Comparing Hand Tracking to Motion Controllers when manipulating small objects in Virtual Reality in terms of usability and user preference

Denis Dimitrov

BSc (Hons) Computer Games Application Development, 2025

Faculty of Design, Informatics and Business
Department of Games Technology and Mathematics
Abertay University

Table of Contents

Table of Figures

Acknowledgements

I would like to thank my academic supervisor, Dr. Andrei Boiko for supporting and guiding me throughout the project, both academically and personally.
Additionally, I would like to thank Gerald High, for his guidance and assistance in building the haptic feedback gloves.
Finally, I would like to thank the Fishbowl friend group - Kyle Singer, Torin Pellegrini, Jan Huss, Liam Laverty, Dominik Wylie, Fraser McCann, David Findlay, Daniel Wooldridge and Michael Ennis for being supportive and amazing, as well as being present thus motivating and helping me throughout the academic year.

Abstract

Context: While hand tracking technology is becoming increasingly accessible, affordable and accurate, it is not yet the primary input method for virtual reality applications. Just like how interactions with mobile phones transitioned from physical buttons to touch and gesture-based input, can virtual reality adopt hand tracking and gesture-based input in the same way, providing a more natural, intuitive and accessible solution.

Aim: The aim of this study is to compare the effectiveness of handheld motion controllers and hand tracking when manipulating small objects in virtual environments in terms of intuitiveness, ease of use and user preference, and examine the impact of haptics being present in the hand tracking setting.

Method: 8 participants were recruited to participate in test sessions, where they had to use motion controllers hand tracking and a custom haptic feedback glove device, to interact with a virtual environment and complete a series of tasks, including object alignment, object sortation, cube stacking and moving an object following a spline. Before the test session, participants were asked to fill out a demographic survey, and after testing each input method they were asked to fill out a system usability survey to gauge the usability of each input configuration. During the test sessions quantitative data was also gathered to allow for statistical analysis to find any discrepancies between the user preference and their performance.

Results: The collected data shows that while motion controllers had the best quantitative results with some statistically significant differences, hand tracking was the generally preferred input method as seen by the system usability scores over motion controllers, because it featured a more natural and accessible way of interacting with the virtual environment. The haptic hand tracking configuration was seen as the worst in this project, as it faced tracking and accuracy issues because of

the size of the haptic device, that intervened with the algorithms driving the hand tracking inputs.

Conclusion: This study found that hand tracking has become a viable alternative to motion controllers in user preference and performance, especially when manipulating small objects. It also found that haptics can improve hand tracking and its usability, but it could not create a perfect device that has no impact on the hand tracking algorithms, while providing haptic feedback, rather than the device implemented in this study had more of a negative impact as it drastically reduced the accuracy of the hand tracking algorithms.

Abbreviations, Symbols and Notation

VR - Virtual Reality
VE - Virtual Environment
HMD - Head-mounted Display
SLAM - Simultaneous Location and Mapping
vSLAM - Visual Simultaneous Location and Mapping
SU – System Usability
SUS – System Usability Scale

Chapter 1 Introduction

Hand tracking has faced many issues throughout the years which have prevented it from becoming the mainstream input modality for VR applications. With accuracy and speed being the biggest challenges. This research compares them against motion controllers to examine if improvements to hand tracking technology have pushed it beyond it being a gimmick and made it a competitive input method for VR applications.

1.1 Virtual Reality and Input Mediums

1.1.1 Virtual Reality Devices

Virtual Reality (VR) is an emerging technology that allows users to experience the feeling of presence, the subjective experience of being in one environment, while physically in another environment (Witmer and Singer, 1998). It achieves this by presenting images on screens that are positioned on a head-mounted display (HMD) device that also blocks the incoming light from the outside world. As HMDs generally use 2 displays, it allows for users to perceive depth. HMDs also uses sensors and cameras to track user's movements translates them into movements in the Virtual Environment (VE). The combination of which allows for a higher feeling of presence than in any other currently available immersive technology. Furthermore, VR has been slowly rising in popularity over the last decade and even more throughout recent years, mainly credited to devices becoming more affordable and powerful, like the Meta Quest 3S, which quickly became the most sold Games Console on Amazon US (Heaney, 2025). VR is now used in entertainment, marketing, education, and many other sectors.

1.1.2 Input Methods: Motion Controllers

While HMDs can translate head movements and display images to users, they do not provide much more than that and interactivity within VR applications is essential to keep the user engaged. As a solution to that, VR systems provide different methods of interacting with the VEs.

Traditionally VR devices relied on handheld motion controllers, a combination of sensors, buttons, and joysticks, allowing them to track users' hand motions and translating them into corresponding actions within the VE. These handheld motion controllers also offer haptic feedback, in the form of vibrations, which can enhance the usability and feeling of presence by providing a feeling connected to physical actions, as well as provide much needed acknowledgement of the users' action within the VE. Because of the technology and algorithms that drive motion controllers, they tend to be quite accurate and prove to be effective at providing users with interactions that would be hard to incorporate in other mediums, like precise object manipulation and natural interaction with environments. As a result, motion controllers have become the standard input method in most commercially available VR devices.

1.1.3 Input Methods: Hand Tracking

In recent years a new input modality has been improving and is starting to be used in more and more VR systems and applications, hand tracking. Previous implementations of hand tracking were clunky and required auxiliary devices, such as gloves designed to track your fingers and hands. Recent improvements have removed the need for those devices and have provided a much more natural interface, by just wearing the HMD and utilizing its cameras and sensors to track the users' hands and fingers. Usually after capturing the data, it is then processed and sent to the VR application to use as input. While current implementations solve many of hand tracking algorithms' previous issues like obstruction, one hand covering the other, fast movements, and overall tracking accuracy due to their complexity, hand tracking still faces precision and speed issues, that are slowly improving but are still present. These improvements and the speed of achieving them, shows that hand tracking can and might have already become a viable alternative input modality for VR applications.

1.1.4 Input Methods: Hand Tracking and Custom Devices

Some hand tracking solutions have attempted to use auxiliary devices to attempt to solve common issues with this input method. Those solutions usually utilize depth sensors, cameras, lighthouses, devices that emit laser beams to track the position of the HMD and the corresponding motion controllers, and different types of gloves. One thing that has not been currently explored or is commercially available is providing haptic feedback when hand tracking is utilized. Haptics on motion controllers can be used to affirm user actions as well as inform them of possible actions, this might apply to hand tracking as well.

1.2 Aim

The aim of this project is to compare the effectiveness of handheld motion controllers and hand tracking in virtual reality applications, specifically in terms of intuitiveness, ease of use, and user preference when manipulating small objects in virtual environments and examine the impact of haptic feedback in the hand tracking setting.

1.3 Research Questions

How does hand tracking compare to motion controllers when manipulating small objects in virtual environments in terms of intuitiveness and user preference?

1.5 Objectives

The primary objectives of this project are:

- To evaluate the intuitiveness, user preference and performance of hand tracking and motion controllers for small object manipulation tasks using data gathered from questionnaires and test sessions
- To develop a VR application that facilitates input through hand tracking and motion controller configurations and ability to go through pre-designed tests
- To design and develop a custom haptic feedback device to allow for examination of the impact of haptics being present in the hand tracking setting

Chapter 2 Literature Review

The following sections review relevant literature to this project. First going into virtual reality, then diving into different input methods and their use cases and finally exploring relevant research and critically analysing it.

2.1 Virtual Reality

Virtual Reality is one of the latest iterations of devices aimed at providing immersive experiences, which focuses on allowing users to explore virtual environments and interact with them in a more intuitive and freeform manner. In the past VR was slowly rising in popularity and has seen some use in in applications, like medicine and military training, entertainment and education. In recent years companies like Meta have made VR more affordable and mainstream, with their latest offering the Meta Quest 3S, being released around Christmas and very well priced, it has managed to bring many new users (Heaney, 2025).

2.1.1 Head-mounted Displays

The most common VR devices are head-mounted displays (HMDs), which present visual information to the user using displays, while blocking the users' vision of the outside world. HMDs can be monocular, one-eye display, biocular, 2 displays showing the same image, and binocular, 2 displays showing stereoscopic images. Binocular HMDs have become the standard nowadays, due to having the added benefit of better depth-perception and scale recognition, which thereby increases immersion and the feeling of presence (Rolland and Hua, 2005).

Along with HMDs VR devices are usually sold with motion controllers included, which feature sophisticated tracking solutions, which provide the ability to update visuals depending on the users' head and hand movements, with more recent devices featuring hand tracking and full-body tracking, which attempt to further increase the feeling presence.

2.2 Input Methods

2.2.1 Motion Controllers

The primary method of providing user input to VR devices is through the use of handheld motion controllers. Motion Controllers are devices that translate the users' movements through the use of a tracking algorithm called Visual Simultaneous Location and Mapping (vSLAM), which is based on the Simultaneous Location and Mapping (SLAM) algorithm, that continuously updates the users' hand position and orientation in the virtual world. The difference between vSLAM and SLAM is that vSLAM in addition to the sensors and the motors, also includes cameras for improved accuracy and initial transform estimation (Taketomi, Uchiyama, and Ikeda, 2017). In addition to translating users' hand movements from the real to the virtual world, motion controllers feature a variety of buttons, triggers, and thumbsticks that allow users to provide digital input. These buttons and triggers are commonly used to map intention to users' actions like picking up objects, pressing buttons, or navigating the virtual environment (VE). Some common examples of button mappings in VR applications include the "grip" button being used to grab an object for as long as it is held, and the "trigger" button to use grabbed objects or perform their action, like shooting a gun. This combination of digital input and hand movement tracking allows VR devices to provide a high degree of freedom in terms of control, precision and user expression. However, while the control and precision might be increased compared to traditional input devices, motion controllers usually feature a smaller number of buttons, which can limit the number of interactions a VR application is able to provide, thus reducing immersion and presence.

2.2.2 Hand Tracking

Recently, an alternative to motion controllers for VR devices has been rapidly improving and growing in popularity, hand tracking. Hand tracking removes the need for auxiliary devices by allowing users to use their hands in the virtual world as they would in the real one.

In order for hand tracking to work, it requires the use of cameras and depth sensors to estimate the position and orientation of the users' hands and fingers in real time. To achieve this, first a bounding box is calculated that encapsulates the users' hands after which, data from the depth sensors and cameras is captured and sent to computer vision algorithms to analyse and recognize specific hand features, poses, and gestures (Han et al., 2020). This allows users to interact with virtual objects as they would with ones in the real world, but even though hand tracking provides a lot of freedom in the way users interact and express themselves, it faces other issues, like locomotion, how does a user signify the intent of moving. One solution to this problem can be by the user moving physically, but there is the limitation of space in the real world. Currently developers solve this problem by assigning specific gestures to intent, for example Half-Life: Alyx (Valve, 2020) has the user point in the direction they want to move but this approach is not ideal, as if a user wants to keep moving, they must keep pointing, which can be tiring. Expressing intent is one of the biggest challenges hand tracking faces. Drakheir (2024) is an example of a project that has attempted to tackle the challenge of expressing intent and does it very well, by providing intuitive controls, like grab and move your hand to move in the VE, as well using different hand gestures for different actions, like choosing a weapon. But their solution is still not perfect, as the gestures are not consistently recognised and sometimes you get the wrong weapon or no action is performed. Another big challenge that hand tracking faces is its inability to provide haptic feedback, which can be crucial for usability and can lead to user feeling unacknowledged by the virtual world.

2.3 Motion controllers vs. hand tracking

Several studies and experiments have compared motion controllers to hand tracking in different conditions and tasks. Johnson (2023) investigated the usability and user agency between motion controllers and hand tracking in an object sorting task, where participants had to sort differently colored balls into the appropriately colored container. Although this study found that motion controllers offer a greater sense of user agency and better usability, the tasks featured unfavourable conditions for hand tracking, such as fast movements. The paper credits the preference towards motion controllers, due to their higher accuracy and haptic feedback. Due to these findings this project will implement tasks with both slow and fast movements, which will help balance the results.

Masurkovsky (2020) researched the user preference and performance in grab-and-place tasks. Participants were asked to grab a box that spawns randomly in a circle around a point and place it on top of said point. The results show that participants showed no preference between hand tracking and motion controllers, but participants collectively performed better using motion controllers. However, since this study, hand tracking technology has improved and continues to improve with recent updates, like v72 on Meta Quest bringing better stability in fast motions, and overall better tracking (Meta, 2024).

Similarly, Khundam (2021) conducted research in medical training scenarios involving intubation. Participants were required to go through 7 procedures, which started with training instruction assistance and after which objectives were provided. Results show that there were no measurable differences in performance or preference between the two input methods. That may be due to the nature of intubation movements, which are slow, precise and controlled, allowing hand tracking algorithms to perform similarly to motion controllers, and the current research builds on that insight by featuring slow and controlled movement tasks.

Another study by Rantamaa (2023) compared mouse, hands, and a controller + stylus combination in the task of annotating 3D objects for medical surgery planning. The annotation task involved marking spots on a skull by touching a virtual pen or a ray in the case of the mouse configuration to points on a 3D skull object in the VE. The main differences between the input configurations are that the mouse one featured a 3D model for it and an always visible ray coming out the front of it. In the hands configuration, participants had to use one of their hands to move the skull and the other to annotate, and finally in the controller +

stylus, the controller was used to manipulate the skull and the stylus to annotate. They found statistically significant differences between the controller + stylus combination and the other two methods, with hand-based and mouse interaction being almost equal in feasibility. Although the controller + stylus combination was favourable in this study, it is hard to draw clear conclusions as the other input devices did not feature a stylus or another object as a point of reference, which might have twisted the odds in favour of the controller configuration.

Luong (2023) compared controllers and hand tracking in a selection and trajectory tracing tasks using touch, having to reach out to interact with the virtual content, and ray casting, the ability to interact at a distance, for input. The selection task featured targets that would spawn in predefined locations and participants were instructed to interact with as many targets as possible, with the appropriate action for each input, pinch for hand tracking and pressing the trigger button on the motion controllers. For the trajectory tracing task, participants had to trace either a circle or a square for 2 minutes by aiming at a target, selecting it, and then following it as closely as possible. Results show that participants felt a higher sense of agency when using motion controllers compared to hand tracking in the ray cast conditions, while in touch conditions they did not find any significant difference in subjective ratings. Perceived exertion was also measured, and it showed that motion controllers were only perceived as significantly less exerting in the ray cast condition. Participants were also asked about their preference, and in touch conditions, hand tracking was generally preferred, while in ray cast conditions it was the opposite with motion controllers being the generally preferred interaction interface. These findings suggest that while motion controllers may be the objectively better input device, hand tracking provides a more natural input modality when it comes to touch-based interactions, which is what this project examines.

Healy (2024) compared motion controllers to hand tracking in a nerve block simulation, where participants were required to manoeuvre an ultrasound probe and injection needle during a nerve block procedure. To complete a procedure participants had to perform 5 tasks, including picking up an ultrasound probe, guiding and placing it in a specific location, picking up a needle, using the simulated ultrasound image to guide the needle to a specific location, and injecting an anaesthetic solution. Results from this research show that motion controllers may offer a slight advantage in terms of ease of use, but hand tracking provides a significantly more realistic experience when controlling the needle, which is crucial for simulating the required precision, thus making it a preferred solution for the given task. This and many of the previous papers suggest that while hand tracking might not be as good as motion controllers in terms of ease of use and accuracy, it offers a more realistic and natural way of interacting with objects, thus it could be a preferred input method if hand tracking's drawbacks were to be addressed.

But these drawbacks are also implementation dependent, as there are many ways to achieve hand tracking, which is why Noghabaei (2020) compared 3 different ways of achieving hand tracking. They tested used construction training tasks, where participants had to grab objects and snap them to specific locations highlighted in green to measure the usability of each implementation. The 3 different configurations tested were Magnetic-Based using Noitom Hi5 gloves, Image-Based using Meta Quest, which is what this project is using, and Infrared-Based using Leap Motion. Results from this research show that the Magnetic-Based gloves performed the best, being closely followed by the Image-Based, and Infrared coming in last. This paper shows that while Meta's Image-Based solution might not have been the best it is comparable. Nowadays Image-Based hand tracking might have become as good as hardware-based solutions with the help of more powerful devices and improvements to the algorithms used.

Considering all the findings mentioned above this project implements different tests requiring both slow and fast movements, controlled movements, and object placing tasks to gauge the improvements in hand tracking technology and implements a haptic feedback solution for hand tracking in the form of custom-built haptic gloves to see the impact of haptic feedback and whether it improves the usability and user preference.

Chapter 3 Methodology

3.1 Application

This project developed an application that allows for object manipulation using three different input configurations: Hand Tracking, Motion Controllers and Hand Tracking with Haptic Gloves, was developed. Along with that a custom device providing haptic feedback was designed and built with the help of the university's engineering services technician Gerald High.

3.1.1 The Setup

The application was developed in Unreal Engine 5.4.4, a general-purpose game engine developed and maintained by Epic Games (2024). Along with the MetaXR, OculusHandTools and OculusUtils plugins developed by Meta (2024) to facilitate the required hand tracking interactions, and the SerialCOM plugin by Ramiro Montes De Oca (2024) to allow for communication between the haptic feedback devices and the application.

3.1.2 Initial Interaction Implementation

Initially the project implemented a grabbing mechanic that intended to simulate a more realistic interaction, by having each grabbable object specify the number of required contact points, points where the object touches a finger, and the required fingers for it to be grabbed. After the object was grabbed, it would follow the hand movement and remain in the same relative position to it. To achieve this functionality, every frame the player pawn would perform a sphere trace per finger per hand, checking for objects in a radius. If an object was overlapped by a sphere, it would be stored in an array along with a bitmask, marking which fingers are in contact with it. The application would then check if enough fingers were in contact as well if the required ones are present in the bitmask, if both conditions are true, the object would be grabbed, otherwise it would be released. While this implementation had more realistic grabbing, it had some issues, like favouring one hand over the other when attempting to transfer objects between hands. These issues could have been cleared up, but due to time limitations as hardware was taking longer than expected to design and create, other solutions were explored.

3.1.3 Final Interaction Implementation

While exploring different solutions for interactions while using hand tracking as an input modality, the Meta Hand Gameplay Sample (Meta, 2021) was discovered, which implemented a gesture-based grabbing mechanic. This implementation was chosen to be used as it solved all the problems the initial implementation had and was well developed and documented, easy to customise, and simple to use. The plugin works by using the provided data by headset about the hand bone rotations and comparing that to a reference hand pose, measuring the error rate, the difference between the current and target rotation, if the error rate is lower than a given threshold the plugin registers the performed pose and sends an appropriate event to the input system in Unreal Engine. A user can then bind to the input events or use Camera Hand Input component to check whether a pinch or a grab action is being performed.

This is utilized by the user through a player pawn that has a component called Hand Character Hand State attached to it, which checks for grab gestures, updates the hand poses, and the hand model to match the real hands.

3.1.3.1 Modifications

The default functionality was not perfect and required modifications to be made to facilitate easy communication between systems. The modifications involved adding delegates that are broadcasted when a grab or release action was performed, so that any other object interested in when those actions happen, is aware that the interaction has happened and can perform any necessary computations. The grab radius was also reduced as initially it was way too large and unintentional grabs could happen. Another key setting that was missing from the default grabbable objects, was the ability to block grabbing, which was crucial for this project, this was added by having a blueprint editable boolean variable and a setter and getter functions for it. The Hands Character Hand State component was also modified, removing the functionality for throwing, selecting objects, ray cast interactions, and teleportation as they were not needed, and it resulted in improved performance.

3.1.4 Tasks and Objectives

In order to test different input modalities and gather results, a framework was designed and developed to allow for easy combinations of different objectives. An objective is an action or a series of actions that a user must perform, and a task is a series of objectives. This is implemented by the HTTask and HTTaskObjective classes.

The HTTaskObjective class is marked as Abstract, Blueprintable, DefaultToInstanced and EditInlineNew to remove the need for creating extra blueprints for each objective and make it easier to add and remove them. The task objectives have functions for activating, completing, checking their state, cleaning up after themselves and some other small utility functions, they also include delegates that are broadcasted when the objective is ready to be completed, has been started and/or completed. Task objectives also collect and send data to the Result Data Writer subsystem, that writes the data to a csv file.

An HTTask is a collection of task objectives, with functionality focused around starting an objective and completing the currently active objective, in addition to that there are delegates that are broadcast when a task objective is started and completed.

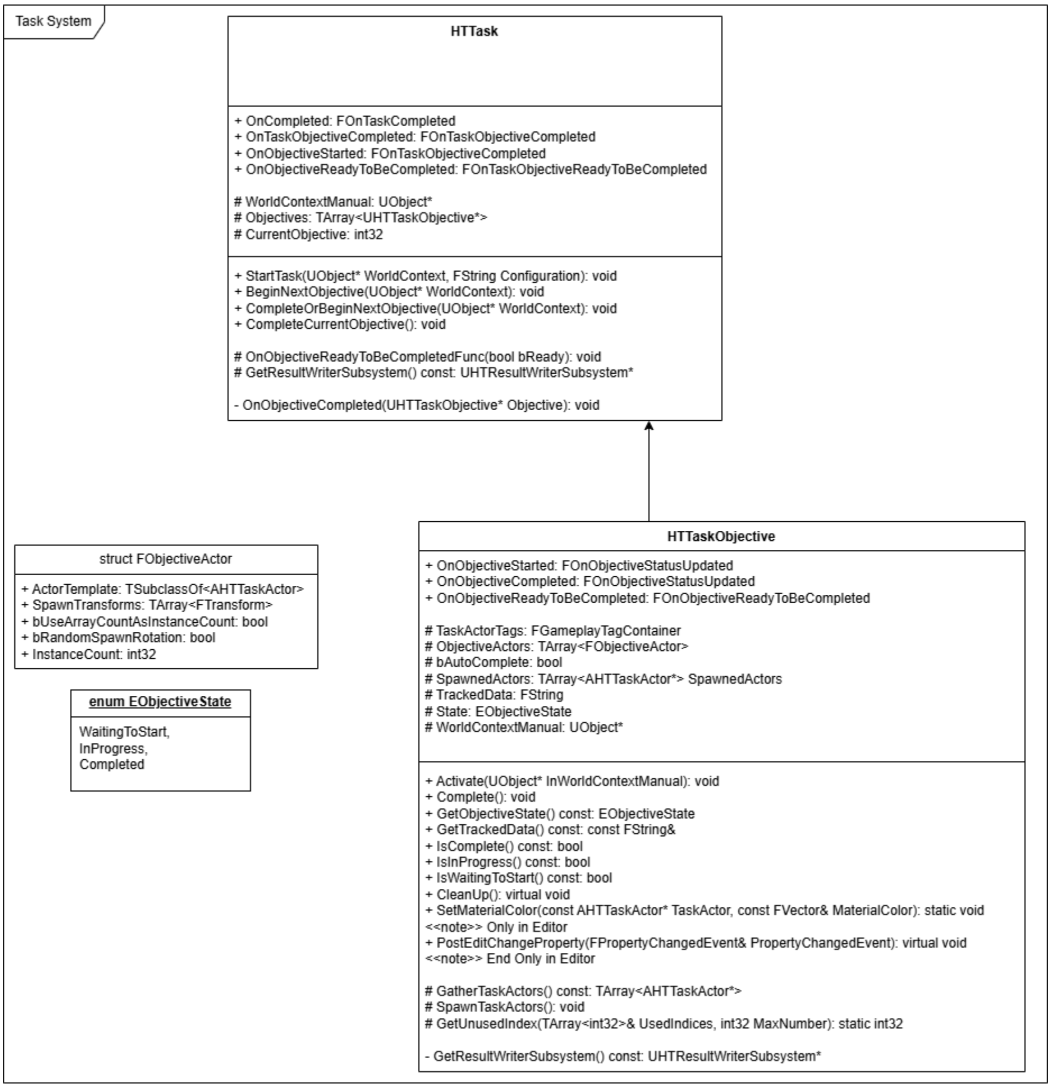

Figure 1 Task System Class Diagram

3.1.4.1 Task Objectives

The currently implemented task objectives are:

- **Location Rotation Alignment**: The user must align objects in corresponding target points in the virtual world marked with the same object. This works by checking if the distance between the position of the released object and its target position is within a threshold. After which it counts it as validly placed object and records the data so that it can be sent to the data writer, once all objects have been placed correctly.
- **Sortation**: The player must sort differently colored balls into their appropriately colored boxes by dropping the object within the box, which is determined by if checking the distance between the ball's

position and its target box position is within a given acceptance radius. This objective tracks the number of correctly and incorrectly sorted balls.

- **Cube Stacking**: The user has to stack cubes in an ascending order, from 1 to 5. The order of the cubes is indicated by a texture on the cubes and has a tag identifying which cube it is and what is its target. To determine if a cube has been stacked correctly, a box sweep is performed downwards, checking if the closest object hit has the correct tag, this check is performed when releasing the cube or when trying to complete the objective. This objective also features settings for enabling physics on the cubes and the hand-displaying model. The objective tracks the number of cubes stacked correctly and the average delta on the x and y axis.
- **Follow Target**: This objective has the user navigate an object around a course determined by a spline. As objectives do not have ticking functionality, a ticking async action was added. The async action would start tracking position delta from the closest point of the spline to the object once the object is grabbed and pauses when it is released. This objective tracks average position delta, max distance delta, time taken to complete, total position samples and total position delta.

3.1.5 Haptic Feedback Subsystem

To allow for testing of the haptic feedback device, the project required a haptic feedback system to facilitate communication between the application and the haptic gloves. The system was developed as a game instance subsystem, meaning that it has the same lifespan as the game instance. The subsystem references 2 SerialCom objects, one for each haptic device, as well as a map of hand locations to time of activation. The system publicly exposes 2 functions, one for enabling the haptics and one for applying feedback. The apply feedback function compares the current time with the last written time plus the duration of the last effect sent in order to make sure that the device is not being overloaded with data, after that it writes 8 bytes on the appropriate serial com port. The first byte signifies the location where the haptic feedback should be applied, the second byte stores the strength, followed by 4 bytes for duration and 2 bytes for padding, to make the information 8-byte aligned, which removes the chance of misaligned access of happening, thus removing the chance of reading wrong data.

3.1.6 Result Writer Subsystem

The project also required quantitative data to measure performance differences between the different input methods, which meant that a result writing subsystem had to be implemented. The system was implemented as a game instance subsystem, that features functionality for writing test results to a Comma Separated Values (CSV) file. The subsystem has functions for beginning the writing of a test task, this specifies the configuration used and stores it in a variable, beginning writing test objective, which accepts the name of the objective and marks a point in the result csv file, writing test results, which takes an appropriately formatted string for the results and ending the writing of test objectives and tasks, once finished writing a task, the system will output everything to a file. This system is fairly simple as all it does is store the configuration and an array of strings for each task and objective, upon finishing a task, it checks results directory for the number of files and diving it by 3 to find the participant number, after which creating the file and writing the data into it.

3.2 Hardware

3.2.1 Virtual Reality Hardware

To allow for users to enter VR and interact with the virtual world, the Meta Quest 3 headset was used, because it has a good hand tracking solution, is readily available, easy to use and develop for, as it has many development samples, software development kits and tools.

3.2.2 Haptic Feedback Gloves

For the project to provide haptic feedback in the hand tracking input configuration a haptic feedback device was designed and developed through an iterative process. It provides feedback by pulsing vibration motors that are attached to the glove at the tip of each finger.

3.2.2.1 Initial Design

The initial design of the haptic feedback glove featured an Arduino Uno Rev3 mainboard, with an Arduino Motor Shield Rev3 on top of it, connected to a lithium-ion battery, that sat under both and provided power. This device stack was then placed in a 3D printed box. Even though the setup succeeded in providing haptic feedback and was used as a development device so that the haptic system included in the application could be tested, it was way too big to be placed on top of a glove and provide a comfortable experience to the user, going from personal observations (Figure 2). This led to a redesign of the containing box, adding a belt attachment, so that the box can be attached to a belt and have wires running from the belt back to the computer and forward to the vibration motors on the gloves. After building one of those boxes, it became apparent that the design is not good enough and another solution must be explored.

Figure 2 Initial Hardware Device

3.2.2.2 Final Design

While exploring alternatives for the haptic feedback gloves, and having multiple conversations with the university technician Gerald High, it was decided to use ESP32-S3-Nano as the mainboard, as it was much smaller and had a richer feature set than the Arduino Uno Rev3. After some testing to ensure that enough power can be provided to the motors a final design was set on. The design featured a lithium-ion battery on the bottom with a 3D printed separator, also designed to hold a WEMOS Battery Shield 1.2.0, allowing the battery to be charged with a USB Micro cable, followed by a layer of 5 n-type MOSFETs mounted on a 3D printed holder and the ESP32-S3-Nano placed on a 3D printed base. This new device stack was placed inside a 3D printed box, with a cutout for pin-out holes that can be connected to the motors that were attached on the tip of the fingers on the glove using superglue. The box was Velcro-strapped to the back of the glove allowing for running shorter cables to the motors. This device stack was much smaller than the initial one as you can see in Figure 3, featured Wi-Fi, the ability to charge the battery, and was strapped straight to the glove, removing all the issues the initial design had as well as providing some nice improvements. But a key limiting factor was the thickness of the vibration motor wires, which would snap during some of the test sessions, meaning that haptic feedback feeling was reduced, but remained present.

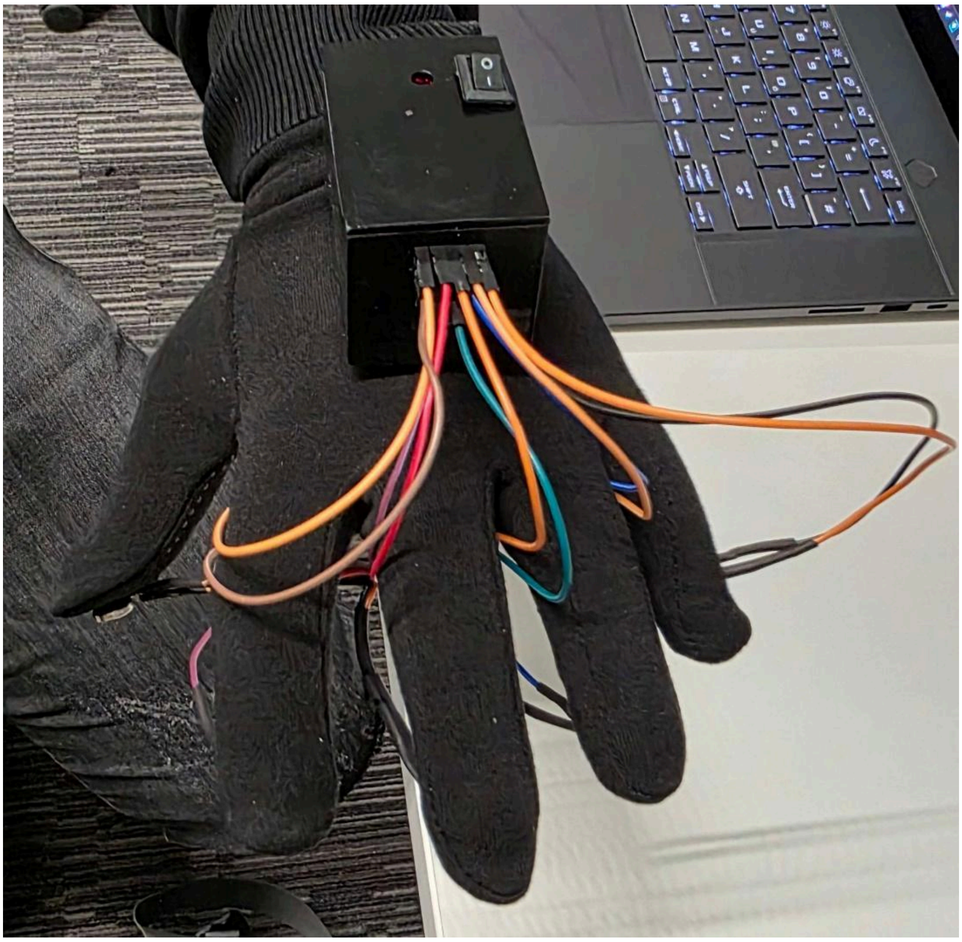

Figure 3 Final Haptic Device

3.2.2.3 Software of the Haptic Device

The software running on the hardware device was written and uploaded to the device using the Arduino IDE. It works by first setting up a structure for the haptic feedback effect that is currently applied, opening the SerialCom port for communication and setting pin modes to OUTPUT for the pins that are controlling the vibration motors. After this the device operates in a loop where it attempts to process data by reading the serial port and if there are at least 8 bytes available to be read it would setup the feedback effect, by reading the appropriate bytes into the appropriate variables. After this the loop continues by iterating over all of the effects and updating the pin values controlling the vibration motors with an analog write and then waiting for a delay of 20 milliseconds ensuring that the device does not overheat by reducing the amount of wasted computation cycles. The 20 milliseconds were chosen as through trial-and-error testing it was determined to be a fast enough update rate to where the delay was not noticeable, but at the same time it did not run too many times, wasting power.

3.3 Evaluation

Finally, to determine the usability of each input configuration, qualitative and quantitative data was collected. All data collected was anonymised and participants were given a number so that their responses can be corelated with the demographic survey.

3.3.1 Questionnaire

In order to gather qualitative data, the system usability scale (SUS) questionnaire was used (Brooke, 1995), because it is a standard and frequently used survey to gauge the usability of systems. It is a Likert scale, intended to be answered quickly and no answers must be omitted. After receiving the participant answers, it is scored 1-100 by taking the positions of answers 1, 3, 5, 7, 9 minus 1, answers 2, 4, 6, 8, 10 contribute 5 minus their scale position, after this their values are summed and multiplied by 2.5 to get the final score. A score of 51 and below means that the system is not usable, 52-80 means that the system is somewhat usable, but should be improved, and 81 and above means that the system is very usable, and there is little room for improvement. (Healy et al., 2024; Oana Rotaru et al., 2020).

3.3.2 Quantitative data

The application collects quantitative data through the test sessions, saving the results in an anonymised format, giving each participant a number, to allow for corelation of results to questionnaire answers and demographic survey answers. Averages of each collected data point will be calculated and compared between different input devices, to examine if the system usability was a placebo effect or just a novelty of experiencing a new way of interacting with the VEs.

Chapter 4 Results

To determine the usability and user preference between Motion Controllers and Hand Tracking, results from a System Usability Scale (SUS) questionnaire, and quantitative data throughout the testing sessions was gathered. The project examines the results of 8 participants, which allowed for testing every possible combination of input devices at least once. A demographic survey was also present, showing that 6 out of 8 of the participants had previous experience with virtual reality and using motion controllers, but only 3 out 8 had used hand tracking as an input modality.

4.1 System Usability Scale Results

In order to follow the SUS requirements, each participant was asked to fill in the survey right after testing with each input configuration and was instructed that they might not omit any answers, if they could not answer any questions they could either withdraw from the study or select the neutral option, and to answer as quickly as possible without thinking much about each statement. Their answers were then collected in an Excel spreadsheet and the SU Score was calculated with the appropriate formula. Results show that participants found the hand tracking configuration to be the most usable with an mean score of 90.94, closely followed by motion controllers with a mean score of 87.5, and the haptic hand tracking configuration being the least usable with a mean of 72.5, and a lower median of around 67. These results can be seen in Figure 4. Significance analysis was not performed as there were only 8 participants present, which would make the results unreliable.

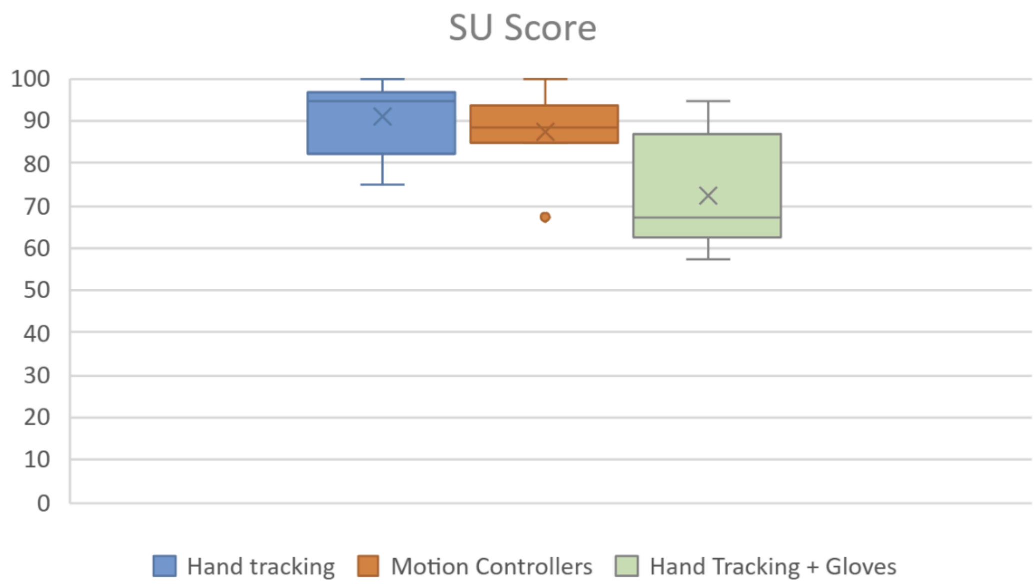

Figure 4 SU Score Box Chart (X - mean)

4.2 Quantitative Data

During the testing sessions quantitative data was collected while participants were going through the tasks. Every task objective recorded different data ranging from position deltas and time taken to complete to number of correctly sorted objects. Results show that participants overall performed slightly better with motion controllers compared to regular hand tracking and much better when comparing to the haptic hand tracking configuration, possibly due to issues with tracking that the box on top of the gloves introduced. These results can be seen in the appendix. The haptic hand tracking input method was also the only one where participants incorrectly sorted cubes and featured an extremely high maximum distance delta during the follow the path task. After performing ANOVA statistical analysis because it can be used to analyse variables spread across more than two configurations (Kim, 2014), it was determined that there were statistically significant overall differences between all configurations as it was tested using multiple input variables(in Pyramid and Ico sphere alignment task with p values of 0.0167 and 0.0174 respectively, time taken to complete, average position deltas and max distance delta were also statistically significant during the follow the target task with p values of 0.0046, 0.0167 and 0.0079. There was no significant difference found only in the sphere location alignment task.

4.3 Summary

Hand tracking was perceived as the most usable solution by users judging from the SUS questionnaire and participants' feedback, but quantitative data proved that when using motion controllers participants performed slightly better, but the small differences in performance, could be credited to the overall lesser experience with hand tracking from most participants. The haptic feedback hand tracking input configuration had the worst performance and usability score, but participants did mention that the feedback sensation was a good addition, and they preferred to have haptics rather than not.

Chapter 5 Discussion

5.1 Motion Controllers

Going from the quantitative data, motion controllers had the best overall performance, which aligns with most of the previously mentioned research like the ones by Luong (2023), Healy (2024) and Han (2020). Data showed that motion controllers had small position deltas in the object alignment task, no incorrectly sorted objects; lowest XY position delta during the cube stacking task, and lowest average position delta during the follow the spline task, as well as fastest time to complete tasks on average. The user performance could be credited to previous exposure and familiarity with motion controllers as they are the default intended input device for the majority of VR applications, and controllers in general, as gaming controllers are a widely used input modality and have similarities with motion controllers, like the thumb sticks and buttons. Experience with regular gaming controllers can translate to a quicker familiarisation process and understanding of the device as shown by Boletsis and Cedergren (2019). Participants also mentioned that the haptic feedback provided by the motion controllers helped them understand if the object was grabbed and then proceeded to perform the required manipulation of it, which can have a positive impact on performance, usability and user preference as proven by Moon, Orr and Jeon (2022).

While motion controllers were the best performing input configuration, it was not rated as the most usable on the SUS questionnaire by users on average, which confirms the findings from previous studies like the comparison made by Luong (2023) where hand tracking was the preferred interaction method in touch-based interactions. This might be credited due to the requirement to use buttons to interact with the VE and the objects within it, as well having to keep holding a button down to continue the interaction. Another possible factor that might have contributed to motion controllers being rated as less usable can be the misaligned visuals, as users saw controllers instead of their own hands or an accurate representation of it.

5.2 Hand Tracking

Hand tracking received the highest system usability score on the SUS questionnaire (Figure 5), as such it is regarded as the most usable input modality for the activity of small object manipulation in VEs, which shows that the improvements hand tracking has received have been sufficient and can now be used as the default input modality, contradicting the results shown by Johnson (2023). This can be credited to a couple of factors, starting with the accurate representation of the users' hands in the VE, which helps them understand the action they are going to or are currently performing much easier, it also provides the opportunity for translating ingrained knowledge of how to interact with objects from the real world to the virtual one, a pinch action or a fist grab is much more intuitive than the press of a button. Another contributing factor can be not having to think about the button layout of the device that is currently used, which can lead to confusion and the need to remove the HMD to check where the required button is located. Finally, not having to hold onto controllers can improve the usability, as it frees your hands to perform the natural motions humans are accustomed to, as well as making it more accessible for people with disabilities, something this project was unable to examine.

Even though hand tracking was regarded as the most usable input modality by the participants for the purposes of this project, it was closely followed by motion controllers, which is also the device participants performed best with. This can be credited to the lack of haptic feedback present in the hand tracking setting and the limitations of hand tracking, requiring users to keep their hands in view of the HMD's cameras and sensors, as well being slightly slower than the motion controllers one.

5.3 Haptic Hand Tracking

The haptic hand tracking input configuration was both rated as the least usable, with a score just under what would be considered a usable system of 72.5, and had the worst participant performance being the only input configuration where participants incorrectly sorted an object and overall, much worse results recorded from every task, with nearly double the time taken to complete the follow the spline task. Some usability and performance issues can be credited to the reduction in tracking accuracy, due to the addition of the box housing the haptic hardware on top of the glove. This object disrupted the machine learning algorithm and sometimes obstructing view to key points required for the algorithms to function properly, making it harder to analyse the image received and find the correct bounding box of the hands and identifying the finger joint rotations, leading to slow updates, wrong poses, and even loss of tracking at certain points. Another contributing factor to the bad rating and performance is the weight and size of the housing of the hardware. It was made as small as possible with the given timeframe, but it was still too large and heavy for its intended purpose.

Although haptic hand tracking performed poorly, it did receive a much higher usability score than expected after seeing all the tracking issues it presented. The high score can be credited to the haptic feedback provided by the gloves as most participants reported that feeling a haptic sensation when interacting with the VEs through their hands felt incredible. If tracking issues due to the addition of the hardware device were to be resolved through the development of a smaller and lighter device that does not impact the effectiveness of hand tracking algorithms, it may become the most usable solution.

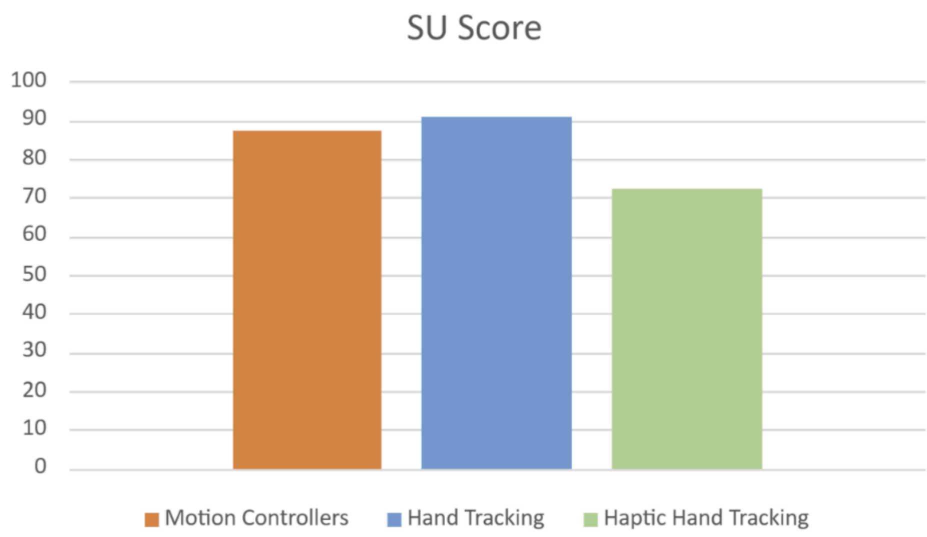

Figure 5 Averaged System Usability Score

5.4 Summary

To summarise, these results align with the results shown by Healy (2024), Luong (2023), Masurovsky (2020), Khundam (2021) and Noghabaei (2020), while also showing that results from other studies comparing hand tracking and motion controllers in tasks that include fast movements, like Johnson (2023) are not as relevant in 2025, as the performance of hand tracking algorithms has improved. This proves that while motion controllers are statistically and physically a better device, are not generally the preferred method of interaction, while the haptic hand tracking input method, was not developed to a point where it can be a competitive alternative, but haptics is definitely an area that should be explored further, as users did mention that haptics were a good addition, but the reduction in tracking accuracy had a bigger impact on usability and user preference as indicated by Moon, Orr and Jeon (2022).

Chapter 6 Conclusion and Future Work

6.1 Overview

Overall, the results from the project were not surprising, except for the usability rating of the haptic hand tracking configuration, due to the observed precision and tracking quality reduction. Hand tracking and motion controllers being close competitors was expected seeing as previous papers had similar results and users performing better when using motion controllers was also an expected result, once again referring to the previously mentioned papers having similar results. Although it seems like earlier papers by Johnson (2023), Rantamaa (2023) and Healy (2024), found a bigger difference in the performance metrics of users utilizing hand tracking and motion controllers, the gap between the two appears to be narrowing and hand tracking becoming more equivalent as algorithms improve in terms of speed and accuracy, due to improvements in both software and hardware.

The SUS score of the haptic hand tracking was surprisingly high, as it faced many tracking issues, like obstructing fingers and altering the shape of the hand, thus confusing the algorithms, causing unexpected behaviour and even loss of tracking throughout the test sessions. The higher usability score can be credited to the haptic feedback that the device provides, aligning with the results found by Hye Sung Moon, 2022.

6.2 Limitations

Some key limitations of the study include, the small sample size of participants, although 8 is an acceptable amount, 12 would have been perfect, as that way all different input combinations would have been tested at least once, thus removing any bias can might be built from using the input methods in different order.

Another limitation was the thickness of the wires that were connected to the vibration motors, as some would snap in the middle of a testing session, thus reducing the amount of haptic feedback felt, which can negatively affect the performance and user perception of the usability of the device.

Finally, the last key limitation was that in the motion controller input configuration collision on the controllers had to be disabled when an object is grabbed, even in the cube stacking where physics should have been present, this was to remove the problem of objects being push-out from the controller's collision with very high speeds. Collision was still present on the cube and when an object was not grabbed.

6.3 Future Work

The current design of the haptic feedback device while smaller and lighter than the initial, is still too large and heavy, disrupting the tracking algorithms and encumbering the user with unnecessary load. Most of the problems come from time constraints, not being able to find or design a custom printed-circuit board (PCB) that can deliver the necessary amount of power so that all motors can be driven with just a PCB and a battery connected to it for power. This should be the ideal solution as it would be small enough to not affect the hand tracking algorithms, as well as being light enough to not be noticeable when wearing. Another limiting factor throughout the experiments was the need for using cables instead of Wi-Fi for communication between the haptic device and the VR application. Using cables can limit the mobility of the users and provide a feeling of unease as you can damage equipment, while when using Wi-Fi those limitations are removed, which can lead to an even better usability and performance scores.

6.4 Summary

Going back to the proposed question at the start of the project: *How does hand tracking compare to motion controllers when manipulating small objects in virtual environments in terms of intuitiveness and user preference?*

It can be answered as hand tracking is now a comparable alternative to motion controllers when manipulating small objects in VEs in both intuitiveness and user preference, indicated by the results from this research, where hand tracking had higher system usability scale score,

but performed slightly worse than motion controllers when comparing quantitative in-application user performance data.
To summarise, with the current state of hand tracking algorithms, hand tracking has improved to a point where it can be the default input method for VR applications, especially when slow movements are involved. A way for hand tracking to keep moving forward other than improving the algorithms that make it possible, is by introducing a way to provide haptic feedback, the obvious and currently most achievable implementation is through designing a haptic feedback device in the form of a glove. Providing haptic sensations to users while using hand tracking will most likely improve its usability, indicated by the feedback provided by participants while using the device, all of them enjoyed the feedback sensation, but the poor tracking drastically reduced its usability, thus receiving a lower rating than the other input methods.

List of References

Witmer, B.G. and Singer, M.J. (1998) ‘Measuring Presence in Virtual Environments: A Presence Questionnaire’, *Presence: Teleoperators and Virtual Environments*, 7(3), pp. 225–240. Available at: https://doi.org/10.1162/105474698565686.

Healy, P., Dmitriy Babichenko, Orebaugh, S., Haase, M.N. and Fuller, Z. (2024) ‘Comparing the Use of Controllers and Hand Tracking in a Virtual Reality Nerve Block Simulation’, pp. 1–8. Available at: https://doi.org/10.1109/segah61285.2024.10639607.

Noghabaei, M. and Han, K. (2020) ‘Object manipulation in immersive virtual environments: Hand Motion tracking technology and snap-to-fit function’, Automation in Construction, 124(103594). Available at: https://doi.org/10.1016/j.autcon.2021.103594.

Luong, T., Cheng, Y.F., Mobus, M., Fender, A. and Holz, C. (2023) Controllers or Bare Hands? A Controlled Evaluation of Input Techniques on Interaction Performance and Exertion in Virtual Reality, IEEE Explore. Available at: https://ieeexplore-ieee-org.libproxy.abertay.ac.uk/stamp/stamp.jsp?tp=&arnumber=10269728&tag=1.

Rantamaa, H.-R., Kangas, J., Kumar, S.K., Mehtonen, H., Järnstedt, J. and Raisamo, R. (2023) ‘Comparison of a VR Stylus with a Controller, Hand Tracking, and a Mouse for Object Manipulation and Medical Marking Tasks in Virtual Reality’, Applied Sciences, 13(4), p. 2251. Available at: https://doi.org/10.3390/app13042251.

Johnson, C.I., Fraulini, N.W., Peterson, E.K., Entinger, J. and Whitmer, D.E. (2023) ‘Exploring Hand Tracking and Controller-Based Interactions in a VR Object Manipulation Task’, pp. 1–18. Available at: https://www.researchgate.net/publication/376071730_Exploring_Hand_Tr acking_and_Controller-Based_Interactions_in_a_VR_Object_Manipulation_Task

Masurovsky, A., Chojecki, P., Runde, D., Lafci, M., Przewozny, D. and Gaebler, M. (2020) ‘Controller-Free Hand Tracking for Grab-and-Place Tasks in Immersive Virtual Reality: Design Elements and Their Empirical Study’, Multimodal Technologies and Interaction, 4(4), p. 91. Available at: https://www.mdpi.com/2414-4088/4/4/91

Khundam, C., Vorachart, V., Preeyawongsakul, P., Hosap, W. and Noël, F. (2021) ‘A Comparative Study of Interaction Time and Usability of Using Controllers and Hand Tracking in Virtual Reality Training’, Informatics, 8(3), p. 60. Available at: https://www.mdpi.com/2227-9709/8/3/60

Han, S., Liu, B., Cabezas, R., Twigg, C.D., Zhang, P., Petkau, J., Yu, T.-H., Tai, C.-J., Akbay, M., Wang, Z., Nitzan, A., Dong, G., Ye, Y., Tao, L., Wan, C. and Wang, R. (2020) ‘MEgATrack’, ACM Transactions on Graphics, 39(4). Available at: https://research.facebook.com/publications/megatrack-monochrome-egocentric-articulated-hand-tracking-for-virtual-reality/

Taketomi, T., Uchiyama, H. and Ikeda, S. (2017) ‘Visual SLAM algorithms: a Survey from 2010 to 2016’, IPSJ Transactions on Computer Vision and Applications, 9(1). [Online] Available at: https://link.springer.com/article/10.1186/s41074-017-0027-2

Valve (2020) *Half-Life: Alyx* [video game], Valve. Available at: https://store.steampowered.com/app/546560/HalfLife_Alyx/ (Accessed: 18 March 2025)

Shambala virtual reality technology (2024) *Drakheir The First Hand Tracking ARPG* [video game], Gallantry Games. Available at: https://www.meta.com/en-gb/experiences/drakheir-the-first-hand-tracking-arpg/8234180549963919/?srsltid=AfmBOorV2MWKLlwa5eQZmSV7ayJ5g4P8A0KcLWdl-I5vwdtWiLAZVLbM (Accessed: 18 March 2025)

Meta (2024b) *Meta Quest v72 Update: Seamless Remote Desktop, Better Hand Tracking, and More | Meta Quest Blog*, *Meta.com*. Available at: https://www.meta.com/en-gb/blog/meta-quest-v72-update-remote-desktop-hand-tracking-keyboard-more/?srsltid=AfmBOopthel7L0wjz6pIX9CTC8Ti_LcKzd4ZTVEKI0UmUqmJcCChelmJ (Accessed: 18 March 2025).

Heaney, D. (2025) Quest 3S Was The Top Selling Console On Amazon US In 2024, Uploadvr.com. Available at: https://www.uploadvr.com/quest-3s-top-selling-console-amazon-2024/ (Accessed: 7 April 2025).

Rolland, J. and Hua, H. (2005). Head-Mounted Display Systems. [online] doi:https://doi.org/10.1081/E-EOE-120009801.

Oana Rotaru, Vert, S., Andone, D. and Andone, D. (2020) 'Standardised Questionnaires in Usability Evaluation. Applying Standardised Usability Questionnaires in Digital Products Evaluation', Information and Software Technologies. ICIST 2020. Communications in Computer and Information Science, 1283, pp. 39–48. Available at: https://doi.org/10.1007/978-3-030-59506-7_4.

Healy, P., Dmitriy Babichenko, Orebaugh, S., Haase, M.N. and Fuller, Z. (2024) 'Comparing the Use of Controllers and Hand Tracking in a Virtual Reality Nerve Block Simulation', pp. 1–8. Available at: https://doi.org/10.1109/segah61285.2024.10639607.

Epic Games Developer. (N.D.). UGameInstance | Unreal Engine 5.5 Documentation | Epic Developer Community. [online] Available at:

https://dev.epicgames.com/documentation/en-us/unreal-engine/API/Runtime/Engine/Engine/UGameInstance.

Brooke, J. (1995). SUS: a Quick and Dirty Usability Scale. [online] ResearchGate. Available at: https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale.

Epic Games (2024) Unreal Engine 5.4 is here! Find out what's new., Unreal Engine. Available at: https://www.unrealengine.com/en-US/blog/unreal-engine-5-4-is-now-available.

https://communityforums.atmeta.com/t5/user/viewprofilepage/user-id/1440692 (2024). Meta Quest build v69 release notes. [online] Atmeta.com. Available at: https://communityforums.atmeta.com/t5/Announcements/Meta-Quest-build-v69-release-notes/ba-p/1237980 (Accessed 24 Apr. 2025).

Kim, H.-Y. (2014). Analysis of Variance (ANOVA) Comparing Means of More than Two Groups. Restorative Dentistry & Endodontics, 39(1), p.74. doi:https://doi.org/10.5395/rde.2014.39.1.74.

Meta (2021). Quest 'Hand Gameplay Showcase'. [online] Meta.com. Available at: https://developers.meta.com/horizon/blog/quest-hand-gameplay-showcase/ (Accessed 24 Apr. 2025).

Oca, R.M.D. (2024). Serial Communication Plugin for Unreal Engine. [online] GitHub. Available at: https://github.com/videofeedback/Unreal_Engine_SerialCOM_Plugin (Accessed 24 Apr. 2025).

Boletsis, C. and Cedergren, J.E. (2019). VR Locomotion in the New Era of Virtual Reality: An Empirical Comparison of Prevalent Techniques. Advances in Human-Computer Interaction, 2019, pp.1–15. doi:https://doi.org/10.1155/2019/7420781.

Moon, H.S., Orr, G. and Jeon, M. (2022) ‘Hand Tracking with Vibrotactile Feedback Enhanced Presence, Engagement, Usability, and Performance in a Virtual Reality Rhythm Game’, International Journal of Human–Computer Interaction, pp. 1–12. Available at: https://doi.org/10.1080/10447318.2022.2087000.

Appendices

Appendix A Box Charts for User Performance Data

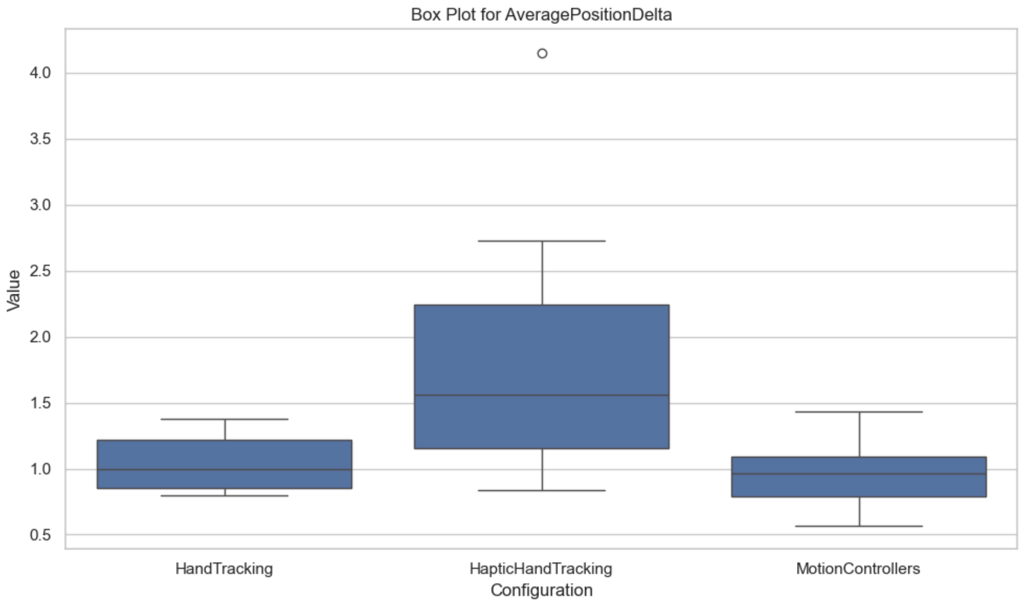

Figure 6 Average Position Delta During Object Alignment Task Box Plot

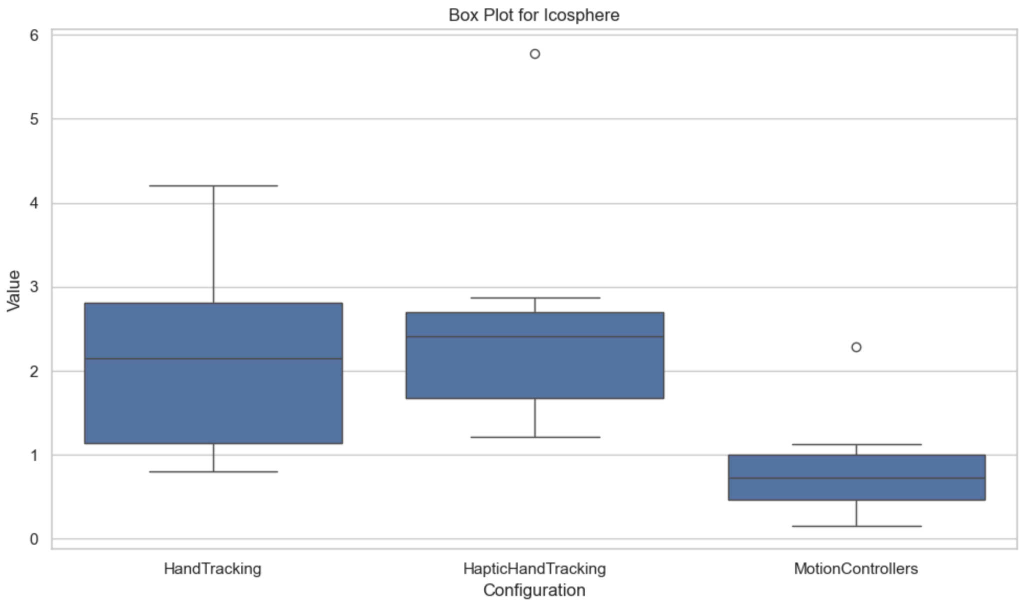

Figure 7 Icosphere Position Delta Box Plot

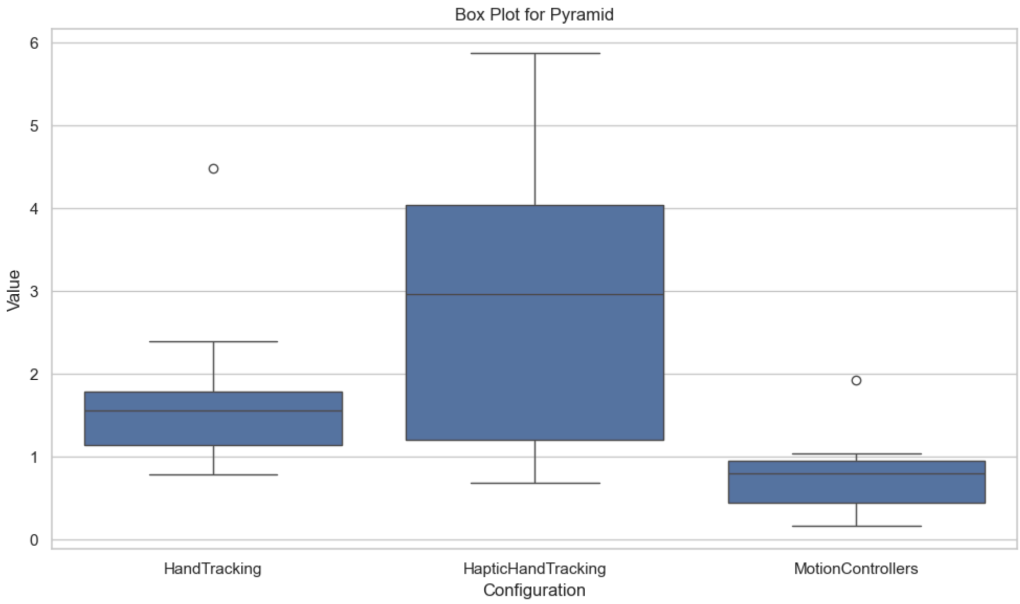

Figure 8 Pyramid Position Delta Box Plot

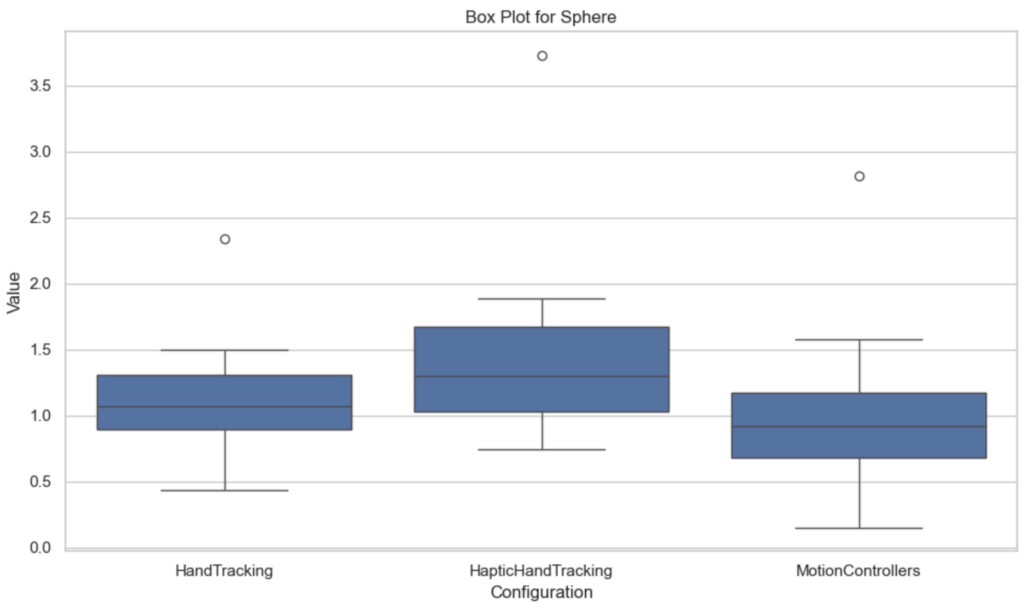

Figure 9 Sphere Position Delta Box Plot

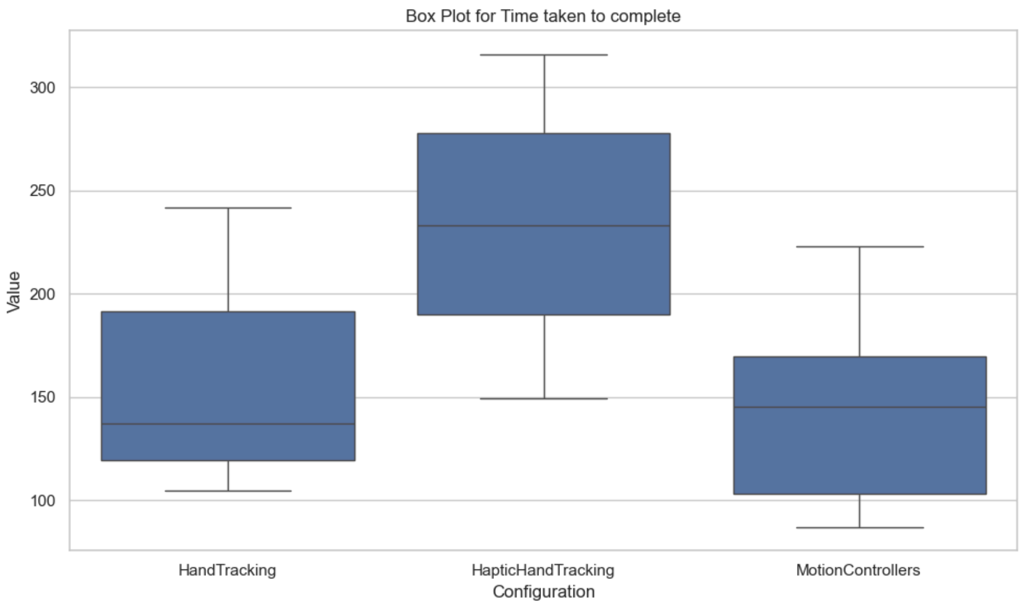

Figure 10 Time Taken to Complete Object Around Spline Task Box Plot

Appendix B Table of Statistical Significance Analysis

Variable	Test	Comparison	p-value	Significant
Sphere	ANOVA	All configurations	0.44175	No
Pyramid	ANOVA	All configurations	0.01674	Yes
Icosphere	ANOVA	All configurations	0.01742	Yes
Time taken to complete	ANOVA	All configurations	0.00463	Yes
AveragePositionDelta	ANOVA	All configurations	0.01668	Yes
MaxDistanceDelta	ANOVA	All configurations	0.00788	Yes
TotalPositionSamples	ANOVA	All configurations	0.96812	No
TotalPositionDelta	ANOVA	All configurations	0.228	No