Aalborg University Copenhagen

Semester:
Medialogy 10th Semester

Title:
Substitutional Reality and its effect on Game Experience in a Virtual Reality Shooter Game

Aalborg University Copenhagen

Frederikskaj 12,

DK-2450 Copenhagen SV

Semester Coordinator: Stefania Serafin

Secretary: Lisbeth Nykjær

Project Period:
1. February - 28. May

Semester Theme:
Master Thesis

Supervisor(s):
Jon Ram Bruun-Pedersen

Members:
Emil Jonathan Wittendorff
Mads Krøll Didriksen
Mathias Månsen Clemmensen

Abstract:

In recent years, virtual reality has become a consumer product and because virtual reality in a domestic setting often has to share space with furniture, the real world can obstruct the virtual experience. Substitutional reality tries to solve this problem by giving the home consumer the possibility to incorporate the intruding real-world objects into the virtual world. In this thesis, we explored the effects of combining a virtual reality action shooter game with substitutional reality in order to investigate whether or not substitutional reality has an effect on the game experience. A simple virtual reality shooter game was developed, to determine if the use of substitutional reality has an effect on the game experience. An evaluation was performed, which showed that incorporating substitutional reality into an action shooter game created for virtual reality, did not show any significant effect on the game experience of the user. This thesis serves as early research on the potential to include substitutional reality in virtual reality video games without disturbing the game experience and, in theory, confirms that it is possible to incorporate substitutional reality into any simple virtual reality action shooter game without disturbing the game experience.

Copies: 0
Pages: 60
Finished: 28th of May - 2019

Substitutional Reality and its effect on Game Experience in a Virtual Reality Shooter Game

Mads K. Didriksen, Emil J. Wittendorff and
Mathias M. Clemmensen

Medialogy 10th Semester, Aalborg University Copenhagen
Email: {mdidri14, ewitte14, mclemm14}@student.aau.dk

May 28th, 2019

AALBORG UNIVERSITY COPENHAGEN

Abstract

In recent years, virtual reality has become a consumer product and because virtual reality in a domestic setting often has to share space with furniture, the real world can obstruct the virtual experience. Substitutional reality tries to solve this problem by giving the home consumer the possibility to incorporate the intruding real-world objects into the virtual world. In this thesis, we explored the effects of combining a virtual reality action shooter game with substitutional reality in order to investigate whether or not substitutional reality has an effect on the game experience. A simple virtual reality shooter game was developed, to determine if the use of substitutional reality has an effect on the game experience. An evaluation was performed, which showed that incorporating substitutional reality into an action shooter game created for virtual reality, did not show any significant effect on the game experience of the user. This thesis serves as early research on the potential to include substitutional reality in virtual reality video games without disturbing the game experience and, in theory, confirms that it is possible to incorporate substitutional reality into any simple virtual reality action shooter game without disturbing the game experience.

Table of Content

1. Introduction

Virtual reality (VR) is not a new concept and has roots far back in history, with concepts of head-mounted displays (HMD), head and hand tracking going back to the late 1980s (Slater, 2018). Over this time span, many different terms used to describe VR, and its qualities, have emerged. The terms used in this paper are the ones proposed by Slater (2009), *Presence*, also known as *Place Illusion*, and *Plausibility Illusion*. Presence refers to what is perceived or the "feeling of being there", which is connected to the number of possible actions you can make in the VR system, that are equivalent to those you can make in real-life, e.g. rotating ones head to see what is around or bending down and looking under things. The level of presence is a function of these possible actions, where the more possible actions you can take leads to a higher level of presence (Slater, 2009). The plausibility illusion refers to what is perceived while immersed in VR and is strongly tied to events in VR that refer to you, but over which you have no control. This could be in the form of virtual characters that react towards you, based on your proximity or behavior (Slater, 2009).

Since the 80s, VR equipment has evolved quite substantially, both in terms of better graphics and processing power but also tracking fidelity. Modern VR equipment has six degrees of freedom for both head, hand and in some cases foot tracking as well (Slater, 2018). The latest generation of VR headsets has made VR a consumer product, taking the VR experience out of laboratories and bringing it into the user's own home. Unfortunately, not all users will have space enough to create dedicated VR play areas. This forces the user to either clear a sufficiently large area of furniture, both Oculus and HTC recommends a play area of 3.5m x 3.5m for room scale tracking (Oculus, 2018; Borrego et al., 2018), or play amongst their furniture. The latter is not recommended as the risk of colliding with objects and hurting oneself is increased, as the furniture is not visible in VR. To lower the chance of colliding with objects within the play area, while being immersed in VR, the concept of Substitutional Reality (SR) has been developed. The concept is that users should be allowed to mark up the physical objects in their play area and by doing so a virtual counterpart is created in the virtual environment (VE) (Simeone et al., 2015; Garcia et al., 2018). Because the field of SR is rather new it is yet to be tested in a game setting and the concern is that, if users are allowed to influence the content of the virtual world, the experience might be negatively impacted. Garcia et al. (2018) suggest

that future research should investigate which circumstances will have an effect on the enjoyment of a VR game when users are allowed to modify the VE using SR.
Inspired by this, the following initial problem statement is formulated.

> ***Initial Problem Statement:***
> *How is the game experience affected if users are allowed to modify the virtual environment of a virtual reality game?*

2. Analysis

In this chapter, we will be exploring several fields related to VR. Where we will mainly focus on the field of SR, as this field is the main inspiration of this paper and will be an important part of the product and experiment creation process.

Besides SR, fields such as passive haptics for VR, VR locomotion methods, current popular VR games, game experience, and current consumer VR equipment will be explored, as they are important for the creation of the project.

2.1. Substitutional Reality

VR is moving out of the laboratory and into the common household, as the equipment is becoming more affordable and because newer headsets are becoming more efficient and require less processing power (Garcia & Simeone, 2018). The challenge of using VR in a home environment is that it often has to share space with normal household furniture, which could increase the chance of damaging objects or oneself. To accommodate these physical objects a VE would normally have to be tailored to a specific room layout, making it less useful for home use (Simeone et al., 2015). SR uses the concept of passive haptics to incorporate objects from the real world into the virtual one in a more flexible manner, where a certain mismatch is allowed between the physical object and its virtual counterpart. Having a visual representation of physical objects in VR will not only help navigate the tracked space but being able to reach out and physically touch an object in virtual reality also increases the feeling of presence (Simeone et al., 2015).

2.1.1. Allowed Mismatch

The study by Simeone et al. (2015) investigated how much mismatch is allowed in a VR experience before the illusion of VR is broken. They created two experiments, where they first manipulated the features of a virtual object while keeping the physical counterpart constant. The object was a mug which was manipulated in different ways in the experiment, see figure 1.

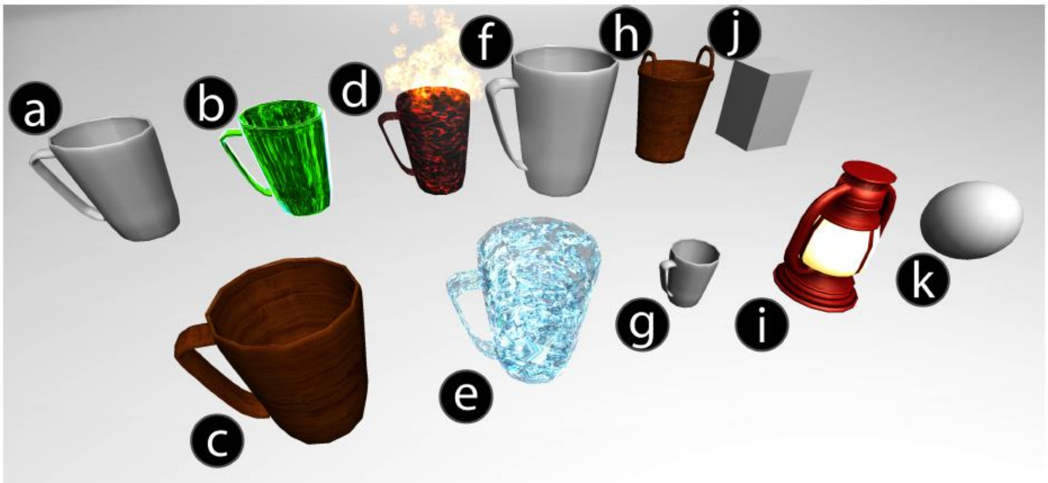

Figure 1:
Different versions of the virtual mug, where version (a) was a replica of the physical mug.

They extracted different features from the virtual substitute that would be manipulated and these were:

- **Aesthetics**: Different materials were used for the mug replica, to alter the appearance and perceived temperature.
- **Size (subtraction/addition)**: The size of the virtual mug replica.
- **Function**: The size and shape were kept similar to the virtual mug, but the placement of interaction points (handles) and virtual model was altered.
- **Category**: The virtual mug replica was exchanged for a simple sphere and box to test the importance of shape.

The results showed that shape and perceived temperature (material), were some of the most important factors, as they caused the largest reduction in the users' suspension of disbelief (Simeone et al., 2015).

In the second experiment, the VE represented the bridge of a spaceship and participants were instructed to hit a flying object with a lightsaber. Different physical objects were used to simulate the lightsaber, these can be seen in figure 2. The goal was to determine if different

Figure 2:
The mug and wine bottle were used in the first experiment and the torch, umbrella and lightsaber were used for the second experiment.

household objects could serve as believable counterparts to the virtual lightsaber if no accurate replica was available. The experiment showed that participants preferred the torch over the lightsaber replica and the umbrella, as the torch had the smallest weight and was easier to handle (Simeone, et al, 2015).

2.1.2. Existing SR systems

Making a VE that has accurate representations of physical objects normally means that the VE has to be tailored to the exact layout and objects located in the real-life environment. This makes it difficult to use such VEs in a home environment for the end user, as their home environment will be different from the tailored VE. Several studies have created tools for overcoming this problem and potentially allow home users to customize the VE based on their home environment (Garcia et al., 2018; Garcia & Simeone, 2017; Sra et al., 2016). The solution that fits best with this project is the one proposed by Garcia et al., (2018), where they created a tool for marking edges of physical objects. The marked objects are first represented in the VE as red boxes and these boxes are later substituted for virtual objects that fit within the measured bounding boxes. The goal of the experiment was to determine the best method for marking the physical objects in the tracked space. One condition, called inside-looking-out, had users immersed in VR, where a front-facing camera streamed footage of the real world into the VE, while the second condition, called outside-looking-in, gave users a tablet that showed the VE while the user marked physical objects, without being immersed in the VE (Garcia et al., 2018).

Garcia et al. (2018) found that users who were immersed in VR while marking the physical objects had an average accuracy of 99.47%, compared to the reference size of the physical objects. The tablet had an average accuracy of 115.65%, meaning that the drawn objects were 15.65% larger than the physical objects. This means that it is possible to make very accurate measurements of the bounding boxes of physical objects, but the accuracy will be highest if the user is immersed in VR. Looking at completion times, for marking the physical objects, the tablet was the fastest, with an average completion time of 160.73 seconds, compared to an average completion time of 276.73 seconds for users immersed in VR.

As our project also uses SR it would be logical to create a similar tool as Garcia et al., (2018), to mark physical objects. Because the marking up phase is a prerequisite for the actual experiment, in our project, it would be more beneficial to use the outside-looking-in method with the faster completion time, even if the measurements are not nearly as accurate. This would give participants more time in the experiment phase, potentially reducing the risk of fatigue and cybersickness, as they overall spend less time immersed in VR. Furthermore, our experiment will be more focused on how substituted objects will affect the enjoyment of a VR game, making the accuracy and mismatch of the generated virtual objects less important.

2.2. Passive Haptic Interfaces

Haptic interfaces are important to SR as these are used in conjunction, more specifically the passive haptic feedback mentioned by Simeone et al. (2015). Passive haptics is an approach to incorporate haptics into a virtual environment, where the focus is to utilize real physical objects to give users haptic feedback (Lindeman et al., 1999; Azmandian et al., 2016; Garcia & Simeone, 2017). In Lindeman et al. (1999) there is a good explanation of what passive haptics entail:

> "*Passive-haptic 'devices' are physical objects which provide feedback to the user simply by their shape, texture, or other inherent properties. In contrast to active-haptic feedback systems, the feedback provided by passive-haptic feedback devices is not controlled by a computer.*"
> (Lindeman et al., 1999)

As both mentioned above and can be seen from the quote, passive haptics is about making use of real physical objects in order to provide users with a form of haptic feedback through the sense of touch (Lindeman et al., 1999; Azmandian et al., 2016; Garcia & Simeone, 2017).
This makes it possible to use simple objects in the nearby environment to give users haptic input instead of creating a large and elaborate setup in order to provide haptic feedback. A user would be able to include the physical space in the virtual space, such as replacing furniture with virtual objects (Azmandian et al., 2016; Garcia & Simeone, 2017).
Passive haptics can be used to perform a direct mapping between real and virtual objects in order to provide an immersive experience that can satisfy the haptic feedback of the users because of the actual physical objects in the same space as the virtual object (Azmandian et al., 2016). There is, however, a downside to passive haptic which is related to the mapping. If the locations of objects in the physical space are moved or altered, then all of the physical objects' locations would have to be recalibrated to ensure that the VE has consistent one-to-one mapping (Azmandian et al., 2016).
Research has been done into this misalignment between physical and virtual object position, and what effect it could have on immersion (Gall & Latoschik, 2018). Gall & Latoschik (2018) created a test with one condition where the physical and virtual objects were aligned and one where they were not aligned (20 cm difference). They found that users had a higher sense of presence in the condition with proper alignment as opposed to non-aligned. It is possible that a smaller mismatch is acceptable but Gall & Latoschik (2018) show that a misalignment affects how users perceive the VE.
Still the interesting aspect of using passive haptics is that home use of VR can be made easier as users would not necessarily need a specific room with no furniture to set up the VR equipment, but they could incorporate any physical object they desire and overlay the position with a corresponding virtual object (Garcia & Simeone, 2017). This very closely resembles the attributes of SR, where the user is given the possibility to perform the action of incorporating real objects into the virtual environment through a tool.

2.3. Locomotion in Virtual Reality

Locomotion is an important component of VR, and there are a lot of different ways of implementing it, each with their own advantages and disadvantages (Boletsis, 2017). The goal of this chapter is to create an analysis of the different locomotion methods, which can be used when deciding on the locomotion method to be implemented in this project. Although locomotion is not the main research topic, of this project, it is still relevant to choose a locomotion method that makes sense for the testing environment.

A literature review was made by Boletsis (2017), which contains a compendium of all the different locomotion methods used in studies and research topics. 36 articles passed the screening process and were used in the review, from which 11 different locomotion methods were found. Boletsis categorized these 11 different locomotion methods into four distinct locomotion types: motion-based, room scale-based, controller-based, and teleportation-based.

2.3.1. Motion-Based Locomotion

This locomotion type is centered around using physical activity to interact with the virtual space. This locomotion type includes methods like walking-in-place and arm swinging, which requires the user to perform a continuous physical movement in order to interact in VR (Boletsis, 2017). Walking-in-place is a common method which either translates head oscillation data or feet tracking into movement in the virtual space. The advantage of this method is that it is similar to real-walking and solves the problem of limited physical space (Feasel et al., 2008). According to the widely accepted theory of sensory conflict, the disadvantage of this type of locomotion is that it could cause cybersickness, as there is a mismatch between the vestibular and visual systems (LaViola, 2000).

2.3.2. Room Scale-Based Locomotion

Like motion-based locomotion, this type of locomotion utilizes continuous physical movement to interact in the virtual environment. However, as the name suggests, the difference is that the virtual space is restricted by the physical space. A typical method that falls under this locomotion type is the real-walking method (Boletsis, 2017). The advantage of real-walking is that it is the most natural of the locomotion methods, as it translates 1:1. The disadvantage is that the traversable area of the virtual environment is limited by the size of the physical space the user is in.

2.3.3. Controller-Based Locomotion

In this locomotion type, the controller is used to move around in the virtual environment. This is also a form of continuous movement, and examples of the typical locomotion methods are joystick-based and head-directed movement (Boletsis, 2017). The benefit of using this locomotion type is that it is easy and quick to implement, but the drawback is similar to the Motion-Based locomotion, as it can cause cybersickness due to the sensory conflict theory (LaViola, 2000).

2.3.4. Teleportation-Based Locomotion

This type of locomotion is one of the most utilized and dominant within VR applications. Unlike the other locomotion types, this is a non-continuous movement, as the user moves from point a to b in one action (Boletsis, 2017). An example of this type of locomotion is the point and teleport method. The advantage of this locomotion type is that users who have tried VR before are most likely familiar with this method as it is widely used.

2.3.5. Locomotion for Substitutional Reality

When pairing physical objects with virtual ones, using SR, the virtual objects created should match the VE to minimize the mismatch and would be connected to the tracking area. This also means that when the virtual objects are created they belong to one specific point in the VE, e.g. substituting a physical dresser with a virtual pile of wood. Since Motion-, Controller- and Teleportation-Based locomotion works by moving the tracking area around in the VE, the substituted virtual objects would also move around to match the new position of the tracking area, e.g. the virtual pile of wood would follow you around when you moved. This would likely feel very unnatural in most cases and affect the suspension of disbelief in a negative way.

While it would be interesting to explore how these locomotion methods can be used with SR, it is not the focus of this project. Instead, our implementation will utilize the Room Scale-Based locomotion method as the tracking area is static in the VE and it gives users the most natural way of navigating inside the VE.

2.4. Selecting a Game Genre

There are a lot of different popular genres to select from in VR, but the most popular one, based on Steam's top-selling titles of 2018, is of the shooter genre (Steam, n.d.). Every shooter game genre employs a simple main game mechanic in VR, which is essentially point and shoot. This mechanic is very simple to implement and is also very intuitive for the user, which makes the learning curve smaller in terms of learning the system. This makes it an optimal candidate for choosing which genre to create a game around.

2.4.1. Popular Virtual Reality Shooter Games

The Steam[1] platform sells many VR games and at the end of 2018, Steam released information on the top-selling VR games of 2018 (Lang, 2018) which shows some of the most popular VR games currently available. In the interest of this paper, games that were of an action shooter genre were selected. In order to consult with the list, a few other sources were drawn in to select a few popular VR action shooter games ("10 Best Virtual Reality Shooter Games", n.d.; Carbotte, 2018).

From Steam's list of most popular VR games, only shooters (Lang, 2018):

- [**Game Name**] **-** [**Positive User Reviews**]
- Space Pirate Trainer - 95%
- Superhot VR - 89%
- Arizona Sunshine - 86%
- Robo Recall - Not from Steam list

Space Pirate Trainer is a VR shooter game set in a futuristic setting, where the user has to fight wave after wave of flying enemy robots, with various types of weapons and equipment at the user's disposal (I-Illusions, 2016). The score, on the list above, is a testament to the games engaging combat and immersive system.

Locomotion: Room Scale-Based Locomotion

Mechanics: Two pieces of equipment, a pistol and a shield can be equipped for each hand by moving the hands over the shoulder, both include a variety of modes of usage, e.g. the gun can

[1] https://store.steampowered.com/

change from single-shot mode to shotgun mode. Shoot the enemies to continue to the next wave, several different types of enemies. Continue for as long as possible.

Superhot VR was initially released as a regular PC game, but because of its mechanics, it translated well into VR (Carbotte, 2018). The game is based on movement when the user moves then the world moves if the user stands still then the world stands still. The objective of the game is simple, kill the enemies with whatever is at your disposal to continue (SUPERHOT Team, 2016).

Locomotion: Room Scale-Based Locomotion

Mechanics: Only when the user moves, do the world move if the player stands still then the world stands still. Several weapons and items that can be shot or thrown. One hit kills, both the player and enemies. Several small events within one level.

Arizona Sunshine is a zombie apocalypse shooter game, which focuses on survival while having minimum resources to defend yourself ("10 Best Virtual Reality Shooter Games", n.d.). The controls in the game are realistic, any weapons used have to be reloaded manually and the ammunition for the weapons have to be found around the map (Vertigo Games & Jaywalkers Interactive, 2016). It is one of the best selling VR game titles that has remained consistent since its release (Carbotte, 2018).

Locomotion: Controller-Based Locomotion, Teleportation-Based Locomotion.

Mechanics: The locomotion method can be selected in terms of preference. The game has a story to follow. Zombies and enemies with a few different types. Weapons are placed at the hip, maximum two at a time, ammunition is placed in between the weapons, to reload move the weapon to this area, manually remove the clip from weapon before reload.

Robo Recall is the only game mentioned here which is not a part of the Steam list, as it is not sold on the Steam platform but on the Oculus instead. The game is an intense fast-paced shooter game which has you destroying robots with a fair amount of weaponry (Epic Games, 2017).

Locomotion: Teleportation-Based Locomotion

Mechanics: Large area to move around in. Wide variety of weaponry to use to kill enemies, weapons are placed at the hip and over the shoulder. A level has a set amount of enemies to defeat before winning. Point-based system. Fast paced. Various types of enemies.

These popular games can provide a better insight on how to implement certain mechanics, and as an overall source of inspiration for design decisions when creating the final game used in our test.

2.5. Measuring Game Experience

To determine if SR has an effect on game experience, it is necessary to find measurements that reflect the user's experience of playing video games. The following sections explore existing methods for measuring game experience.

2.5.1. Continuation Desire

Continuation desire is the willingness to continue engaging in or keep playing a game (Schoenau-Fog, 2014). This is mostly based on the motivation that a user could have to continue playing a game, which could be a motivator included by the game or a motivator the user comes up with on their own (Schoenau-Fog, 2014).

Schoenau-Fog (2014) goes through different prior papers that have investigated engagement and what motivates people to continue engaging in games. In most of the research found by Schoenau-Fog, the evaluations seem to have happened post-testing. Therefore, he investigates if it is possible to use continuation desire during play-testing, and finds that it is definitely possible (Schoenau-Fog, 2014).

Continuation desire can be measured through questions that appear while the user is in the game, these questions would then require simple answers based on a numbered list (Schoenau-Fog, 2014). This was called an intrusive method because it is possible to get information from a user while he/she is engaged in the experience, which could provide more accurate measurements as the user does not have to remember the experience for a post-questionnaire (Schoenau-Fog, 2014).

2.5.2. Game Experience Questionnaire

Another method for measuring game experience is the Game Experience Questionnaire (GEQ), it was created by IJsselsteijn et al. (2007) because at the time it was problematic to measure experience related to video games. This was because there was no definitive method created to assist in collecting data. Although there does not seem to be any direct confirmation on the validity of the GEQ from IJsselsteijn et al. (2007), it is still widely used for getting measurements on game experience (Norman, 2013). The GEQ has been analysed in other papers and compared with different methods of measurement for games, but has been found to be a tool for assessing what it is in a game that contributes to the game experience (Norman, 2013).

The GEQ can be used in several different scenarios depending on when an experiment wishes to deploy it (IJsselsteijn et al., 2013), as for this paper there is an interest in deploying it after the test session, which leads to the possible use of the core module of the GEQ. The core module contains 33 statements that have to be rated from 0-4 by the user. The 33 statements belong to one of seven categories, these categories are related to different aspects of experience in games (IJsselsteijn et al., 2013). This will assist in measuring the experience in a large aspect, and also get some information on the categories if some were more effective than others.

2.6. Virtual Reality Equipment

Selecting the proper VR equipment for SR experiences is important since the two most renowned HMDs, the Oculus Rift and HTC Vive, use different setups for tracking the headset and controllers. Both HMDs use additional equipment to track their positions; the Vive uses two Lighthouses that are placed diagonally and the Rift has two infrared cameras that are placed in front of the user (Borrego et al., 2018). This means that the Rift was never meant to do room scale tracking, as the user would occlude the controllers if they faced the opposite direction of the two cameras. This would make the Vive the best choice out-of-the-box, but because there has been an increasing interest in using the Rift for room scale tracking, Oculus has released a guide that should allow room scale tracking with either two or three infrared cameras (Oculus, 2018). According to the guide, the setup should be able to track a 12ft by 12ft space which creates the same play area as the Vive, but the guide does warn that the performance may vary.

Given this information, both HMDs seem like equally good options in terms of tracking. Since we decided to use outside-looking-in for marking the physical objects both the Vive and Rift can be used. If we had decided to use inside-looking-out, the Vive would have been a natural choice, because of its front-facing camera, which the Rift is missing. It is then a question of availability and because we have an Oculus Rift readily available and have experience working with the SDK, this project will utilize the Rift for creating the VR experience.

2.7. Analysis Conclusion

In this section, we will shortly summarize the various sections of the analysis and explain how we will be utilizing the information we have gathered. This will follow a chronological pattern similar to when the information was presented in the analysis.

The topic of SR was explored and described where the mismatch between physically and virtually paired objects is controlled. As the participants in this experiment are not going to directly interact with the substituted objects, a large part of findings, done by Simeone et al. (2015), regarding features can be ignored. The most prominent feature of the substituted objects will, therefore, be the size. A system for adding physical objects to the VE was also investigated. Two different methods of viewing the VE, while marking the objects, were tested, inside-looking-out and outside-looking-in. We chose to use the fastest of the two methods, outside-looking-in, even though it had slightly lower accuracy, as the main experiment comes after the marking of objects.

The chapter on SR is followed by the research of passive haptic feedback. As was explained in the SR chapter, it creates a virtual representation of a real object, which means that if a user were to reach out and touch the virtual object the user would also touch the real object which is what passive haptics is. Therefore, we know that passive haptics is included with our SR system by default and thus proper alignment between the virtual and physical objects is important.

When working with VR, it is necessary to consider locomotion as it controls how the user will navigate inside the VE. When working with SR the natural locomotion method is Room Scale-Based locomotion as it will not move the tracked area around in the VE, which would otherwise also move the substituted objects, and it gives users the most natural method for navigating amongst the physical objects.

Following the locomotion, the current popular VR games were explored. All of the mentioned games have been researched in order to have a point of reference in terms of design. This research was for the sake of getting inspiration from games that have been rather successful and incorporate relevant aspects into the game which could be useful for the experiment.

In our initial problem statement, we mentioned game experience and whether or not it could be affected by modifying the virtual space. This led to researching methods used for games that can measure the user experience. The two methods mentioned in section 2.5. were the ones found to be of most relevance to the paper.

Finally, the selection of HMD system was shortly explored, the Oculus Rift and HTC Vive. The two systems were compared to one another in order to be able to select one. In the end, we opted for the Oculus Rift, as it fulfilled all the requirements regarding tracking space, the possibility of room scale tracking, availability, and experience developing for this platform.

This led to the following problem statement.

> ***Final Problem Statement:***
>
> *To what degree is game experience affected by incorporating a substitutional reality system in a virtual reality action shooter game?*

3. Methods

The following chapter will establish a hypothesis, determine what data will be measured and how it is analysed. The type of experimental design will also be discussed.

3.1. Experimental Measurements

The focus of this section will be to determine what measurements are needed to measure the performance of the experiment.

3.1.1. Game Experience

The main goal of the experiment is to determine if a user's experience will differ between a normal VR experience and an SR experience in the same shooter game. In the analysis two different measurements were explored, continuation desire and the GEQ, both measure the user's experience, but in different ways. As the GEQ is made for measuring experience, it will be the primary measurement that will be used to determine the effect of the experiment. The GEQ includes 33 questions divided into seven factors where two of these are negatively worded (IJsselsteijn et al., 2013). Because we are interested in the complete experience and not the different factors, a single value will be calculated by taking the average of the 33 questions, after flipping the negatively worded scores. This gives us a single score for each participant that can be used for statistical testing to determine if a significant difference exists between scores. The GEQ questionnaire is given to participants after the experiment is done.

The GEQ will serve as the primary measurement, but users will also encounter the continuation desire measurement while they are doing the experiment. The participants will be asked to rate how much they wish to continue, on a scale of 1 to 5, at three evenly spread time points throughout the experiment. The measurement will be done while the user is immersed in VR, hopefully minimising the effect the measurement has on the user's immersion, while in VR. The continuation desire scores will serve as complementary data, potentially revealing how entertaining the game is over the course of the experiment.

3.2. Hypothesis

If SR is to be used in commercial VR games, it is important that adding support for SR does not affect the game experience. It is likely that game developers will be less inclined to add SR to their games if it affects the game experience, especially if it does so negatively. To evaluate whether there is a difference in user's game experience between VR and SR in an action shooter game, the following hypothesis was created:

> **H0**: *For a virtual reality shooter game, there will be a significant difference in game experience, between the regular game and a version with substitutional reality.*
> **H1**: *For a virtual reality shooter game, there will be no significant difference in game experience, between the regular game and a version with substitutional reality.*

3.3. Active & Passive use of SR

When a home user sets up their VR equipment, they would have to select a space that has sufficient room for movement. Most rooms in a home would have pieces of furniture and these would likely overlap with the play area to some extent. Because physical objects added to the VE using SR becomes a part of the game experience, several scenarios are possible when preparing the play area, before a gaming session:

Scenario one, the user would move the furniture away in order to have as clean a space as possible to play in, removing all furniture from the play area, this could be considered the regular optimal VR gaming experience.
Scenario two, the user would let the furniture stay in place inside the play area. This requires the user to either remember the position of the furniture in order to not hit it while playing, or use SR to create virtual counterparts. Using SR would make the scenario easier to handle as the user would be able to see the objects even when inside VR.

As SR is able to include furniture as virtual objects that become part of the VE, could the user then not take advantage of this feature? This was the subject we came upon, which leads us to:

Scenario three, the user would move the furniture to specific positions in their play area, so that the virtual counterparts, created using SR, would give them an advantage in the VR game. In the case of an action shooter, the user could utilize the physical objects to create walls around them for protection, thereby giving themself an advantage that was not a part of the game's design.

Scenario two and three create two separate ways for the user to incorporate SR in their VR experience. In order to distinguish between the two scenarios the following two terms were created:

- Scenario two: **Passive-Simple Object Usage** (Simple Approach), the user that utilizes their environment as-is. They do not manipulate the physical objects in the play area in any way.
- Scenario three: **Active-Advantage Object Usage** (Active Approach), the user that actively tries to use the physical objects to their advantage by moving furniture in and out to get the most use out of the created virtual counterparts inside the VE.

We believed that these two approaches would emerge if SR systems became a regular tool used at home. Comparing these two approaches could create an interesting research subject but for this experiment, we decided to use the Simple Approach. We believe that before the Active Approach can be tested, it is necessary to determine whether SR alone has an effect on the game experience.

3.4. Experimental Design & Conditions

The following two sections will cover the different conditions of the experiment and the experiment design.

3.4.1. Experiment Conditions

The experiment will have two conditions, a control condition, and an experimental condition. The control condition will have the normal version of our game, where participants will play it, as they would any other VR game, in terms of the physical surroundings. The experimental condition will have an extra step before the gameplay, where participants utilize the SR tool to create a virtual counterpart of the physical objects in the play area.
As can be seen in table 1, this created a single independent variable condition, with two levels (Control and Experimental). Where the dependent variable is game experience, measured with the GEQ questionnaire (Field & Hole, 2003).

Independent Variable	**Dependent Variable**	**Description**
Control Condition: Game version with no SR tool.	Game experience	The regular version of the VR game, where physical objects in the play area are not marked.
Experimental Condition: Game version with SR tool, prior to gameplay	Game experience	VR game version, where an SR tool step is added prior to the gameplay, allowing participants to mark objects in their play area.

Table 1: Overview and description of the different experimental variables.

In order to avoid introducing another variable to the experiment, the physical objects in the play area were placed in fixed locations for both conditions, in accordance with the Simple Approach for using SR. Although the objects are only utilized in the experimental condition, with the SR tool, they are still meant to represent a potential physical play area in a domestic setting. This forces participants to have some awareness of their surroundings, in the control condition, even if the cannot see the objects.

3.4.2. Between-Group Design

A between-group design was chosen for this experiment, as we believe the learning effect of playing the game multiple times would have a negative impact on the game experience and the data gathered. This meant that participants could not be used for both conditions, resulting in a larger required participant pool (Field & Hole, 2003).
The main drawback of using a between-group design is that the variance between participants might be very large, as some participants might have more experience with VR and might play video games more frequently where other participants might not. This means that in order to detect a significant difference between the conditions the effect of the experiment will have to be relatively large (Field & Hole, 2003)

3.5. Data Analysis

The experiment has a single independent variable on two levels, creating two groups of data, a simple *t*-Test can be used to analyse the data. Because the experiment uses a between-group design the Independent *t*-Test should be used, as this compares the mean of two different groups (Field & Hole, 2003).
Because the *t*-Test is a parametric test three assumptions must be met:

- The data must be measured on an interval or ratio level. As both the GEQ and continuation desire uses Likert items on a range of 0 to 4 and 1 to 5, respectively, the data collected is interval data and our data meets this requirement (Field & Hole, 2003).
- The data gathered from the experiment must be normally distributed. This can be tested using the Shapiro-Wilk test of normality (Field & Hole, 2003).
- The final assumption is the *homogeneity of variance*, meaning that the variance of the mean should be equal for the two groups (Control and Experimental). This can be tested using Levene's Test (Field & Hole, 2003).

If any of these three assumptions are not met there are methods for correcting the data, and if that does not solve the issues within the data sets, non-parametric tests can be used instead (Field & Hole, 2003).

4. Design

Using the research found during the Analysis chapter, this chapter is split into two, the first part will discuss how the SR tool should work and the second part discusses the design of the VR action shooter game, in which the SR tool will be tested.

4.1. Tool Design

This section will explain the functionalities that are expected from the implementation of the tool. The aim of the tool is to make it easy and intuitive for the user to utilize, while still remaining accurate in terms of representing the object in the virtual environment.

The tool's design is based on the implementation of Garcia et al. (2018), but it will be slightly altered to make it faster and simpler to use. The tool that Garcia et al. (2018) made uses an approach of having the user draw the volume of the object. The system requires either four or eight points from the user, depending on the size of the object. For our system, we decide to only use four points, as this will make it faster for the user, and also creates less room for error. The tool will provide the user with feedback when substituting the objects both visually and through haptics. Every point placed will be visualized with a sphere and the tool will provide a short vibration through the Oculus controllers to let the user know that the input was received. The tool will also include an undo command, which can clear any placed points, or delete the previously substituted object if any has been placed. In addition to correcting mistakes, the tool should also have measures to detect wrong inputs, such as if the user accidentally provides duplicate inputs.

The tool will not require the user to handle an HMD or a tablet as they do in Garcia et al. (2018), as we do not feel it is necessary, in terms of accuracy. Instead, the system will only require a tracked controller to provide the system with the points in space. The virtual environment will be displayed on the computer monitor where the user can see the substituted objects.

4.2. Game Design

This chapter will explain the different design decisions that were made, both regarding the game itself but also the SR tool. As the VR platform for games is relatively new and is still being explored, no literature on VR game development was found. Alternatively, we have used other popular VR titles as inspiration throughout the different design topics.

4.2.1. Game Type

There are many different game types within the shooter game genre to pick from. A lot of the games mentioned in chapter 2.4.1 are somewhat similar game types using wave-based enemies. Both Space Pirate Trainer (I-Illusions, 2016) and Superhot (SUPERHOT Team, 2016) utilize a form of wave-based game type where you have to kill the current enemies in order to progress to the next ones. As we will be using room scale tracking for locomotion it makes the most sense to draw inspiration from Superhot, as it uses the same locomotion type. This means that the game should be created as a fully wave-based game.

4.2.2. Substituted Objects

The virtual objects that are substituted in the VE, are supposed to become part of the virtual world in a way that allows the player to both see and feel the virtual object. To avoid breaking the plausibility illusion the virtual objects also have to become a part of the VE, more than what was the case in the experiment by Garcia et al. (2018) where they only marked physical objects and added virtual substitutes. The reason is that our implementation will have non-player-characters (NPCs), in the form of enemies, that will walk towards the player. If substituted virtual objects are not incorporated in the VE properly, the NPCs will simply walk through them instead of around them. Properly incorporated objects can, therefore, work as cover from enemies, adding additional functionality to the physical objects in the play area.

4.2.3. Weapon Interaction

The weapon interaction draws a lot of inspiration from the VR game Robo Recall (Epic Games, 2017), as it has an intuitive and straightforward interaction system. In 'Robo Recall' the interaction for equipping a weapon is to move the hands down to the sides of the legs, or to the shoulders, and then press and hold a button on the VR controller, this simulates the experience of 'grabbing' a weapon as if it was really located in these positions (Epic Games, 2017).

We decided to copy this same approach of equipping a weapon and chose to copy the placement of weapons by the side of the user's legs, this made it so that a user only had to move his/her hands down the side of the body, and press the associated button on the controller to grab the weapons, if the button was released then the weapon would also be dropped in the game.

Another design decision that was impacted by 'Robo Recall' was that weapons do not reload. This meant that whenever a weapon had no more ammunition the weapon was useless. Basically, there is no reloading in the game, the user just has to grab a new weapon in the holster. This interaction was also copied into our game.

Because there are two controllers for VR to simulate two hands, we decided that a weapon could be held in each hand, this meant that the grabbing interaction was located on either side of the user, as both hands were able to grab a weapon to use in the game. This was also a feature that 'Robo Recall' had.

4.2.4. User Interface

User Interfaces (UI) for VR are not handled in the same way as for regular computer screens, this is because of how VR works, and it cannot incorporate a UI which is bound directly to the camera's view ("User Interfaces for VR", n.d.). This does not mean it is impossible to have a UI in VR, so instead of placing the UI in what is called screen space it is instead placed in world space ("User Interfaces for VR", n.d.). Our VR game is very short and will only need UI for the continuation desire measurement, for displaying damage to the player and showing weapon placement, creating a very minimalistic UI.

As the continuation desire will appear as an intrusive measurement, it is given that it can possibly break immersion for the duration it is there. Therefore, the measurement should fit in with the virtual environment visually, so that it does not feel too out of place. It should also not appear during gameplay, which means that a good time to take each measurement is when the player is waiting for the next wave of enemies to spawn.

To display when the user is damaged and how the damage has been sustained, simple screen effects can be used such as in Arizona Sunshine or Robo Recall. More specifically, a vignette effect could be used. This would appear as an overlay on the user's view with a red edge appearing and increasing if damage accumulates. This UI would disappear after some time has passed to signify the regaining of health.

Lastly, the weapon spawn locations can be a visual representation of the weapons to provide an indication of where to place the controllers in order to grab new weapons, much like how Robo Recall implemented it.

4.2.5. Enemies

As the test is rather short we will limit the different types of enemies to two in order to minimize the learning curve. The two different types of enemies will namely be a ranged unit and a melee unit, which will increase the diversity of the gameplay and add more of a challenge. This was also inspired by the two VR games Robo Recall and Superhot, as they both have melee and ranged enemies. The ranged enemy will fire projectiles at the user which the user has to avoid, and in the experimental condition, there will also be SR objects that can be utilized to dodge the arrows. The ranged units will spawn in stationary positions around the environment, making them harder to hit as they are some distance away but should require fewer hits to kill. The melee enemies should spawn some distance away first and then move towards the user in their designated paths. For this reason, these enemies should also be more armored than the ranged unit and should, therefore, take more hits before dying.

As the game will be made to spawn enemies in waves, the number of enemies should vary in between waves in order to slowly progress the difficulty to maintain the challenge. For example, the first wave could have five melee- and one ranged enemies, and in the last wave it could be 20 melee- and five ranged enemies.

4.2.6. Environment

The environment will be designed with the game mechanics and the various other assets in mind. Therefore, the graphics on the environment should be made with the same visual style as everything else. The environment should also support wave-based levels, which means various paths have to be made for the enemies to traverse, and space to represent the VR play area. Inspiration can be taken from Unity's demo Book of the Dead (Unity, 2018), which is a sizeable forest environment with many different high-quality nature assets. These assets are free to the public, which means that they can be imported into the project.

5. Implementation

The following chapter will describe how different aspects of the experiment were implemented, this is both in regard to the SR tool and the game it was tested in.

5.1. Substitutional Reality Tool

The Substitutional Reality Tool (SR tool) implementation works by the user drawing points in space (vertices), and these vertices are then used by the tool to approximate a cuboid representation of the drawn object. Both the position and orientation of the substituted object are approximated based on the four vertices, and a few assumptions.

5.1.1. Position & Orientation

As seen in figure 3, the position of the substituted object is based on the average position of each of the four vertices placed by the user. It adds up the XYZ coordinates of each vertex and divides it by the sum of vertices placed, resulting in the center position of the four vertices.
The assumptions made for the position is that stacked substituted objects are drawn by the user from the bottom and up. This allows the function to only require an input of four points, as the user only has to draw the topmost surface. The height of the object is then calculated automatically by ray-casting downward to find any intersecting objects. The height of the substituted object is then set to the distance of the ray-cast when it hits either another object or the floor.

```
//Returns the averaged center position of the given vertices
private Vector3 GetAveragedCenter(Vector3[] vertexPositions)
{
    Vector3 center = Vector3.zero;
    //Add up all the positions
    for (int i = 0; i < vertexPositions.Length; i++)
    {
        center += vertexPositions[i];
    }
    //Average the positions to get the center
    center = center / 4;

    return center;
}
```

Figure 3:
Code snippet responsible for the X&Z position of the substituted object.

As shown in figure 4, the rotation of the substituted object is calculated by choosing the first vertex as the reference point. The shortest line segment is then found by looping through the other vertices and comparing their distances to the reference point. The facing direction is then calculated by subtracting the position of the reference point and the position of the vertex with the shortest distance.

```
//Returns the facing direction of the shape made by the placed vertices which are used to rotate it accordingly.
    private Vector3 GetVerticesFacingDirection(Vector3[] vertexPositions, float averageHeight)
    {
        Vector3 startPos = vertexPositions[0];
        startPos.y = 0;

        float shortestDistance = 10000;
        Vector3 shortestPoint = Vector3.zero;

        for (int i = 1; i < vertexPositions.Length; i++)
        {
            Vector3 vertex = vertexPositions[i];
            vertex.y = 0;

            float distanceToPoint = Vector3.Distance(startPos, vertex);

            if (distanceToPoint < shortestDistance)
            {
                shortestDistance = distanceToPoint;
                shortestPoint = vertex;
            }
        }
        return (shortestPoint - startPos).normalized;
    }
```

Figure 4:
Code snippet responsible for approximating the rotation of the substituted object.

5.1.2. Undo Functionality

The undo function in figure 5 first checks if any vertices have been placed by the user. If any vertices have been placed, it will then remove all of them. However, if none of the vertices have currently been placed, it will check if there are any substituted objects currently placed. If there are any substituted objects in the virtual environment, it will delete the most recently placed one.

```
private void Undo()
{
    if (anyVerticesPlaced() == true)
    {
        ResetVertices();
    }
    else if (spawnedObjs.Count > 0)
    {
        GameObject toDestroy = spawnedObjs[spawnedObjs.Count - 1];
        spawnedObjs.RemoveAt(spawnedObjs.Count - 1);
        Destroy(toDestroy);
    }
}
```

Figure 5:
Code snippet responsible for the undo command.

5.1.3. Accuracy

A small test was conducted to measure the error rate of the SR tool. The test consisted of measuring the topmost surface area of different objects with a ruler and then comparing it with the surface area of the substituted object in the virtual environment. As seen in figure 6, the average error percentage was roughly 0.6876%.

Measurement no.	Percent Error (%)
1	1.58
2	0.008
3	1.108
4	0.082
5	0.66
Sum Average	0.6876

Figure 6:
Percent error of the substituted object's surface area.

5.2. Virtual Reality Shooter Game

This section will explain the process of creating the game needed for the experiment. This includes the creation of the environment, the enemies and their behaviour, the weapons available to the user, and how it included the SR substituted objects into the game.

5.2.1. Environment

The environment base is made using the Unity Terrain Editor, where the terrain mesh is drawn in with a brush based on a heightmap. The nature assets in the environment (figure 7) are made from a pack of prefabs with different trees, rocks, plants, etc. which were taken from Unity's demo Book of the Dead (Unity, 2018). The majority of the nature assets and textures are made from 3D scanning technology.

Figure 7:
Screenshot of the assets used to build the environment.

As a dense forest environment can be very hard to render, especially in VR, it can be a good idea to use a few tricks. We added fog to hide the horizon, and also blend in with the trees in the distance. As seen in figure 8, the trees in the background are actually quads with textures on them, to give the illusion of density, while keeping a high frame-rate.

Figure 8:
Quad with a texture creating the illusion of tree meshes in the distance.

5.2.2. Enemies

The game had two types of enemies, an undead melee unit and a skeleton archer, see figures 9 and 10. The melee unit moved using root motion animations. This means that the animation clip is responsible for moving the character. This creates smooth motion without footsliding, making the motion of the enemies more realistic.

Figure 9:
Melee unit, that will walk towards the player and attack when they are close.

Figure 10:
Ranged unit, will attack the player from the position they are spawned in.

To navigate the environment the melee unit used Unity's built-in pathfinding, the NavMesh Agent. The NavMesh Agent uses the predefined NavMesh, see figure 11, to calculate a path to the player and will also avoid obstacles along the way. Using the agent's desired velocity, a vector pointed towards the next path waypoint, an appropriate root motion animation is selected and played. When the melee unity was within range of the player it would play an attack animation and damage the player. To create some variability between the melee units, they would spawn with a random weapon and have different pieces of armor disabled. The ranged unit did not move around the map and would stay where it was instantiated. From this position, it would rotate towards the player and shoot arrows at a constant pace.

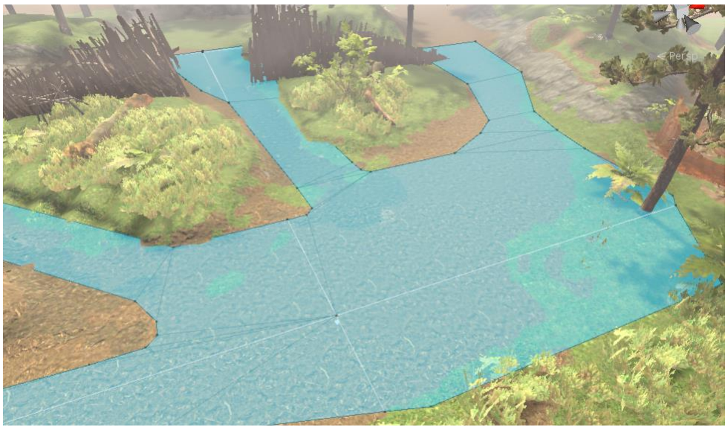

Figure 11:
The blue NavMesh is used by the melee units' NavMesh Agent to calculate a path to the player.

Because the skeleton model did not follow the standard humanoid joint layout, it was not possible to utilize inverse kinematics to point the character towards the player, as if it was aiming. Instead, the rotation of the units spine and neck bones were rotated in LateUpdate, after the animations transformation was applied for the current frame, this is illustrated in figure 12.

```
void LateUpdate()
{
    if (AimingChest) //Rotate chest
    {
        ChestBone.LookAt(Target.position);
        ChestBone.rotation *= Quaternion.Euler(ChestOffset);
    }
    if (AimingNeck) //Rotate neck
    {
        NeckBone.LookAt(Target.position);
        NeckBone.rotation *= Quaternion.Euler(NeckOffset);
    }
}
```

Figure 12:
If aiming is enabled, the chest and/or neck bone will rotate towards the target (player). Because the bones had offset rotation when imported, an offset is added, to ensure that the unit is pointed towards the target.

Unity's physics system was used for both enemies to simulate death. This could have been done using animations but in an effort to create some variability in the very limited content in the game, active physics were used. When the archer was defeated it would appear to explode, this was done by disabling the animated model and activating individual bone models in the pose the skeleton had when it was killed. A small force was then added to these bones, spreading them in the opposite direction they were hit from. A ragdoll effect was used for the melee unit when it was killed, disabling the animator and enabling the rigidbodies on all the limbs, making the character interact with the terrain and surround enemies.

5.2.3. Weapon

The weapon will be spawned very frequently, as you need to spawn new weapons every time you reload. This can quickly become very performance intensive as every time something is instantiated it creates a lot of "garbage", which eventually will need to be cleaned up by the garbage collector, resulting in potential lag spikes. To solve this issue we implemented a system called "object pooling". As seen in figure 13, the object pooling works by preloading a large number of weapons when the game first starts, and then adding them all to a list. Whenever a request for a weapon is made, the object pool will supply a weapon from the list of weapons that are not active in the environment and activate it.

```
public void Awake()
{
    SharedInstance = this;
    m_freeList = new List<T>(m_size);
    m_usedList = new List<T>(m_size);

    // Instantiate the pooled objects and disable them.
    for (var i = 0; i < m_size; i++)
    {
        var pooledObject = Instantiate(m_prefab, transform);
        pooledObject.gameObject.SetActive(false);
        m_freeList.Add(pooledObject);
    }
}
```

Figure 13:
Code snippet responsible for creating the pool of weapons.

This makes sure that zero garbage collecting is being done as everything is cached at the start of the game when everything is created. The same system is used for the bullets, as this is also something that is required to be instantiated every time the player shoots.

5.2.4. Substitutional Reality Walls

The object used for substituting in the virtual environment is a stone block, which has been pre-scaled to fit in a unit sized box. This means that the base size of the stone block is 1x1x1m in size, as visualized by the orange volume box in figure 14. The reason for this is that the unit size allows us to scale it 1:1 directly from the input measurements of the user when substituting.

	
Figure 14: Screenshot of the object used for substitution.	Figure 15: Screenshot of the navigation mesh intersecting with the substituted objects.

Lastly, after the substituted objects have been created, they are baked together with the navigation mesh used for the enemies AI movement as seen in figure 15. This prevents the enemies from walking through the substituted objects, instead, forcing them to go around them.

6. Experiment

The following chapter will summarize the details of the final experiment, as well as the experimental procedure that the participants had to go through.

6.1. Participants

A total of 30 participants were collected for the experiment, five females and 25 males. All 30 participants were undergraduates of AAU CPH, therefore the sampling was done using non-probability and collected through convenience. These 30 were equally split between the two conditions, 15/15, which was done through a predetermined randomized list.

The participants had an average age of 24.83 years. All of the participants had tried VR applications before the test, of these, 11 had experienced cybersickness and another three were unsure. None of the participants got sick from our test. Of the 30 participants, seven did not play video games on a regular basis and four were unsure, of these 11 there were five that did not play video games at all.

6.2. Setup

For the experimental setup, we needed to have plenty of tracking space. This was mostly for the SR as we needed to be able to place objects and mark them with the SR tool within the space and the trackers should be able to see the controllers even with the objects there.

The VR equipment that was used for the experiment was an Oculus Rift, CV1. Which includes the headset, two controllers, and two tracking units, however, we found that because we wanted a large space we needed an additional tracking unit. The tracking space that was set up was 3x3 meters large, which is slightly lower than the measures mentioned in section 2.6 (12 feet, which approximates to 3.6 meters).

Figure 16 & 17 shows the experiment setup, which was used in both the control and experimental condition of testing. Everything was stationary for every participant and was corrected back into proper positions if moved around by accident, such as from participants walking into the boxes.

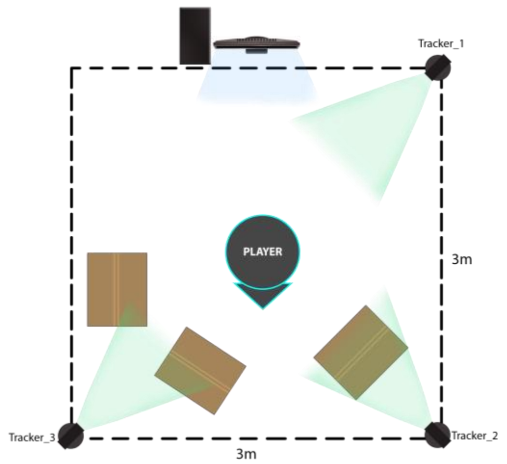

Figure 16:
The layout of the experiment setup. With three trackers and three physical objects in the play area.

Figure 17:
This image shows the entire setup. Included the PC, HMD, three tracking units (placed on the chairs), three objects in the tracking space, and laptop used for questionnaires.

When participants were wearing the HMD they would be faced away from the PC, such that the HMD cable would not get in their way while testing. Because the front was on the opposite side of the PC it was where two of the tracking units were located in order to track the front properly. The third tracking unit was placed next to the PC and behind the user, this was to be sure that tracking would occur even if the controllers were behind the boxes. In order to track properly, we elevated the trackers, as can be seen in figure 17, this was to make sure that the tracking units could look past the boxes.

The experiment was comprised of the following hardware:

- Oculus Rift, CV1
  - With one additional Oculus sensor.
- Windows 10 PC
  - Nvidia Geforce GTX 1080
  - Intel i5 8400 processor
  - 16 GB - DDR4 - 2666MHZ
  - 256 GB SSD

6.3. Procedure

In this section, we will explain the experimental procedure for our experiment. It will include, how we introduce the experiment, how we assigned participants to conditions, the tutorials needed to use the product, and the measurements.

Introduction to the experiment

Introductions to the experiment were given as a participant was recruited. The participant was told that he/she would participate in a VR action shooter experience, which would last up to 20 minutes. The participant was not told what was being measured for the experiment beforehand, as to avoid influencing the results.

After the introduction, the participant would be guided to take the Demographics questionnaire. This would collect data on age, gender, VR experience, gaming experience, and permission for whether or not filming and photos were allowed during testing. This was done on a laptop placed outside of the VR tracking space.

Assigning conditions

After the participant had finished the Demographics questionnaire, he/she would continue to the VR tracking space. The experiment had two conditions and the condition for each participant was determined beforehand with a prepared randomized list. If a participant was in the experimental condition they were given an introduction to the SR tool.

Before any of the following steps began the participant was given an explanation on the tracking space available. The participant was informed of the size of the space and the presence of the boxes was explained. Participants in the control condition were told that the boxes would be present in the area but they could not be seen, while participants in the experimental condition were told that they would be including them in VR.

SR Tool Tutorial

Only the participants that were assigned to the experimental condition went through this step of the procedure. The participant was told that he/she had to mark four corners on the six boxes in the play area, and the tutorial on the controls for this segment was given verbally to

the participant by the experimenter. Figure 17 shows the six boxes in question, where the four corners were the top four in the bottom box, and the top four corners on the upper box.
The participant was allowed to touch the boxes as much as needed to place the corners. When a participant had placed four corners, a virtual object was spawned which could be seen on the PC screen, and the experimenter would have the participant notice this after one placement. The participant was then asked to proceed with marking the remaining boxes and the experimenter would make sure all six would be present and no input errors were made before continuing with the test.

Video Game Tutorial

Participants that were assigned to the control condition went directly to this step after finishing the demographics questionnaire. This step was initiated by the experimenter after the SR tool step was concluded.
Before the tutorial started the participant would be given the VR equipment and it was made sure that these were fastened to the participant. The experimenter would then move the participant to the middle of the tracking space and make sure that they faced in the correct direction before beginning. The participant was told that the tutorial would introduce him/her to the mechanics of the game, and after the tutorial was finished it would continue into the game itself. The tutorial was then started by the experimenter. From here on the experimenter took the opportunity to observe and write down details of the participant's playthrough, also the experimenter would take photos and record videos, if the participant had given permission.

Continuation Desire during playtime

During the participants' playtime, a scale would appear before them, which posed the question of how much they wanted to continue, and then had them giving a score from 1-5, where the higher the score the more the desire to continue.
The experimenter would notify participants of this if they did not spot it by themselves or would instruct them in how to interact with it if this was confusing. The participants had to shoot the score they wanted with the gun in the game, in order to continue the experience.
The score did not decide whether or not they would continue but was a measurement for data purposes. Some participants did question this but were quickly given an answer from the experimenter that it was only a measure and did not affect gameplay directly.

The continuation desire question would appear three times and every time the participant had to shoot a score to continue. This would be instructed if the participant did not figure it out.

GEQ after the play session

After the participant had finished his/her playthrough of the game, the experimenter would come to help remove the VR equipment.

The participant would then be directed back to the laptop to answer the GEQ questionnaire.

End of the experiment

After the participant had finished the GEQ the testing was over. The experimenter asked the participant how the whole experience had been at this point to round off the process. The participant's remarks at this point could be used in observation notes.

7. Results

The following chapter will present the results of the experiment. Data analysis was done using SPSS Statistics (v. 25.0) and diagrams were created using Microsoft Excel (v. 1904).

7.1. Descriptive Statistics

Figures 18 and 19 show the average GEQ score for all participants and an average for each condition. Figure 20 shows the average scores for the seven factors of the GEQ questionnaire.

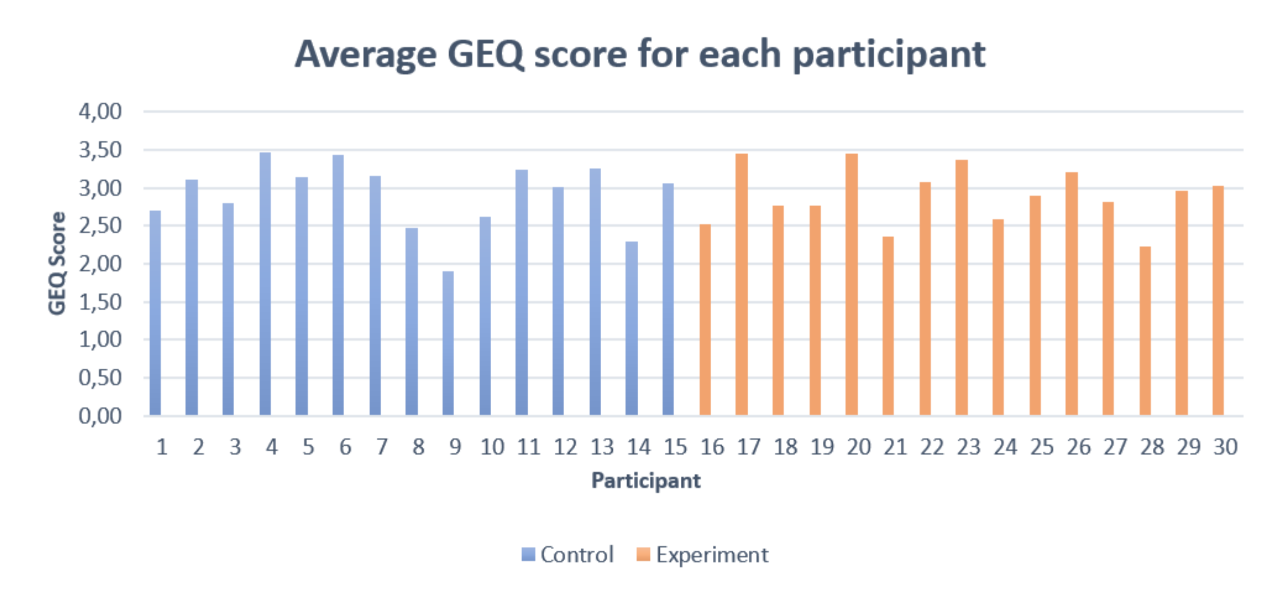

Figure 18:
Bar chart of each participants average GEQ score. Bars colored blue belong to the Control group and the orange bars belong to the Experimental group.

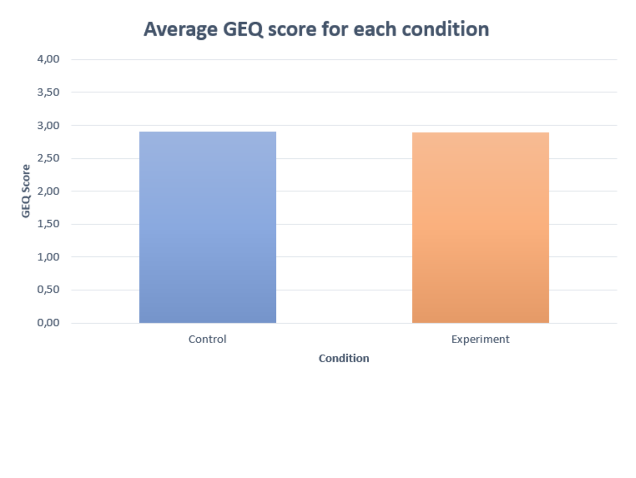	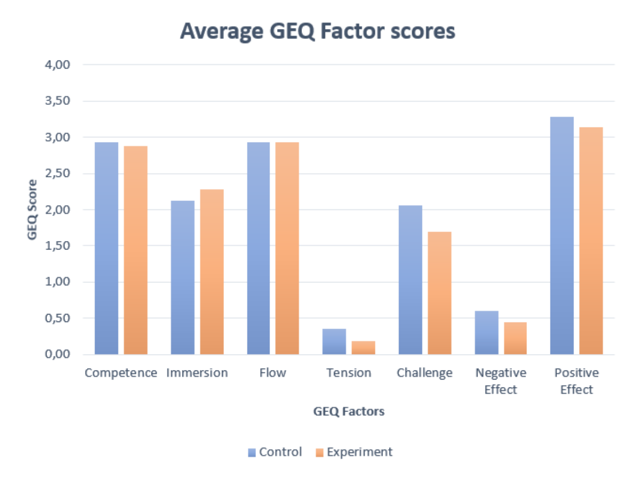
Figure 19: Bar chart of each condition's average GEQ score. Bars colored blue belong to the control group and orange bars belong to the experimental group.	Figure 20: Bar chart of each GEQ Factor's average score. Blue bars belong to the control condition and orange bars belong to the experimental condition.

Figure 21 shows the development of the continuation desire scores throughout the experiment for both conditions.

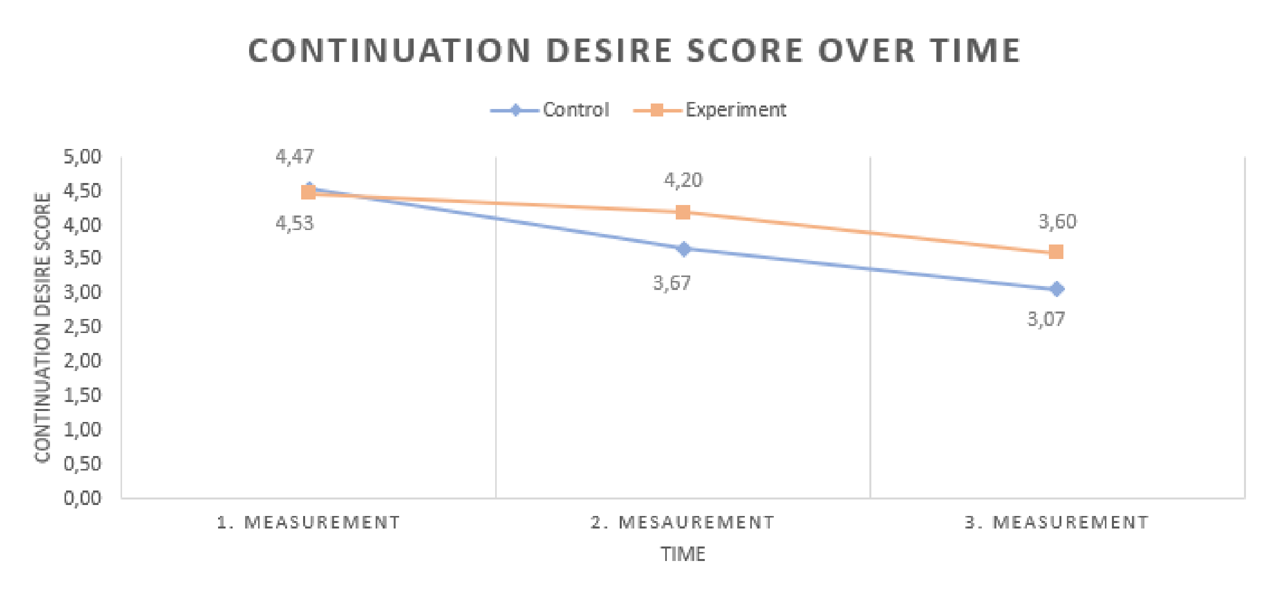

Figure 21:
Graph showing the development of the continuation desire score. Each of the three measurement points is an average of the participants' scores.

7.2. Determination of Parametric Data

In order to determine that the gathered data is parametric, it must meet three requirements, the first requirement is the level of the data, and it has already been discussed and determined to be of the interval level for both the GEQ and continuation desire data. The second requirement is that the data is *normally distribution*, the data was tested using the Shapiro-Wilk Test and the results showed that both the control group ($D(15) = 0.929$, $p = 0.266$) and the experimental group ($D(15) = 0.965$, $p = 0.779$), in the GEQ dataset, were normally distributed. The test showed that the continuation desire data, measured over time, were all non-normally distributed, Measurement 1 ($D(15) = 0.387$, $p <.001$), Measurement 2 ($D(15) = 0.346$, $p < .001$) and Measurement 3 ($D(15) = 0.224$, $p = 0.01$).

The third requirement, is *homogeneity of variance* and this was, for the GEQ data, tested using a Levene's Test, were the result indicated an equal variance ($F = 4.4$, $p = 0.51$), meaning that the GEQ data is parametric. For the continuation desire data was tested using the *Mauchly's Test of Sphericity*, as the data was collected using repeated measures. The results showed that the assumption of sphericity was violated, $X^2(2) = 16.1$, $p <.001$. The degrees of freedom was therefore corrected using Greenhouse-Geisser ($\varepsilon = 0.69$) (Field & Hole, 2003).

7.3. Statistical Test

To determine if there exists a statistically significant difference between the average GEQ scores of the two conditions, an Independent *t*-Test was used. The result showed that there was no significant difference between the control condition ($M = 2.9$, $SD = .44$) and the experimental condition ($M = 2.9$, $SD = .38$), $t(28) = .6$, $p = .95$.

The continuation desire data was tested using a Mixed ANOVA, with a Greenhouse-Geisser correction, using the two factors, Condition (Control and experimental) and Time (Measurement 1, 2 and 3). The results showed that there was a statistically significant difference for the Time factor, $F(1.38, 38.67) = 15.41$, $p < .001$. There was no statistically significant difference for the Condition factor, $F(1, 28) = .81$, $p = .38$, and the interaction between Condition and Time, $F(1.38, 38.67) = 1.36$, $p = .26$.

Because the data is non-normally distributed, the result might not be completely accurate and should be interpreted with that in mind (Field & Hole, 2003).

8. Discussion

In this chapter, we will be discussing the results as well as the different factors that might have affected the results, and how they could have possibly been avoided or alleviated.

8.1. The Effect of SR on Game Experience

As the analysis of the GEQ data shows that there is no significant difference between the two conditions. This means that we reject the null hypothesis and accept the alternative hypothesis, that there is no difference in game experience when adding SR to a VR shooter game. This means that it should be possible to add SR to any VR shooter game, in theory. We believe that the very simple nature of our game is not sufficient to determine whether SR might have an effect on all existing and future VR shooter games.

SR did not interfere with the game experience for our game, likely because it only contained very simple game mechanics that did not change throughout the game. The simple nature of our game could also explain the decrease in continuation desire, as no new elements or mechanics were introduced over the course of the game. This also means that more advanced games might have different game mechanics that would be affected by adding unintended objects using SR.

The best solution to thoroughly test SR would be to implement it into existing popular VR shooter games and measure if the game experience was affected. There are quite a few problems with this approach, as it would require access to the games' source code and the locomotion method might not be Room Scale-Based. With these limitations, we believe that our experiment has contributed valid data that can help explore the field of SR and its effect on domestic VR experiences.

8.2. A significant difference in the Time factor

In figure 21 of the Results chapter, there is a figure showing the change in the continuation desire score over time for each of the two conditions, as can be seen, it has a fairly high score at the beginning but it drops by a fair amount by the end of the game. It was found that there was a significant difference in the Time factor which suggests that something about the game actually made the participants want to continue less as they progressed through the test. This is likely due to boredom as the game does not present any new elements. The game was too static in its gameplay which is what could have led to the fall in continuation desire.

The problem with the fall in continuation desire is that whatever affected this score could also have affected the results of the GEQ. If this was the case, then one would expect to see high values in the GEQ factors Tension and Negative Effect, however from figure 20 of the Results chapter it can be seen that the scores for these two factors are on the very lower end. This would indicate that the lack of new content in the game, should not have affected the GEQ in a significant way. Even if it had, the lack of a significant interaction for Condition * Time, in the continuation desire scores, means that the boredom effect would have affected both conditions somewhat equally.

If the game was to attempt to keep the continuation desire at the same level, throughout the experiment, various alterations could be made to hopefully keep the continuation desire from falling. A few examples of changes could be, to introduce some new enemies giving the user more to look out for, or giving the user new weapons which would make the user able to have more to strategize with. Extending the game with new experiences to discovers as time continued could have led to the user wanting to continue more to find the next new element, which could help with the continuation desire not dropping over time as much as it did in this experiment.

8.3. Potential Occlusion Problem

Both the Oculus Rift and HTC Vive uses external tracking stations to track the HMD and controllers. This means that if the path between the HMD/controller and the tracking stations is blocked, then the tracking will stop working. Because the Oculus Rift and HTC Vive have two external tracking stations, the event where the path to a controller or the headset is blocked from both is very unlikely, if the play area is cleared of objects.

Because SR utilizes the objects in the play area, users could potentially hide behind substituted objects during gameplay, making it much more likely that the tracking stations are unable to see the equipment worn by the user. When the equipment cannot be tracked the best-case scenario is that the position of the controllers are not moved when the user moves the hands, this would affect the game experience but not the user's health. The worst-case scenario, however, is that the position of the HMD is not tracked properly making it unresponsive and jump between positions. This would both affect the game experience and the user's health as cybersickness might occur.

8.3.1. Newer Generation of Oculus

A solution to the occlusion problem might already be here, as Oculus[2] has developed a new line of equipment since the Oculus Rift was released. The new Oculus Rift S uses a different tracking method called *Inside-out tracking* (Oculus, n.d.; Hillmann, 2019). This tracking method uses several cameras inside the HMD to track the controllers and to map the user's surroundings in order to track the HMD's movement and translate it to VR. This makes the external tracking sensors obsolete as the HMD is tracking the controllers, potentially making it much less likely that the controllers are occluded. This would, however, need to be tested in a separate experiment.

In relation to SR, because the equipment uses inside-out tracking and actually maps the play area on its own, then this tracking method could potentially be utilized to also perform SR substitution of objects, as the HMD creates many points of reference to track the room. If these cameras were also able to distinguish objects, then it would be possible to perform SR as the room is being mapped. However, this is purely speculation, rather than actual knowledge, but could be an interesting research topic.

8.4. Added Collision Protection with SR

Because the experiment was created to incorporate boxes in the play area, it was expected that the participants would collide with the boxes during the testing, especially in the control condition as they were not able to see the boxes while immersed in VR. It was noted during the experiment, see Appendix 1, that 8 out of 15 participants in the control condition collided with the boxes at some point during the test, however the specific amount of collisions was not recorded for each participant, only wording such as; "several" or "few" was used, not specific numbers.

No collisions, however, where observed in the experimental condition indicating that the substituted objects were able to give the participants somewhat of a reference to the real

[2] https://www.oculus.com/

objects' positions. While we cannot say with certainty what type of effect SR had on the participants, it at least shows that SR provides some safety to the user while playing in an area with real objects. It was even noted that 7 out of 15 SR participants actually used the virtual objects for cover. Which also means that SR actually added something different into the experience than just safety.

9. Conclusion

This thesis was based on the need for a safe way to engage with virtual reality (VR) content in a domestic setting, where the VR play area often shares space with furniture or other objects. Our analysis showed that research in the field has already been commenced under the name substitutional reality (SR), where previous research has uncovered different ways to substitute real-world objects with similar virtual ones. From this, our research question emerged, which focused on how SR would impact the experience of VR gameplay when the user is allowed to alter the virtual environment (VE). From this research, a simple VR action shooter game, featuring room scale-based locomotion, two types of enemies, and simple weapon mechanics, was created. The experience of the game was compared between two conditions; the control condition where no new objects were added to the VE, and the experimental condition where participants substituted physical objects with similar virtual ones. The results showed that there was no significant difference in the game experience between the two conditions, suggesting that it should be possible to use SR with VR action shooter games without affecting the experience.

It should be noted that the game designed for this project was very simple and the findings might not translate into a fully fledged VR experience, as this could contain mechanics that are not compatible with SR.

10. Future Work

In this chapter, we will mention various ways of continuing research on the subject of mixing VR video games and SR, including various research topics derived from this thesis.

10.1. Testing SR with other game genres

In this paper, only the shooter game genre was tested, with a relatively simple game specifically developed for testing. Although our results showed that SR can be applied to the shooter game genre and not disturb the experience, it would be worth investigating how SR might work in other types of game genres, such as puzzle games where height and placement might make a difference. An example of this could be if a key was placed upon a taller location than the player can reach, then with SR, the player might be able to place an object in the game and stand on it to reach the key. In VR this might require a longer process to get the key down, but SR can provide a different depth or have an actual impact on game design decisions.

10.2. Testing the Simple & Active approach

In section 3.3, we create the terminology Simple and Active Approach to help us determine how the physical objects could be used in the experiment. In this paper, we decided to go with the Simple Approach, where the physical objects are placed in static locations. We imagine that allowing users to freely alter the VE, with the Active Approach, by placing physical objects in opportune places, will create a very different game experience. Future research could compare the Active and Simple Approaches but could also try to determine, if future SR experiences should have a maximum allowed amount of substituted objects.

10.3. Substituted objects have in-game effects

Another subject that came up during the design phase of the game was whether or not the user should be able to interact with the substituted objects. The idea was that the game should contain certain virtual elements that were interactable. If the game was augmented using SR these interactable elements should be overlayed on the substituted objects, adding passive haptics to essential virtual elements. An example could be a virtual touch interface where the user can press buttons and use sliders, this interface could then be added to a substituted table or similar. This would be a direct inclusion of real objects in the virtual environment which promotes the usage of objects, rather than forgetting them or not using them, which might have been the case for the experiment in this paper in terms of actually making direct usage of the substituted objects.

10.4. Using different locomotion methods with SR

The Room Scale-Based Locomotion method was chosen because it was simple to implement and easy to use, and the method seemed to be the most fitting when utilizing SR. There are, however, other types of locomotion for VR but research in regard to SR is lacking and the only good solution is currently Room Scale-Based locomotion. More research into SR locomotion could reveal new and interesting methods for incorporating substituted objects in the user's movement.

Let us take a look at the Teleportation-Based Locomotion, after a user has substituted the real objects with virtual counterparts, this entire user-created environment will always surround the user, so if the user was to teleport around an environment created by other designers it would not necessarily fit with the user's immediate environment. A potential way of actually being able to use teleportation would be to designate special "empty" locations within the game which the user could teleport to together with the substituted objects, without it bothering the rest of the environment, of course, this would greatly limit the freedom of teleportation but it would allow a user to use SR and not be restricted to one position, as with the room scale-based locomotion.

11. Bibliography

10 Best Virtual Reality Shooter Games. (n.d.). Hentet 21. maj 2019, fra Fraghero website: https://www.fraghero.com/10-best-virtual-reality-shooter-games/

Azmandian, M., Hancock, M., Benko, H., Ofek, E., & Wilson, A. D. (2016). Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 1968–1979. https://doi.org/10.1145/2858036.2858226

Boletsis, C. (2017). The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology. *Multimodal Technologies and Interaction*, 1(4), 24. https://doi.org/10.3390/mti1040024

Borrego, A., Latorre, J., Alcañiz, M., & Llorens, R. (2018). Comparison of Oculus Rift and HTC Vive: Feasibility for Virtual Reality-Based Exploration, Navigation, Exergaming, and Rehabilitation. *Games for Health Journal*, 7(3), 151–156. https://doi.org/10.1089/g4h.2017.0114

Carbotte, K. (2018, juli 14). The Best Selling SteamVR Games of 2018. Hentet 21. maj 2019, fra Tom's Hardware website: https://www.tomshardware.com/picturestory/848-bestselling-steam-vr-games.html

Epic Games. (2017). *Robo Recall [Computer video game]*. North Carolina.

Feasel, J., Whitton, M. C., & Wendt, J. D. (2008). LLCM-WIP: Low-Latency, Continuous-Motion Walking-in-Place. *2008 IEEE Symposium on 3D User Interfaces*, 97–104. https://doi.org/10.1109/3DUI.2008.4476598

Field, A. P., & Hole, G. (2003). *How to design and report experiments*. London; Thousand Oaks, Calif: Sage publications Ltd.

Gall, D., & Latoschik, M. E. (2018). The Effect of Haptic Prediction Accuracy on Presence. 2018 *IEEE Conference on Virtual Reality and 3D User Interfaces* (VR), 73–80. https://doi.org/10.1109/VR.2018.8446153

Garcia Estrada, J., & Simeone, A. L. (2017). Recommender system for physical object substitution in VR. *2017 IEEE Virtual Reality* (VR), 359–360. https://doi.org/10.1109/VR.2017.7892325

Garcia, J. F., Simeone, A. L., Higgins, M., Powell, W., & Powell, V. (2018). Inside looking out or outside looking in?: an evaluation of visualisation modalities to support the creation of a substitutional virtual environment. *Proceedings of the 2018 International Conference on Advanced Visual Interfaces* - AVI '18, 1–8. https://doi.org/10.1145/3206505.3206529

Hillmann, C. (2019). Comparing the Gear VR, Oculus Go, and Oculus Quest. I C. Hillmann (Red.), *Unreal for Mobile and Standalone VR: Create Professional VR Apps Without Coding* (s. 141–167). https://doi.org/10.1007/978-1-4842-4360-2_5

I-Illusions. (2016). *Space Pirate Trainer [Computer video game]*. Brussels.

IJsselsteijn, W. A., Kort, Y. A. W. de, & Poels, K. (2013). *The Game Experience Questionnaire*. Hentet fra https://research.tue.nl/en/publications/the-game-experience-questionnaire

IJsselsteijn, W. A., Y. A.W. Kort, D., Poels, K., Jurgelionis, A., & Bellotti, F. (2007). Characterising and measuring user experiences in digital games. *Proceedings of the International Conference on Advances in Computer Entertainment Technology* (ACE 2007), *June 13-15, 2007*, 1–4.

Lang, B. (2018, december 28). Valve Reveals Top Selling VR Games on Steam in 2018. Hentet 21. maj 2019, fra Road to VR website: https://www.roadtovr.com/valve-reveals-top-selling-vr-games-on-steam-2018/

LaViola, J. J. (2000). A discussion of cybersickness in virtual environments. *ACM SIGCHI Bulletin*, 32(1), 47–56. https://doi.org/10.1145/333329.333344

Lindeman, R. W., Sibert, J. L., & Hahn, J. K. (1999). Hand-held windows: towards effective 2D interaction in immersive virtual environments. *Proceedings IEEE Virtual Reality* (Cat. No. 99CB36316), 205–212. https://doi.org/10.1109/VR.1999.756952

Norman, K. L. (2013). GEQ (Game Engagement/Experience Questionnaire): A Review of Two Papers. *Interacting with Computers*, 25(4), 278–283. https://doi.org/10.1093/iwc/iwt009

Oculus. (n.d). Oculus Rift S Features. Hentet 27. maj 2019, fra https://www.oculus.com/rift-s/features/

Oculus, V. (2018, januar 5). Roomscale Revisited: Getting the Most Out of Your Rift. Hentet 12. marts 2019, fra https://www.oculus.com/blog/roomscale-revisited-getting-the-most-out-of-your-rift/

Schoenau-Fog, H. (2014). At the Core of Player Experience: Continuation Desire in Digital Games. I *Handbook of Digital Games* (s. 388–410). https://doi.org/10.1002/9781118796443.ch14

Simeone, A. L., Velloso, E., & Gellersen, H. (2015). Substitutional Reality: Using the Physical Environment to Design Virtual Reality Experiences. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, 3307–3316. https://doi.org/10.1145/2702123.2702389

Slater, M. (2009). Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1535), 3549–3557. https://doi.org/10.1098/rstb.2009.0138

Slater, M. (2018). Immersion and the illusion of presence in virtual reality. *British Journal of Psychology*, 109(3), 431–433. https://doi.org/10.1111/bjop.12305

Sra, M., Garrido-Jurado, S., & Schmandt, C. (2016). Procedurally generated virtual reality from 3D reconstructed physical space. *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology* - VRST '16, 191–200. https://doi.org/10.1145/2993369.2993372

Steam. (n.d.). The Top VR Titles of 2018. Hentet 28. maj 2019, fra https://store.steampowered.com/sale/2018_top_vr/

SUPERHOT Team. (2016). *Superhot* VR *[Computer video game]*. Lodz.

Unity. (2018). Book of the Dead. Hentet 28. maj 2019, fra Unity website: https://unity3d.com/book-of-the-dead

User Interfaces for VR. (n.d.). Hentet 16. maj 2019, fra Unity website: https://unity3d.com/learn/tutorials/topics/virtual-reality/user-interfaces-vr

Vertigo Games, & Jaywalkers Interactive. (2016). *Arizona Sunshine [Computer video game]*. Rotterdam.

12. Appendix

12.1. Appendix 1

Participant collision with objects in the play area, during testing.

<u>Control Condition</u>

Participant	Collision Description
Participant 2	had several times where she collided with the boxes, mostly with the hands.
Participant 3	No collision with the boxes.
Participant 7	hit the boxes a few times.
C Participant 8	walked into boxes a few times
Participant 10	No collision with the boxes.
Participant 11	knocked into to the boxes a few times.
Participant 13	One collision with boxes.
Participant 15	Hit the boxes sometimes during this exploring. Never hit them when not exploring.
Participant 18	No collision with the boxes.
Participant 19	No collision with the boxes.
Participant 22	Walked into to the boxes several times, probably because he forgot about them.
Participant 23	No collision with the boxes.
Participant 26	A few collisions with the boxes happened. Commented that he forgot about the boxes while moving.
Participant 27	No collision with the boxes.
Participant 30	When dodging he walked or knocked into the boxes, almost every time.

Experimental Condition

Participant	Collision Description
Participant 1	No collision.
Participant 4	No collision. Used cover a lot to hide.
Participant 5	No collision.
Participant 6	No collision. Hides behind the boxes when necessary which was when there were too many arrows.
Participant 9	No collision. Hides behind the boxes from arrows, did this several times.
Participant 12	No collision. Stayed behind the boxes most of the time, hid behind them to focus on melee units rather than ranged units.
Participant 14	No collision. As it got more difficult the person got down on the knees to hide.
Participant 16	No collision.
Participant 17	No collision.
Participant 20	No collision
Participant 21	No collision. Used the boxes whenever necessary, as the amount of archers increased the participant hid behind the boxes more and more to stay safe.
Participant 24	No collision.
Participant 25	No collision.
Participant 28	No collision.
Participant 29	No collision. Getting down on the knees and hides between the boxes for cover.