A Comparative Evaluation of Walking Methods for Reducing Simulator Sickness for a High-Fidelity Archery Virtual Reality Simulation

Exploring the use of haptic feedback to reduce simulator sickness in virtual reality

Hayden Norrie
University of Cape Town
Cape Town, South Africa
NRRHAY001@myuct.ac.za

Figure 1: Robin Hood Virtual Environment

ABSTRACT

Virtual reality has become a more developed and advanced technology and is only geared to grow more in the future. Furthermore, virtual reality is becoming more widely available to consumers rather than enthusiasts only who are willing to pay high prices to use new virtual reality equipment. With companies releasing cheaper consumer grade headsets virtually reality is becoming more accessible to the average person. Given this greater availability of consumer grade virtual reality equipment it is imperative to address the major factors preventing a further rise in popularity of virtual reality. This paper aims to address the major issue regarding virtual reality, simulator sickness. Simulator sickness is often caused by locomotion in virtual reality and results in symptoms like headaches, nausea and dizziness. This paper aims to address simulator sickness by designing a new locomotion method that relies on a form of haptic feedback, called the cradle locomotion method. The cradle method was compared against two other common locomotion methods, namely the arm-swinging and teleportation methods. Through a series of human trials (n = 22) we exposed participants to all three locomotion methods(treatments) in a archery virtual environment named the Robin Hood Environment. Using the SSQ, GEQ, and target hit ratio as measurements no statistical significance ($\alpha = 0.05$) was found for SSQ and GEQ components for each locomotion method. We found that the cradle reduced performance.

KEYWORDS

Virtual reality, Simulator sickness, Locomotion, Haptic feedback, Archery

1 INTRODUCTION

Over it lifespan Virtual Reality(VR) has gained greater acceptance by the general public. Nowadays it has achieved its status as a usable and easily accessible technology with companies releasing cheaper consumer grade headsets. Contributing to this wider acceptance and gain in popularity is new computing hardware with increased power and decreased price, such as the NVIDIA 3000 graphics card series, which with its increased power can easily take on running virtual reality content. More so, with advancements in technology we now find cheaper and more powerful stand-alone VR headsets like the Oculus Quest 2, a tetherless virtual reality device. However, there is a significant set back against the wider adoption of VR, simulator sickness. Simulator sickness is defined as a subset of motion sickness[12]. When a user experiences simulator sickness they often experience symptoms such as nausea, headaches, and dizziness and other symptoms being similar to that of motion sickness. It is often experienced during VR experiences, but not always. Most often it can be caused by locomotion in VR. There are three underlying theories explaining why one would experience simulator sickness.

(1) Sensory Conflict Theory
(2) Postural Instability Theory
(3) Poison Theory

The Sensory Conflict Theory was created at a time when there was no understanding of motion sickness's mechanisms. It suggests that sensory theory can be modelled mathematically. Furthermore, it states that if stimuli from the outside environment are being perceived differently by different human senses, as often is the case with virtual reality, simulator sickness will ensue and symptoms

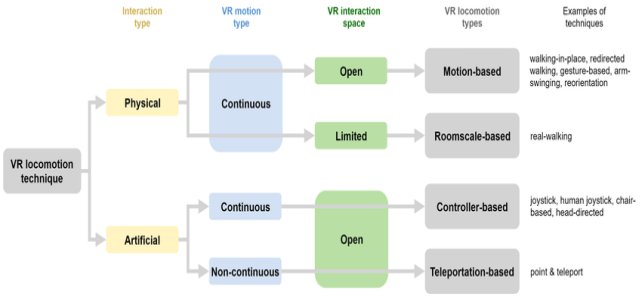

Figure 2: Boletsis' and Cedergen's Locomotion Technique Typology[3][4]

will then occur[14]. Previous research has shown that there are a variety of factors that affect the degree to which one experiences simulator sickness, both hardware and individual factors.

For this study the sensory conflict theory was used as the underlying theory for our technical and design choices, as will be explained in section 3.

According to Boletsis' and Cedergen's Locomotion Technique Typology[3] locomotion methods can be clearly defined into four distinct categories that will be further explored under section 2:

Using the sensory conflict theory and what we already know about locomotion this research was focused on designing and developing a locomotion system that takes advantage of haptic feedback when used, and to compare this novel method against a motion-based locomotion method and a controller-based locomotion method, arm-swinging and teleportation respectively, both of which are industry standard locomotion methods. Thus, we will compare the novel method against both a motion-based method and a controller-based method.

The haptic feedback in the novel method aligns itself with sensory conflict theory in that it tries to align what the different senses, in this case touch and sight, experience. This new method was named the 'Cradle Method'. The cradle method is a new controller-based method, though not a controller in the classic VR sense. In this method a user is surrounded by a ring which controls locomotion. This locomotion method makes use of haptic feedback by having the ring push back against a user when a user walks into it. Thus, by using the haptic feedback from the cradle method we aim to reduce simulator sickness when using locomotion in virtual reality. Thus we can define our main research questions and further define our research hypotheses:

(1) **Aim:** Will a novel locomotion method, namely the cradle method, result in lower simulator sickness scores after executing tasks in the RHE in comparison to arm-swinging and teleportation methods?
Hypothesis 1.1: The cradle method will reduce simulator sickness in an archery based virtual environment in comparison to the arm-swinging method.
Hypothesis 1.2: The cradle method will reduce simulator sickness in an archery based virtual environment in comparison to the teleportation method.
(2) **Aim:** Will a novel locmotion method, namely the cradle method, result in greater performance scores after executing tasks in the RHE in comparison to arm-swinging and teleportation methods?
Hypothesis 2.1: The cradle method will result in higher performance scores in an archery based virtual environment in comparison to the arm-swinging method.
Hypothesis 2.2: The cradle method will result in higher performance scores in an archery based virtual environment in comparison to the teleportation method.

2 BACKGROUND AND RELATED WORK

2.1 Locomotion in VR

VR. Boletsis' and Cedergen's Locomotion Technique Typology[3] clearly defines the different types of locomotion in virtual reality categorized into four locomotion types.

(1) Motion-based
(2) Room-scale-based
(3) Controller-based
(4) Teleportation-based

2.1.1 Motion-based. Motion-based locomotion makes use of a user physically moving parts of their body while remaining in place. This movement is tracked and gets translated into movement in the virtual environment. These methods can use basic trackers such as HTC's Vive Tracker or trackers that come with a given headset, thus making tracker integration fairly easy. Examples of motion-based locomotion methods are the arm-swinging and walking-in-place methods. In arm-swinging the arms of a user are tracked and are used to cause locomotion in the virtual environment. The walking-in-place method tracks the legs of a user and that is then translated into locomotion in the virtual environment. This can be done using simple trackers or even an omni-directional treadmill, like the Omni by Virtuix. Motion-based locomotion is useful when one has a VR setup in a small room as one remains in place and thus physical boundaries do not pose an issuesidco.

2.1.2 Room-Scale-based. Room-scale-based locomotion requires a user to walk within a space in real life and their exact movement is translated into the virtual world. Thus, the physical size of a room limits the distance one can traverse in a virtual environment and therefore limits someone designing a VR environment for a room-scale-based locomotion as well. VR systems often include to measures to prevent users from walking into the real life boundaries. An example of this is the chaperone system within the HTC Vive, where a virtual boundary appears when you are about to walk into a physical boundary in real life.

2.1.3 Controller-based. Controller-based locomotion uses common controller hardware like touch pads and joysticks to enable movement in a virtual environment. This locomotion type can also use unconventional controllers as will be seen with the cradle method. Interaction with the controller translates into locomotion in the virtual environment. One can be limited to where one can move in the virtual environment by the software itself by using either typical programming techniques or more complicated methods like Unitys Navigation Mesh. This method is thus suited to when one is limited by the size of a physical room or space.

2.1.4 Teleportation-based. Teleportation-based locomotion is a very common way to navigate a virtual environment fast. It requires a user to point their controller at the point in the virtual space in which they wish to travel. After some controller interaction they will appear at the point they were pointing at. Thus, teleportation is non-continuous. This method is thus suited to when one is limited by the size of a physical room or space.

2.2 Simulator Sickness Triggers

Simulator Sickness, a subset of motion sickness[12], is often experienced by a user during a virtual reality experience for a variety of different reasons and resembles motion sickness in its symptoms. Users often experience motion-sickness-like symptoms such as nausea, headaches, dizziness, stomach bloating and more. There are many factors that can trigger simulator sickness as are discussed below.

2.2.1 Technical Factors. We see from previous research that one of the greatest technical factors contributing to the experience of simulator sickness is the field of view of the VR headset. It has been shown that an increased field of view directly correlates with an increase in simulator sickness.[19]

We also find that latency contributes towards the amount of simulator sickness a user will experience[7]. Latency is what a user experiences when there is a lag between their input and what is rendered on their screen. Thus, we would expect this given it aligns with sensory conflict theory.

2.2.2 Application Design Factors. For this research we used a virtual environment that we named the Robin Hood Environment. Different factors regarding a virtual environment can influence simulator sickness

Though there has been little research on realism and its affects on simulator sickness, a recent study has shown that, contrary to what one may think, higher levels of realism result in higher levels of simulator sickness[16].

Vection is defined as the sensation of movement of the body in space produced purely by visual stimulation. Thus, vection would be a very common idea to deal with in VR. Vection is highly correlated to the experience of simulator sickness[5][9]. Thus, when one is immersed in a virtual environment they experience movement visually while their body remains still. While a user remains still the vestibular system indicates that a users body is remaining still while they see that they are moving. According to sensory conflict theory, this sensory conflict is the cause of simulator sickness.

2.2.3 Individual Factors. Humans are defined greatly by their individual characteristics. These characteristics have been shown to affect the amount of simulator sickness an individual will experience.

Age has shown to contribute greatly to the degree to which one will experience simulator sickness, with older folk being affected most. This is theorized to be due to an older person slower visual processing[11][15]. A more recent study has challenged this idea and has shown that younger people experience SS more than the older[18].

Results from multiple studies have also shown females to be more susceptible to SS than men[1][8][2].

2.3 Simulator Sickness Measures

In doing research regarding simulator sickness it is essential there be a standardized way to measure ones experienced level of simulator sickness. Measuring simulator sickness can be either objective or subjective. In this section we will analyze existing methods, both subjective and objective, to measure simulator sickness.

2.3.1 Subjective. The theory of simulator sickness was been developed in 1968 and quarter century later, in order to measure simulator sickness more accurately, the Simulator Sickness Questionnaire(SSQ) was developed by Kennedy et al.[17]. The SSQ expanded onto the original measure of simulator sickness, the MSQ. The MSQ did not do any form of factor analysis and therefore it did not take into account the different factors of motion sickness[13] and for this reason it was not an accurate measure of simulator sickness. For this reason Kennedy et al. developed the SSQ that is now used in measuring simulator sickness in most VR research. One of the main aims of the paper by Kennedy et al. was "to provide a more valid index of overall simulator sickness severity as distinguished from motion sickness"[17]. According to research, the SSQ is the most common tool for measuring simulator sickness[10].

The SSQ requires one to rate 16 symptoms on a level from one to four. After evaluating the questionnaire the SSQ provides an overall score to describe the level of simulator sickness a participant experiences. More so, it can describe the effects on the sub-components that make up simulator sickness: nausea, oculumotor and disorientation.

2.3.2 Objective. Though the SSQ is most commonly used in simulator sickness research[10], it is based on what a user reports they felt, and can therefore be inaccurate simply due to human error or bias. This begs the question, is there a more accurate/precise way one can measure simulator sickness? The idea of using physiology to determine a measure for simulator sickness has been suggested in older research[6]. Some studies have even used physiological measures like blink rate and heart rate to determine simulator sickness[13]. For this research, due to time constraints and equipment limitations, physiological measures were not taken.

3 DESIGN AND IMPLEMENTATION

For a full outline of the division of labour for this research, please see Appendix C

3.1 The Environment

Given the project aim of evaluating simulator sickness for different locomotion methods, the virtual environment that our participants will traverse should be designed in such a way that it will induce a base level of simulator sickness. This is done such that we can compare all three locomotion methods properly and compare levels of simulator sickness to a baseline of when the participant was feeling ordinary. Thus, we designed a virtual environment that met the following criteria:

(1) The environment must to induce a base level simulator sickness in participants.
(2) The forest environment should be realistic in its look and feel. Thus, the environment should be a high fidelity environment.

Figure 3: A bow in the Robin Hood Environment

(3) One should be able to navigate the virtual environment using all three walking methods, without having to adapt the environment to a given loocmotion method.

Using Unity3D, C#, Blender, SteamVR and the above criteria, we developed a forest environment in which a user would get a bow and arrow and were be required to traverse the environment and shoot randomly appearing targets. The reasons for said design and implementation details are discussed below.

3.1.1 Forest Environment. Using the Sensory Conflict Theory as a base for our design decisions, we implemented the virtual environment as a forest. A forest environment would encourage vection and therefore help produce a base level of simulator sickness. Furthermore, the vast amount of trees in the wide field of view would help induce a base level of simulator sickness as increased chances of vection will occur.

The ground of a forest tends to be uneven in real life and was thus designed to reflect as such in our environment. Therefore, when walking around the virtual environment a user will visually experience an up-and-down motion as one would walking in real life. This up-and-down motion is without the accompanying sensory experience they would get in real life. Therefore, there will be a mismatch between the visual and sensory systems and induce simulator sickness. This directly aligns with sensory conflict theory.

To increase the realism of the environment, realistic tree, bush and ground models and textures were used. Furthermore, forest sounds such as animal sounds and bird chirps, were played in the background to fully immerse a user in the game.

3.1.2 Archery. Archery has been chosen as the game mechanic as it requires traversal of the environment while simultaneously shooting targets. This combination requires the user to focus on using their controls to aim and shoot while having to traverse the environment at the same time. Combined with uneven ground, this will provide a mild level of simulator sickness as a baseline. The majority of simulator sickness is then thought to come from the locomotion methods themselves.

When a user enters the virtual environment they must pick up a bow and arrow using their dominant hand. When a user shoots an arrow, another arrow will appear in their hand immediately after the previous has been shot. To increase the realism of the game the bow and arrow has two very important design features. Both features are implemented for when a user draws their arrow. When a user draws their arrow before shooting, not only will they hear a sound of a bow string stretching but they will also feel a rumble on their remote thus making drawing of an arrow feel as if they were drawing and arrow in real life and therefore increasing the realism. A sound will also play when the arrow is shot and moving and once it hits it's intended target. Shooting the arrow required users to use the HTC Vive controllers.

3.1.3 Targets. In the Robin Hood Environment a user is required to find and shoot targets that were pre-placed in the trees. The targets would appear one at a time and then once shot the next target would appear. Thus, a user was directed to their next target by a floating arrow that pointed them in the correct direction. For a user to experience simulator sickness they are required to traverse the virtual environment. Thus, the design of the environment had to be such that it encouraged movement. Therefore, when a target was shot, before the next one appeared, a calculation was made to find all targets in the environment within a certain radius from the player. Thus, using a set radius, targets never appeared close to a user. Thus, the distance between a user and a target was too great and a user would be required to move for each target.

Figure 4: Pre-placed targets in the Robin Hood Environment

Figure 5: Floating arrow pointing towards the next target

Figure 6: Arm-swinging movement

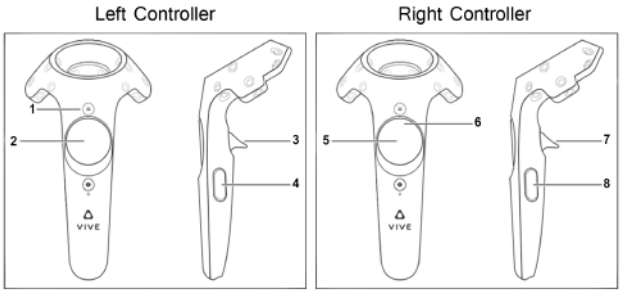

Figure 7: HTC Vive Pro controller

3.2 Arm Swinging

Arm swinging is a very commonly used locomotion method in VR. It has been used in games like VindictaVR, and many others. Thus, to compare our novel cradle method we decided to use the industry standard locomotion technique of arm-swinging such that we could compare our novel method to a commonly used locomotion technique.

The arm swinging technique is continuous motion-based locomotion method. To use the arm-swinging method in the Robin Hood Environment, a user is required to hold down both controller triggers (As shown by figure X, items 3 and 7). While a user holds down both triggers they must move their arms by swinging them back and forth, as if they are marching. The movement can be seen in the figure X below.

Essential to this locomotion method is controller tracking. Tracking the controllers is the cause for movement in the virtual environment. The difference in the controller position from one frame to the next determines the speed and therefore distance a user will move from frame-to-frame. This is done by storing data regarding the positions of the controllers from two frames, namely the previous frame and the current frame. To get the player to move using this data we were required to calculate the distance between the hands in the current frame and the hands in the previous frame. This was done as follows using Unity's own Vector3.Distance method which allows us to calculate the distance between to 3-dimensional vectors:

$$Ds = Vector3.Distance(T_{currentframe}, T_{previousframe}) \quad (1)$$

D_s is the distance between the left or right hands between two frames, s represents the left or right hand and T_n is the position of a controller at frame n. Following this we calculate the hand speed to be used later in the movement equation. What's important to note is that though the controller distances are being tracked within the virtual environment and not the real world, thus the actual distance moved will be larger as the player position moves in the virtual world as well. Thus the distance between controllers in the current frame and previous frame is larger than we expect. To offset this movement we must subtract the player position when calculating the S as follows:

$$S = (D_{left} - P_{player}) + (D_{right} - P_{player}) \quad (2)$$

where S is the hand speed and P_i is the position of the object i, in this case the player.

Finally, to transform the players position we simply set the player position to a $forwardComponent$ of a 3-dimensional vector multiplied by S, which is then scaled by a constant speed factor and multiplied by Unity's Time.deltaTime. The reason for multiplying by Time.DeltaTime is such that performance of the game, mainly frame rate, will not affect the movement based on game performance. The equation below describes the movement of the player.

$$P_{player} \mathrel{+}= Vector3.forwardComponent \times S \times speed \times Time.deltaTime \quad (3)$$

where in this case variable P_{player} denotes the position of the player, $Vector3.forwardComponent$ is a forward component of a 3-dimensional vector, S is the hand speed, $speed$ is a constant set by the developers and $Time.deltaTime$ is a Unity time value to maintain smooth movement at different frame rates, thus performance of the game is not important when developing the arm-swinging mechanics.

3.3 Teleportation

Teleportation is a non-continuous teleportation-based locomotion method. It requires a user to point their controller at the position in the virtual environment they wish to move to. In our case, the pointer was a yellow sphere to clearly differentiate itself from the environment. After clicking a designated button on the controller, in our case the trackpad, they will appear at the point at which they pointed without having experienced the full movement, hence non-continous. Using SteamVR, we implemented our own teleportation mechanic to handle a terrain such as the Robin Hood terrain as the standard SteamVR teleportation could not handle a hilly environment without modification. Though, building a new script posed many complications. Thus we will discuss the teleportation design implementation below.

3.3.1 Distance. The first issue that arose with teleportation is that one was able to teleport as far as they wished. Allowing this would've skewed experiment results as a user would be able to traverse the environment at speeds much greater than that of arm-swinging or cradle methods. Given this increase in traversal speed, one would be able to shoot targets at a faster rate and thus skew

performance results and provide us with false conclusions on the performance hypothesis as laid out earlier.

Thus, to ensure we would not attain skewed results a travel distance limit was implemented. When a user moved the pointer into a distance to far them the pointer would turn red, this providing feedback to the user and indicating that they could not teleport to the pointer position. Users were unable to teleport into trees, though this same method was not implemented for tree teleportation issue.

3.3.2 Teleportation Area. When teleporting, a user was able to not only teleport around the terrain but a user was also able to teleport up a tree. This would often happen by mistake and would cause many issues in the game and would also cause an extra sense of simulator sickness, and in some cases also made one fall off balance. Because of the Unity terrain settings, trees were indistinguishable from the ground itself, thus simple programming techniques would not suffice to fix this issue. Using Unity's in-built navigation mesh generator we were able to generate a path along the terrain that the user was able to navigate. Other points in the environment, like trees, were then disabled as walking paths. If one placed their pointer onto a tree and tried to teleport they would simply remain in place until until they placed their pointer onto valid location on the navigation mesh.

3.3.3 Teleportation Pointer. The last issue with the teleportation method is that the teleport pointer appeared right in front of your eyes when you drew the arrow. This is because the controller was pointing onto the bow and thus attaching the pointer to the bow collider. This required us to disable the renderer whenever one was drawing the arrow and once shot the renderer was enabled again the pointer would appear without having bothered the users view while shooting the arrow.

3.4 The Cradle

The cradle is a novel walking method designed specifically for this study. The cradle is quite simply a ring attached to the walls via tether cables and surrounds a user. When a user walks into to the ring they will move in the virtual environment in the direction that the ring is being pushed. This was done through the use of ring displacement and 3-dimensional vector sub-components. The ring thus provides haptic feedback as a user will feel the push back of the ring against their stomach. The reason we chose to design the cradle in such a way is because, as mentioned earlier, the use of a form of haptic feedback is theorized to reduce simulator sickness. The ring pushing back against one's body acts as haptic feedback and is therefore theorized to reduce the simulator sickness. This is backed up by sensory conflict theory in that a user will experience motion by their visual system, but because they will be moving their vestibular system will experience it too. More so, the force from the ring against their body will act as a form of haptic feedback. Thus, the novel cradle method is theorised to reduce simulator sickness in a user while locomoting in the Robin Hood Environment.

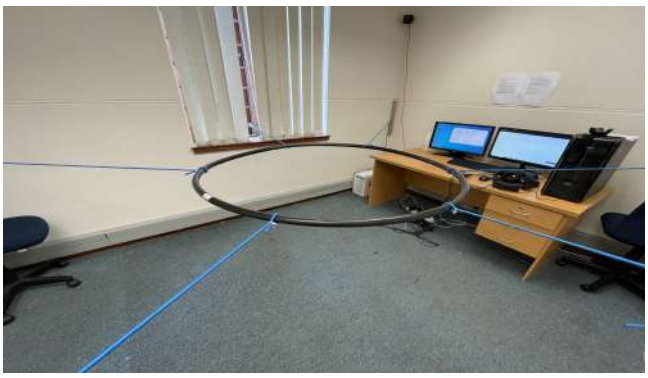

Figure 8: Novel cradle

4 EXPERIMENTAL DESIGN

To compare simulator sickness levels across the three walking methods, human trials were run. By inviting users to play the archery game by using all three walking methods individually we were able to discover the affect each method had on simulator sickness levels. We therefore conducted a Single-Factor Repeated Measures ANOVA.

4.1 Participants

Given that individual factors contribute greatly towards the degree of simulator sickness that one experiences defining a clear and precise participant inclusion criteria was very important. With research having shown that age contributes greatly toward the degree of simulator sickness one experiences we limited the age of the participants to 18 - 30 years old.

Moreover, for COVID reasons we limited the participants to be UCT students for the reason that for someone to get access to the UCT campus to come and take part in the trial they were required to fill in the UCT COVID health check questionnaire, which is only available to UCT students. They were also required to live in Cape Town as trials were to take place on the UCT campus. Also for COVID reasons students were required to not have any comorbidities. This is because those with comorbidities have a higher risk of hospitalization if they were to contract COVID. Thus, we can define the inclusion criteria as follows:

(1) $18 < \text{Age} < 30$
(2) Neurotypical
(3) No comorbidities
(4) University student
(5) Cape Town based

Pre trial screening was conducted using a Google Form and all data was double checked with the participant upon starting the trial. Screening included collecting other information such as: Experience using VR, how often they play PC/Console games, and if they own a VR headset themselves. This data was collected specifically as it could help make deductions in our analysis of the recorded data during the trial.

Finally, participants were required to sign an informed consent form stating that they were allowed to withdraw themselves from the trial at anytime and that they understood the risks of taking part in the trial, namely contracting COVID or experiencing simulator sickness.

4.2 Apparatus

To conduct the Single-Factor Repeated Measures ANOVA certain equipment and hardware were required and provided by the University of Cape Town.

4.2.1 Hardware. Most obviously, participants were required to use a VR headset to immerse themselves in the Robin Hood Environment. The headset provided was the HTC Vive Pro. The choice for this VR device was due to it's high resolution eye panels and ability to run high fidelity content seamlessly, given enough computer power. This is mainly due to it's tether which means it connects directly to the computer and therefore has a higher bandwidth to transfer data in comparison to a tetherless VR device. Thus, the HTC Vive Pro can run content at high resolutions with high quality textures. The HTC Vive Pro also uses mounted base stations for headmount display and controller tracking. These base stations provide more accurate tracking unlike the Oculus Quest 2 which uses cameras on the headset and can lose tracking accuracy when hands or controllers are placed behind the headset. Thus the more accurate tracking was important for the arm-swinging and cradle methods as the tracker or controller, especially in the cradle method, often appeared behind the participants headset.

The HTC Vive Pro requires a high powered GPU in order to run at appropriate frame rates. The minimum requirement to run a HTC Vive Pro according to HTC is a NVIDIA GeForce GTX 970, at least 4gb of ram, and an Intel Core i5-4590/AMD FX 8350 equivalent or better. Thus, we used a computer with a high powered NVIDIA GeForce GTX 1060 and 8gb of RAM. Using an Intel Core i5-**INSERT MODEL HERE** also provided us with plenty power to run our virtual environment smoothly. Moreover, another requirement from HTC is that the computer be booted into Windows, thus our computer booted into Windows 10. Thus, the computer specifications can be summarised as follows:

- CPU: Intel Core i5-**INSERT MODEL HERE**
- RAM: 8GB
- GPU: NVIDIA GeForce GTX 1060 **XGB**
- Operating System: Windows 10

4.2.2 Measurements. Participants were required to fill out 2 questionnaires after each treatment, namely the SSQ and GEQ. Keyboards are a particularly difficult hardware to clean properly, thus to prevent participants from sharing a keyboard and therefore lowering risk of contracting COVID, we printed out the SSQ and GEQ. Thus, a participant could fill in their questionnaire using a pen that had been fully sanitized, or their own pen

4.2.3 COVID PPE. To adhere to strict COVID guidelines and to lower the risk of contracting COVID the following PPE was needed to ensure proper hygiene during the human trials:

- Disposable face masks
- Sanitizing wipes
- Sanitizing spray
- Operating System: Windows 10

All PPE was supplied by the University of Cape Town

4.3 Measures

During each individual trial, in order to achieve our research aims we are required to track certain data.

4.3.1 Simulator Sickness. The first measurement we needed to track and calculate was a participants simulator sickness score. This was done by administering the SSQ to a participant . The questionnaire is subjective measure of a how a participant feels and requires a participants to rank 16 symptoms from none to severe. These symptoms are further categorized into the following categories: Nausea, Ocular and Disorientation . By filling out the Simulator Sickness Questionnaire after a participants uses each respective locomotion method we will be able to calculate a participants total simulator sickness score as well as a score for each sub-category that make up simulator sickness: Nausea, oculumotor and dizziness. We can compare these scores across the different treatments to make deductions about the effectiveness of the cradle method in reducing simulator sickness.

4.3.2 Game Experience. We will also be measuring a participants experience of the game using the Game Experience Questionnaire(GEQ). This questionnaire uses a likert-scale where a user rates their agreement with a statement from none to extremely. The GEQ, like the SSQ, provides us with sub-categories scores which we can compare against the different treatments. The sub-categories are as follows: Competence, Sensory and Imaginative Immersion, Flow, Tension/Annoyance, Challenge, Negative affect and Positive affect. The GEQ does not provide us with an overall game experience score.

4.3.3 Performance. In order to compare performance of a participant across the different treatment we also track the participants target hit ratio, their shot accuracy. This is simply the number of successful shots they took divided by the number shots they shoot. It can be defined as the equation below:

$$TargetHitRatio = \frac{\sum (SuccessfulTargetHit)}{\sum (ArrowShots)} \tag{4}$$

4.4 Procedure

To run our trials efficiently and without any human error we broke down each trial into four phases where each phase was clearly defined in an experiment protocol document. The four phases are laid out as follows:

(1) Phase 1: Hardware preparation
(2) Phase 2: Information, safety consent
(3) Phase 3: Task execution evaluation
(4) Phase 4: Feedback and remuneration

4.4.1 Phase 1: Hardware Preparation. In line with COVID protocols, all hardware and necessary objects were sanitized at the beginning of the day and between each participant. Items that were sanitized include: HTC Vive head mounted display, HTC Vive controllers, pens, and the cradle. Ensuring such hygiene standards was crucial in order to minimize the risk of a participant contracting COVID.

4.4.2 Phase 2: Information, safety consent. Before a participant was administered any treatment, the individual running the trial was required to double check a participants UCT Health Check App

status and confirm they were legitimately let onto the UCT campus. Following this all relevant details they entered in the Google Form were double checked as if these details weren't checked the data analysis could be skewed. If all participant details were correct the individual running the trial would then clearly outline the experiment procedure and the risks of partaking to the participant, ensuring that they were fully informed, before asking for their consent to continue with the experiment. If a participant chose to consent and accept the terms they were required to sign the informed consent form and move on to phase 3. If they declined to accept the terms laid out by the individual running the experiment they were then thanked for their time and the trial would end. The next participant would then be attended to.

4.4.3 Phase 3: Task Execution Evaluation. To begin the third phase of the trial, the individual conducting the trial would generate a random order in which to administered to treatments to a participant. Once an order had been generated a participant would begin by using the first locomotion method. A user was then entered in the virtual training environment where they could practice the archery mechanics of the Robin Hood Environment. A training environment was made for each respective locomotion method. After 4 minutes in the training environment they were transferred into the main game environment. In the main game environment a timer would begin immediately and they were to shoot as many targets as they could in the given 10 minute time frame. Once their time had run out the participants data was recorded and they were required to fill out the SSQ and GEQ. The participant was required to repeat the process for the other two treatments, filling out the SSQ and GEQ for both.

4.4.4 Phase 4: Feedback and Remuneration. After finishing all the experiment, the user was encouraged to provide feedback, either orally or written. Following this, a participant was required to sign a form confirming they had received their remuneration after which they were given R50 and thanked for there time. Finally, all equipment was put back onto their respective chargers such that the equipment is ready for the next trial/day.

5 RESULTS AND DISCUSSION

Once trials were completed all data was compiled and sorted such that the data would suit a statistical analysis software suite. This required digital data capture as both the SSQ and GEQ data were hard copys.

Using the sorted data we conducted statistical analyses at a significance level of $\alpha = 0.05$. Due to the small sample size it was crucial we first run the Shapiro-Wilk normality test, a test that informs one whether their data follows a normal distribution. This step is crucial as if one were to have collected normally distributed data they could get more accurate results by using more precise statistical tests. After having run the Shapiro-Wilk normality test on all data sets it was found that no data was normally distributed. Thus, the decision was made to use non-parametric data analysis tools. Following the normality tests the same procedure was undertaken for each data set as data sets have the same structure. Friedman tests were run to determine if there were statistically significant differences between treatments in all data sets. Using the resulting p-values from the Friedman tests, one could determine whether there were significant differences between treatments. This is done by either rejecting or not rejecting the null hypothesis. To gain further insight between different combinations of pairs of the treatments a Wilcoxon Signed Rank Sum was run on each pair. This test is specifically used on paired-non-parametric data and was thus useful given our data distribution and small sample size. Furthermore, box plots were generated for an easily understandable visual representation of the given data.

5.1 Simulator Sickness Measures

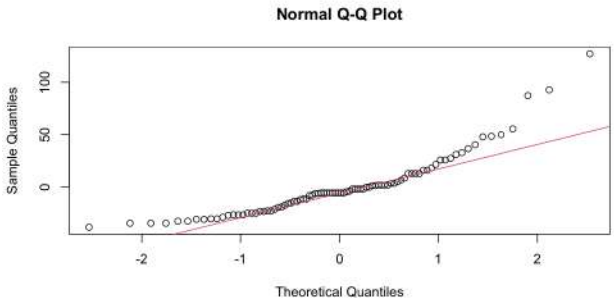

Figure 9: Example of Q-Q plot describing non-normal SSQ data

	χ^2	p-value
N	25.33	**0.0000138**
O	21.635	**0.00007769**
D	27.281	**0.000005141**
T	26.17	**0.000008787**

Table 1: Friedman test: Chi-Squared and p-value statistics for SSQ

Using a Friedman test it was shown that for the total SSQ score ($p = 8.787 \times 10^{-6}$, $\chi^2(2) = 26.17$) and for all simulator sickness sub-component scores, Nausea ($p = 1.38 \times 10^{-5}$, $\chi^2(2) = 25.33$), Oculomotor ($p = 7.769 \times 10^{-5}$, $\chi^2(2) = 21.635$), and Disorientation ($p = 5.141 \times 10^{-6}$, $\chi^2(2) = 27.281$), there was a statistically significant difference between all three locomotion methods.

It was observed that there was a statistical difference between the locomotion methods. Following this a statistical comparison was made between all possible locomotion method pair combinations. It provided us with clearer picture of whether the locomotion methods were truly statistically significantly different from one another, as the Friedman tests only tells us if there is a different between all three methods, not specific pairs. This was done by executing further Wilcoxon Signed Rank Sum tests between each respective locomotion method combination. After running this test on each respective combination of treatments a bonferroni adjustment was applied to the p-values. With all p-values being equal to 1 they suggest that no statistical difference was found in total SSQ scores between each respective combination of the treatments.

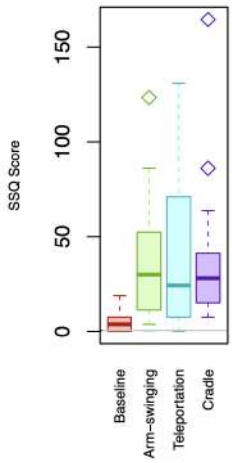

Figure 10: SSQ

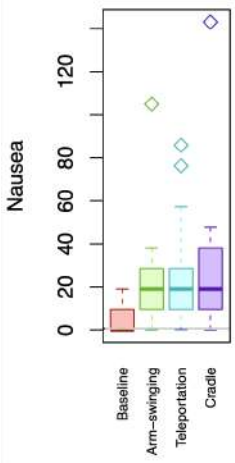

Figure 11: Nausea

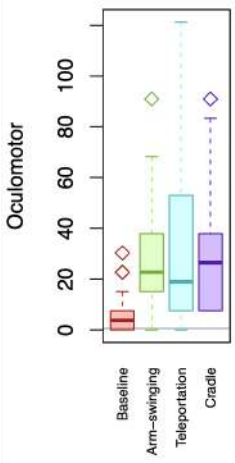

Figure 12: Oculomotor

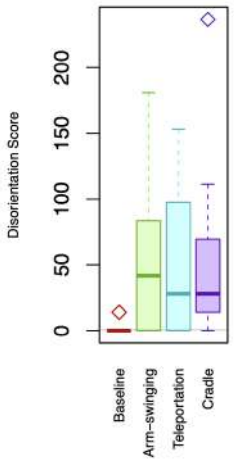

Figure 13: Disorientation

Figure 14: SSQ sub-component scores

Further insight was then gained into the sub-component scores of the SSQ for each locomotion method order to give us a more refined picture. Having used the results from the Wilcoxon Signed Rank Sum tests run for each sub-component of the SSQ it was shown that the sub-component scores of the SSQ were not statistically different amongst the different treatments. Therefore according to the analysis the cradle did not reduce simulator sickness as this research had initially hypothesized by Hypothesis 1.1 and Hypothesis 1.2. This showed that the cradle did not help reduce SSQ scores in comparison to the arm-swinging or teleportation methods. A reason for this might be due to a very small sample size which would reduce the accuracy of the results. Or simply, it could be indication that the cradle method is simply not effective in reducing simulator sickness. It is important not to conclude that haptic feedback is not useful. The cradle may just not provide enough or good enough haptic feedback to reduce SSQ scores. Though not statistically significant, when looking at boxplot **??** it seems that the SSQ do cluster at a lower point that the arm-swinging and teleportation method. This may indicate that with a larger sample size we would see a reduction in SSQ scores when using the cradle. Further research would need to be done to explore this.

5.2 Game Experience Measures

Though game experience was not initially set out to be measured, we believed it may contribute toward the overall analysis of the data and so participants were encouraged to fill out the GEQ. But to no avail, it yielded no significant results with Friedman tests across all sub-categories reporting no significant difference between the treatments. This is shown by table 2 where all p-values were greater than the significance level and thus no null hypotheses were rejected. We therefore saw that game experience remained constant across the different treatments.

	χ^2	p-value
Competence	0.39506	0.8208
Sensory	2.1224	0.346
Flow	0.19718	0.9061
Negative Affect	0.88571	0.6422
Positive Affect	2.6076	0.2715
Tension	1.8462	0.3973
Challenge	0.33333	0.8465

Table 2: Friedman test: Chi-Squared and p-value statistics for GEQ

5.3 Performance Measures

To measure the performance of a user we decided to track their shot accuracy, which we called a users target hit ratio. There target hit ratio has been defined in equation 4. This was simply a measure of the accuracy of shooting targets. We ran the same set of tests on the performance data as was done on the SSQ and GEQ data. The data was not shown to be normal ($p = 3.056 \times 10^{-2}$). Since the hit ratio data was shown not to be normal a Friedman test was applied and it showed that there was a statistically significant difference between the hit ratio ($p = 1.576 \times 10^{-3}$, $\chi^2(2) = 12.905$) between the treatments. After running a series of Wilcoxon Signed Rank Sum test we were able to see that there were statistical difference between the arm-swinging and cradle methods as well as the teleportation and cradle methods as shown by $p = 4.543\,134 \times 10^{-3}$ and $p = 1.552\,067 \times 10^{-3}$ for each pair respectively. An issue arises is that the test doesn't tell us which locomotion method had a higher or lower average target hit ratio. It was then shown in figure 15 that the cradle produced lower target hit ratios in comparison to the arm-swinging and teleportation methods. Thus we have used the box plot to further backup our statistical tests. These results

contradicted Hypothesis 2.1 and Hypothesis 2.2 in that the cradle method was shown to have reduced a users target hit ratio. Therefore, based of this research, one should not use the cradle method a performance based game or task.

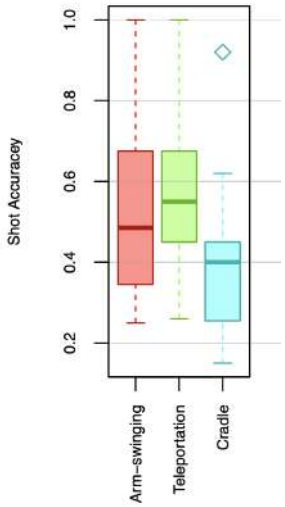

Figure 15: Hit ratio box plot

6 CONCLUSIONS

In this research, we sought to reduce simulator sickness using a novel locomotion method, the cradle method. We designed and conducted a Single-Factor Repeated Measures ANOVA to show whether the cradle would reduce simulator sickness and increase performance of a user. The experiment was run with 22 participants having took part in the trials with their SSQ, GEQ and target hit ratio data having been recorded. Haven taken the small sample size into account, the results showed that there were no statistically significant results with regard to simulator sickness differences across the three treatments. We can conclude that the cradle was ineffective in reducing simulator sickness, though this does not say anything about haptic feedbacks effectiveness in general.

A statistically significant difference was found in the target hit ratios between the arm-swinging and cradle methods as well as the teleportation and cradle methods. We therefore conclude that the cradle method had the opposite effect that was expected in that it reduced the target hit ratio of users.

Potential future work ideas include adjusting the size of the cradle, using a greater sample size for increased accuracy, including another locomotion method for greater comparison and results.

7 LIMITATIONS AND FUTURE WORK

Simulator sickness research is crucial to the wider adaption of VR. Though, simulator sickness research also requires human trials if one wants to make accurate measurements. In this research time the time frame given for human trials was a week. Given the time frame combined with the time it takes to run a single human trial this research was limited to a small sample size. To gain more accurate results and further insight it is suggested this research be run over a longer period of time with a larger sample size.

Moreover, with the sample size affecting the accuracy of the data so to do our participants. With a potential larger sample size in the future it is suggested to use a balanced amount of males and females as in this research only 40% of participants were female. This is because according to previous research females experience much greater degrees of simulator sickness, and this could affect our results.

To get a greater overall picture of what a user was experienced it was important to measure their simulator sickness using an SSQ. This is a subjective measure and can therefore be inaccurate. Thus, future work could include using ECG equipment which would provide objective data and might provide greater insight to the degree to which the user experienced SSQ.

We were limited to the size of the experiment room used for the trials. A greater sized room would allow one to use the cradle method with a different sized diameter. Thus, a future work could include using a cradle which can have it's diameter adjusted and thus one could test the cradle with varying diameters.

Finally, for greater accuracy and comparison it would be useful to compare to against another walking method, one that comes from a different section of Boletsis' and Cedergen's Locomotion Technique Typology[3].

ACKNOWLEDGMENTS

I would like to thank Ryan Acton without whom this research would not have been made possible. We would like to thank Sam Chetty for constantly helping us organize the necessary virtual reality equipment and the materials needed to construct the cradle, even if he had to delivery to our door when we had COVID. I would like to thank my family for their constant support with a special mention to my twin, Grant Norrie, for all his help and ideas he thought of to help this paper be the best it could be. Most importantly, thank you to Batsheva Channa Brand. Without her constant support and constant care, I don't know if I would have finished this paper. With all my heart, **thank you**. Finally, we would like to thank Prof. James Gain for supervising our researching and guiding us throughout this project.

A SSQ EXTRA DATA

Dataset	N		O		D		T	
	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev
Arm-swinging	22.12	21.71	28.94	22.02	48.72	48.55	36.04	29.64
Teleportation	23.42	24.41	30.66	30.36	51.25	53.03	38.08	36.56
Cradle	26.45	29.87	29.29	23.89	47.45	54.11	37.57	35.21

Table 3: SSQ Data

B GEQ EXTRA DATA

Dataset	Competence		Sensory		Flow		Negative Affect		Positive Affect		Tension		Challenge	
	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev
Arm-swinging	2.68	0.76	1.82	0.50	2.61	1.05	0.49	0.35	2.75	0.86	0.41	0.91	0.86	0.58
Teleportation	2.65	0.83	1.84	0.59	2.50	0.93	0.43	0.35	2.72	0.72	0.59	0.80	0.80	0.58
Cradle	2.66	0.75	1.75	0.55	2.48	0.87	0.60	0.54	3.02	0.64	0.36	0.49	0.81	0.58

Table 4: GEQ Data

C DIVISION OF LABOUR

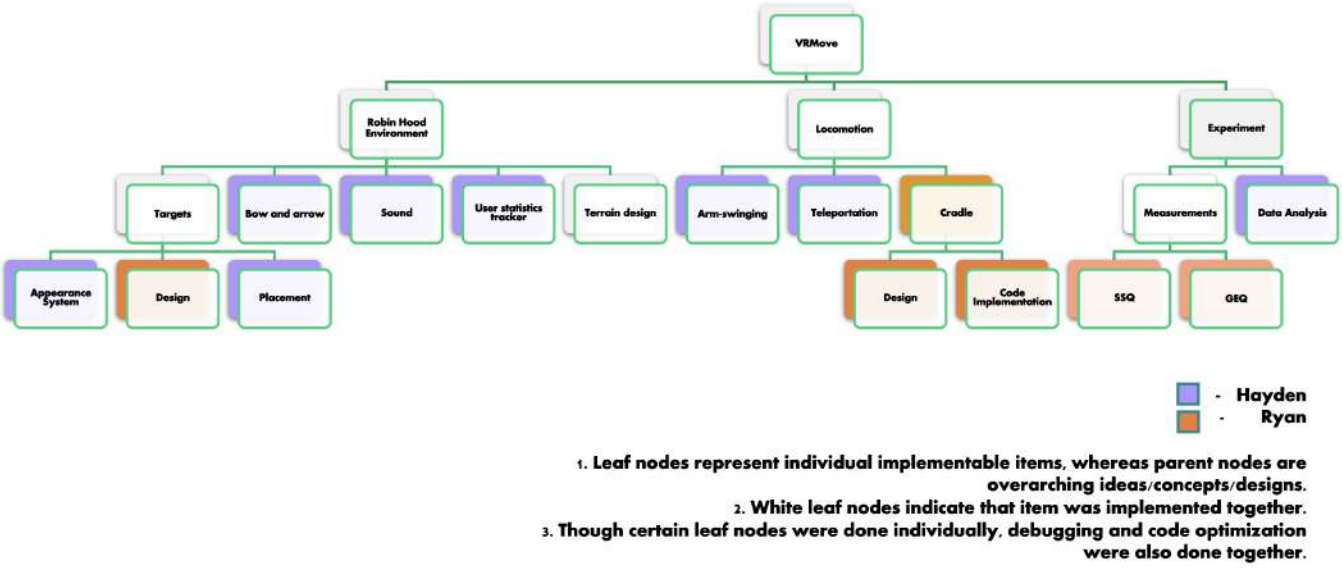

Figure 16: Division of labour diagram

REFERENCES

[1] Mustafa Almallah, Qinaat Hussain, Nora Reinolsmann, and Wael Alhajyaseen. 2021. Driving simulation sickness and the sense of presence: Correlation and contributing factors. *Transportation Research Part F Traffic Psychology and Behaviour* 78 (03 2021), 180–193. DOI: http://dx.doi.org/10.1016/j.trf.2021.02.005

[2] F. Biocca. 1992. Will Simulation Sickness Slow Down the Diffusion of Virtual Environment Technology? *Presence: Teleoperators Virtual Environments* 1 (1992), 334–343.

[3] Costas Boletsis. 2017. The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology. *Multimodal Technologies and Interaction* 1 (Sept. 2017), 24. DOI: http://dx.doi.org/10.3390/mti1040024

[4] Costas Boletsis and Jarl Cedergren. 2019. VR Locomotion in the New Era of Virtual Reality: An Empirical Comparison of Prevalent Techniques. *Advances in Human-Computer Interaction* 2019 (apr 2019), 1–15. DOI: http://dx.doi.org/10.1155/2019/7420781

[5] Frederick Bonato, Andrea Bubka, Stephen Palmisano, Danielle Phillip, and Giselle Moreno. 2008. Vection Change Exacerbates Simulator Sickness in Virtual Environments. *Presence: Teleoperators and Virtual Environments* 17, 3 (06 2008), 283–292. DOI: http://dx.doi.org/10.1162/pres.17.3.283

[6] Patricia S Cowings, Steve Suter, William B Toscano, Joe Kamiya, and Karen Naifeh. 1986. General autonomic components of motion sickness. *Psychophysiology* 23, 5 (1986), 542–551.

[7] Paul Dizio and James Lackner. 2000. Motion Sickness Side Effects and Aftereffects of Immersive Virtual Environments Created with Helmet-Mounted Visual Displays. (11 2000), 4.

[8] Andre Garcia, Carryl Baldwin, and Matt Dworsky. 2010. Gender Differences in Simulator Sickness in Fixed- versus Rotating-Base Driving Simulator. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 54 (09 2010), 1551–1555. DOI: http://dx.doi.org/10.1177/154193121005401941

[9] Lawrence J. Hettinger, Kevin S. Berbaum, Robert S. Kennedy, William P. Dunlap, and Margaret D. Nolan. 1990. Vection and Simulator Sickness. *Military Psychology* 2, 3 (1990), 171–181. DOI: http://dx.doi.org/10.1207/s15327876mp0203_4 PMID: 11537522.

[10] Teresa Hirzle, Maurice Cordts, Enrico Rukzio, Jan Gugenheimer, and Andreas Bulling. 2021. A Critical Assessment of the Use of SSQ as a Measure of General Discomfort in VR Head-Mounted Displays. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 530, 14 pages. DOI: http://dx.doi.org/10.1145/3411764.3445361

[11] Jukka Häkkinen, T. Vuori, and M. Paakka. 2002. Postural stability and sickness symptoms after HMD use. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics* 1 (11 2002), 147– 152. DOI: http://dx.doi.org/10.1109/ICSMC.2002.1167964

[12] H. Kim, J. Park, Yeongcheol Choi, and Mungyeong Choe. 2018. Virtual reality sickness questionnaire (VRSQ): Motion sickness measurement index in a virtual reality environment. *Applied ergonomics* 69 (2018), 66–73.

[13] Young Kim, Hyun Kim, Eun Kim, Heedong Ko, and Hyun-Taek Kim. 2005. Characteristic changes in the physiological components of cybersickness. *Psychophysiology* 42 (10 2005), 616–25. DOI: http://dx.doi.org/10.1111/j.1469-8986.2005.00349.x

[14] Charles M. Oman. 1990. Motion sickness: a synthesis and evaluation of the sensory conflict theory. *Canadian Journal of Physiology and Pharmacology* 68, 2 (1990), 294–303. DOI: http://dx.doi.org/10.1139/y90-044 PMID: 2178753.

[15] George Park, Richard Allen, Dary Fiorentino, Theodore Rosenthal, and Marcia Cook. 2006. Simulator Sickness Scores According to Symptom Susceptibility, Age, and Gender for an Older Driver Assessment Study. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 50 (10 2006), 2702–2706. DOI: http://dx.doi.org/10.1177/154193120605002607

[16] Matti Pouke, Arttu Tiiro, Steven M. LaValle, and Timo Ojala. 2018. Effects of Visual Realism and Moving Detail on Cybersickness. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 665–666. DOI: http://dx.doi.org/10.1109/VR.2018.8446078

[17] Kevin S. Berbaum Michael G. Lilienthal Robert S. Kennedy, Norman E. Lane. 1993. Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness. *The International Journal of Aviation Psychology* (3 1993), 203–220. DOI: http://dx.doi.org/10.1207/s15327108ijap0303_3

[18] Dimitrios Saredakis, Ancret Szpak, Brandon Birckhead, Hannah Keage, Albert Rizzo, and Tobias Loetscher. 2020. Factors Associated With Virtual Reality Sickness in Head-Mounted Displays: A Systematic Review and Meta-Analysis. *Frontiers in Human Neuroscience* 14 (03 2020). DOI: http://dx.doi.org/10.3389/fnhum.2020.00096

[19] Hiroaki Shigemasu, Toshiya Morita, Naoyuki Matsuzaki, Takao Sato, Masamitsu Harasawa, and Kiyoharu Aizawa. 2006. Effects of physical display size and amplitude of oscillation on visually induced motion sickness. 372–375. DOI: http://dx.doi.org/10.1145/1180495.1180571