ALTERNATIVE CONTROL METHODS IN VIRTUAL REALITY APPLICATION: ASSESSING USABILITY OF KEYBOARD AND MOUSE, AND TRADITIONAL GAME CONTROLLERS IN VIRTUAL REALITY

Lappeenranta–Lahti University of Technology LUT

Software Engineering, Master's Thesis

2021

Niko Mattila

Examiners: Associate Professor Jussi Kasurinen

Erno Vanhala, D.Sc.

ABSTRACT

Lappeenranta–Lahti University of Technology LUT

LUT School of Engineering Science

Software Engineering

Niko Mattila

Alternative control methods in virtual reality applications: Assessing usability of keyboard and mouse, and traditional game controllers in virtual reality

Master’s thesis

2021

72 pages, 10 figures and 2 appendices

Examiners: Associate Professor Jussi Kasurinen and Erno Vanhala, D.Sc.

Keywords: Virtual Reality, Keyboard and Mouse, Motion Controls, Game Controller, Oculus Quest 2, Unity, Game Controls, VRTK, Cybersickness

Virtual Reality has been making its appearance into consumer markets since its introduction to in 2016 in a form of Oculus Rift VR-system. The combination of being able to move one’s head in virtual environment with the help of an HMD and simulation of one’s hands to interact with objects with the accompanied motion controllers have open possibilities for vastly higher immersion in video games. However, the adoption rate has been exceedingly slow and has often been cited to being due to the notable cost of the hardware required, the lack of high-profile games, problems with cybersickness making prolonged gaming sessions difficult and issues with larger scale movement capabilities making gaming on the system undesirable. As such new solutions are needed to make the technology more desirable and to smooth out the curve of transitioning to the system. The emphasis on roomscale-VR has resulted in making most if not all pre-existing games not compatible with virtual reality without massive changes to the way the games are played. However, traditional control methods and styles have been successful and there might be season to implementing these in VR-games to make creating virtual reality experiences easier. And outside of issues with rotating one’s player character this control style appears to have some merit and people have expressed their interest towards bringing keyboards to VR. However, the issues with Covid-19 pandemic resulted in postponing parts of the research for later date, making it difficult to form definite results for now.

ABBREVIATIONS

VR	Virtual Reality
AR	Altered Reality
HMD	Head-Mounted Display
IR	Infrared
WMR	Windows Mixed Reality
FoV	Field of View
2D	Two-dimensional
3D	Three-dimensional
PC	Personal Computer
VRTK	Virtual Reality Toolkit
XR	Mixed Reality

Table of contents

1. Introduction

Computer Science and the factors related to it have advanced drastically over the previous half of a century and computers themselves became a common part of people's everyday lives especially in the last two decades with virtually every device in people's homes slowly becoming a miniature computer. This computation has also affected of how we see and acquire our own entertainment. Since gaming consoles arrived into peoples living rooms in the late 70s and 80s there has been a drive to provide more immersive, realistic, and visually appealing games to the consumers.

While advancements to computer graphics and game physics over the years have resulted to the games becoming more realistic looking and more immersive as a result, consumers were still by most accounts still just bystanders looking at the events and happenings in the game from the side-lines, even if they were in control of the game characters to a degree. This has resulted in some people starting to explore alternative methods to not only control the game characters but to also perceive the game world players interact in, some way to properly inject the player into the game world and make them feel like they are truly part of it.

One such path to achieving this is virtual reality (VR), where user can control not only the movement of their character and but also the camera that shows where they are looking with their bodily movements. (Gigante, 1993) The basic idea is that the player sees their visual feed from the perspective of the game characters eyes and as they move their head the game character also moves their head. This gives the player the perception that they are the character they are controlling.

VR has been available for consumers since 2016, when Oculus Rift VR-headset entered the consumer market but has been fairly slow to catch on. In late 2020 only around two percent of users of Steam, the largest digital game marketplace and library, owned any kind of VR-headset. This has largely been due to two factors, the price of the VR-headsets and a relative lack of high-profile videogames, with Elite: Dangerous, Half-Life: Alyx, Microsoft Flight Simulator and Star Wars Squadrons being the most notable ones, with three of them being flight simulators with fairly niche audience.

One of the limitations of VR is that it effectively requires the user to use head-mounted displays to provide the visual feed and camera movements for the game. This effectively limits the usage to first-person perspectives that allows to gamer to see as if they are the game character that they are controlling, eliminating its usefulness in games that use third person view, birds eye view or a side view of the game area. As result the most natural inclusion for VR in games falls to first-person shooters, first-person adventure games, and games where the player is seated the whole time like driving games and flight simulators. Additionally, the 3D-movement provided is known to cause nausea to most user after extended usage. (Groth, et. al., 2021)

The fact that VR-application use head-movements to perform camera movements instead of more typical control methods that use a computer mouse or an analogue stick on a game controller allows game cameras to use more complex controls. Additionally, motion controls have become a viable alternative to more traditional control devices as they allow people to simulate individual hand movements in games, bringing ahead even more immersion. This allows the control point to no longer be forced to be in the middle of the screen as the freed controls can be used to move the cursor itself independently from the camera itself. This makes the controls more complex when compared to before, effectively allowing more immersive and realistic control scheme than before. This however comes as a trade of as more complexity means more points of failure for the controls. Especially with motion controls that are notorious for suffering from inaccuracies when compared to more traditional and more static control methods. (Lugrin, et. al., 2013) As a result, creating functional control schemes becomes notably more difficult but more rewarding when they work correctly. However, how does one make more immersive controls that are also comfortable to the user and do not lose on the efficiency when compared to the earlier ones.

Is having more freedom in controls the optimal way to better gaming experience or should we consider limiting the controls on VR-systems in order to bring more pleasant experience to the gamers? That is the main question that is being answered in this paper. Where the ideal line for freedom of movement is when compared to the potential sacrifices in the quality of the gaming ability that the new controls scheme imposes on the player. As games have been slow to include VR-capabilities has gamers motivations to get a VR-headset been relatively low. Why get an expensive piece of technology if there is barely anything to use it on? It ultimately brings up the question of what can be done to make it easier for game developers to include VR-capabilities in their games.

1.1. Research Questions

The purpose of this thesis is to determine various issues that VR-applications can suffer from when it comes to the controlling methods and devices used to operate in a Virtual Reality world-space. Virtual Reality offers new types of possibilities both in entertainment and education purposes. While previously our visual feeds have mostly been limited to separate screens that provide us with information and that we could interact with using separate static input devices, VR allows us to use our bodies to control the feed that we see independently of the input devices thus granting us broader view of everything in that virtual world. This subsequently means that the lessons, methods, and rules that have been previously established for the way we, for example, move around in video games might no longer be applicable to this new paradigm. And even if they would be, there might now be new methods or systems that the new technology has opened for us and are better suited to the needs of the VR-applications.

Therefore, there is a need to take a look at what do we in actuality already have in our disposal and how does that function in this new workspace and consider what pitfalls could these rules have that would hinder the efficacy of VR-devices and the virtual worlds that they open up for us. And when these potential pitfalls have been discovered there would be a need to find potential solutions to these issues be it through altering the pre-existing rules and methods to fit into these new needs or by creating completely new rules that would solve these issues. But these new rules are bound to also have their own set of benefits and limitations that determine in which types of situations they are mostly beneficial and where they are less so.

This paper is focused on the ways we control ourselves in virtual worlds and what different controlling methods can be used in conjunction with VR mainly on the software side. By determining this we can form a basis on where subsequent research should be focused on and what aspects of Virtual Reality are in need of improvement before it can become a mainstream medium for delivering us entertainment be it form of games or otherwise.

Henceforth, the actual Research Questions for this work are as follows:

- How do control systems perform in VR-space and how does this differ from the more typical usage of control methods used in for example contemporary games?
- What controls schemes can be used with VR-systems and why?
- What control methods could provide more benefits in certain types of use cases?
- What compromises can and should be taken in ensure that the VR-experience not only provides adequate improvement to user immersion but also doesn't heavily detract from the user's speed and accuracy when using a VR-device or cause any other types of undesired side-effect?

1.2. Structure of content

This paper opens up by explaining the main research methodology used in this thesis, what is done, why and how.

After this it will delve into existing research to look at what VR properly is to explain its basics and looks at what potential issues have already been found in relation to the controlling methods in virtual space as well as other potential issues that could be related to the controls. Additionally, it will take a look at what potential solutions to the problems that are presented have been suggested and/or attempted.

Then the paper will go into detail on what type of program was built to perform the testing required by the research, how it was built, why it was built the way that it was and what compromises were made in order to perform the research in satisfactory manner.

Then the paper will go to onto parsing of the data that was received from the testing phase of the research. What it tells us about different styles of potential controlling schemes and what are reasonable to be used in which type of VR-application. Additionally, it will explain how the research was in actuality conducted, what steps were taken and what potential problems were run into while performing the testing itself.

Finally, the paper will form a summary on what the research was based around and what was discovered as well as form some discussion on what the results could be used on and what types on further research could or should be performed in the future to further develop the controlling schemes used in Virtual Reality.

1.3. Research Methods

Basis of this research is to create a fairly simple test program in VR that can be used to test various different controlling methods and schemes. These controlling methods would be split into two separate categories: physical and programmical control schemes.

Physical control schemes consist of the actual physical controller used to play the games. These would be keyboard and mouse combinations used in personal computers, that are considered fast and accurate albeit somewhat cumbersome, game console controllers that are easy to handle and use but suffer from speed and accuracy issues in first-person games and motion controllers that are commonly paired with VR-headsets to simulate hand movements to provide additional immersion. (Reski & Alissandrakis, 2020)

The programmical control schemes consist of methods and ways that the controls are coded into the game itself. This relates to how to camera movement is handled and how the player targets and selects things in the screen as well as how the movement of the player character is handled in the game space. In VR-applications the camera movement is handled with the head-mounted device that tracks players head movements and provides the visual feed to the player. Therefore, this does not need to and realistically also should not be changed between tests. However, the targeting handling and/or body movements handling can be changed so this is where our research will be focused on.

The program that is produced is then tested with different combination of control schemes and studied with speed, accuracy, intuitiveness, and immersion in mind. Also, other VR-related issues are taken into accord. Speed and accuracy are gained from by making the tester do certain tasks and observing how fast and well they are able to complete those tasks, while the rest is produced through a simple questionnaire that the tester answers after they have tested the program.

With these results we will try to come into a conclusion of what types of controlling methods are the most viable with VR in which situations and how these methods could potentially be further developed in the future and could some of them be potentially combined in some way to produce a more desirable result.

2. Basics of Virtual Reality

Since the stories have been told to one another there has been a desire from people to become part of them and experience them first-hand. While the rise of videogames achieved some of these desires, it has still left people experiencing the stories as bystander through a surrogate. As such there been a need to increase the feeling of involvement in videogames. Virtual Reality has provided a potential solution for this by allowing players to perceive and control their game characters through their own body movements.

2.1. History of Virtual Reality

The roots of VR back down into the 1960s with Morton Helligs Sensorama-device (Gigante, 1993) This device was a large cabinet that simulated a bike trip and fed the user realistic wide-angle video feed, sounds, smells and vibrations to provide the user as authentic of an experience as possible. The enormous size of this device made it cumbersome to use and ultimately VR-devices remained in medical and military usages in the following decades.

There were some attempts to introduce VR-systems by the large gaming companies in the 1990s, but most of these either failed to make it past the prototype phases or were changed so drastically in design that they barely had anything to do with VR anymore. The most infamous is the Virtualboy-console created by Nintendo that resembled modern VR-headsets but was not intended to be held on the head and was instead placed on a stand. To use it, player had to lean their head onto the display of the device, making it uncomfortable to use and the red-coloured vector graphics were not comfortable on the eyes. (Boyer, 2009) It was originally intended to be a VR-device, hence the VR-headset like design but due to insufficient processing power and rushed development it was ultimately unable handle proper Virtual Reality features.

The VR-devices intended for consumer market ultimately did not appear until 2016 when the Oculus Rift and HTC Vive VR-devices were released. In these the player was able to move the game characters viewpoint by moving their head with the head-tracking is handled with base-beacons that use infrared to keep track of where and in which position users head and by extension their HMD is in. (Arndt, et.al., 2018) These beacons are also able keep track of the motion controller positions used by motion controller that simulate the hand movements used in the VR-space. The beacon then sends the signal of these positions to the computer to be interpreted by the VR API.

2.2. Function of Virtual Reality

VR HMDs operate by tracking the movement and rotation of the HMD and delivering the information to the device the HMD is connected for that device to then supply it to the VR-application running on it. From here the application can adjust the camera position according to the information supplied to in. In return, application then supplies the camera feed to the HMD to be displayed through its inbuilt screens, allowing the user to see what the game character would see.

This tracking is handled by sending Infrared light that is then used by the tracking device to detect the changes in the movement of the HMD as well as the controllers. There are two methods to handle the sanding of infrared light as well as its detection. These are the outside-in and inside-out tracking methods. In outside-in tracking the base-station beacons send infrared light that is then detected by the sensors on the HMD and controller. The information from detected IR-lights is then sent back to the base-station to be interpreted to tell the system the position of the HMD and the controllers.

Inside-out tracking on the other hand uses cameras and trackers on the HMD to determine the changes in its position. Additionally, the controllers employ IR light that are sent out to be detected by the HMD to tell their position in relation to it. This helps reduce the reliance in the external trackers and makes the system more portable but can suffer from tracking issues for example with controllers if the IR light is unable to reach the HMD itself.

Actual movement controls in VR are typically handled by either game controllers such as Xbox controllers or motion controllers designed for the VR-experience. (Reski & Alissandrakis, 2020) With game controllers the controls resemble the typical non-VR control schemes, with the difference that the camera controls are done by the HMD. This would fix some of the issues related to somewhat slow camera controls that using controllers typically causes, but the fact that the pointer that normally would be at the center of the screen would now be separate from the camera would easily lead to losing track of the pointer leaving to unideal game experience.

Motion controllers on the other hand can simulate more natural hand movements creating fairly realistic feeling control over the game character. This can be seen by the player picking up virtual object with their hands, using them or even throwing them around the play field. These however also need to be tracked by the VR-system, typically through the lighthouse trackers or by the HMD itself if the system uses no lighthouse tracking. This adds additional complexity to the VR-environment that could potentially create more point of failure to the system.

VR systems however suffers from the problem where neither the HMD nor the motion controller are able to properly simulate player leg movements. This results in players being quite often unable to move from one place to one another fluently. Common workaround for this is to make the player teleport around in VR to be able to make them reach from one place to another. This however causes VR games to lose some of that immersion factor that would be gained from being able to move the upper body freely.

2.3. Differences between VR devices

Since the coming of the modern VR-systems there have been multiple distinct types of VR-devices that while mainly performing operations for the VR-experience have some differences to the methods used. While a lot the VR-systems, like the original Oculus Rift, HTC Vive and Valve Index, use lighthouse beacons to track the movement of the HMD and motion controllers, some of them forgo the beacon approach and instead use built-in cameras to handle the tracking of the HMD and controllers. The HMDs using this technology are the WMR devices and the Oculus Quest devices. These devices are often more intended for either stand-alone operation without a dedicated computer of console to handle the image rendering and information handling or they are intended for augmented reality applications that instead of using dedicated virtual worlds, add virtual elements into the real world that the person using the HMD can interact with.

In addition to these the HMDs offer several types of displays in themselves. Most of headsets offer somewhere between 100 and 130 degrees of field-of-view, with varying resolutions to the screens. The original Oculus Rift and HTC Vive included 1080x1200 resolution displays per eye. This had however the problem that the image seemed to have a “narrow net” on top of it. Since the users had their eyes extremely close to the screen, they were able to see the dark areas between the pixels. This phenomenon is called the “screen door effect.” (Hoffman, 2019) This issue has since been partially mitigated by offering higher resolution displays and modern displays offer at least either 1440x1440 or 1440x1600 resolution display per eye. Even if the problem still ultimately present, it has become less noticeable.

On top of this practically every VR-headset supports at least 90Hz refresh rate to offer sufficient smoothness to the image and not suffer from delays that would hinder the using experience, with most modern high-end HMDs also being capable of higher refresh rates for improved viewing experience.

Other difference between VR-devices are the motion controllers intended to be used with them. The controller used by HTC Vive is bulky wand-design that ends in a ring that is intended to send out infrared light to the tracking system to determine their location. They employ limited number of buttons with most buttons on the topside of the device intended for navigating menus, while the trigger button on the backside of the controllers serves as the operator button that is used to interact with the environment.

Oculus Touch controllers on the other hand offer smaller footprint than the Rift controller and offer more typical button layout compared to traditional controllers, offering three buttons of the top of the controller alongside an analogue stick and two triggers on the side and bottom of the controller. These offer larger array of possible actions to be performed on them, for example using one trigger to grab onto object and other to use them. On later versions of the controller the IR rings were placed on the topside of the controller to improve its compatibility with the inside-out tracking used in Oculus Rift S and Quest lines.

Windows Mixed Reality controllers similarly to the HMDs are built by different computer component manufacturers and offer different takes on the controller depending on the manufacturers creating them. As WMR devices use similar inside-out tracking as Oculus devices the controllers built for WMR are quite similar to the Oculus Touch with the IR ring being placed on top of the controller and using similar button layouts although the exact number of buttons differs from one controller to another.

Valve Index uses different take on VR motion controller compared to the others with instead of using an IR ring it uses a handguard style of design to handle IR tracking. The middle part of the controller is quite similar to Oculus touch and WMR controller with three buttons, an analogue stick, and a trigger on the back. It does forgo the second trigger used in Touch but also includes a touchpad on the top of controller next to the buttons. Additionally, the controller supports the tracking of individual fingers to allow players to perform varying gestures while using the controller. This in effect replaces the need for separate grabbing button on the controller.

Most other VR-devices are commonly designed to be used in other devises than PCs. These include Sonys PlaystationVR -devices and various different VR-solutions intended for the mobile market. As the focus is on PC VR-gaming and controls, these are not of particular interest for this paper even though most of the teaching learned can be transferred between devices.

2.4. Motion sickness

One major issue that VR has faced as long as it has existed is that it has caused quite a lot of people to suffer from motion sickness, especially after extended play sessions. (Munafo et. al., 2017) The exact reasons for why VR causes the motion are mostly individual for every person and are on top of that fairly poorly understood.

Some of the aspects that affect how susceptible people are to motion sickness in VR have to do with people's physiology and genetics. These for example are the gender and the race of the person using VR-headsets. (Munafo et. al., 2017) As an example, females are found to be more susceptible to motion sickness in VR-settings than males are. This is theorised to be due to hormonal differences between genders but the exact reasons for it are still ultimately unknown. Also, it has been noted that people of Asian-decent are more susceptible to VR related motion sickness.

Additionally, also the age of the person wearing the HMD affects both the likelihood as well as the severity of the motion sickness symptoms on people using VR. (Arns & Cerney, 2005) It has been noted that younger people are less likely to suffer from VR-sickness. Interestingly, this opposite to how motion sickness susceptibility typically relates to persons age as normally younger people are more susceptible to motion sickness.

Outside of the physical reasons related to one's own body there are also aspects related to the VR-devices themselves. This is because of how much people normally rely on their visual feedback to operate in the normally. When they wear the HMD on their heads their vision is limited to the inside of the HMD, including the display screens. As a result, they cannot perceive the actual world around them. Therefore, if the visual feed they receive from the display of the HMD does not match with the movements their body is doing, their brains get confused, resulting in motion sickness.

These are things that can be affected and controlled with the design and attributes of each VR-system in use. For example, the attributes of the display can affect how susceptible the person wearing the HMD is to VR-induced motion sickness. The first aspect of the displays that affect motion sickness is the framerate that the display operates in. This has to do with two separate attributes related to framerate. The first is how even the supplied framerate from the system to the HMD is. This is problem even with normal displays as uneven framerate results in stuttering video feed that is unpleasant to look at. With VR-headsets this also contributes to motion sickness. The second is whether the HMD operates on high enough framerate to begin with. (Zhang, 2020) Most VR HMDs operate with at least 90 frames per seconds for this very reason as otherwise the display would be unable to provide sufficient updates in the image for the eyes to process.

Another aspect that causes motion sickness is the field of view provided by the HMD. Most HMDs offer around 100 and 115 degrees of vision. However, while higher field of views would provide better vision to the users allowing for better execution of tasks, higher FoV also correlates to larger risk of experiencing motion sickness symptoms for the user. (Lin et. al., 2002) This is especially noticeable when the FoV grows higher than 130 degrees. This is because lower FoVs allow persons eyes to rest or focus on the sides of the HMD instead of the display itself, while higher FoVs result in the user only seeing the feed from the display resulting in confusion between what is real or not.

This is an issue that needs to be solved if VR wants to become mainstay in videogame market eventually.

3. Previous research

As controlling your player characters is what gives videogames their characteristic immersive feel, the methods to control those characters are placed into the limelight. And each game has its own way of handling this to provide the player as fluent and enjoyable experience as possible. This does not differ with VR-games in any way from the more traditional videogames. However, as controlling the camera is handled with the HMD held on the players head instead of the controlling device on the players hands this offers new challenges that need to be solved and circumvented to provide the new experience and make it feel both fluent and efficient.

These challenges can be divided into two intertwined sections that affect the experience in VR-space. The first one is the methods and functions that handles the controlling of the movement and actions and how they perform on the programmical ways within the game code itself. These are the ways that the game interprets the signals including the movement related data from the control devices and translates it into the actions happening within the game world itself. The second sections then are the physical devices used to insert the desired actions into the game. This includes how easy it is for people to hold these devices on their hands (or heads as is the case with HMDs) and use them to perform the desired actions, how easy it is to move from one action inducing set of button presses to another and what actions can be handled with what sets of buttons and controls.

As any method for interpreting the signals from controllers to perform the actions are useless without the controllers themselves and the controllers themselves are useless if the application is unable to receive the commands from them or fail to handle the actions in satisfactory ways the two parts of control methods are inseparable from one another. And changes to one side requires attention from the other side to make sure that the two are in perfect harmony with one another and they can provide the best possible service to the gaming experience itself. Therefore, one side cannot be completely ignored in favour of the other if an optimal gaming experience is desired to be offered to the player playing the game.

3.1. Control Devices

In previous research related to control devices used in gaming it has been noted that most people consider more natural control devices being more intuitive and contributing to the improved gaming experience. (Williams, 2014) These natural control devices could be defined as anything other than keyboard and mouse combination as well as console game controllers. The notable part about them is the fact that they are typically designed for singular purpose and singular type of game. For example, steering wheels and pedals create more realistic and natural feel when playing racing games. Likewise, flight sticks provide more natural feel in games where the player controls either an airplane or a spaceship. On the other hand, when the player plays a game that involves moving one's body in a three-dimensional space it would be more natural to use motion controls to input these controls to the system. And as VR-games are portrayed from the first-person view of a game avatar it would be safe to assume that these games could be ideal place to utilize motion controls to simulate the bodily movements of the game characters.

Motion controls however suffer from two particular issues that hinders its usefulness in VR-applications and games in general. And thus, making it less optimal in every situation. This is the fact that motion controls are less accurate than the control schemes provided by the more typical control devices. (Lugrin, et. al., 2013) This results in players actions being slower and more clunky than would be desired. A problem that is especially noticeable in fast-paced first-person shooters as well as other games that would require rapid accurate responses from the player.

This raises a question on itself on whether motion controls or more typical controls would be more desirable in VR-space to provide the optimal gaming experience. As most of the modern VR-games have been more involved to providing more immersive gaming experiences it is safe to say that the motion controls, or driving wheel for example in the driving games, have been able to provide better gaming experience together with the HMD controlled camera movements. But should VR-games also try to expand more towards the competitive side of gaming these motion controls might not be able to anymore cut it.

Additionally, there is notable difference between the traditional control methods. In first-person games, which are the ones of interest in terms of VR-games, keyboard and mouse combination is more accurate and faster compared to the game controllers. (Isokoski, 2007) This could point to the keyboard and mouse combinations potentially being useful to achieving higher performance even in VR-application. However, keyboards also suffer from a particular issue when used together with the VR HMD. The fact that the player is unable to see the keyboard which could potentially result in the player pressing the wrong buttons due to the close proximity and the vast number of button that the keyboards provide. Should this happen, it would result in loss of control and thus reduced immersion and/or performance and as such reducing enjoyment, which is undesirable.

Game controllers are of no stranger when it comes to VR-gaming. For example, the original Oculus Rift was originally designed to be used with the Xbox controller with the Oculus Touch motion controllers being sold as separate accessory. And while they are not as accurate or rapid as mouse or keyboard would be in controlling the player character, they are simpler to use and can omit the need for line of sight to the device making them something that should be always considered when creating VR-applications and the control methods built on them.

3.2. Other control devices

There has been some research towards the usefulness of various control devices in VR-environment.

One common theme in a lot of researches related to VR and more natural style of controls is that most people often consider them to provide more immersive and pleasing experience when compared to more traditional control schemes and most people that have tried out VR and the related technologies have indicated that they would be interested in using the technology in the future despite the perceived problems that the technologies were showed to have. (Lugrin, et. al., 2013 & Munafo et. al., 2017)

However, just because VR and more natural controls like motion controls are available to be used in applications, they do not automatically make the experience of using those applications better because they use them. This is well represented in research performed by Seibert and Shafer (2018) where they were testing the naturalness and enjoyment between keyboard and mouse interface and motion controls in both.

These tests were conducted using Razer Hydra, a third-party motion controller and not the first-party controllers intended to be used with VR-systems, but it does provide a generalised look into the comparison between the two control schemes. One of the notable results of this research was that a vast majority of people preferred traditional controls over the motion controls provided by Razer Hydra regardless of whether they were used in typical or VR gaming environments.

While much of this was attributed to the shortcoming of the Hydra itself, it does show that motion controls can fail to provide additional immersion if the system does not function sufficiently enough. This in return can raise the question of whether more traditional control schemes could provide better gaming experience in some situations as they could supplant the issues that motion controls for example suffer from.

Nonetheless, a lot of people that have not played videogames regularly and have tried out motion controls in VR environments have expressed their preference towards them compared to the more traditional control schemes. (Roupe, et. al., 2014) This is because the actions performed on the motion controllers resemble the actions performed in real life giving them a point of reference that they can relate to compared to the ultimately artificial controls that gaming has used until now. On the other hand, people that play games regularly have more of a preference towards these game controllers and control schemes that have been used outside of VR-space. This is also one of the reasons why the Wii-console performed extremely well with non-gaming people but ultimately failed to properly gain traction with the more traditional gaming crowd.

However, most natural control devices have limited use cases there they can be used. For example, steering wheels have little use outside of driving games and flight sticks have limited use outside of flight simulators and space simulators. They have however showed to being highly effective in providing better gaming experience in their limited use areas. (Williams, 2014) This is because they are able to simulate their use cases extremely well and as such, they are able to provide prominent levels of immersion and enjoyment to the user. Motion controls on the other hand do not have this benefit due to the issues related to their accuracies and the requirement for more complex move sets and attributes than they can provide currently.

One potential path for the motion controls to take could be to evolve them towards actually simulating the actions of the people using them. This is similar to how the steering wheels enhance the immersion in driving games and flight sticks in flying games. There is however some thought put into how more minute controls could be simulated in VR-space. (Sinclair et. al., 2018) In this research three different devices were presented for inducing distinct types of motion controls.

The first one is a “Claw,” that includes an outward pushing arm with a ring at the end of it that could simulate the movement of users’ index finger. To achieve this a set of servo motors and force sensors to detect when the index finger moves the ring around and controls the counterforce that would push the object back to neutral state. This could simulate grasping objects with one’s hand, touching object with outward pointed finger as well as simulate the pulling of a trigger.

Second device is an actuated wheel at the end controller arm that allows user to simulate both the feeling of touching objects with one fingertip as well as moving one’s finger around on the virtual surface. This, called the “Haptic Revolver” allows person to move their finger in sideways motion by rolling the wheel and the wheel is able to move up and down depending on the surface’s location relative to the actual hand location in the virtual space.

The last device offered is not as much of a new controlling method as a linking tool between multiple separate motion controller. This, called “Haptic Link” allows the controllers to be connected with one another making it possible to better simulate actions done with both hands simultaneously. Such actions would for example be using a bat in a baseball game, bow in archery or firing a two-handed weapon in VR-space. The purpose of the device is to synchronise the feedback from the virtual space to the controllers as well as to limit the movement area of the controller in conjunction to one another to simulate holding a solid piece of equipment with both hands.

The purpose of these devices is foremost to provide better feedback to the user about their actions in VR-space and while they do offer some solutions to more minute movements such as finger movements, they are limited to just one extended finger at a time making the use cases fairly specific.

To offer more accurate simulation about the minute movements of one’s hand, one realistically has been in need to be wearing haptic gloves that allow for accurate tracking of finger movements. (Sinclair et. al., 2018) These tools however are generally limited from the consumer market meaning that games have ignored the potential of finger moments as there has been no tangible way of tracking them anyway. This at least partially changed when the Valve Index was released in 2019 with its own motion controllers that allows the system to track the movement of each individual finger. This created new possibilities of VR-games, but as other VR-controllers have not really followed suit the feature has not really gained much attention outside of few games designed specifically for the Index, like Valve’s Half-Life:Alyx. However, the push towards controllerless controls by Oculus could provide a new avenue for this as one would no longer be limited by the controller provided in this case.

3.3. Eye-tracking

Outside of external controllers there is also research been made on using the HMD itself to handle the movements or at least some of it in VR-space. In particular a lot of research has been done on the field of eye-tracking and controlling virtual elements this way. (Qian & Teather, 2017) The main purpose of eye-tracking is to allow those with disabilities to experience and enjoy interacting with the world, be it virtual or the real one. This is especially true with video games and similar interactive media as they commonly rely on the person being able to input controls to the system using their hands with some kind of controller as a medium. And if one is unable to make use of the control schemes required, they are also unable to properly experience the media.

However, even though eye-tracking has mainly been focussed on creating possibilities for the disabled, it could also provide use for the common users and gamers in some situations. (Hou & Chen, 2020) The key issues however are that the existing eye-tracking technology and especially those that could be used in VR-devices is still in its infancy and suffers from notable inaccuracies and partial synchronization and calibration with the rest of the system. For example, in Hou & Chens research to they noted found out that using an external controller in VR offered better accuracy and throughput as well as was preferred by most test participants compared to the eye-tracking methods, especially when dealing with 2D-environments, but that the gap diminished notably when people moved to three-dimensional space even if in most cases it was still there and in terms of accuracy the eye-tracking occasionally offered better performance in 3D-space.

Also, a comparison between using head-tracking with the HMD, eye-tracking and mix of the two to control the scene in VR-environment has been performed by Qian & Teather (2017). They were surprised to find that just using head-tracking was not only the fastest but also the most accurate and the most pleasing control method to use in their tests, with eye-tracking offering the worst performance and the mixture of the two somewhere in between the two. This was presumed to be due to the accuracies of the eye-tracking that ultimately proved to be bad enough to hinder even the expected benefits of the technology. Despite this however, most test participant found that they for the most part liked the mixed control scheme as using it felt fairly natural for first person environments such as first-person shooters, while both eye-tracking and head-tracking were equally preferred for menus, showing that people saw potential in eye-tracking, but the adolescence of the technology was hindering it and preventing proper utilization still.

3.4. Comparison between Control Devices

In regards to the commonly used control methods in gaming one can split them into three categories: Mouse and keyboard controls on PCs and Macs, Game controllers on game consoles as well as PCs and motion controls commonly associated with the Wii-console as well as Virtual Reality gaming. However, while Motion Controls have been associated with VR due to their both involvement in creating more natural and immersive experience for people this does not automatically mean that they are the optimal method to feed inputs into the VR-application. Alongside immersion the accuracy and throughput of the controls also matter significantly.

In research performed by Ali & Cardona-Rivera (2020) Xbox Controllers were found to be significantly slower, less accurate and less engaging than HTC Vive controllers in VR environments. This was largely attributed to the limitations of the analogue sticks that are either slow and accurate or fast and inaccurate. Additionally, as the targeting reticule is not tied to the bounds of the screen it can be difficult to keep track of the reticule when it is outside of the view of the camera even with contextual clues of its location provided to the user.

Another controller research in VR is research comparing the performance of pointing tasks in VR and AR environments with conventional computer mouse, VR controller and a special made pointer pen. (Pham & Stuerzlinger, 2019) This experiment demonstrated the inaccuracies and slowness that the typical VR controllers suffer in comparison to the computer mouse and the purpose made pen pointer that was made for this experiment. Also, the research found that every control functioned better in VR space compared to the AR experience.

These researches could point towards the fact that the typical mouse and keyboard approach could potentially provide the best experience in VR experience outside of the fact that does not really contribute to the immersion factor provided by VR-systems. But just because something should offer high immersion does not mean that the control scheme offers the optimal experience especially in games that require high degree of accuracy and throughput of inputs.

3.5. Problem of seeing one's hands

One of the fundamental issues of with using VR HMDs is that their user is unable to see their hands and as a result cannot see what buttons they are pressing in actuality. This complicates the usage of keyboards that use and offer enormous number of different sets of control schemes in terms of key bindings.

A solution to this is to display a visualization of the keyboard in the virtual space to the user. (Bovet et. al., 2018) This is particularly useful when one needs to write text in VR to help the user know what buttons they are pressing and orient themselves accordingly. The main issue ends up being how to project the keyboard into the VR-space to make it useful. In Bovet’s research they used a Logitech BRIDGE to synchronize the keyboard and its button presses into the virtual environment. This however relies on physical hardware and compatible keyboards to display the keyboard and while it provided effective and mostly unintrusive experience it cannot be used in every situation due to the tie to hardware.

Another problem with virtual keyboards is the fact that they need to be displayed to make use of. A floating keyboard disconnected from the rest of the world affects immersion negatively and distracts from the rest of the experience. While this is not as notable of an issue with writing related tasks as the person can justify its existence with being a helpful tool to perform the writing on it would be a problem should one try to use it to control their avatar. And as one of the main tenets of VR is its immersion factor having objects not related to the VR-world in the user's view takes away from the experience. As such virtual keyboards are not particularly suitable for more generalized VR-experience at least in constant use.

Virtual keyboards are also usable in VR without needing a keyboard to input the keystrokes. However, in these cases the input needs to be provided in some other way. In non-VR cases controllers can be used as an input device through using a menu-style keyboard to select the desired symbols to be inputted. This could also be expanded to be used in VR-space. The crucial issues with this are just the relative slowness to the input as well as the problem of how one should change between the text input mode and the general movement mode as seamlessly as possible.

4. Creating the test program

To test the usability of various different possible control schemes in VR applications, there is a need to create a VR-application to test them with. To create this application the Unity game engine is used. Unity is one of the most popular game engines to create videogames and prototypes as well as is used in the creation of various real-life applications. Released in 2005, Unity quickly rose to the forefront of game development for its relative ease of use compared to other engines, its versatility in creating vast array of different types of games and support for large number of different platforms from PCs to Macs and Mobile devices.

For the purposes of this research Unity is notably relevant for its native support for virtual reality solutions especially for the Oculus devices that are being used as the tools to conduct this research. VR devices created by Oculus are among the most common and widespread VR systems created in the PC market. Notable is that Oculus devices and especially their Quest line up is not just intended for gaming purposes but also offers uses in productivity and AR environments. Therefore, they are on the forefront when the one wants to determine the usability of the VR devices in VR space. As such an Oculus Quest 2 is used to conduct this research to serve as a reasonable ground of reference for existing VR systems. This makes Unity with its innate support for Oculus systems ideal for this type of research.

Additionally, Unity has considerable number of tools that have been made for it to ease content creation and game development with the engine. This includes tools to streamline and assist in creation of VR-applications on the platform.

One of the notable sets to assist in VR-creation is the VRTK or Virtual Reality Tool Kit. This is a set of tools made by group called Extend Reality and have been used to make large array of different VR-games available through Steam, Oculus store and even the PS Store. The modern version of VRTK is comprised of independent tools and scripts called Tilia Packages from which the developer can choose the tools they need to help create their VR-software. However, due to the fact that these tools are dependent on the function the XR system in Unity itself the tools require the developer to use version 2019.4.19f1 of Unity for these tools to function properly as the XR system was changed in 2020.1.

The toolkit is comprised of tools to handle “CameraRigs” in Unity, including a simulator tool that allows the VR camera and controllers to be controlled even without actually using them for development purposes, pointers to determine what the player to pointing at, tools to handle input management in VR, methods to create interactable areas within the game world, as well as tools to handle character movement in VR and wrappers for SDKs used by the VR devices with the main ones being Oculus SDK that is being used in this research as well as the SteamVR/OpenVR SDK that is used with HTC and Valve devices and is needed to use the software through Steam. Additionally, there are tools to handle object collisions and their handling and tools to modify the visual feed for the player to provide him with information within the VR-space.

4.1. Test cases

To find out the optimal solutions for VR control schemes a few different test cases are needed to compare between the suggested solutions as well as to find out their effectiveness when compared to the control scenarios designed to represent the current situation with control schemes both in VR-space and to those outside of it. These scenarios test different aspects of VR movement and controls and provides answers to questions related to them.

The scenarios can be divided into a few different categories. Firstly, the general movement in VR space including moving within the VR-environment using different input methods and movement styles as well as controlling the camera with the HMD and how they relate to one another.

Second category includes interaction with different virtual objects. This consists of how picking up objects, using them and aiming functions on different control devices, as well as pressing or pushing game objects to perform various interactions and how they function in different control schemes.

‘The third category consist of performing various writing assignments within VR-space testing how easy and how fast it is to write text within the VR-space where one is unable to see the keyboard or uses different types of controllers in VR.

Finally, there is the last category where aspects of the previous categories are put together. This means making people to move around the VR-space doing various tasks from writing to pressing buttons and shooting targets. This is to determine how easily one can transition from one type of task to another with different schemes and what schemes functions the best in different functions simultaneously.

Additionally, the test cases are also split between the various test devices to control the scene. The first set is the keyboard and mouse combination:

- The targeting reticule/cursor is tied to the center of the display and moving one's head moves both the camera as well as the cursor.
- Moving the mouse moves the cursor within the display but cannot leave the confines of the display area.
- Cursor can leave the confines of the display.
- Additionally, movement can be handled with keyboard as normal or as a teleportation using the mouse to select the teleportation location.
- For the writing task:
    - Keyboard is used
    - Using mouse to click the button on a virtual keyboard.

For the game controller tests the same rules and options are used with the following exceptions:

- For general movement analogue sticks are used in addition to the teleportation.
- Writing is handled through menu selection style similar to how writing is handled in general with controllers.

Motion controllers on the other hand offer a broader set of different possible options:

- Controllers can be used to simulate hand movements thus picking up objects and moving them with their hands independently from the camera.
- Alternatively, when handling tools limiting the movement of motion controllers could be prudent to assist in creating more controlled environment.
    - For example, when handling two-handed weapons with the separate controls for each hand might not the best solution.
    - This for example, means that the reticule is constrained within the display and controller by tilting and spatial movements of the controller.

- The motion controllers could also be used as a general controller, simulating the experience that traditional controllers provide.
- Writing task can be handled using through pointing the at the virtual keyboard with the controller and selecting desired inputs that way.
    - Alternatively, one could use drumming techniques to input text.
        - Hitting keys on virtual keyboard with balls attached to the typical visualization of persons hands controlled by the motion controls

While performing these tests we evaluate them in two separate categories, objective performance metrics and subjective questionnaires done by the testers after completing the tests. In the performance metrics two aspects are largely looked at: the throughput of the tester for each control solution as well as the accuracy of the inputs for those solutions. For general movement this means that how fast one was able to navigate from one location to the other. For object usage task it corresponds to how fast were the tester able to perform their task and how many times they missed what they were supposed to do. In writing tasks these metrics are how long it look to write each passage and how many writing errors tester conducted along the way.

For the questionnaire there are questions about how they felt the solution was to use. How intuitive it is to use, how easy it is to use, what was its immersion factor and did the solution take the tester away from the experience, did they feel motion sickness symptoms and how bad they were? These are ultimately conducted for each set of testcase as well as the control methods used in that particular testcase.

4.2. Rationale

Most virtual worlds used in VR are created to convey a certain style of atmosphere that would fit the narrative of the game or environments. This is done to leverage on the increased immersion provided by the HMD-movement and motion controls offered by the VR system. However, as this research is interested in the movement aspects of the VR space and not on the narrative ones, there is no real need to create a complicated visual world for the people to traverse on. Therefore, the test program contains fairly basic geometry for the testers to traverse in.

Visually the main aspect that would be created for the program are ones that would help the tester to understand where they are needed to go, what they are needed to do and offer the necessary tools to perform these tasks successfully. This means creating clear indicators of which direction the tester is moving in and in which direction the things they are interacting with are located at. This is necessary as the HMD-reliant camera means that the point of interaction and the direction of the camera are no longer linked with one another.

Additionally, as the purpose is comparison between multiple control schemes that could be used in VR, there is a need to create fairly equal and comparable comparison between the schemes. This means creating an environment does not inherently favour one scheme over the others due to the design as creating an unequal environment would result in the test results being skewed and as such being not entirely valid or usable.

4.3. General Program

The basics of the test program lie in creating a VR program where the user or player is capable of moving around, interacting with objects, and using those objects to do some tasks in VR-space. To achieve this the program needs to be built from the ground up with the intent of testing these aspects.

However, as the program is intended to be built using Unity and VRTK, one needs to learn how these two in actuality function in terms of Virtual Reality and how do they function in conjunction of one another. As VRTK is built specifically to function with the Unity's own “Extended Reality” -system as well as to expand on it and make it simpler to use, there was not really questions about whether it would work but more on the idea of how it works for the purpose of this program. To achieve this, it was decided to start by creating a small testbed program with the purpose to test the various functions by themselves that could then be expanded to cover the intended testcases for the research should they be found both functional and desirable for creating the testcases.

To start the development of the program it was decided to use one of the VRTKs own test scenes as a basis for the testbed where the program would eventually take shape from. As such the “Bowling Game” -tutorial scene was chosen as the starting point for the program. This is because it helps set up the necessities of the VR environment while not adding much excess clutter that would be in the way of the rest of the program or would be considered plagiarising. The tutorial scene sets up the VR camera rig that allows the program to track the location and rotation of the VR system itself, the alias system that allow rest of the program to interface with the VR system without interfering with the system as well as setting up interactable objects and a way to interact with them as well as setting up buttons into the game scene that can be interacted with.

After setting these initial systems into place, the player is able to move their camera position and rotation in relation to the HMDs position in the play area determined to the Oculus Quest. Additionally, the player is able to pick up, move and release interactable objects as well as press spatial buttons placed into the scene that would perform predetermined tasks. In this case the only object that is interactable by the player for now is the bowling ball that the player can start rolling towards the bowling pins in the inside the bowling lane. The buttons are able to reset the position and the velocity of the bowling ball and the pins to allow the player to bowl without needing to reset the scene after every throw.

The next task was to enable movement of the player character in the VR space. Although tracking of the HMD does allow the player to move in the game, this is limited to roomscale VR-setting that allows the player to physically move around the room one is playing the game in. As such using it in a stationary style becomes difficult as would likely be necessary for people to use keyboard and mouse to control their characters. Additionally, even in roomscale the player will face the edge of the gaming area eventually as they move around. This means that additional methods of travelling need to be implemented to enable people to properly move around.

The first method to handle movement in VR is more typical method of using keyboard keys or analogue sticks of a controller to handle the movement. The main difference with this method when used in VR-space compared to more typical is that camera movement in mainly handled through the HMD rotation. This means that the method only needs to handle positional changes in most cases. However, depending on how the rotation of the player character is handled there might be a need to rotate the player separately. This is especially true in stationary VR-setups where the player is sitting down as the player is unlikely to turn around completely and as such limiting the tracking capabilities of the HMD.

To conduct this effect the play area projected by the HMD to the program is rotated to simulate the movement within the VR-space. This is because trying to move the player camera itself would interfere with the tracking of the HMD handled through the VRTK. The VRTK itself would have tools to handle this called “AxisMove.” However, those were not used in this test program due to the need to interface with multiple different control schemes and control devices.

The second traversing method added into the program is the teleporting method. In this method the player teleports from one position to another by pointing towards the intended teleportation direction and pressing a button to conduct the teleportation. This method is fairly commonly used in contemporary room scale VR-games due to it allowing players to move larger distances while still allowing them to enjoy the freedom of movement that VR-games that physically moving in room scale VR provides.

To determine the intended position for the teleportation a curved pointer is used that starts turning downwards towards the ground is typically used. This is to guarantee that the pointer ends up pointing to the floor instead of walls or the ceiling. Additionally, it allows the distance that the player could teleport in to be limited to a more reasonable distance as well as mostly preventing teleporting to higher elevation allowing the movement to feel more fluent and prevent players from breaking the boundaries of the game area.

VRTK offers its own methods for handling teleportation and curved pointers. As these do not conflict with the desired control schemes and are not dependant of the exact control devices used, they were used in the program to handle teleportation movement as an option for all control schemes. Additionally, the teleportation method does check whether there is something in between the player character and the intended teleportation location, preventing the player from passing through obstacles that one should not be able to pass through, which helps avoid players ending up in unintended locations in the test scene.

While this solved the problems related to creating movement schemes for the test there was still another major issue to solve before test cases could be created. This is the fact that the player does not have a hitbox or a tangible form. As a result, the player would not be affected by gravity and would be able to move through objects. The Unity itself contains a “character controller”-object that can be used to make a player character tangible. This would make the player affected by gravity and be able to track collisions with other objects within the scene and prevent clipping into walls and other kinematic objects with its built-in movement functions. However, this is intended to be used with traditional non-VR games and as such it fails to account for the needs of movement provided by VR-environment. Especially the room scale functions of VR fail to handle the collisions detected by HMD movement with character controller. This is because the VR movement is in relation to the play area and not the scene itself. As a result, to simulate movement that is not related to the movement of the HMD one needs to move the play area itself that then would move the player character along it. The issue is that moving objects directly on the scene ignores collisions which happens when the position of the player is updated in accordance with the HMD.

To solve this issue the collision handling needs to be added separately for the character. Fortunately, VRTK does offer its own player character container that can handle the collision handling while offering the other benefits of the Unity's own “character controller” like gravity tracking. Thus, it was used in the test prototype to handle player character physics. The collision detection does suffer from a rubber band effect near walls as it tries to place the player further away from the wall that can be presumed to be slightly distracting but it was decided to be an acceptable issue for this test that should not affect the test results.

Finally, before the test cases can be crated one last problem needs to be solved. That is the way for players to interact with the environment outside of the motion controls that were setup in the initial VR-setup. The usual way of handling this is using a type of cursor that is locked to the middle of the screen. The reason this works in non-VR first person games is because the mouse or controller is used to control the camera, meaning that the player receives rabid feedback to the movement produced by one's hands. Additionally, these devices are also used to input the commands themselves, so everything is concentrated in a single place. However, as the camera movement is relegated to the HMD this would now result in the inputs being fed through a separate device potentially causing a disconnect in the way the player would input their commands from the controls themselves.

The inclusion of an HMD does however offer a potential opportunity of untying the cursor from the middle of the screen. This would allow the player to use their input device to move the cursor around the screen as they would in desktop environment. This would allow more finetuned control over the point of interaction to the player and bring back some of the feedback that the was lost with the inclusion of the HMD camera controls. On the downside, the player will likely have problems keeping track of the cursors position, especially if the cursor is not noticeable enough and if they have no object that they are controlling at any given time.

As the VRTK-tools are concentrated on creating a competent VR experience using the motion controllers associated normally with them, they do not offer tools for handling cursor placements with controllers or a mouse. Therefore, a system to provide this functionality had to be built from scratch using tools offered by Unity itself. The cursor itself was created through a hit scan check that originates a small distance behind the camera and interjects through a plane at the cameras position. This interjection point is the position where the cursor is in relation to the camera allowing the object touching the hit scan ray passing through the cursor position to be interacted with.

Three separate setting were created to test what type of a cursor could offer the most optimal gaming experience. In the first one the cursor is locked to the middle of the camera simulating a situation that also appears in traditional first-person gaming. This also serves as a control case for the other settings to be compared to. The second setting is where the cursor is allowed to move within the confines of the screen while following the direction that the camera is pointing towards. This results in the player always knowing that the cursor is pointing to somewhere in the direction they are looking at while also offering more control options to the player. The third setting is like the second one except the cursor direction is tied to the direction of the player character instead of the camera direction. This would allow the player to point their cursor forward from their character while also allowing the player to turn their head and look around without causing the objects they want to interact with being in a single position.

A fourth setting that was considered but ultimately was scrapped was the ability to move the cursor completely independently. The issue that was perceived with this was that it would have resulted in the cursor direction, camera direction and the character direction being disconnected from one another, making controlling the player character exceedingly cumbersome as well as keeping track of the different directions difficult. This is however something that could be considered again in further research.

4.4. Movement Test

With the movement schemes, camera controls and interaction pointer established arose the need the start creating the tests to establish if more traditional movement schemes could be still used in VR environment despite the different challenges and constraints caused by the shift in the way the game is perceived. The main determining factor in this is whether the controls function in general at satisfactory levels in VR environment while the player is moving about and performing whatever the task is that they are attempting to perform in the VR.

To test the function of the general movement faculties there technically might not be a particular need for separate test suite as the testers could conduct this alongside completing other test scenarios. However, this would result in vastly varying test depth and therefore results due to the testers performing quite different movement patterns and methods while moving around completely freely with no particular goals related to testing the movement itself. The results and feedback offered by the testers would likely become somewhat inconsistent and difficult to compare between each and every tester. The results would not be invalid, but they might hide some more general benefits and negatives of each control scheme behind the more personalised playstyles that the testers might develop.

To avoid this issue from arising when retrieving the feedback from the testers a straightforward and fairly simple test was developed into the prototype that forces the tester to move around and perform simple tasks alongside it. To achieve this a simple maze task was developed that the player would need to navigate in order to test the different control schemes. This maze was not created to be particularly difficult to prevent the testers from getting lost in their first attempts at it and thus affecting their feedback as result due to aspects not related to the thing being tested at the time.

Inside this maze the player would need to find three buttons that they would need to press in order to complete the test and move onto the next. As they press the buttons inside the maze a light will light up above each button to indicate that the button has been pressed. Additionally, there are guiding arrows inside the maze to help the player find the buttons they have not pressed yet. These would be then changed accordingly when the buttons are pressed leading the player to the next one.

Additionally, there is an aspect of requiring the player to hold an object to properly navigate the test. This allows rudimentary test to the handling of an object and while moving around the scene and how holding an object and moving around interact with one another with different control schemes. To achieve this the maze was made extremely dark to require the player to use a flashlight to navigate it. Therefore, the player is required to pick up a flashlight at the entrance to the maze and use the beam of light provided by it to illuminate their path and see the guiding arrows that show the player the path towards the buttons.

Also, as each of the tests are located on the same scene with some distance between them, this allows for additional testing of the movement schemes while the player moves between the test areas for different tests. As the tester does not need to hold on into an object i.e., the flashlight from the maze, deal with limited spaces or the darkness. It allows them to gain a perspective from the movement scheme that they cannot by just navigating the maze due to the varying attributes of the environment.

This test is affected by each of the testing settings from control device to movement style and cursor control style. Especially the one setting that solely affects this this test is the difference between teleportation and free movement as the other test do not require the tester to move around other than to move from one test site to another. This means that to generate sufficient information a lot of different permutations of the settings needs to be gone through. Therefore, the test case is left fairly generalised, and it does not account for specific action and detail. This can be potentially accounted for in a future, narrower research that looks into the subject in a more detail.

One major aspect that was left out of this research is the verticality factor of the test scene. This decision was made because the verticality is not entirely necessary to establish whether one of the control schemes functions even in the first place. Additionally, the inclusion of verticality could cause the testers to be affected by vertigo and therefore produce nausea and cyber sickness that would cause the schemes to be perceived worse than what they would otherwise. This aspect should be included in future research as verticality has been a part of three-dimensional games since the early 90s and the lack of it causes lessened immersion.

4.5. Aiming Test

While the movement test contains most of the aspects related to the moving players position and rotation around the scene it fails to comprehensively stress the cursor movement aspect of the control schemes. To this purpose a second test case needs to be created to find which to the potential methods of simulating one's hand movements and aiming towards different objects in actuality functions. As most first-person games are various types of shooters it warrants to test how the system is able to handle handling a weapon in VR environment.

To conduct this test, it was decided that the testers would be required to shoot a number of targets with the emphasis being on how long it would take the testers to hit every target as well as see how many times they would miss their target in order to get some objective number related to the accuracy of each control scheme. To do this a shooting range was created into the Unity scene used by the program. This shooting range was initially decided to contain 10 targets for the testers to shoot at with each of them offering different challenge for the tester to face. Some of the target would be closer to the player, some farther away and some in an angle to make them harder to shoot while standing at the middle of the opening at the front of the range. Additionally, there were some obstacles added into the range, in form of pillars and walls, to make aiming more difficult. Finally, of the targets were made to move around the range in varying speeds to offer further challenge to the testers.

However, ultimately it was decided that the test should also include an element of the turning around to further test the elements of the control schemes offered to the players. To facilitate this, initially additional 5 targets were placed around the player that they would also need to shoot in order finish the test. Most of these were placed in an alcove behind the player to force the player to turn a complete 180 degrees. Additionally, some of these targets were also not only made to move around but also follow more erratic patterns or move notable faster than the ones in the range itself. However, some of these targets were found to not really serve significant additional purpose compared to compared to the 10 initial targets within the range so three of these targets were ultimately cut, but the rest were made to be shot multiple times to complete the test. This left a total of 12 target with two of them needed to be hit multiple times.

The shooting is handled by an object resembling a pistol that the players would need to pick up in front of the range. While being held by the player the object would send a hit scan ray outwards from the top of the object representing the barrel of the weapon when they pressed an action button to denote that they want to shoot at a target. To assist the player at aiming the weapon a bright dot appears in front of the players weapon to show where the imaginary bullet would be headed towards. This is to simulate crosshairs that most shooting games use to show where to weapon is pointed at.

The test begins when the player hits the first target. When this happens, a sound is played and the target the player shot would disappear, and the timer would start to advance. At the side of the test area there is two plates showing the time elapsed since the start of the test as well as the number of shots the player has taken. These will keep running until all of the targets are shot and have disappeared. It should be noted that the targets that would need to hit multiple times would also play a sound to denote that they have been hit but they would not disappear unless they have been hit a sufficient number of times.

Some aspects that were considered for the shooting test but were ultimately not included were the inclusion of targets that would only be hittable for a short window before they disappear for good as well as the ability to disable the laser pointer from the weapon to make shooting it "more realistic." The first was not done mainly because it causes a disproportionate difficulty difference between those doing the task the first time without pre-existing knowledge of the task and those who know what is going to happen. This would cause some of the test results to be skewed due to this knowledge as people would be testing multiple control schemes to have them determine their opinion between them. The second was left undone due to the fact that the weapons middle point is not directly on the middle of the screen due to the weapon obstructing the screen. This causes aiming to become exceedingly difficult on mouse and controller setups due to their reliance on the cursor movement for aiming.

4.6. Writing Test

Not all problems with VR software are related to the moving the player character around the playfield. The aiming test detailed previously describes a way to interact with the playfield in further detail. This type of test is especially notable in finding solutions for moving larger objects around or interacting with them in a basic detail. However, it misses a lot of more specific tasks that would require more than just moving one's hands around. Additionally, lot of the actions one would perform in a traditional game environment do not have specific singularly determined methods to conduct them. Some of these come down to the technology not being mature enough to handle these tasks, while some like writing text do not have a standardized way of handling it.

Writing has been vital part of videogames for a long time. In the early days of gaming, it was a way to communicate the actions the player wanted to do to the game especially when the games did not have visual components to convey the potential actions to the player. In later days it has become a method of communication between players and the prevalence of the internet has increased its importance even further. VR games however do not have a central method of handling writing text. Notable reason for this is the lack of a keyboard that is normally used to write text in PC environment as motion controllers are typically used as control devices in VR environment.

The writing test offers three straightforward methods of inducing text into VR games that could theoretically be easily developed further should they show promise in testing. The first two methods use virtual keyboards placed into the game environment that can be interacted with. The first method consists of pointing the interacting pointer towards a keyboard and clicking the keys on it through a hit scan ray checks that determine which key the player pressed. The second method on the other hand uses interactable game objects in a shape of a mallet to hit these keys. The final method input the text through a physical keyboard like is typically used in contemporary non-VR games.

The mallet method consists of using an elongated virtual object hit button or keys on a virtual keyboard. This can be helpful for writing applications. The keys on a virtual keyboard need to be fairly large to prevent the user from hitting incorrect keys. This results in the keyboard itself requiring notable amount of space in the scene potentially putting the some of the keys far away from the user when the keyboard is laid out horizontally. Using a mallet gives the user extended reach to reach these further away keys. Additionally, only the tip of the mallet is able to hit these keys preventing other objects or even the shafts of the mallets from hitting the keys on a keyboard unintentionally.

The hit scan method on the other hand allows the user to just point at the keys and select which keys he wants to press. This is similar to how virtual keyboard have been used with a mouse in PC-environments. The main issue is that if one tries to use the keyboard this way from an angle, it becomes easier to press wrong keys as even minute adjustment could change the point the person is pointing at quite notably. As such the method is most suitable for being used when the keys are vertically and in front of the player.

The virtual keyboard used in the first two methods is from a downloadable Unity asset called "VR Keys." It offers a tangible keyboard that can be spawned into the VR scene for text writing, supporting multiple keyboard layouts from the typical English QWERTY-layout to DVORAK and the French AZERTY-layouts. It also offers capitalization and limited special character support as well a text field that houses the text written with it, as well as buttons for sending messages forward, clearing the text field and closing the keyboard down when it is not needed. As such it made an easy and convenient addition into the test scene to conduct the testing with.

Notable aspect of the "VR Keys" is that it was created to be used with the drumming writing method in mind and as such includes its own mallets to hit the keys with. This however also meant that the tool had to be slightly modified to also function with the pointer clicking method also used in the writing test mainly to make it detect the hit scan rays sent from the control pointer.

Additionally, to conduct the writing testing a few additional additions had to be added to the system to make it able to keep track of the testing progress. These additions include two information tables similar to the ones used in the aiming test. The tables are intended to track the time it takes for the tester to conduct the test with each of the methods as well as the number of times they make a mistake and must press the backspace key on the keyboard. To start the test a button was added to open the keyboard if it was closed and start the timer and track the written text.

For the physical keyboard test there is no need for the virtual keyboard so only a text field to contain the written text was provided. Otherwise, the setup is similar to the other two with information screens for time and mistakes and a button to start the test. The other main difference to the other methods is that when the keyboard is inputting text to the text field the player character stops receiving movement commands to prevent accidental movement during the test from pressing keys on the physical keyboard. Control is regained by pressing the button that started the test again to decouple the player from the writing mode.

The test itself consist of three predetermined phrases of varying difficulty that the tester needs to input into the text field and then press the "submit" button or hit the Enter-key in the physical keyboard test. If the phrase is written correctly a corresponding light lights up green to signify that the phrase was correct. When all three phrases are inputted correctly, the test concludes.

All of these methods are expected to be of different efficacy in different situations. The mallet test is expected to be more suitable to motion controllers as swinging the mallet around with other controls is likely to be limited, while the physical keyboard is likely more suitable when one is using the keyboard anyway as their control device.

5. Results

While creating a prototype program is necessary to be able test various different control schemes, they are incapable of testing how they function in practice. Furthermore, as most things in life, every person has different preferences on to what they believe feel more natural, responsive, or intuitive. Therefore, it becomes impossible to confidently determine whether something work or not without having multiple people testing it and giving their own personal feedback on the matter.

To find out whether which of the control schemes laid out in the prototype are of satisfactory performance, a notable subset of people needs to test the software in a controlled environment to make sure that the testers are conducting the testing in a correct manner, without cheating the results and that they do not miss some aspects of the software and cause themselves to gain misinformed opinion about the subject they are testing. This means conducting the testing in a lab or test site where knowledgeable personnel are able to monitor and assist the testers in their task to try out the software.

There are some things to take into account when starting to conduct testing on prototype software with a random subset of people. The first one is that testing takes time, and the more thorough the testing is the more time it takes to conclude. Therefore, there is a need for a balance in the testing methodology that allows as sufficient testing as possible while also allowing it to be concluded in a reasonable about of time. The second issue that might arise is that the people who are intended to be testing the software might be arriving to test it with preconceived notions about the subject matter. This can cause the results that the people might give to be skewed one way or another due to their personal biases. This can be especially notable in cases where people are required to compare two or more objects to one another resulting in people being more favourable to one in favour of others even if they did not really find it actually better than the other.

5.1. Setting the questionnaires

The testing result on this research a formed by two distinct parts: The objective results and the subjective results. The objective results in this case are the performance metrics that result from people conducting their tests. These are the amount of time that it takes for people to complete each test as well as the number of mistakes they conduct while completing the test case. These numbers are available from the aiming and writing test to help determine which schemes cope better in which situations. They are not collected from movement test as variation there is more likely to be caused by general differences in playstyles, more “freeflowing” test parameters and from people testing things in varying ways even if they are given the same task to complete.

The subjective results are collected from the testers after they have finished their test suites and have seemingly evaluated the system to determine their opinions about the subject. To collect these results the testers are asked to fill out a questionnaire that collects feeling and opinions about what they just had tested. The questions inside the questionnaire are in a form of grading various aspects of the system offered, for example asking how would the tester rate their experience and whether they would use it in the future if offered and whether they would recommend the system to other people. Additionally, the questionnaire also includes the possibility to give open ended answers to allow them to elaborate on the reasons of why they gave a certain grade as well as offer notes about things that are not asked in the questionnaire, but they noticed during the duration of the test.

The questions asked in this particular questionnaire looking for answers to the combability of various control schemes VR-systems are listed in Appendix A consisting of questions about their enjoyment of particular control method, whether they would use it, if possible, would they recommend the method further and whether they suffered from cybersickness.

In addition to the questionnaire that the people testing the prototype are required to fill out, another questionnaire was also created. This questionnaire is aimed towards people that already own VR-systems or use them in regular basis regardless of whether they are going to be testing the prototype or not, spread through the internet for people to answer to regardless of who they are or where they are. The purpose of this questionnaire is to gauge the pre-existing opinions about the VR-systems, people's satisfaction with them, what improvements people would like to see in current VR-devices or VR-software. Additionally, it also gauges whether people would be interested in alternative control methods to the current motion controllers and whether they would recommend VR devices to other people or not and what prevents them from recommending them for others.

The questions asked in the internet questionnaire looking for answers people's attitudes towards Virtual Reality are listed in Appendix B.

The results from these sources can then be used to determine if there is reason to conduct further research on these control methods and whether it would be reasonable to consider alternative control systems to be used in VR-applications and games in the future to make it easier for people to be willing to commit to owning the system than what the current situation is.

5.2. Results from the internet questionnaire

To find out whether the research planned in this thesis has actual reasoning behind it that could benefit the actual real-world applications of VR an internet survey was conducted to find out what people think about the currently existing VR-systems and the software run on them and ask what are the aspects that people would like to see improved on these devices or their implementation. The questions for the survey were explained in previous section with the hope that they would shed some light into the current state of VR.

The survey was spread around through the LUT University social media channels as well as directly to the computer science students through advertisement on classes as well as spread to the Computer Science department staff through the Thesis supervisor. Additionally, some advertisements were made toward the students and staff at Aalto University as well through contact there. In the end the survey received a total of 21 answers over the period of a few weeks. As the survey was focused on receiving answers from people who already own a VR-device or have used one in somewhat regularity through other means and the technology is still somewhat niche this number of recipients is found to be sufficient in order to find answers on the topic.

The first question was centred around which VR-devices people had used or owned. From the 21 surveyed people a total of 14 (66.67%) were using various versions of Oculus devices with a fairly even split between the older Oculus Rift devices and the newer standalone Quest devices, 6 and 7 respectively with the last person using Oculus Go. Of the remaining 7 people 5 were using PlayStation VR, with the last two using HTC Vive and a WMR-device. (Figure 1) The small number of the more gaming oriented HTC Vive devices and lack of Valve Index devices is notable as a lot of the current use cases for VR-devices for consumers currently are various gaming application. This likely could be attributed to the relatively higher cost of these devices.

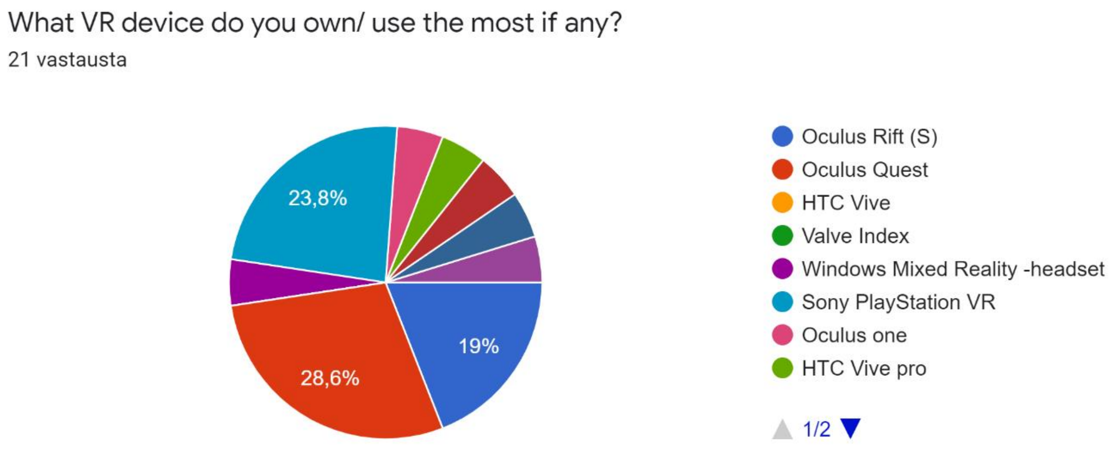

Figure 1. VR-device ownership. Small sections are 4.8%

When asked how often people use their devices 38% of the people said that they were using their VR devices weekly, with 29% saying that they only used device when something new that interested them, like a game, was released and another 29% using the device even less often signifying that their devices were left on the shelves and cupboards as they failed to find a clever use for them. (Figure 2) This denotes the relatively slow growth rate of VR when even those that have gotten their hands on one are finding trouble in finding use for them and why it should be pivotal to find ways for people to be able to find use for them. It should also be noted that of the people that use VR somewhat often nearly all of them are using various versions of Oculus Quest. Whether this is because of the standalone nature of the devices needing not to be tied to another machine or some other aspect related to their design is unclear.

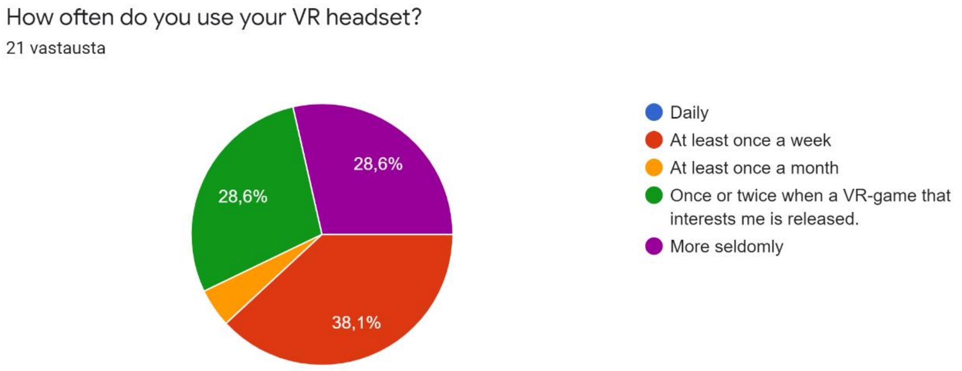

Figure 2. Use ratio of VR-devices. Small section is 4.8%

When asked about the satisfaction about their devices the people in the survey relayed slight lean towards being satisfied with their devices. On a scale from 1 to 7 the average score came to 4.71 with 7 being extremely satisfied and 1 being extremely unsatisfied. Of the 21 answers only three people expressed being unsatisfied with their devices by giving a score of 3 or less, while the plurality of answers (8) landed on the score of 5 with five people giving a neutral answer of 4. (Figure 3) The main note about the people with negative feeling about the devices is that they largely complained about problems with the controllers as well as a lack of usable software of the device. This has an interesting effect that even though most people do not use their devices very often they do generally like what they offer.

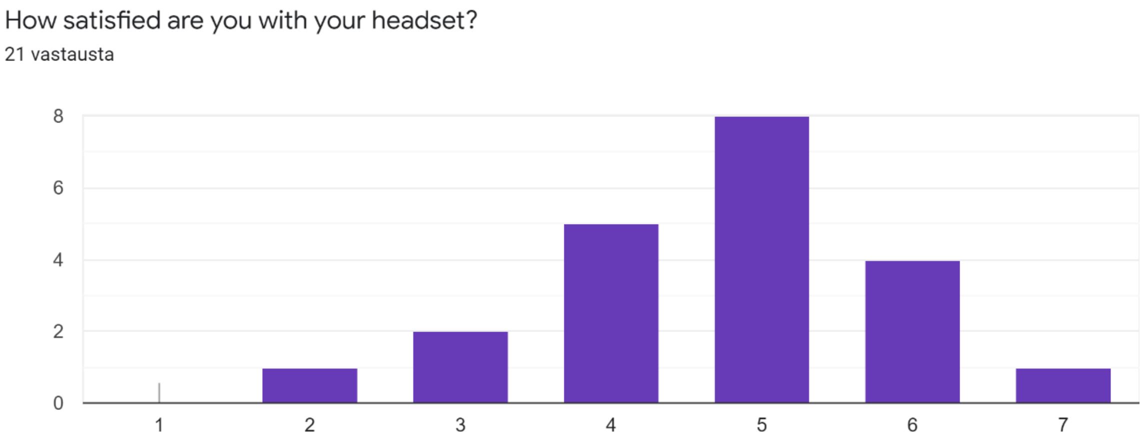

Figure 3. VR-system satisfaction.

When asked about what part of the headsets themselves people would like to see improved most people (52%) voiced their desire to see the resolution offered by the screens to be improved. (Figure 4) This ultimately makes sense as due to the screen being notably close to a person's eyes it is liable to be affected by the Screen-door effect, where the person is able to see the dark areas between pixels. Also, text readability, image clarity, and cybersickness were raided as issues on this topic when asked to deliberate further. Notable is that most of the people mentioning this were using the older Oculus Rift devices or PlayStation VR, but also some of the newer Quest users also noted the need for better resolution hinting that while the older devices in particular are affected by clarity issues the newer devices have not fixed the issue yet entirely. The second largest point of improvement was noted to be on the controllers (14%) which was also notable with people who were less satisfied with their devices. Other points of improvement were noted towards the tracking capabilities, especially in regard to leaning these devices towards AR uses. Also, the general screen quality and the weight of the device were brought up by different people.

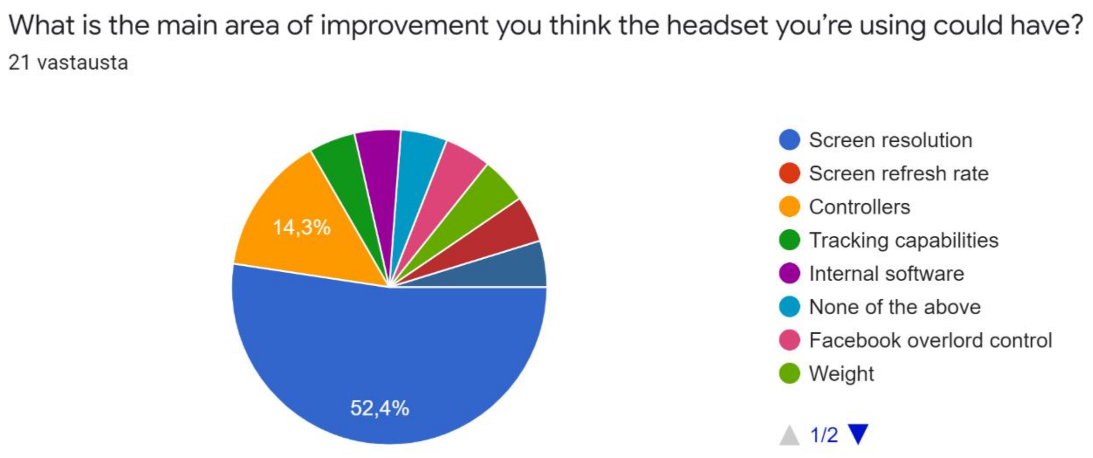

Figure 4. Desired hardware improvements. Small sections are 4.8%

When asked about the satisfaction towards the software used on the VR devices the reception was noted to having been slightly less favourable than with the satisfaction towards the devices themselves. The average of scores was 4.62, a .09 drop from what the devices scored. (Figure 5) The most notable aspect is that plurality of people (8) gave a score of 4, with 5 receiving six votes and 6 receiving 5 votes. However only two people gave a score of 3 or less. This leads to believe that the current software in VR might be lacking in any particular attributes that would make some people excited to use them, however people have no particular issues with them either.

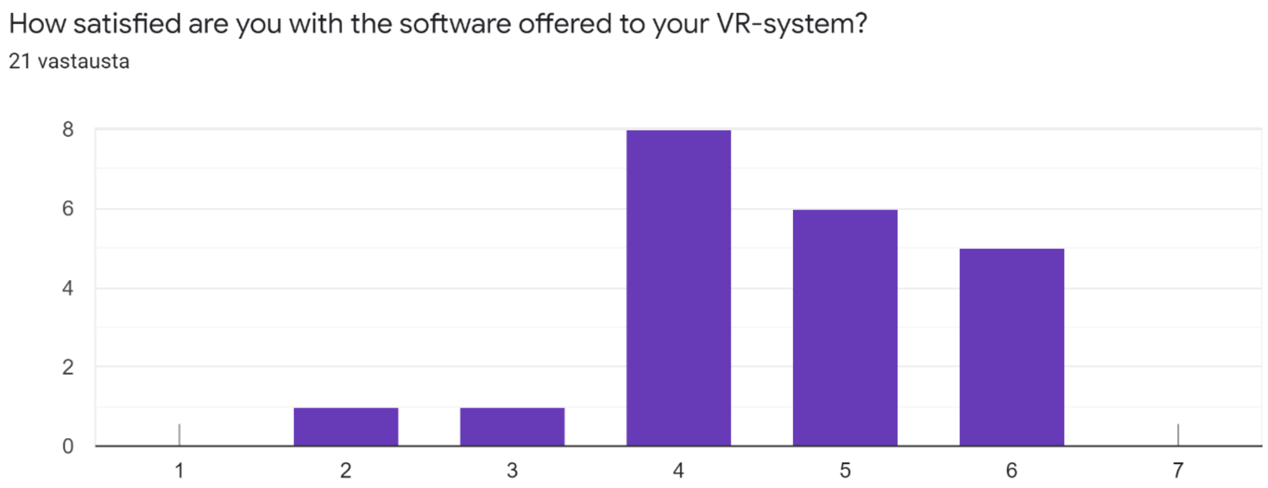

Figure 5. Satisfaction of VR-software

When asked about the aspect of improvement people would like to see on the VR-software the answers were notably more varied than what the device question constituted of. The most desired improvements constituted of Menu navigation improvements (19%), Controller Accuracy from the software side (14%) and Movement related improvements (10%). Also, UI and text writing were noted as issues. (Figure 6) This leads credence to the statement that the VR imposes notably different challenges and needs for how everything is presented to the user with the person being able to move their head and controllers around freely independent from one another. As a result, the combining of UI elements with the motion controllers and the game world leaves something to be desired for, pointing that either this aspect in particular needs to be improved or a new method to handling these needs to be found. Additionally, the movement systems are also in need of improving to make VR more desirable. These parts are pointing towards that the ideas of taking a step back and trying to go back towards what is proved to be functional before might be reasonable at least temporarily as the issues with current methods are ironed out. It should be noted however that a third of the surveyed people did not have anything particular to add to this topic.

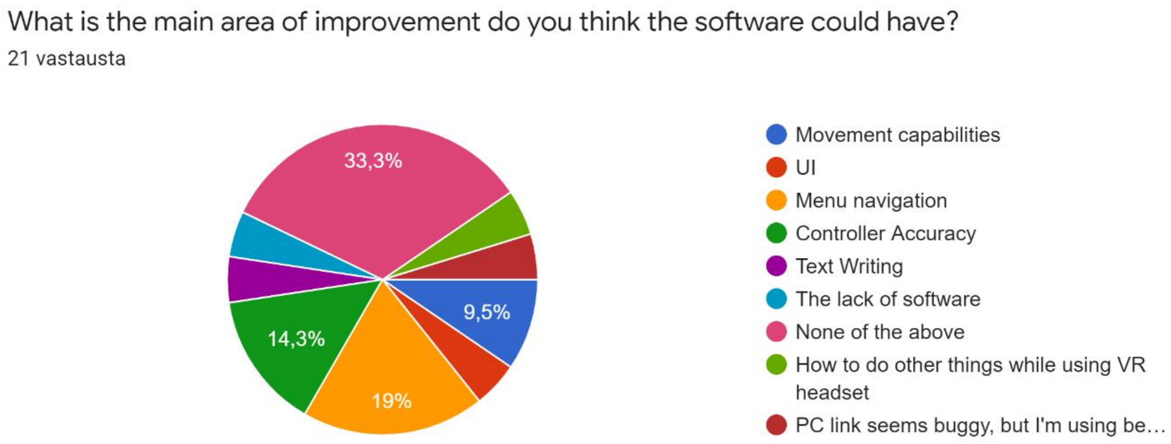

Figure 6. Desired improvements for VR-software. Small sections are 4.8%

On the topic of taking a step back towards the proved control methods, people were asked whether they were interested in using keyboard and mouse combination with VR software, provided that is functional and can be used. This question split people quite notably with people giving answers from all areas of the answer spectrum. However, most people ended up being open to the idea of using a traditional PC controls with the average of 4.4 in spectrum from 1 to 7, with 4 and 5 receiving the most votes with five each, respectively. (Figure 7) Notable is that both extremes received votes and two for 1 and three for 7. While this highlights to divisiveness of the issue it also brings up question about what would cause people to answer this differently. The was no correlation found between this answer and any of the others, people using any device gave vastly varying answers as well as usage did not have real effect to the answers. Additionally, taking the question to the other direction, it was also asked whether people would be interested in using VR without separate motion controllers and instead use their body movement directly to control the experience. This could be seen as an advancement over the current motion controllers, and something often displayed in sci-fi films and literature. This received extremely high interest among the surveyed people, with an average of 5.9 and no answer being lower than 4. (Figure 8) This could be interpreted as the ideal that people imagine VR being capable of eventually. This however is still far away from being a reality on consumer devices, so this avenue is largely closed for now.

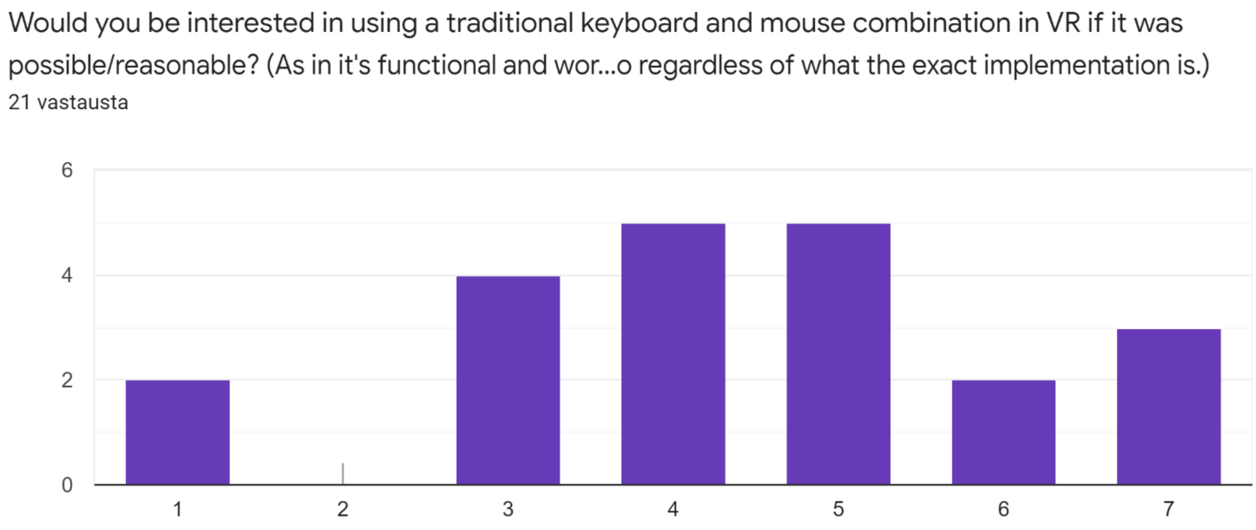

Figure 7. Interest in usage of keyboard and mouse in VR.

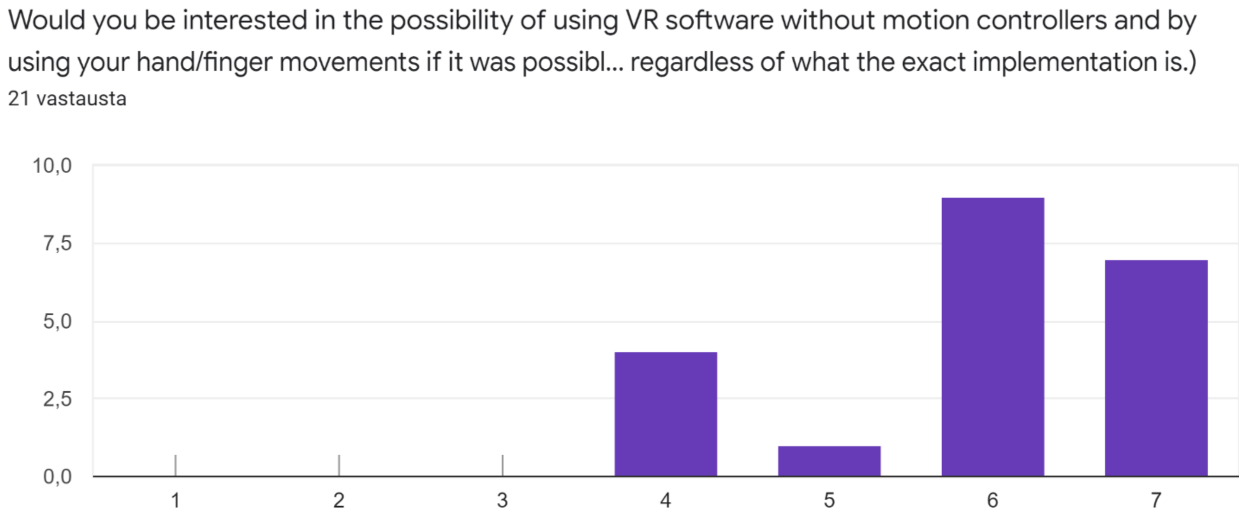

Figure 8. Interest in usage of motion control without controllers.

Finally, people were asked how likely they were to recommend VR to other people. The idea of recommending VR to other received a warm welcome with an average of 5.0, no answers lower than 3 and 6 receiving the most answers (7). (Figure 9) When asked what aspect would be preventing them from recommending the devices, the most common concerns were raised about the cybersickness caused by the devices (33%), the somewhat steep cost of the devices (21%) as well as the general immaturity of the technology (21%). (Figure 10) These are in previous articles about contemporary VR raised as the most relevant issues as well that the current VR devices are facing and as this question also shows these are some of the factors that a VR-systems need to fix to attain higher adoption rate.

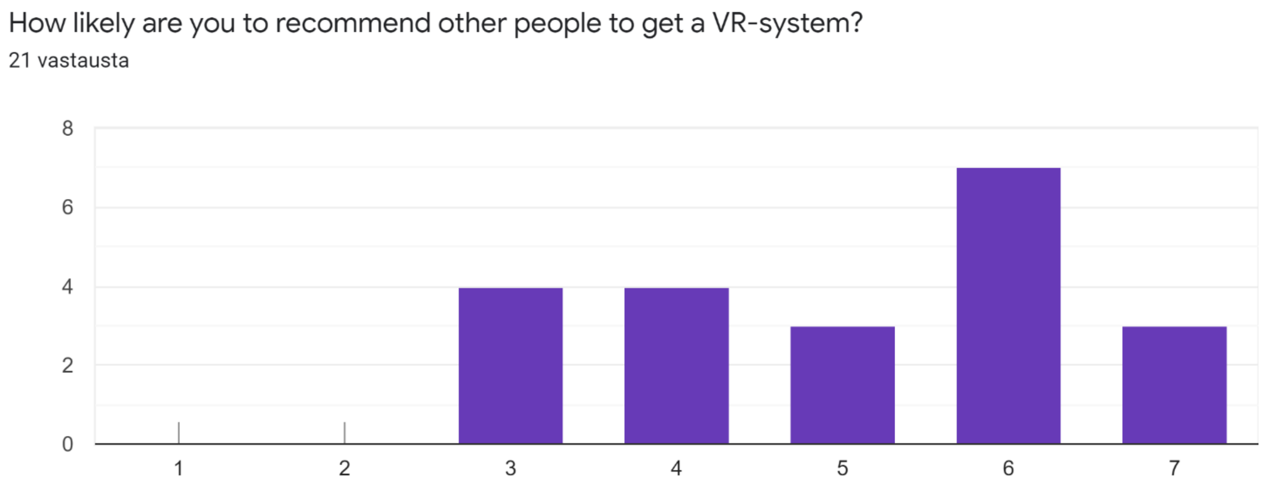

Figure 9. Recommendation likelihood of VR-systems.

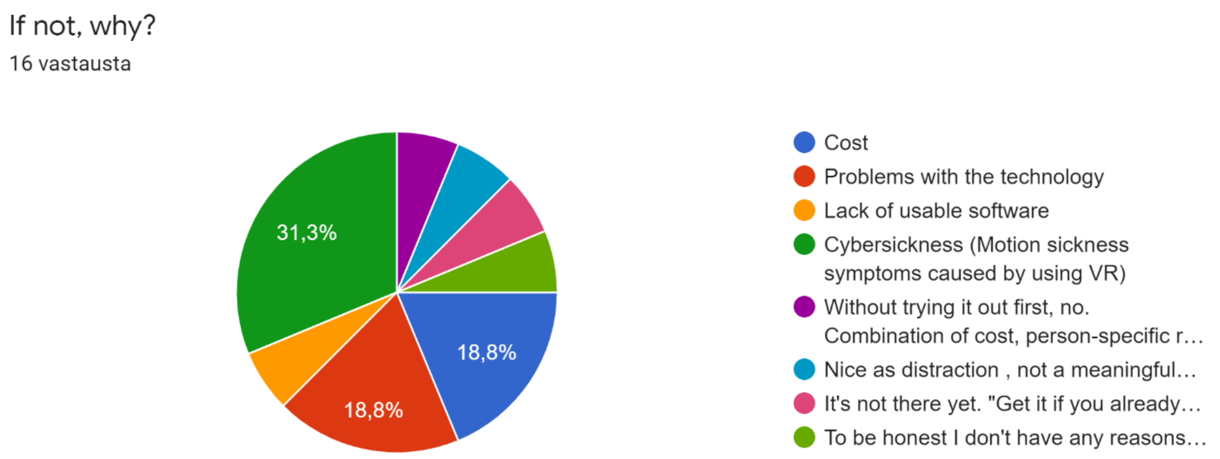

Figure 10. Reasons to not recommend VR. Small sections are 6.3%.

5.3. Postponed testing due to the Covid-19 pandemic

To test out whether the control schemes laid out in the prototype are truly functional they need to be tested. And to do this the best idea is to get a set of people that are interested in the technology or at the very least work operate regularly within the computer and software environment so that they have at least so kind of basis in the subject of Virtual Reality to produce reasonable results. The main difference between people with little knowledge with VR to the people with at least moderate amount of it is that the former can have incorrect expectations of the subject and expect wrong things from it, thus resulting in giving skewed results if things are not working the way they expect, while the latter is likely to give more accurate results in general although some answers will be skewed due to their personal biases.

To handle this the testers were intended to be picked up from volunteers from the computer science students at LUT University, especially from the Bachelor students. They would have been tasked to use an Oculus Quest 2 in a testing lab at LUT University and run the prototype to conduct the three tests laid out earlier in this thesis using some of the various control scheme settings in order to find out what they think of the settings they are testing. After completing the testing itself, the tester would fill out the survey to find out what they thought about the schemes they tried out.

Since there are tens of different variations to try out it is not reasonable for every tester to try out every single one out. Additionally, as people complete the tests they are tasked to complete, they will become more familiar in the layout of the environment which would result in later test passed being easier or faster to complete. This would result the later tests having better results from the tester, causing a potential skew in the results not related to the scheme itself. As such having each tester only conduct a few test passes with as different schemes as possible can help avoid this.

However, due to delays in setting up the testing lab, the complications of the Covid-19 pandemic causing multiple lockdowns and reduced personnel at the University, including the students themselves outside of mandatory lectures, with another remote working recommendation happening right as the testing was supposed to happen as well as the time constraints related to actually finishing this thesis in a reasonable timeframe, it was decided that the testing portion of the this thesis would be omitted for now and postponed to a later date. This would happen when the pandemic situation eases up again and people could be in good conscience brought in to test the prototype and as such gather the test results. As there is no telling when this would exactly happen it was decided that there is no point in holding back this thesis and that the test results would be held in a separate paper when it does happen.

In its stead it was also decided to note some of the observations from the internal testing done during the development of the prototype to set what could be expected from the actual test results when they are eventually gathered. Additionally, these notes and observations could be used in the future to further develop the testing environment to get even more comprehensive results from future research.

Firstly, with the “free movement”-style it was found that the movement itself was decently fluid on every control device with the main exception being that due to the camera being separated from the character direction resulted in needing to rotate the player character separately, something that is not needed in non-VR games and resulting in potentially cumbersome controls. This was not found to be a large issue with the limited button setup used for the test but could be an issue when a large amount to different buttons is needed. Also, the disassociation of camera direction with the direction of the character can result in difficulties in tracking the actual of the direction where the character would move at times this could be fixed easily with some kind of marker that shows the actual direction of the player character. Additionally, using free movement in roomscale VR can result in cybersickness symptoms due to the disassociation of movement from the bodily movements that the user brain is expecting. This did not happen when using VR sitting down, this however limits the movement of the motion controllers.

The cursor movement was found to be fairly slow on controllers, which was to be expected when compared to existing first-person controls on a controller. This does however make using the cursor hard to use. This was not an issue with a mouse, and it could be moved fast and accurately across the screen. However, if the cursor is not tied to the screen, it can become difficult to track the position of the cursor when the player turns their head and does not look forward. Additionally, it was noticed that the when the cursor was tied to the screen that when the player moved their head, they were likely to stop moving the cursor simultaneously. This corresponds to the finding made by Qian & Teather (2017), though in this case the reason is likely due to the unfamiliarity with the method and difficulty with tracking the cursor and controlling the head simultaneously.

Notably, the keyboard and mouse are practically unusable in roomscale setup due to their size and weight consigning the user to be next to a desk making them unable to take benefit from the freedom of roomscale VR. Controller does not suffer from this but being unable to use one's hand like with motion controllers does raise the question of whether there is real point in using roomscale without motion controllers.

In shooting test, it was found that aiming through moving one's head was an effective way of aiming as well especially with a controller due to the slowness of aiming with an analogue stick. With a mouse, moving the gun with the mouse was also snappy and accurate providing there is some sort of targeting reticule in helping with aiming.

With the writing the drumming was found to mostly be suitable with just the motion controllers due to needing 6 degrees of freedom movement that the motion controller provides to use it properly. Otherwise, it causes a lot of accidental hits on the keys. It also seemed notably slower than pointer writing. Additionally, using the physical keyboard to write was found to be less cumbersome than expected, as even though being unable to see the keyboard will result in a lot of incorrect inputs it was found to be fairly fast still and with experience it could be possible to notably reduce the mistakes. Switching from other control devices to use keyboard is unintuitive however, resulting in this method only being suitable when already using keyboard. The likely best method is the pointer style keyboard as pointing at the desired keys and pressing them was relatively easy and reasonably fast.

Overall, the movement with keyboard and mouse combo was found to be mostly pleasant even though the immersion factor that is one of the big factors with VR did ultimately suffer, with the only real issue being related to the rotation of the player character. Controllers' slowness likely hinders it enough to make it not be reasonable to be used unless keyboard and mouse are not available. On the other hand, the motion controller offers the best immersion factor offers good control of the players hands but does notably suffer in the movement factor due to the teleporting likely causing disconnect in the flow of the game.

5.4. Conclusions

While the lack of testing does hinder making assertions about whether this research is actually helpful or needed some key points have rose up from the internet survey and the limited internal testing that was conducted. These points should help setting up the following research subjects as well as the testing that was postponed from this thesis and potentially even improve upon it.

On the potential usage of keyboard and mouse combination or a typical game controller in VR setting there are notable markings that could suggest that the idea is not entirely unreasonable, at least on the keyboard side of things. The results of the internet survey showed that people find the motion controllers currently used in Virtual Reality environments as one of its current weak points. While they are intended to improve the immersion factor offered by the VR HMDs and successfully do so quite often, they do suffer from significant issues then used outside of simply picking up objects and operating them, one of the main ones being menu navigation, text writing and accuracy especially when operating objects with tighter constraints.

A lot of this to do with the fact that the controller location varies constantly due to the nature of motion controls which makes it difficult to design interactable UI elements when one does not know where the point of origin for different signals are coming from. In contrast, the origin with mouse controls is located on a more static position on the player's character, typically around the location from where the player is looking from assisting in determining which way the user needs to adjust themselves to make using menus easier if needed. Additionally, the menu elements and worlds game objects need to be larger due to the controllers being suspended by the users' hands, part of the body that is hard to be kept still due to the force of gravity. This often results in the controller objects swaying when people try to use them due to the swaying of their arms, resulting in decreased accuracy. More traditional control devices do not suffer from this due to them either not needing to be in motion to used or by being supported by a table.

Additionally, the relative lack of buttons makes using more complex commands more difficult on motion controllers. With the trigger on each controller being typically reserved as an “action”-button and the side button used to grab objects conducting other actions is relegated to the two button, two system buttons and analogue stick on each controller. This limits the number of actions that a player can conduct quickly, this became clear in internal testing as most of the buttons ended up being used despite the narrow set of actions that player could conduct in the VR-scene.

Another factor that limits the usability of the VR motion controllers is the lack of capability to track finger movements of a person. This results on a lack of minute adjustment and interaction with small elements in the VR environment that is available in the real world. This results in the various element needing to be larger than desirable so that the player is able to interact with them.

This leads to the idea that using keyboard and mouse for example could fix a notable amount of these issues, keyboards have a large number of buttons that actions could be bind into and mouse could offer very accurate control over the environment. This would result in room scale VR becoming unfeasible due to the bulkiness of these devices and the need of them being placed on a table or a desk but would likely offer notably better experience when using VR is front of a desk. This would come at the price of immersion, but immersion does not help if the game itself is tedious to play. The main downside of using keyboard specifically is the fact that due to the HMD covering the users eyes they are unable to see what buttons they are pressing. This could lead to the user resting their hand in the wrong place and as a result cause them to press wrong buttons. Additionally, rotating one's character with a keyboard was noticed to being a small issue. This is due to not being able to use the HMD to rotate around especially when sitting in front of a desk and the mouse being limited to controlling the cursor. This requires a new way of rotating one around, considering that the sideways movement keys are typically reserved for strafing the character, raising the buttons needed to control one's horizontal movement from four to six in all likelihood, making the controls more complex than before.

The controller was found to suffer from sluggishness in the when controlling the cursor pointer with an analogue stick. This could be improved by making the cursor faster, but this would come at the cost of accuracy which is something that is not desired. This makes game controller worse option when compared to the mouse to control the cursor, likely making keyboard and mouse combination a better option when using VR in PC environment. However, the inclusion of a D-pad buttons could likely help with navigating the various menus leading the typical controller being valid option for use when mouse cannot be used, and the user is not in trying to achieve room scale experience.

However, no matter what the method of controlling the character in VR ultimately is, it does not particularly help with the issue of text writing very much. The pointer method was found to be functional with any controller but ultimately also be a little slow, while the drumming was found to be usable only with motion controls and while being pleasant to use also more error prone and thus slower. Using an actual keyboard to write on the other hand was found to be fast but extremely error prone due to being unable to see the keyboard. This combined with the fact that using it is very awkward when using other control methods than a keyboard means that this method might also be undesirable unless a way of seeing the keyboard to write on is found out. As a result, other alternatives should be developed and observed for people to be able to write text in VR-environment.

Considering that the reception to the idea of bringing keyboards and mouse into the VR-experience was slightly positive, it shows that there is desire for more accurate and versatile control scheme than what the motion controls currently offer. And as the idea of eliminating the controller in its entirety for actual body tracking was very highly received it shows that the current controller situation is not sufficient enough. This point to the idea of using an alternative to be valid and require more experimentation and research.

6. Summary and conclusion

Virtual Reality has been making its entry into the digital market since the original Oculus Rift was released in 2016. The technology offers numerous new opportunities for both entertainment industry in a form of more immersive experiences in video games as well as placing the user into the virtual environment as if they are actually there. Additionally, the VR allows the educational sector to offer practical training for people without needing to travel from one place to another or needing to have the tools needed to handle the task in real life. However, the adoption rate of VR-systems has been slow and even the people who do own a VR-system do not use them particularly often.

The slow adoption rates can be attributed to the relative recentness of the technology, the issues that they still suffer from, the relatively high cost associated with them, the amount of room space needed to use them in room scale as well as the notable lack of high-profile applications to be used on them. Most of the notable VR-games are space simulators that are already in a niche and even they do not require people to use them. The one notable more traditional high-profile VR-game is the Half-Life: Alyx from Valve Corporation.

This slow adoption rate then results in most large companies not being willing to commit to this new technology that does not have a high user base to buy their products, especially in the gaming market. This lack of games then further feeds the slow adoption rate, creating a self-feeding loop that holds the market back. A lot of this has to do with VR, creating a completely new type of section of games with room scale VR where the player moves around by moving their own body physically and using motion controllers to control the players upper limbs to interact with the environment. This makes it incompatible with most pre-existing games due to the drastic departure from the way people move and handle the camera.

The question then lies whether it would be possible to use these “old” methods of traversing the environment used in contemporary non-VR games in actual VR-games. This is what this research started to investigate, whether these traditional controls could be used in VR to offer an alternative to the room scale VR and help both the developers with being able to offer VR capabilities in games for those that own an VR-system and the consumer to ease them into owning an VR-system by potentially offering higher number of videogames where the system could be used in to enhance the experience provided to them.

To find answers to this question a prototype program utilizing VR capabilities was created with the usage of the Unity-engine, one of the most commonly used game-engines in contemporary gaming market and VRTK a set of tools and components that can be used to help in the creation of VR-software on Unity-engine and its editor. This program would then enable people to use various different control schemes and devices in a multitude of tests to determine which ones of these could offer possibilities to improve the VR-experience or functionality. The control devices usable in the program were keyboard and mouse combo commonly used in PC environment, game controllers commonly used on game consoles and motion controllers that are currently used in conjunction with VR-devices to simulate realistic body movements.

The program consists of three tests that are intended to test different aspects of the VR experience. These tests were the labyrinth test, where the tester was tasked to navigate a pitch-black labyrinth and find three buttons within to complete the test, the shooting test where the tester was tasked to hit 15 targets with a pistol and a writing test where the tester was tasked to write three phrases with various different writing methods.

These tests were put to be conducted by a set of testers gathered from the students of LUT University, with the intent of finding out which control devices and control schemes would perform the best. To figure this out they were set to fill out a survey to find out what they thought about the schemes and which they found out to function the best in their opinion. These results would be subjective and would change from person to person but also offer perspective onto which solution would likely offer the best results in more grand scale, such as a full-blown videogame utilizing VR-technology. In addition, a set of objective results were to be gathered from the tests in terms of speed and accuracy to help making these conclusions.

However, due to the delays in setting up the testing environment and constraints caused by the Covid-19 pandemic combined with the timelines to conduct this testing it was decided that the testing would be omitted from this thesis and postponed to be conducted on a later date when the testing could be done in a safe and controlled fashion. The internal testing however did show the keyboard controls to offer a decent experience although finding the exact button prompts that would be ideal for this type of system would need to be found in further research. Additionally, the pointer keyboard offered fairly decent writing experience although maybe a touch slow, while using the physical keyboard was found to be fairly fast but also very error prone and difficult to transition into unless the user was using the keyboard to begin with.

Additionally, alongside the testing of the actual prototype an internet survey was conducted that was focused on finding out people's opinions on the current state of VR-technology and the systems it has produced. The main focus was to find which aspects people believe are in the most need of improvement and what hampers their enjoyment of the technology, if any were to be found. This would offer perspective on what direction and sections of VR-technology should be researched and improved so that these systems could be eventually adopted by the common people on mass as well as whether this type of research into VR controls is needed or whether it should be deepened to truly fuse the existing control devices into the VR environments to offer more enjoyable experiences to people.

The main takeaways from the internet survey were that people had issues with the screens of VR devices while the controllers were the other aspect people were desiring improvements for. On the software side of things people were desiring improvements to menu navigation, and controller accuracy, while text writing, movement capabilities and the UI provided to the player were also mentioned. On the hypothetical controller questions people were very interested in the possibility of removing the controller aspect altogether and use their hands to control the VR-experience, while the idea of “reintroducing” the keyboard and mouse combo received moderately positive reception. Finally, when asked whether people would be willing to recommend VR to others the answer was mostly positive with most issues being related to Cybersickness and the cost associated with the devises.

6.1. Discussion

Using VR nowadays in consumer applications has been largely relegated to the uses in room scale VR with the usage of the motion controllers. This style of gameplay however suffers notably from the issues related to movement as moving the player character separately from players down movements while to player is standing up or moving themselves is liable to cause cybersickness symptoms for the user, making prolonged usage next to impossible. As a result, most VR games have used teleportation movements to circumvent this issue and allow people to move around. This however causes the gameplay to become jagged and jarring as the viewpoint changes notably from one point to the next just to induce movement. Combine this with the limited space people have to use to move around without teleporting in Room scale VR and suddenly fast paced movement become impossible to conduct without being jarring and massively breaking immersion, the main reason why VR is being advertised for gamers.

To find solutions for this there is a need to take a step back and consider is all the immersion effect always needed or could the games also be successfully run with just slight improvement in immersion without losing from the general enjoyment aspect that players play games for and simultaneously make the games easier to develop for developers and allow for them to include VR capabilities in games that currently do not take use of them without sacrificing the core gameplay in the process. This is the general premise from where this research was started from.

To find out potential solutions for this a step was taken back from the roomscale VR towards more traditional gaming solutions. These have proven to be functional over the years and are used in games outside of VR environment. So maybe they could also be functional with VR applications despite the notable paradigm shift the possibilities of freely moving one's camera with HMD has provided. As such this thesis started to look into the possibility of using keyboard and mouse combo or alternatively a traditional game controller to control players actions inside the VR environment and find out whether they could also be used in VR and potentially fill in with the issues that motion controllers suffer from, allowing VR to reach the hands to larger audience pool than what it currently has achieved.

The results that were gotten on this subject were promising, the prototype showed in internal testing to offer good movement capabilities with both keyboard and mouse as well as a typical controller for VR environment. This notably allowed the player to not need to rely on teleportation capabilities used with roomscale and mostly move as they have in contemporary games. The main downside came from needing a new method to rotate one’s character around as the mouse could no longer be used for it as it had been before. This was not a large issue in the prototype but in a more complex videogame this could prove out to be a hinderance. As such this aspect would require more research and should a suitable solution be found using keyboard in VR would prove to be beneficial. Another issue was that people would be unable to see the keyboard and hence which buttons they are holding their fingers on. This could result pressing incorrect buttons from time to time, but also especially proved to be the main issue that makes writing with keyboard difficult in VR. Finding a way to see the keyboard somehow through the HMD would help with this issue but that could be difficult to make happen without compromising the gameplay itself to make the keyboard appear on the screen. However, making the keyboard visible might be beneficial when writing with keyboard as the gameplay likely would not matter as much.

The traditional controller on the other hand notably suffers from slow movement related to the limitations of an analogue stick. This makes using a cursor with it sluggish and not entirely practical. However, due to the compact size and limited number of buttons means that it does not suffer from the same visibility issue as the keyboard does.

Overall, the testing would suggest that motion controls are preferable when using roomscale VR due to other methods being unsuitable for the method. This would also give the highest possible immersion for the player. Keyboard and mouse would be suitable for VR applications where the player is sitting down in front of a desk where they can use the controls. This results in lower immersion compared to roomscale but would still benefit from the HMD camera movements. This could also allow for adapting current first-person games into VR without too much of a hassle, provided the issues noted earlier could be solved.

Writing in VR is a notable issue and the solutions provided in this seem to not offer an optimal one. The keyboard input is awkward to transition into and suffers from excessive amounts of errors, drumming style only works with motion controllers and does not really offer much more speed compared to the pointer style which can be used with any control style. The pointer method offers similar type of speed as many of the contemporary writing methods on game consoles but does suffer from the fact that the keys on it need to be of considerable size to avoid miss-clicking on neighbouring keys when writing. This results in none of the methods being optimal and another way to handle writing should still be found.

However, the validation of these findings is not properly possible at this time due to the postponement of the testing phase that was supposed to test out the control methods with the prototype. But for now, all that can be said that the idea of using keyboard in VR in particular does not seem to be an unreasonable one and could potentially be implemented in the future when the issues are ironed out.

6.2. Future Research Topics

Virtual Reality is a vast and largely unexplored area of computer science that people are only starting to dip their toes into. However, movement in VR space and what ways could it be made more usable and effective are some of the most crucial issues facing the current VR technology. As such trying to develop the current methods further while also looking for alternatives is crucial.

Firstly, the testing phase that was postponed from this thesis should be conducted to validate and evaluate the ideas and finding from this research. Since the testing itself was postponed, some consideration should be given to the idea of improving and expanding on the test cases already laid out in this paper. These improvements could include the inclusion of verticality into the VR movement that is omitted currently, including new ways to handling the writing aspect in VR, adding new methods of handle the rotation when using keyboard or traditional controller as well as expanding onto the interaction between object and the player with these “new” control methods. The inclusion of verticality should preferably be kept as a separate test and not add it into the current movement test. This is due to the potential issues with cybersickness that verticality could cause and not let it interfere with the rest of the movement testing, at least initially to find a good base line for these control methods to move forward. On the subject of writing one potential way of handling it is through the usage of two radial dial –type selectors that would allow the person to add character depending on which position the dials are set to. These dials could be easily controlled by the analogue sticks on various controllers. By splitting each dial, for example into six sections this would allow 36 or 49 different combinations for characters depending on if no input is considered a valid input also. Considering that the English alphabet is the size of 26 letters. This would leave a significant amount of space for special characters or characters used in other languages.

Another research topic related to movement and controls in VR would take a form of investigating all new methods of one to control themselves in VR with completely new control devices and methods specifically tailored for VR but also avoiding the issues that the current controllers suffer from in terms of accuracy and detail. For example, being able to use one's fingers in VR could allow significantly more detailed control over the environment and allow the user to interact with significantly smaller objects.

Further research subjects would include the research into controls without using external controllers even in the first place. This likely is an eventual direction that both VR and AR applications are headed into as an evolution of the current motion controllers and as the survey conducted shows, there is notable interest towards this direction with the way of controlling oneself. This research would take still a long time to properly materialise and even then, it would take a while to step into consumer applications, but this advancement would revolutionize the way people conduct their work and how they are able to enjoy entertainment in their free time.

Finally, more research should be devoted to improving the HMD-devices. The current devices are bulky and heavy causing strain of the user's neck. Additionally, each person's head is of different size and shape from the next causing adjusting VR-devises being difficult at times and time consuming if it is used by multiple people. Creating ways for people to carry their VR-devices around more easily and making them easy to just put on one's head and use them would make the much more usable in various situations.

6.3. Final Words

Virtual reality is something that has been presented to the general public in terms of various fiction stories where the technology is available in the future and offers numerous opportunities as well as risk associated with it. And while the idea has been teased multiple times over the years to us it took until 2016 for the technology to take its first steps and enter the consumer markets with great anticipation.

The five years that has been elapsed between that and the time of this paper being written have been relatively slow however. Adaptation has been slow, and the software and games have been mostly just small-scale indie games that essentially serve as a proof-of-concept creations that allow people to experience the technology but is missing the grandness that the technology has promised to offer to the players. The main large-scale games offered to the ever-anticipant gamers can be counted on one hand, with majority of the being space-simulators or flight simulators that fall into a fairly narrow niche and are unable to attract the general consumer. The only real splash of grandness in the VR-market was made by Half-Life: Alyx showing us that large scale games are possible to be made for Virtual Reality but outside of it the VR-gaming field has been quiet.

With the adaptation of VR being as slow as it is, large game companies are unwilling to commit to creating these grand experiences as there is uncertainty whether they are able to recoup their expenses back and people are unwilling to commit to the device as there is uncertainty whether they have games to use their new devices on. Therefore, finding ways to make gaming on VR-devices more appealing is necessary to for the technology advance on the gaming side.

One way to make this happen is to offer ways to make VR accessible for pre-existing games or by making it easy to add the technology to contemporary first-person games. To constitute this, Virtual Reality need to be able to be used on the control devices used currently to play these games. And as the research done on this paper shows there is interest towards this among the people that already own VR-devices and while the testing had to be postponed the internal testing made while creating the prototype did offer some promising results, even though some issues were identified at the same time. However, the testing would need to be conducted to find a proper answer for this.

Down the line the likely path to head towards with VR controls is the omission of physical controllers altogether and using person's body movement themselves to control the characters in the virtual world and there is interest towards this as shown by the internet survey but getting there is going to take time and if using more traditional controls can be used as an alternative to help people adopt VR-systems to make developing for them more reasonable and viable then they should not be ignored as an middle step.

VR systems and technology are currently in constant flux and many of the current VR-companies are at least partially shifting their focus towards Altered Reality application with Meta (company previously known as Facebook), the company that is creating the Oculus-devices many use for VR nowadays, spearheading this shift. This shift aims to use these devices on productivity and real-life applications, instead of entertainment as VR would. These two technologies do however develop together with each other and any improvement to HMDs with AR in mind would also trickle towards VR, so the technology is likely not going anywhere. However, if we want it to be a part of a gamers repertoire and be used in peoples living rooms or bedrooms for entertainment something needs to be done to assist this. Nevertheless, those that have used the technology see the promise it offers; it just needs to be utilised.

References

Gigante, Michael, (1993), Virtual Reality Systems, London: Academic Press, Print

Growth, Colin, et. al. (2021), "Visual Techniques to Reduce Cybersickness in Virtual Reality", *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 486-487. Web. 21 Aug. 2021

Lugrin, Jean-Luc et.al., (2013), "Immersive FPS Games: User Experience and Performance" *ImmersiveMe '13: Proceedings of the 2013 ACM international workshop on Immersive media experiences*, pp. 7-12. Web. 15 Apr. 2021

Wiesing, Michael, et. al. (2020), "Accuracy and precision of stimulus timing and reaction times with Unreal Engine and SteamVR", *PLoS ONE*, 15(4), Web 26 Mar. 2021

Ng, Adrian, et. al., (2017), "A low-cost lighthouse-based virtual reality head tracking system", *2017 International Conference on 3D Immersion (IC3D)*, Web. 25 Mar. 2021

Boyer, Steven, (2009), "A Virtual Failure: Evaluating the Success of Nintendo's Virtual Boy" *Velvet Light Trap: A Critical Journal of Film & Television*, pp. 23-33. Web. 15 Apr. 2021

Arndt, Sebastian, et.al. (2018), "Using Virtual Reality and Head-Mounted Displays to Increase Performance in Rowing Workouts", *MMSports'18: Proceedings of the 1st International Workshop on Multimedia Content Analysis in Sports*, pp. 45-50 Web. 7 Apr. 2021

Hoffman, Chris, (2019), "What is the 'Screen Door Effect' in VR?", How-To Geek, 11 February 2019, Web. 15 Apr. 2021 https://www.howtogeek.com/404491/what-is-the-screen-door-effect-in-vr/

Reski, Nico, Alissandrakis, Aris, (2020), "Open data exploration in virtual reality: a comparative study of input technology", *Virtual Reality*, pp. 1-22, Web. 5 Apr. 2021

Munafo, Justin, et. al. (2017), "The virtual reality head-mounted display Oculus Rift induces motion sickness and is sexist in its effects", *Experimental Brain Reseach*, pp. 889-901 Web. 12 Feb. 2021

Niehorster, Diederick, et. al., (2017) "The Accuracy and Precision of Position and Orientation Tracking in the HTC Vive Virtual Reality System for Scientific Research", *i-Perception*, Web. 7 Apr. 2021

Zhang, Chen, (2020), “Investigation on Motion Sickness in Virtual Reality Environment from the Perspective of User Experience”, *2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE),* Web. 7 Apr. 2021

Williams, Kevin D., (2014), “The effects of dissociation, game controllers, and 3D versus 2D on presence and enjoyment” *Computers in Human Behavior,* 38, pp.142-150. Web. 15 Apr. 2021.

Isokoski, Poika, Benoît, Martin, (2007), “Performance of input devices in FPS target acquisition”, *ACE '07: Proceedings of the international conference on Advances in computer entertainment technology,* pp. 240-241 Web. 15 Apr. 2021

Thomas, Bruce, (2012), “A survey of visual, mixed, and augmented reality gaming”, *Computers in Entertainment,* Web. 02 Apr. 2021

Bovet, Sidney, et. al., (2018), “Using Traditional Keyboards in VR: SteamVR Developer Kit and Pilot Game User Study”, *2018 IEEE Games, Entertainment, Media Conference (GEM),* Web. 20 Apr. 2021

Sinclair, Mike, et.al., (2018), “Three Haptic Shape-Feedback Controllers for Virtual Reality”, *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR),* Web. 20 Apr. 2021

Seibert, Jonmichael, Shafer, Daniel, (2018), “Control mapping in virtual reality: effects on spatial presence and controller naturalness”, *Virtual Reality*, 22, pp. 79-88, Web. 26 Apr. 2021

Roupé, Mattias, et. al., (2014), “Interactive navigation interface for Virtual Reality using the human body”, *Computers, Environments and Urban systems,* 43, pp. 42-50 Web. 27 Apr. 2021

Qian, Yuan Yuan, Teather, Robert, (2017) “The eyes don't have it: an empirical comparison of head-based and eye-based selection in virtual reality”, *SUI '17: Proceedings of the 5th Symposium on Spatial User Interaction,* pp. 91-98, Web. 27 Apr. 2021

Hou, Wen-jun, Chen, Xiao-lin, (2020), “Comparison of Eye-Based and Controller-Based Selection in Virtual Reality”, International Journal of Human-Computer Interaction, 37(5), pp. 484-495, Web. 27 Apr. 2021

Ali, Monthir, Cardona-Rivera, Rogelio, (2020), "Comparing Gamepad and Naturally-mapped Controller Effects on Perceived Virtual Reality Experiences", *SAP '20: ACM Symposium on Applied Perception 2020*, pp. 1-10, Web. 27 Apr. 2021

Pham, Duc-Minh, Stuerzlinger, Wolfgang, "Is the Pen Mightier than the Controller? A Comparison of Input Devices for Selection in Virtual and Augmented Reality", *VRST '19: 25th ACM Symposium on Virtual Reality Software and Technology*, pp. 1-11, Web. 27 Apr. 2021

Virtual Reality Toolkit, https://www.vrtk.io/tilia.html, Web. 19 Sep. 2021

Appendix A.

- Gender
- How would you rate your experience using *control method here*? (1=unusable, 10= excellent)
- How intuitive do you feel like the control method is?
- Did the control method affect the immersion you felt during the test?
- How likely would you see yourself using this control method in an actual game if it was offered? (1 unlikely - 5 very likely)
- How did you feel that the teleport controls functioned compared to free movement? (If both movement methods are used.)
- Did you experience motion sickness during the test? If yes, how severe, and how quickly did it develop and during which test?
- How would you rate the aiming methods used in the test? (1-10)
- How likely would you see yourself using this aiming method in an actual game if it was offered? (1 unlikely - 5 very likely)
- How would you rate your experience writing text with *method here*?
- Which one of the methods could you see yourself using within VR-games? (This isn’t asked if only one control scheme is tested.)
- How cumbersome was it to change to the physical keyboard?
- Would you recommend people to try out using VR without motion controls if that opportunity was offered to you? (This is asked only if the tester uses methods other than Motion Controls)
- Any comments related to the test or the control methods? (Open ended answer)

Appendix B.

- What VR device do you own/ use the most if any?
    - Oculus Rift (S)
    - Oculus Quest
    - HTC Vive
    - Valve Index
    - Windows Mixed Reality -headset
    - Sony PlayStation VR
    - Other, What?
- How often do you use your VR headset?
- How satisfied are you with your headset?
- What areas of improvements do you think the headset you are using could have?
    - Screen resolution
    - Screen refresh rate
    - Controllers
    - Tracking capabilities
    - Internal software
    - Other, What?
    - None of the above
- What in particular would you like to see improved? (Free form)
- How satisfied are you with the software offered to your VR-system?
- What areas of improvements do you think the software could have?
    - Movement capabilities
    - UI
    - Menu navigation
    - Controller Accuracy
    - Text Writing
    - The lack of software
    - Other, What?
    - None of the above

- Would you be interested in using a traditional keyboard and mouse combination in VR if it was possible/reasonable? (1-5)
- Would you be interested in the possibility of using VR software without motion controllers and by using your hand/finger movements if it was possible/reasonable?
- Would you recommend other people to get a VR-system? (1-5)
- If not, why?
    - Cost
    - Problems with the technology
    - Lack of usable software
    - Cybersickness (Motion sickness symptoms caused by using VR)
    - Other, What?