzaque, Sharif, Zachariah Kohn, and Mary C Whitton (2005). *Redirected Walking*.

Reason, J T (1978). "Motion Sickness Adaptation: A Neural Mismatch Model". In: *Journal of the Royal Society of Medicine* 11 (71), pp. 819–829. DOI: 10.1177/014107687807101109.

Rebelo, Ana Rita, Pedro A. Ferreira, and Rui Nóbrega (2025). "Techniques for Multiple Room Connection in Virtual Reality: Walking Within Small Physical Spaces". In: *IEEE Transactions on Visualization and Computer Graphics* 31.5, pp. 2310–2319. DOI: 10.1109/TVCG.2025.3549895.

Rebenitsch, Lisa and Charles Owen (2016). "Review on cybersickness in applications and visual displays". In: *Virtual Reality* (20), pp. 101–125. DOI: 10.1007/s10055-016-0285-9.

Reimer, Dennis et al. (Mar. 2020). "The Influence of Full-Body Representation on Translation and Curvature Gain". In: *Proceedings - 2020 IEEE Conference on Virtual Reality and 3D User Interfaces, VRW 2020*. Institute of Electrical and Electronics Engineers Inc., pp. 154–159. ISBN: 9781728165325. DOI: 10.1109/VRW50115.2020.00032.

Renner, Rebekka S., Boris M. Velichkovsky, and Jens R. Helmert (2013). "The perception of egocentric distances in virtual environments - A review". In: *ACM Computing Surveys* 46 (2). ISSN: 03600300. DOI: 10.1145/2543581.2543590.

Rietzler, Michael et al. (Jan. 2019). "Rethinking Redirected Walking: On the Use of Curvature Gains beyond Perceptual Limitations and Revisiting Bending Gains". In: *Proceedings of the 2018 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2018*. Institute of Electrical and Electronics Engineers Inc., pp. 115–122. ISBN: 9781538674598. DOI: 10.1109/ISMAR.2018.00041.

Robb, Andrew and Catherine Barwulor (2021). "Spatial Judgments in Impossible Spaces Preserve Important Relative Information". In: *ACM Symposium on Applied Perception 2021*. SAP '21. Virtual Event, France: Association for Computing Machinery. ISBN: 9781450386630. DOI: 10.1145/3474451.3476231.

Robb, Andrew, Kristopher Kohm, and John Porter (Nov. 2022). "Experience Matters: Longitudinal Changes in Sensitivity to Rotational Gains in Virtual Reality". In: *ACM Transactions on Applied Perception* 19 (4). ISSN: 15443965. DOI: 10.1145/3560818.

Ropelato, S, M. Menozzi, and M.YY. Huang (2022). "Hyper-reoriented walking in minimal space." In: *Springer Virtual Reality* 26, 1009–1017. DOI: doi.org/10.1007/s10055-021-00608-0.

Rothacher, Yannick et al. (Dec. 2018). “Visual capture of gait during redirected walking”. In: *Scientific Reports* 8 (1). ISSN: 20452322. DOI: 10.1038/s41598-018-36035-6.

Ruddle, Roy A. and Simon Lessels (Apr. 2009). “The benefits of using a walking interface to navigate virtual environments”. In: *ACM Transactions on Computer-Human Interaction* 16 (1). ISSN: 10730516. DOI: 10.1145/1502800.1502805.

Safikhani, Saeed et al. (2022). “Immersive virtual reality for extending the potential of building information modeling in architecture, engineering, and construction sector: systematic review”. In: *International Journal of Digital Earth* 15.1, pp. 503–526. DOI: 10.1080/17538947.2022.2038291.

Sakono, Hiroaki et al. (2021). “Redirected Walking using Continuous Curvature Manipulation”. In: *IEEE Transactions on Visualization and Computer Graphics* 27.11, pp. 4278–4288. DOI: 10.1109/TVCG.2021.3106501.

Schmitz, Patric et al. (Apr. 2018). “You spin my head right round: Threshold of limited immersion for rotation gains in redirected walking”. In: *IEEE Transactions on Visualization and Computer Graphics* 24 (4), pp. 1623–1632. ISSN: 10772626. DOI: 10.1109/TVCG.2018.2793671.

Selzer, Matias Nicolas, Martin Leonardo Larrea, and Silvia Mabel Castro (Dec. 2022). “Analysis of translation gains in virtual reality: the limits of space manipulation”. In: *Virtual Reality* 26 (4), pp. 1459–1469. ISSN: 14349957. DOI: 10.1007/s10055-022-00640-8.

Serubugo, Sule et al. (Nov. 2017). “Walkable self-overlapping virtual reality maze and map visualization demo”. In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*. Vol. Part F131944. Association for Computing Machinery. ISBN: 9781450355483. DOI: 10.1145/3139131.3141774.

Simons, Daniel J. and Michael S. Ambinder (2005). “Change Blindness: Theory and Consequences”. In: *Current Directions in Psychological Science* 14.1, pp. 44–48. DOI: 10.1111/j.0963-7214.2005.00332.x.

Simons, Daniel J. and Ronald A. Rensink (2005). “Change blindness: Past, present, and future”. In: *Trends in Cognitive Sciences* 9 (1), pp. 16–20. ISSN: 13646613. DOI: 10.1016/j.tics.2004.11.006.

Sin, Zackary P. T. et al. (2019). "Transferring Object Layouts from Virtual to Physical Rooms: Towards Adapting a Virtual Scene to a Physical Scene for VR". In: *Advances in Computer Graphics*. Ed. by Marina Gavrilova et al. Cham: Springer International Publishing, pp. 253–265. ISBN: 978-3-030-22514-8.

Slater, Mel (2009). "Place illusion and plausability can lead to realistic behaviour in immersive virtual environments". In: *Phil. Trans. R. Soc. B 364:3549-3557*. DOI: https://doi.org/10.1098/rstb.2009.0138.

Steinicke, Frank et al. (2008). "Taxonomy and implementation of redirection techniques for ubiquitous passive haptic feedback". In: *Proceedings of the 2008 International Conference on Cyberworlds, CW 2008*, pp. 217–223. ISBN: 9780769533810. DOI: 10.1109/CW.2008.53.

Steinicke, Frank et al. (Jan. 2010). "Estimation of detection thresholds for redirected walking techniques". In: *IEEE Transactions on Visualization and Computer Graphics* 16 (1), pp. 17–27. ISSN: 10772626. DOI: 10.1109/TVCG.2009.62.

Strauss, Ryan R. et al. (May 2020). "A Steering Algorithm for Redirected Walking Using Reinforcement Learning". In: *IEEE Transactions on Visualization and Computer Graphics* 26 (5), pp. 1955–1963. ISSN: 19410506. DOI: 10.1109/TVCG.2020.2973060.

Suma, Evan A. et al. (2011). "Leveraging change blindness for redirection in virtual environments". In: *Proceedings - IEEE Virtual Reality*, pp. 159–166. ISBN: 9781457700361. DOI: 10.1109/VR.2011.5759455.

Suma, Evan A. et al. (2012). "Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture". In: *IEEE Transactions on Visualization and Computer Graphics* 18 (4), pp. 555–564. ISSN: 10772626. DOI: 10.1109/TVCG.2012.47.

Sun, Qi, Li Yi Wei, and Arie Kaufman (July 2016). "Mapping virtual and physical reality". In: *ACM Transactions on Graphics* 35 (4). ISSN: 15577368. DOI: 10.1145/2897824.2925883.

Thomas, Jerald and Evan Suma Rosenberg (2019a). "A General Reactive Algorithm for Redirected Walking Using Artificial Potential Functions". In: *2019 IEEE Conference* *on Virtual Reality and 3D User Interfaces (VR)*, pp. 56–62. DOI: 10.1109/VR.2019.8797983.

— (2019b). "A General Reactive Algorithm for Redirected Walking Using Artificial Potential Functions". In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. Washington, DC, USA: IEEE, pp. 56–62. DOI: 10.1109/VR.2019.8797983.

Thomas, Jerald, Seraphina Yong, and Evan Suma Rosenberg (2022). "Inverse Kinematics Assistance for the Creation of Redirected Walking Paths". In: *Proceedings - 2022 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 593–602. ISBN: 9781665453257. DOI: 10.1109/ISMAR55827.2022.00076.

Usoh, Martin et al. (1999). "Walking > Walking-in-Place > Flying, in Virtual Environments". In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pp. 359–364.

Vasser, Madis, Markus Kängsepp, and Jaan Aru (2015). *Change Blindness in 3D Virtual Reality*. DOI: 10.48550/arXiv.1508.05782.

Vasylevska, Khrystyna and Hannes Kaufmann (2015). "Influence of Path Complexity on Spatial Overlap Perception in Virtual Environments". In: *ICAT-EGVE 2015 - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*. The Eurographics Association. ISBN: 978-3-905674-84-2. DOI: 10.2312/egve.20151324.

Vasylevska, Khrystyna et al. (2013). "Flexible spaces: Dynamic layout generation for infinite walking in virtual environments". In: *IEEE Symposium on 3D User Interface 2013, 3DUI 2013 - Proceedings*, pp. 39–42. ISBN: 9781467360975. DOI: 10.1109/3DUI.2013.6550194.

Waldow, Kristoffer, Arnulph Fuhrmann, and Stefan M. Grunvogel (Aug. 2018). "Do Textures and Global Illumination Influence the Perception of Redirected Walking Based on Translational Gain?" In: *25th IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2018 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 717–718. ISBN: 9781538633656. DOI: 10.1109/VR.2018.8446587.

Wang, Jialin et al. (May 2022). "Real-time Prediction of Simulator Sickness in Virtual Reality Games". In: *IEEE Transactions on Games*. ISSN: 2475-1502. DOI: 10.1109/tg.2022.3178539.

Williams, Betsy et al. (2006). "Updating orientation in large virtual environments using scaled translational gain". In: *Proceedings of the 3rd Symposium on Applied Perception in Graphics and Visualization*. APGV '06. Boston, Massachusetts, USA: Association for Computing Machinery, 21–28. ISBN: 1595934294. DOI: 10.1145/1140491.1140495.

Williams, Betsy et al. (2007). "Exploring Large Virtual Environments with an HMD When Physical Space is Limited". In: *Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*. APGV '07. Tubingen, Germany: Association for Computing Machinery, 41–48. ISBN: 9781595936707. DOI: 10.1145/1272582.1272590.

Williams, Niall L., Aniket Bera, and Dinesh Manocha (May 2021a). "ARC: Alignment-based Redirection Controller for Redirected Walking in Complex Environments". In: *IEEE Transactions on Visualization and Computer Graphics* 27 (5), pp. 2535–2544. ISSN: 19410506. DOI: 10.1109/TVCG.2021.3067781.

— (2021b). "Redirected Walking in Static and Dynamic Scenes Using Visibility Polygons". In: *IEEE Transactions on Visualization and Computer Graphics* 27.11, pp. 4267–4277. DOI: 10.1109/TVCG.2021.3106432.

— (2022). "ENI: Quantifying Environment Compatibility for Natural Walking in Virtual Reality". In: *Proceedings - 2022 IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2022*. Institute of Electrical and Electronics Engineers Inc., pp. 419–427. ISBN: 9781665496179. DOI: 10.1109/VR51125.2022.00061.

Williams, Niall L. and Tabitha C. Peck (Nov. 2019). "Estimation of Rotation Gain Thresholds Considering FOV, Gender, and Distractors". In: *IEEE Transactions on Visualization and Computer Graphics* 25 (11), pp. 3158–3168. ISSN: 19410506. DOI: 10.1109/TVCG.2019.2932213.

Williams, Niall L. et al. (2025). "Sensitivity to Redirected Walking Considering Gaze, Posture, and Luminance". In: *IEEE Transactions on Visualization and Computer Graphics* 31.5, pp. 3223–3234. DOI: 10.1109/TVCG.2025.3549908.

Wu, Xue-Liang et al. (2023). "Novel Design and Evaluation of Redirection Controllers Using Optimized Alignment and Artificial Potential Field". In: *IEEE Transactions on Visualization and Computer Graphics* 29.11, pp. 4556–4566. DOI: 10.1109/TVCG.2023.3320208.

Xiong, Yuan et al. (2024). "DreamWalk: Dynamic remapping and multiperspectivity for large-scale redirected walking". In: *Computer Animation and Virtual Worlds* 35.1, e2196. DOI: https://doi.org/10.1002/cav.2196.

Xu, Sen Zhe et al. (Nov. 2022). "Making Resets away from Targets: POI aware Redirected Walking". In: *IEEE Transactions on Visualization and Computer Graphics* 28 (11), pp. 3778–3787. ISSN: 19410506. DOI: 10.1109/TVCG.2022.3203095.

Xu, Sen-Zhe et al. (2024). "Spatial Contraction Based on Velocity Variation for Natural Walking in Virtual Reality". In: *IEEE Transactions on Visualization and Computer Graphics* 30.5, pp. 2444–2453. DOI: 10.1109/TVCG.2024.3372109.

Yu, Run et al. (Apr. 2017). "Bookshelf and Bird: Enabling real walking in large VR spaces". In: *2017 IEEE Symposium on 3D User Interfaces, 3DUI 2017 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 116–119. ISBN: 9781509067169. DOI: 10.1109/3DUI.2017.7893327.

Zank, Markus and Andreas Kunz (2017). "Optimized graph extraction and locomotion prediction for redirected walking". In: *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 120–129. DOI: 10.1109/3DUI.2017.7893328.

Zhang, Chaoning et al. (2024). *A Survey on Segment Anything Model (SAM): Vision Foundation Model Meets Prompt Engineering*. arXiv: 2306.06211 [cs.CV]. URL: https://arxiv.org/abs/2306.06211.

Zhang, Ruimin and Scott A. Kuhl (2013). "Flexible and general redirected walking for head-mounted displays". In: *Proceedings - IEEE Virtual Reality*, pp. 127–128. ISBN: 9781467347952. DOI: 10.1109/VR.2013.6549395.

Zhang, Song Hai, Chia Hao Chen, and Stefanie Zollmann (July 2022). "One-step out-of-place resetting for redirected walking in VR". In: *IEEE Transactions on Visualization and Computer Graphics*. ISSN: 19410506. DOI: 10.1109/TVCG.2022.3158609.

Zhang, Song Hai et al. (Apr. 2023). "Adaptive Optimization Algorithm for Resetting Techniques in Obstacle-Ridden Environments". In: *IEEE Transactions on Visualization and Computer Graphics* 29 (4), pp. 2080–2092. ISSN: 19410506. DOI: 10.1109/TVCG.2021.3139990.

Zhao, Yue, Robert W. Lindeman, and Thammathip Piumsomboon (2025). "Daddy Long Legs: A Scale and Speed Up Virtual Reality Locomotion Technique for Medium-scale Scenarios". In: *2025 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 329–339. DOI: 10.1109/VR59515.2025.00057.

Zmuda, Michael A. et al. (2013). "Optimizing constrained-environment redirected walking instructions using search techniques". In: *IEEE Transactions on Visualization and Computer Graphics* 19 (11), pp. 1872–1884. ISSN: 10772626. DOI: 10.1109/TVCG.2013.88.