Tangible input technology and camera-based tracking for interactions in virtual environments

Cato Aleksander Goffeng
Østfold University Collage, Applied Computer Science

May 2016

Abstract

Designing software solutions using virtual environments to train personnel in different real life tasks have seen a rise in popularity whit the release of new and promising input technologies. Businesses see a chance to cut down on otherwise expensive procedures, or just simply to speed up their onboarding process. This study aim to test the Razer Hydra tangible controllers, and the camera-based tracking system Leap Motion for use in high precision interaction tasks in virtual environments. A testbed was designed to measure how both technologies able to handle the grab, move and release phases of a positioning task. A group of ten subjects with varying amounts of experience with human computer interactions where gathered to perform the task presented. The findings of this study indicated that the advantage of haptic feedback present in the Hydra controller makes it the preferred choice for these kinds of interactions. The Leap controllers reliance on a complex gesture logic complicated the release portion of the interaction, which in turn reduces the users ability to achieve the same kind of efficiency whit the device.

Contents

1 Introduction **5**
- 1.1 Motivation 6
- 1.2 Related Work 7
  - 1.2.1 Modal feedback 9
  - 1.2.2 Depth Perception 10
  - 1.2.3 Two-Handed interactions 11
- 1.3 Research Question 12

2 Selected Technology **13**
- 2.1 Depth-camera-based tracking 13
- 2.2 Tangible Input Controller 15
- 2.3 Head-Mounted Display (HMD) 16

3 Method **18**
- 3.1 Participants Selection 18
  - 3.1.1 Participants Information 18
- 3.2 Prototype Design 18
  - 3.2.1 Development Tools 19
  - 3.2.2 Third-party Application Programming Interface (API) 19
  - 3.2.3 Task 20
  - 3.2.4 Data Collection 22
- 3.3 System Usability Scale 24
- 3.4 Think-aloud protocol (TA) 24
- 3.5 Interview 24

4 Evaluation **26**
- 4.1 Technology Setup 26
- 4.2 Procedure 26

5 Results **28**
- 5.1 Testbed Logs 28
- 5.2 System Usability Scale 31
- 5.3 Axis difference 33
- 5.4 Think-Aloud, Interview and Observations 34

6 Discussion **36**
- 6.1 SQ1 36
- 6.2 SQ2 37
- 6.3 RQ 37
  - 6.3.1 Pinch Gesture Drawback 38
  - 6.3.2 Benefit of Haptic Feedback 38
  - 6.3.3 Compounding Issues 39

7 Conclusion **41**

8 Future Work 41

9 Acknowledgements 41

A
Think-Aloud, interview, Observations 45

B
Form of Consent 50

C
Subject Information 51

D
System Usability Scale 52

1 Introduction

As virtual environment(VE) technology is becoming more affordable to the public new input technology emerge to compliment the immersion of the users. One of these areas is camera based tracking where Leap Motion[1], Kinect[2] and Wii Remotes[3] are often used to track motions of the user, and excel at different areas. Depth-camera-based tracking differ from tangible controllers in that the user is completely free from tethered devices by utilizing their hands to directly influence the interactions. This freedom could possibly provide benefits for the user, contrary to Hinckley, Pausch, Goble, and Kassell proposal that using physical props is preferable for conducting interactions in a three-dimensional user interface [4].

Virtual environments pose new challenges to the design of interaction interfaces, compared to the standard WIMP (Windows, Icons, Menus and Pointers) interface we are used to from the inception of graphical computer interfaces. The mouse and keyboard composition have been the standard for traditional computer interactions, and will likely not change in the immediate future as they are effective and familiar to the users. From its inception this interface complemented the processing power of its contemporary computers and the limitations of display technology. Computers have come a long way in terms of their graphical power and we are beginning to see more interest in the VE field as several manufacturers are developing new display and input technology. Attempts to bring VE to the consumer market in the past have made varying success, but the improved computational power now enables virtual environments that can completely immerse the user in an artificial scene. The user is no longer limited to a two-dimensional output that the WIMP interface excelled at controlling, but VE interactions differ from traditional WIMP interaction metaphors. VE interactions on the other hand are not formally defined and new technology is being developed in a fast pace. Formal descriptions of multimodal interaction techniques have been proposed [5] [6], but there is no set standard.

Two types of input-technology have been generating a lot of interest and show some potential in different areas. The first is camera based tracking of the users hand gestures. This approach allows the user to manipulate the objects in a scene directly with their hands, and does not tether the user to a physical input-device. The second approach is tangible 6 degree-of-freedom controllers that tracks position using magnetic fields. This approach allows is similar to the props proposed by Hinckley, Pausch, Goble, and Kassell [4] as the controller offer a tangible device with predefined buttons for interactions similar to the traditional mouse and keyboard composition in contrast to the camera based tracking solution.

1.1 Motivation

My motivations for selecting this area in my Master theses come from a fascination of new technology and its possible applications. My interest in game design and 3D modelling sparked my curiosity for experimenting with input devices like Razer hydra and Leap Motion to see how they would perform in a virtual environment setting. During the computer graphics course conducted at the Institute for Energy Technology (IFE), we were showed some examples of what they were working on at their facilities. This impressed me and probably motivated me to select this field of study. I contacted the software engineers that were responsible for the class to see if they were interested in a collaboration with my masters thesis. They pitched some ideas that they were interested in exploring, one being object manipulation in virtual environments. This aligned perfectly with what i wanted to explore, and they would supply the needed hardware for the project.

As the popularity of virtual environments grows in conjunction with the new and compelling input technology that is being developed, companies and organizations explore the possibilities to incorporate this type of technology into their proprietary systems. IFE as a part of the OECD Halden Reactor Project (HRP) is actively pursuing ways to utilize virtual environment to train personnel in various operations. By creating virtual representations of expensive equipment and facilities a reduction in costs may be possible to achieve, provided that the technology is able to create believable interaction metaphors and realistic scenarios. Research has been conducted at IFE on various ways of conducting training in safe environments, limiting potential hazards related to radiation exposure. IFE is experimenting with non-tangible controllers for this purpose, and have incorporated the Microsoft Kinnect sensor for full body tracking and modeling in several of their solutions. Although the Kinnect is great for large scale tracking, the Leap motion sensor is supposedly better suited for detailed interactions with higher accuracy [7].

1.2 Related Work

Although the field of VE in its current form have gathered lots of interest the last couple of years, virtual environments have been tried multiple times with varying results. General research in the field is plentiful and go back decades with various technology. Modal feedback, two-handed interactions and depth perception are areas that constantly emerge when researching work done on virtual environment interactions, and will probably be central elements of this study.

Card, Moran and Newell propose that a computer is a tool for multiple uses compared to for example a hammer that is designed with one specific task in mind [8]. These interactions take place as an open-ended dialog with the computer as the user will issue a command through the available input devices that the computer processes and returns a response as shown in figure 1.

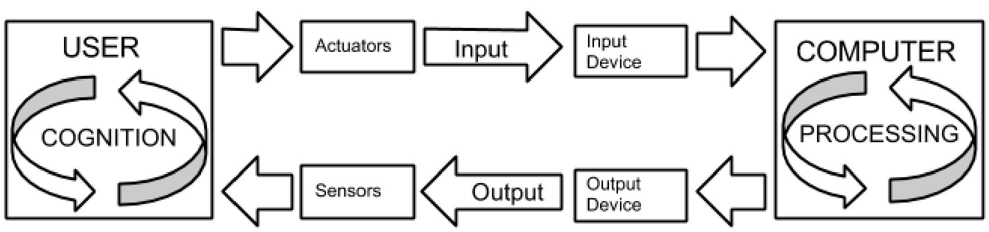


Figure 1: Human-Computer Interaction

The primary input devices in use today is the keyboard and mouse combination, and compliment each other well. Using the mouse a user can select, manipulate and navigate the visual components of the computer interface, while the keyboard provide textual input, modifications to the mouse input and dedicated buttons for specific operations e.g volume. Additional input through voice and gestures is increasingly used to compliment the traditional input devices. Voice and gestures reduce the amount of interactions needed for specific predefined tasks i.e voice search and pinch to zoom, but does not replace the mouse and keyboard as the preferred computer input devices. The put-That-There system developed by Bolt showcase the combination of speech input and gestures to communicate with the computer [9]. In virtual environments a goal is to provide the user with an intuitive physical manipulation of virtual environments, and one of the leading collaborators in the field of Tangible User Interfaces (TUI) Hiroshi Ishii describe his vision for a TUI in his paper Tangible Bits: Beyond Pixels [10]. Early in development TUI was called Graspable User Interfaces but later changed to Tangible User Interfaces, probably to avoid confusion regarding its acronym. A TUI is probably best described as a system where the user interacts with digital information through the manipulation of physical objects directly representing elements present in the system. Ishiis goal was to fully utilize the human ability to grasp physical objects and intuitively manipulate them in order to communicate with a computer. Early examples of his work whiting the TUI field incorporated scaled physical models designed for a given purpose as the Urban Planing Workbench (URP) developed by the Tangible Media Group back in 1999 [11] as an architectural simulation tool. The URP utilized tangible tools in order to manipulate position and rotation of buildings while others controllers like the clock tool where used to alter the position of the sun. Video projection then cast shadows onto the workbench in order to simulate the resulting shadows corresponding to the time of day. Ishii collaborated with George Fitzmaurice and William Buxton to propose a system called "Bricks" [12]. Bricks where introduced as physical objects that could be coupled with virtual counterparts. This is comparable to modern tangible controllers in the way they directly couple the users hand with virtual cursors, and enables the user to directly influence the parameters or objects inside a simulation.

1.2.1 Modal feedback

Feedback is an important part of the communication between a human and a computer and multimodal human-computer interactions can be described as "interaction with the virtual and physical environment through natural modes of communication" [13]. As humans we typically use three of the five senses to receive feedback from the computer, these senses are sight, hearing and touch. When the user interacts with the system there are multiple ways to communicate back to the user that an action has occurred, and that the action is properly handled by the system. If we dissect the common interaction of clicking a button on a web page we see a lots of feedback to the user in every step of the interaction. Compared to some of the devices used in this project the mouse is a tangible device that the user can feel to verify and alter its position. When the user clicks on this button he will first get haptic and audio feedback from the mechanical components of the mouse button. Manipulating the mouse will constantly update the cursor on the screen and give the user a feeling of control. If the cursor is non-responsive or erratic we can easily deduce that something is wrong. This is not limited to motion, other aspects like visual ques of color and size is regularly used by web designers to convey a set of rules that even novice users are familiar with. When the cursor enters the boundaries of a button it can change color or size to indicate its is interactive. Audio is also frequently used to convey interactions or errors and combining both visual and auditory feedback can lead to an increase in performance [14]. Multi-modal interfaces have also showed improved error handling and reliability as users caused 36% less errors compared to a uni-modal interface. This rich feedback was also preferred over the uni-modal interface by over 95% of the users [15].

Adam Faeth also recorded an improvement in using a combination of visual, auditory and haptic feedback, with results indicating that haptic feedback alone would probably not be sufficient in improving interactions whit button clicks [16]. He concludes that a rich amount of different feedback avenues was an effective way to reduce user errors.

There is an entire segment of input technologies in use classified as "wearables" that are designed to be directly connected to the user in some way. These wearable devices convey information back to the user in multiple ways, but most prominently by the use of vibration. This simple vibration can be used for a wide array of different feedback possibilities e.g. immersion, notification and haptics. Camera-based tracking systems like the Leap controller lack this important form of tactile feedback. Pairing the Leap controller with wearable devices to provide the user with tactile feedback may have positive effects on the overall user experience [17][18] by limiting one of its most obvious flaws. This enables the system to give the user feedback in the form of pressure or vibration applied to the fingertips, providing the sensation of touching a virtual object. Later research conducted at University of Siena indicated that the tactile feedback was natural and paramount in offering a realistic experience [19].

1.2.2 Depth Perception

Distance between an object/button and your pointer(i.e your finger) in the scene can be hard to estimate. Research done on depth perception indicates that distance to an object in a virtual environment is usually underestimated [20]. This can pose problems when performing most interactions with buttons or objects. Sugiura, Toyoura and Mao constructed a prototype click interface for artificial reality systems, and conducted a study to try and understand the common patterns related to a click gesture. The study utilized no aural, visual of haptic feedback, and the test subject had difficulty placing their finger on a given button in the scene. Data tracked by a Leap motion controller and post-test interviews indicated that the subjects had difficulty perceiving depth, which made locating the button difficult. Leap motion video also revealed that a click gesture resembles an exaggerated general tapping gesture. It was also evident that the three-dimensional click gesture did not only travel in the Z-direction, but also in the X and Y-directions depending on the button and hand locations. After the preliminary study they started developing the clicking interface. Firstly they defined a set of states for the fingertip.

- STILL: stop at a position
- MOVE: move at a normal speed.
- FAST: move quickly
- Sudden SD: slow down suddenly

State transition is recognized by the system based on the magnitude of speed tracked from the finger. Now a click could be defined from a series of these states happening in rapid succession. Firstly the user confirms an object to interact on with a STILL or MOVE state, if the FAST state is observed the system interprets this to the raising and lowering the finger, then a sudden STOP state followed by STILL to indicate the click. The lack of haptic feedback made it difficult for the users to perceive the distance between their finger and the button, and subjects used additional time to confirm the connection before performing a click. This selection method show precision ratio above 90% (true clicks compared to all clicks detected) [21].

1.2.3 Two-Handed interactions

Both input devices offer a natural way for the user to utilize both hands in solving the task presented. Buxton and Myers conducted an experiment where the subjects where given two tasks that could be performed using either one or two-handed input. The input devices given to the subjects was a graphics tablet and a 4-button puck in the dominant hand, and a slider in the non-dominant hand. The slider offered one degree of freedom and functions similar to the mouse wheel, while the graphics tablet tracked the position of the puck on the screen.

The first task was a selection/positioning task with two separate sub-tasks. The first task had the subjects position an object with one hand while scaling it with the other. The researchers tried to influence the users to do the operations in a sequential manner, hoping the users would intuitively gravitate towards doing the sub-tasks in parallel. All but one of the subjects used both hands simultaneously, with six of the 14 participants doing so on the first trial. Comparing the completion time of the task with the usage of both hands in parallel, a trend that parallel manipulation reduced the amount of time required to complete the task appeared. They concluded that the subjects had no difficulty performing the tasks given, due to the natural comparison with the real world equivalent.

The second was a navigation and selection task where the users would navigate and locate a word in a list. Every test started out with the target word not visible on the screen, so the subject would have to first navigate the list before selecting with the puck. The slider from experiment one was now swapped with a touch-sensitive tablet covered in a cardboard box with two separate vertical strips exposed. One strip would smoothly scroll the document while the other did a jump navigation to a location on the list corresponding to the finger location on the strip. A new set of subjects where used consisting of twelve mouse usage experts, and the rest being novices. These subjects where evenly divided into four groups, where two groups conducted the one-handed version of the test while the others did the two-handed version. Experts from the two-handed group performed 15% better than the one-handed group, while the two-handed novices outperformed the one-handed by 25% [22]. This makes a strong case for improving performance when utilizing a two-handed approach to solve multiple related tasks as shown in experiment two. Experiment one on the other hand show a natural gravitation to two-handed task solutions if designed correctly. As a result the test-bed is designed to offer the option of both one-handed and two-handed interaction.

Fitzmaurice and Buxton [23] experimented with tangible input controllers working together to perform what they described as ”space-multiplexing” in contrast to ”time-multiplexing” input schemes.

> Graspable User Interface that employs a “space-multiplexing” input scheme in which each function to be controlled has a dedicated physical transducer, each occupying its own space. This input style contrasts the more traditional “time-multiplexing” input scheme which

uses one device (such as the mouse) to control different functions at different points in time.

Their work on this subject concluded that a space-multiplexing input scheme could achieve better results for certain tasks compared to the time-multiplexing approach and designing systems tailor made for the task in question may improve performance.

1.3 Research Question

In conjunction with IFE we settled on two different types of input technology that are potentially good for completing precise manipulation tasks. The two input devices selected for this project differ in the amount of feedback the user is awarded. The Leap Motion controller does not require the user to directly touch the controller and therefore does not provide any tactile or auditory feedback from the controller, relying only on the visual and auditory feedback generated by the system. The Hydra controller on the other hand offers the user an experience closer to the traditional mouse, converted to a full 3-Dimensional space.

Two areas where chosen for comparing the selected devices that was felt to be the most important areas to explore, precision and efficiency. First we will take a look at how precise the user can control an object in the virtual environment with the devices. In this precision part the focus lies on the resulting difference of object positioning after the user have completed a grab, move and release action. The efficiency is defines as the amount of time a user spend manipulating the objects in conjunction with the amount of interactions needed to complete the task. The Leap and Hydra differs greatly in the type of tactile feedback that is offered to the user. Tactile feedback is a subset of Haptics that focus on the direct feel of textures, touch, pressure etc., applied to the users fingers. This fundamental difference is haptic feedback and the more general depth-perception issues are estimated to be the major focus points of this study, and the technologies are compared as is from the manufacturers when pertaining to intended consumer use.

- RQ: What benefits and drawbacks are prevalent in the selected input technologies for performing positioning tasks in a virtual environment, with a focus on precision and efficiency?
- SQ1: How does the inherent change in haptic feedback from each technology influence the result?
- SQ2: How prevalent is the traditional depth-perception issue in conjunction with these new types of 6-DOF input devices?

2 Selected Technology

2.1 Depth-camera-based tracking

Figure 2: Leap Motion controller.

The Leap Motion sensor generated a lot of interest after its developer release in 2015, and its software is continuously being improved awaiting the consumer release of the product. When first confronted with the device it seemed so interesting and extremely well suited for intuitive computer interaction. This sparked a curiosity of how well it would compare to other technologies that seemed cumbersome in comparison.

The device is a small sensor that track infrared lights emitted to map the users hands in a set space approximately two cubic feet large. The device produce a 3-dimensional point cloud that is processed by the developer API (Application programming interface) to recognize hands and fingers. This enables a full rendition of the users hands with high precision, and fast response time. Research conducted by Frank Weichert for the University of Dortmund indicated an average accuracy of 0.7mm. This provides a superior accuracy compared to similar consumer products like the Microsoft Kinetic [7] and possibly makes the Leap controller better suited for detailed interactions. It is a relatively cheep solution but still present a few problems as Depth-camera tracking is prone to occlusion errors, and there is no guarantee that all fingers will be recognized even at close proximity to the sensor. If fingers are obscured by the palm, finger or another hand the reference is lost and can produce erratic behavior. This behavior also presents itself when fingers are close to each other and the camera cant clearly identify where one finger end and the other begins. The software is continuously revised to optimize the algorithms performance, and hopefully improve on some of its limitations. Camera-based tracking do suffer from a lack of haptic feedback which is a none-verbal communication between the computer and the user involving touch. This presence is hard to incorporate in virtual environment interactions where objects exists only inside the system, and the user manipulates them as intangible objects. There is no way for the user to get a tangible relationship with the objects in the scene, and this can lead to failure in the interaction loop as the user wont get the amount of feedback they are used to. As mentioned in section 1.2.1 this limitation can perhaps be mitigated by introducing other solutions in cooperation with the Leap controller [19].

Oshita, Senju and Morishige developed a control interface inspired by puppet mechanics using the Leap Motion sensor for the Kyushu Institute of Technology. The system utilized both hands of the user to emulate the interactions of a puppeteer and translated the gestures into movement on a computer model. When input is registered the models pelvis, hand and foot positions along with the head/body orientation is determined. The system then computes the models pose using inverse kinematics based on the constrains given by the user. This experiment was testes on a group of ten undergraduate and graduate students, where a portion of the test subjects where familiar with computer animation. Occlusion errors presented a problem for the subjects when utilizing both hands to accomplish complicated tasks. When conducting the same interactions on a puppet the subjects also experienced difficulty, but stated that is was more cumbersome using the system. This was thought to be a result of the errors in tracking and estimation. The lack of haptic feedback compared to the puppets physical presence made controlling the model more difficult. However the system provided the user with less restrictions than the strings of the physical puppet, and the hand could be controlled freely. The subjects required more training to get comfortable with the unstable recognition, and having to learn the range of recognizable hand movements. The systems viewpoint also differed from the real world counterpart. The viewpoint of the puppeteer is positioned above the puppet and obstructs the subject from seeing the results when manipulating the controls. On the system however the user is given a view directly in front of the model and can monitor the motions more freely, this perspective was favored by some of the subjects [24].

The tablet game Cut the Rope was used to compare touch input on a screen with mid-air gestures on a Leap controller in a study comprised of 20 children aged 11 to 14 years. Observations of the study reviled issues regarding accuracy when using the Leap controller compared to the touch screen that the game was originally developed for. They study concluded that a lack of hardware-based physical feedback may have been the deciding factor when interacting with the Leap based solution [25].

2.2 Tangible Input Controller

Figure 3: Razer Hydra controller.

The measurement degrees of freedom derived from mechanics is often used to convey the way an input device is able to operate and interact with a virtual environment. The following definition was proposed by G. N. Sandor and A. G. Erdman

> "By degrees of freedom we mean the number of independent inputs required to determine the position of all links of the mechanism with respect to ground" [26].

The classic mouse is a two degrees of freedom(DOF) input device, this corresponds to its cursor counterpart that is defined by two components of translation. This is sufficient when used on a 2-Dimensional canvas, like your computer screen, but is less than optimal in a 3-Dimensional environment as you lack the direct manipulation in the Z axis. A 3-Dimensional mouse on the other hand can have six degrees of freedom as it enables the user to manipulate both position and rotation in all three dimensions. The cursor can now be defined by three components of translation and three components of rotation.

The Razer Hydra controller is a six degrees of freedom device, but in comparison to camera based tracking the user manipulates a pointer in the scene with a tangible controller tethered to a base-station. The controllers reacts to a magnetic field generated by the base-station to determine the controllers position relative to the base-station. These tangible controllers offer the user superior feedback through haptic feedback. Each hand is assisted with eight separate buttons and a navigational control stick, enabling a plethora of additional interaction options. The Hydra is a licensed product utilizing technology developed by Sixence [27]. Tuukka Takala‘s work on the RUIS (Reality-based User Interface System) toolkit indicates that tangible controllers like the Razer Hydra are more suited than depth-camera based tracking (Here represented by the Microsoft Kinnect 2) for precise tasks where responsiveness and accuracy are vital to the results [28].

2.3 Head-Mounted Display (HMD)

Virtual environments have been explored for year, and an HMD is a natural and integral part of this field. This technology is prone to cybersickness (also refereed to as simulator sickness) [29] which is a problem that needs to be remedied in order to fully embrace this technology. An HMD is a complex combination of electronic, optical, mechanical and sometimes auditory components, that provide an immersing virtual experience. The quality of screen technology and graphical fidelity have increased significantly since the first generations of virtual environment devices, and today computing power have made it possible to create satisfactory graphical fidelity to fully immerse the user in a virtual environment. A study conducted at the Vanderbilt University compared the low-cost first generation Oculus Rift with a high-cost Nvis SX60. The study show that the Rift performed better or equal to its rival in multiple perception and selection tasks, including distance estimation and navigation [30]. Some Head-Mounted Displays offer a field of view as low as 30 to 60 degrees, this can cause multiple problems including distortion in perception of size and distance [31]. This project will utilize the Oculus Rift DK2 (Development Kit 2). This device offers a 100 degree field of view along with upgraded screen resolution and technology. Also the first generation only supported rotation in three axis, while the improved DK2 version also enables position tracking [32] giving the user freedom to manipulate the viewpoint independent of the interactions performed by the users.

Figure 4: Oculus Rift, virtual environment headset.

3 Method

In order to test the research question, three different methods was selected to gather data for further analysis. The selected methods are listed below, and described in the following sub-sections.

- User testing on a prototype testbed
- The system usability scale developed by Brooke
- Think-aloud protocol

3.1 Participants Selection

The adoption of virtual training software is no longer restricted to high-end technology companies or institutions, and as such the computer competence of end users may vary greatly. In order to get a wider view of the use of these devices the group of ten test-subjects was divided into two groups of five. The first group consisted of users with general computer experience, and little to none virtual environment experience. This novice group was contrasted with a group of computer experts working with software engineering and virtual environments on a daily basis. This diversity may show a difference in use-patterns between novice and experienced users based on their preconceptions and habits for interactions with a computer. Ages of Novice group participants ranged from 24 to 26, while the Expert group was more varied ranging from 26 on the low end all the way to 41 on the other. Both groups included only one female as a consequence of the male dominated profession, however this was not a goal.

All participants signed a consent form before the user testing was executed.

3.1.1 Participants Information

Each participant completes a form contain the relevant information related to the subject, and his self-proclaimed relevant experience as shown in figure 5.

3.2 Prototype Design

Data collected from the prototype will form the foundation of this study as it will log a set of parameters during the interaction tasks. These parameters are described closer in section 3.2.4, and will be used to represent the accuracy and efficiency that the users are able to achieve with each technology. The scenes are constructed to be as plain as possible with limited distractions apart from the task at hand. No visual aids like grids or numerical distance data is given to closely resemble a task performed in a virtual environment that is usually made to simulate a real world task. The task is solely relying on visual comparison of the two objects by the user.

Subject Information

Name: ....

Age: ....

Gender: ....

Experience with virtual or 3-Dimensional environment in a professional capacity?

(Yes/No)

....

Self-proclaimed proficiency with virtual or 3-Dimensional environment?

(Scale from 0 to 10)

....

Figure 5: Subject information form

3.2.1 Development Tools

For this project the Unity pro platform version 4.6.0f3 was chosen as the preferred development tool. The Unity community is large and users contribute solutions and experience even to newly released technology as the Leap, Hydra and Rift that is still under development[33]. Application Programming Interfaces (API) for all devices are already available for this platform making it easy to incorporate the technology.

The Unity engine comes bundled with the MonoDevelop 4.0.1 Integrated Development Environment (IDE) with multi-platform support and code completion for the C# programming language [34].

3.2.2 Third-party Application Programming Interface (API)

Each of the devices selected for this project required API‘s developed by the manufacturer in order to function. As the devices are under constant revision the following versions was available at the start of this project and have remained throughout its lifetime.

- Sixsence SDK 062612 (Razor Hydra)
- Leap Motion version 2.2.5+26752
- Oculus Rift version 0.5.0.1

The Unity engine handles the game-loop and game-object management internally, with relationships being defined in the editor or using Unity search methods. Most of the scripts used derive from the MonoBehaviour class [35], and is attached to a game-object. In order to assure that every scene was identical the same scene is used multiple times, only changing the setup of input and output devices. This ensures that all variables remain the same as it limits the possibility of human error during development.

3.2.3 Task

Selection in a 3-Dimensional scene is traditionally done by ray-casting techniques as explained by Bowman, Kruijff, LaViola and Poupyrev [36] where a ray is cast from the user that intersects with objects in the scene. Selection can then be determined as the first object hit or all objects colliding with the ray, as it can penetrate objects and travel indefinitely throughout the hole scene. In most cases the first and closest object is selected, but this selection technique can traverse all possible candidates if desired. This technique can also be used in virtual environments to enable selecting objects outside the range of the user, but the task presented here is done in a confined area within the users grasp. For the Razor Hydra controller "cursors" are created in the scene to represent the users hands inside the scene. Instead of casting a ray to select objects a rigid body collider is used to detect intersections of the pointers and the objects. The Leap on the other hand utilizes distance between objects in the scene to determine interactions.

The test-bed consists of three scenes with identical tasks configured for different technology combinations. The user is presented with two geometric primitives, a cube and a sphere located in the center of the scene, with an information panel in the background containing general feedback about the scene as shown in figure 6. To complete the task the user have to align the origins of both objects before performing a Boolean Difference operation on them. This Boolean Difference operation will subtract the sphere from the cube and provide visual feedback to the user as the result shape is updated to reflect the objects positioning. The user is free to update his result shape as many times as he feels necessary, and when satisfied commit to the positioning.

The first scene is a benchmark where the user performs the task using the traditional 2D-mouse on widgets connected to the objects which is well known from current 3D modeling tools. These widgets consists of three bars spanning from the objects origin and out in all three dimensions of the scene (X, Y and

Z). Clicking and dragging these bars will move the object in the corresponding direction, or the user can move the object freely in the X and Y directions mapping the movement of the mouse input by interacting with the objects mesh directly. This scene is completed with the traditional monitor output, while the remaining scenes utilizes the Oculus Rift head mounted display.

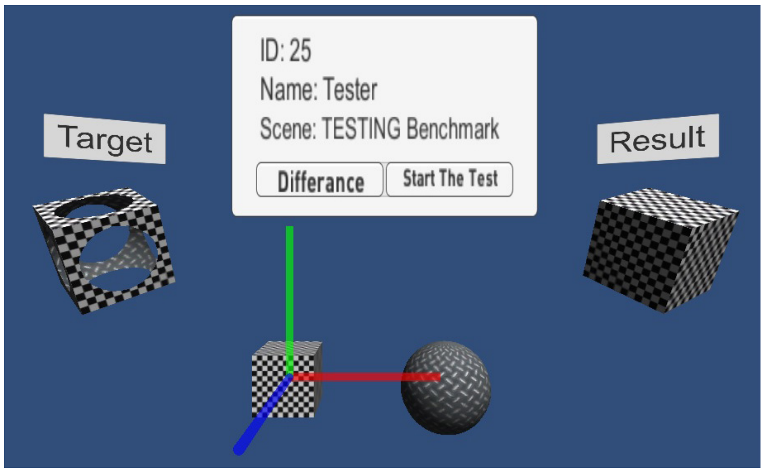


Figure 6: Benchmark Scene

The second scene enables the Leap-motion controller (Figure 7). The third-party API delivered from Leap Motion for the Unity engine defines a set of adjustable values for grabbing and releasing intractable objects. All distances are expressed in millimeters from the Leap motion API[37] and the pinch gesture utilized in this task is predefined to trigger when the tip of the thump and another finger are closer than a given distance from one-another. This pinch gesture will generate an anchor point between the thumb and the respective finger and attach the object closest to the anchor point. The object is now parented to the anchor point and follows the positioning of the user hand model representation in the scene. The release of the object is done when the distance between the finger and thumb exceed the same threshold as defined in the pinch gesture.

The third and final scene is for the Razor Hydra controller(Figure 8). In addition to the objects described above two pointers corresponding to each of the two Hydra controllers are visible in the scene. After a fast calibration these pointers are used to determine the collision meshes that enables interactions with the objects. The controllers triggers are used to initiate the interaction while buttons one and two on the head of the controller are dynamically changed for general purpose options (Yes, No, Commit). In order for the system to register an interaction one of the pointers must collide with the meshcollider of an object as the user initiates a grab with the corresponding trigger.

For each of the test scenes the subjects is allowed up to 5 minutes of test-

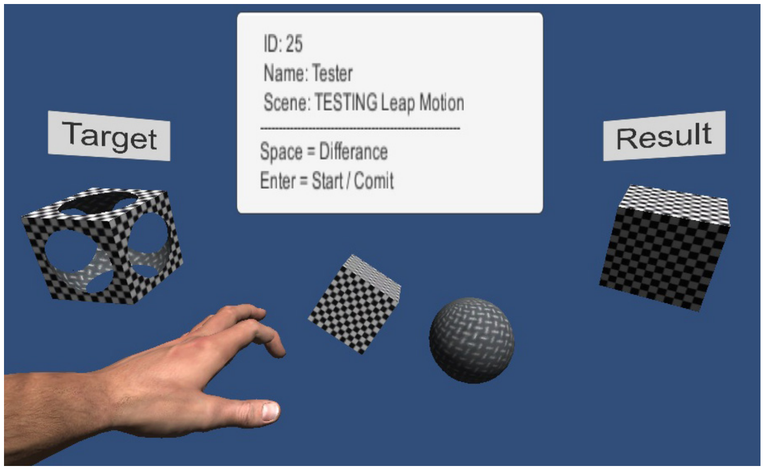


Figure 7: Leap Scene

Figure 8: Hydra Scene

ing to get familiar with the input devices, inspired by Bowman's selection and manipulation experiments [38]. This is believed to provide a more accurate representation of normal use of the technology as the initial confusion associated with new hardware can be limited.

3.2.4 Data Collection

For each interaction the system stores a set of parameters and write them out to a log at the end of a scene. The log-file is titled with ID, name and scene-name. The log is in the standard XML version 1.0 format, and each interaction is stored as follows.


```
<Interaction>
    <interactionCount>7</interactionCount>
    <InteractedObject>SphereMouse(Clone)</InteractedObject>
    <clickDownTime>9/9/2015 3:37:07 AM</clickDownTime>
    <clickUpTime>9/9/2015 3:37:07 AM</clickUpTime>
    <clickDurationTime>00:00:00.2160123</clickDurationTime>
    <objectsDistance>0.1088</objectsDistance>
    <cubeX>-0.8825</cubeX>
    <cubeY>0.7646</cubeY>
    <cubeZ>0.4000</cubeZ>
    <sphereX>-0.9083</sphereX>
    <sphereY>0.6775</sphereY>
    <sphereZ>0.4600</sphereZ>
    <direction>y</direction>
</Interaction>
```


At the end of the file a new type of node is created called summary. This Node contains generalized data about the overall task.


```
<Summary>
    <ID>25</ID>
    <Name>Tester</Name>
    <Experiance>Expert</Experiance>
    <InteractionsTotal>7</InteractionsTotal>
    <Started>9/9/2015 3:37:00 AM</Started>
    <Ended>9/9/2015 3:37:09 AM</Ended>
    <DurationTotal>00:00:09.1015206</DurationTotal>
    <InteractionsTimeTotal>00:00:03.4301962</InteractionsTimeTotal>
    <FinalDistanceTotal>0.1088079</FinalDistanceTotal>
    <cubeX>-0.882548</cubeX>
    <sphereX>-0.9083241</sphereX>
    <finalDistanceX>0.02577603</finalDistanceX>
    <cubeY>0.7645724</cubeY>
    <sphereZ>0.6775393</sphereZ>
    <finalDistanceY>0.08703309</finalDistanceY>
    <cubeZ>0.4</cubeZ>
    <sphereZ>0.46</sphereZ>
    <finalDistanceZ>0.06</finalDistanceZ>
</Summary>
```


Each axis is separated from the Vector3 object used in Unity to describe the objects position and extracted as a floating point numerical value with a four decimal precision. This allows precise comparison between each axis to find out any difference in the Z-axis (Depth) compared to the X and Y-axis.

3.3 System Usability Scale

In order to calculate the users preference of input technology the System usability scale (SUS) is administered directly after the user finish a scene. The SUS was developed by Brooke [39] and have since been a well used and reliable tool for measuring system usability. Bangor, Kortum and Miller conducted a study on data collects from this usability tool spanning almost ten years. They concluded that the SUS was "a highly robust and versatile tool for usability professionals" [40]. The SUS present the user with a simple ten-item questioner using the Likert scale to estimate the users subjective attitude towards the usability of the system as seen in figure 9

Brooke recommends administering the SUS to the user directly after the user have tested the system in question, and before any further discussions take place. The user should not be given ample time to think about their response, but rather record their intuitive response. If the user fails to answer a question the middle mark of the scale should be used.

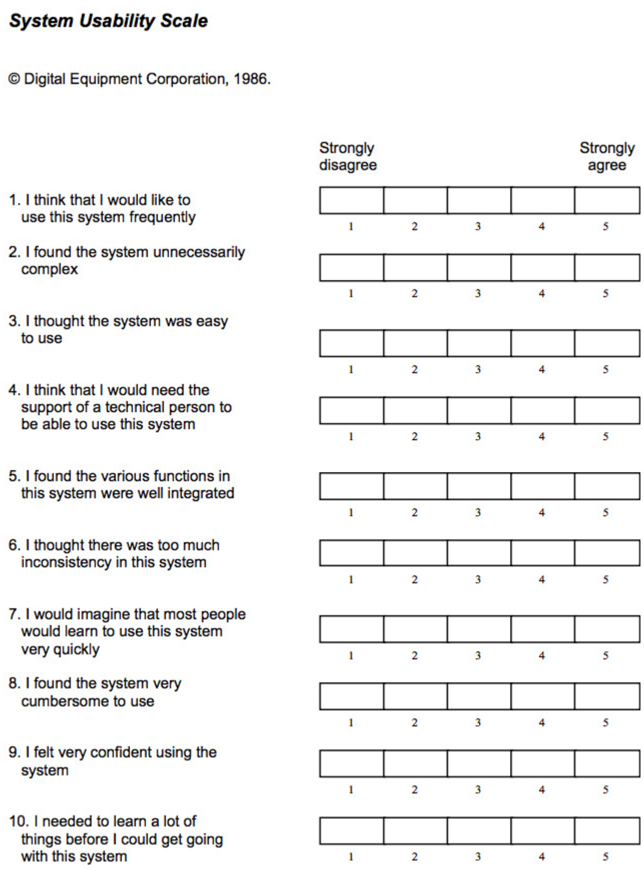
The SUS is scored on a 0 to 100 scale, with 100 being the maximum score. Individual answers are not important on their own, only the combination of all answers are of interest. Brooke explains the calculations of scoring the SUS as "Each item's score contribution will range from 0 to 4. For items 1,3,5,7,and 9 the score contribution is the scale position minus 1. For items 2,4,6,8 and 10, the contribution is 5 minus the scale position. Multiply the sum of the scores by 2.5 to obtain the overall value of the usability [39].

3.4 Think-aloud protocol (TA)

The think-aloud protocol is a widely used tool when conducting usability tests. The TA protocol enables the participants to articulate their thinking as the observer has a hard time knowing what is going on inside the head of each participant [41]. Difficulties related to the technologies in question are hopefully easier to detect in conjunction with other data when the participants are allowed to share their perspective.

3.5 Interview

After the subject completes the system usability scale any observations or interesting comments that emerge during testing are discussed. This is done to ensure that any points unintentionally left out by the subject can be examined.



System Usability Scale

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

Figure 9: System Usability Scale (SUS)

4 Evaluation

This section aims to provide a detailed description of the user-testing session. Technology setup and procedure is divided into two separate subsections.

4.1 Technology Setup

The testbed is presented to the user on a computer running the Windows operating system. The table will only contain the devices pertaining to the experiment (Mouse, Keyboard, Leap, Hydra and Rift).

In order to capture the session for later evaluation the entire session will be video and audio recorded. The camera is positioned on a tripod behind the user, providing an over-the-shoulder perspective. This allows the camera to record all movement within the users reach over the table surface, as well as the result on the computers monitor. This will aid in the later work especially when dissecting the results of the Think-aloud protocol.

4.2 Procedure

Before the session starts all devices must be accurately set up, and tested to ensure they work properly. New documents are prepared for the next session according the the following pre-session checklist.

- Computer/device setup
- Camera setup
- Test all devices
- Documentation (consent, 3x SUS, experience form)

After the initial introduction the subject will be informed about the evaluation process and the goal of the project before signing the for of consent. The subject is explicitly informed that the focus of the experiment is on the performance of the devices, and not the subjects results. The user will then fill inn a short form to record all relevant information about the subject as well as experience with virtual and 3D environments and interaction techniques according to the pre-experiment checklist.

- Signing consent.
- Project information
- Project focus
- Gather subject information

The experiment process follows according to the experiment checklist below.

- Train Benchmark scene

- Perform Benchmark scene
- SUS Benchmark scene
- Train Leap Scene
- Perform Leap Scene
- SUS Leap Scene
- Train Hydra Scene
- Perform Hydra Scene
- SUS Hydra Scene

Post session the data is backed up to ensure no loss of data can occur.

- File documentation
- Create backup of recording
- Create backup of log data

After the subject has filled inn the SUS questioner, a short post interview is conducted to discuss any observations done during the task. The goal of this step is to get any additional information that the user may have forgot to mention during the think-aloud portion of the task, as some of the users might not be to vocal in their assessment of their experiences.

5 Results

This section will iterate through the different data gathered during the study. The two first sections will aim to describe the pertinent data from the system logs and system usability scale questioner, while the following three sections pertain to the data gathered from observation, interviews and think-aloud protocols collected during testing.

The Leap controller required a large amount of bandwidth in order to work properly, and precautions where taken to provide optimal testing condition. The original USB 3.0 ports did not manage to transfer data on an acceptable rate so a separate PCI USB 3.0 card was needed. Even with this contingency problems related to transfer rate still emerged. When working under optimal conditions on the other hand responsiveness did not seem to be an issue, and seeing as the technology is still in development these inconsistencies are not being considered as the main problem.

The testbed originally included an iteration using the standard mouse and keyboard setup which only examined the positioning capabilities of this input technology, and during testing it was obvious that this was not comparable to the the position and rotation capabilities of the leap and hydra. Due to this limitation in the testbed results from the mouse and keyboard part of the user testing is excluded from the study. This does not effect the original research question as it only focus on the new types of input technologies for virtual environments. This was originally implemented as an extra feature in case something stood out in the test results, and could possibly contribute to any opinions regarding tangible input technologies.

5.1 Testbed Logs

The data-set logged from the testbed consists of two separate XML files. The interactions data consists of 843 interaction observations generated by all users in both groups, also including the mouse and keyboard data. The task-summaries consists of 90 task observations comprising of 3 task attempts for each of the 3 technologies for all 10 users.

When comparing the final distance achieved with both technologies, over the full data-set of experts and novice users we find a significant difference between the two. The Hydra controller manages a median of 0.0485 which was a noticeable improvement compared to the Leap controllers Median of 0.0789. The range of observations for the Leap controller also fluctuates more than its Hydra counterpart as shown in Figure 10 indicating a more consistent use pattern.

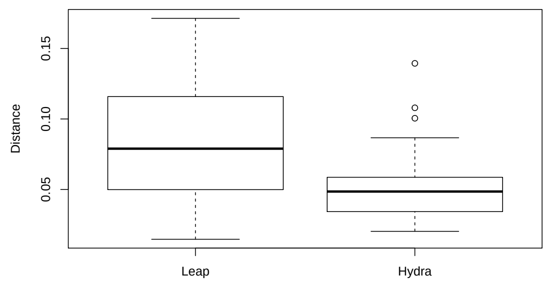


Figure 10: final distance

This trend continues when we take a look at the task duration, interaction time and interaction count. The task duration (Figure 11) on the Leap controller ranged all the way from 10.2 to 207.3 in the upper extremes, with a mean of 73.7. While the Hydra managed a range from 11.9 to 90.7, averaging a 35.1.

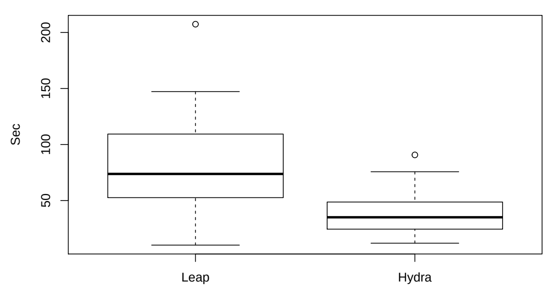


Figure 11: Task duration

The durations logged per interaction with the Leap controller averaged 38.6 compared to the Hydras 13.7 as seen in figure 12.

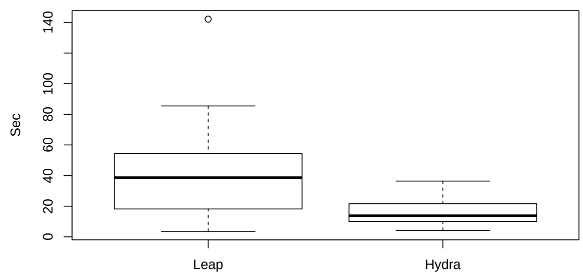


Figure 12: Interaction duration

The number of interaction (Figure 13) also suffered with the Leap controller logging 38 on the high end with an average of 12, compared to the Hydra's average of 4.

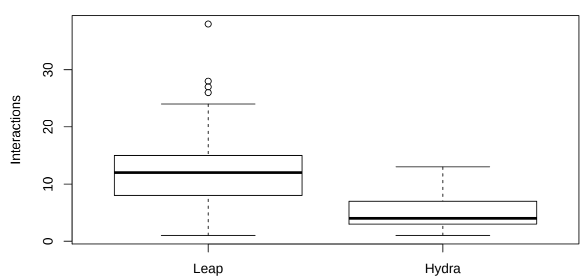


Figure 13: Interaction count

5.2 System Usability Scale

Results of the system usability scale (SUS) questioner also show an affinity for the tactile Hydra controller. It is tempting to interpret the questioner on a point by point basis, but Brook stated that the system usability scale is to be used as an accumulation of the entire questioner.

> "SUS yields a single number representing a composite measure of the overall usability of the system being studied. Note that scores for individual items are not meaningful on their own." [39]

Therefore the individual items are not discussed at greater length, but rather interpreted as the users general preference to try and catch any discrepancies between technologies and/or groups.

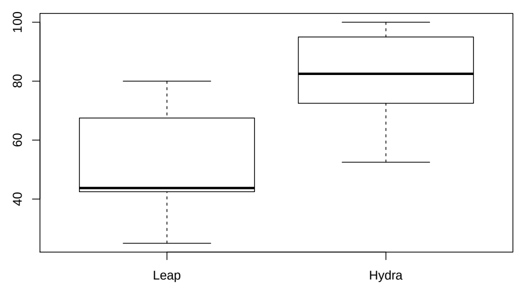


Figure 14: System Usability Scale for both groups

The Leap scored an mean of 43.7 when combining both group, while the Hydra fared much better with a mean of 82.5 (Higher is better).

In contrast to the log data variables, the SUS scores differed more between the groups. Although the Hydra controller was a clear overall winner, a disparity emerged between the two groups on the leap controller results as shown in figure 15.

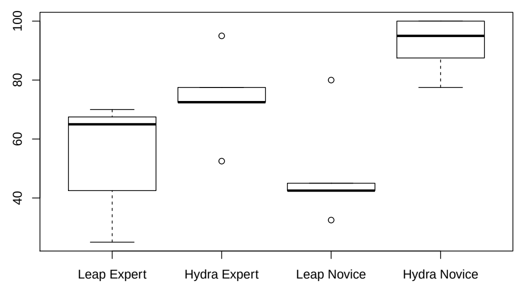


Figure 15: System Usability Scale divided by groups

While the expert group ranged more in their opinion of the Leap controller from 25 on the low end to 70 with a mean of 65, the novice group where situated in the lower spectrum of the scale bottoming out on 32.5 whit a mean of 42.5. The novice group drove down the overall score of the Leap, while the Experts seemed more intrigued by the technology and allowed more leeway in their initial impressions of the technology. The Hydra scored more consistently in the top spectrum of the scale with a mean of 72.5 from the experts to 95 with the novice users.

5.3 Axis difference

The final distance across all axis does not indicate a noticeable difference to support any evidence of depth-estimation problems during testing. The Leap controllers media ranges from 0.038 to 0.032 which is less than noticeable and well within acceptable limits.

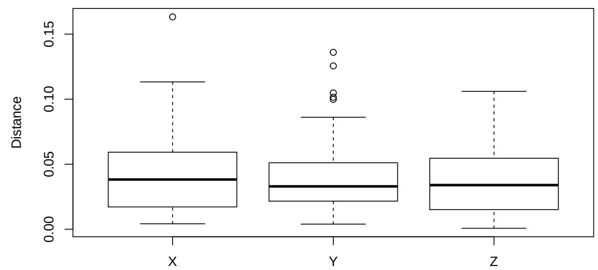


Figure 16: Leap Axis difference for both groups

The Hydra did show a greater discrepancy but not in the Z-Axis. The axis that stand out here is the vertical Y-Axis with a median of 0.027, and even here the difference is not great enough to warrant concerns. The remaining X and Z-Axis fluctuates between 0.016 and 0.019.

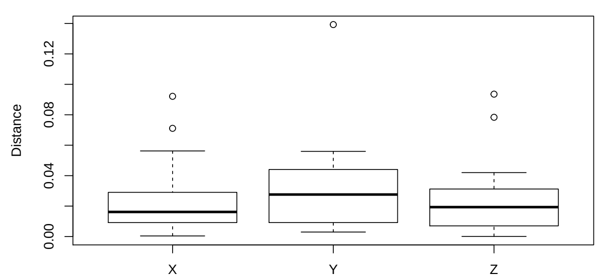


Figure 17: Hydra Axis difference for both groups

5.4 Think-Aloud, Interview and Observations

Initially most of the users expressed some excitement to the intuitive nature of the Leap motion controller. The subjects perception on its responsiveness ranged from "very responsive" to "feels imprecise" [Appendix D], and this may have some connection to the high bandwidth requirements inherent in the technology. The shadowing issue where fingers where not recognised due to being obscured by other fingers did not hinder most of the users. It occasionally complicated the work-flow of the users, but they quickly understood how to limit the issue. Some of the subjects where fascinated with the full hand rendering inside the simulation, but their attitude towards the technology soon changed in a negative direction. As intuitive and fun as the camera based tracking and live hand modeling where, the constraints posed by clumsy gesture control soon outweighed the positives when trying to accomplish the task at hand.

Due to the different ways each technology handled a grab, there where comments on the Leap controllers difficulties to pick the wanted object when both objects where in close proximity of each other. As described in section 3.2.3 the Leap does a distance check to determine its grab target while the Hydra operates on a bounding box collision, making it seem a lot more precise in the selection phase as well. After transitioning to the Hydra controller the majority felt a vast improvement, and exhibited great enthusiasm for the Hydra controller.

> "Very easy to use, precision is ...wow" - Magnus

A reoccurring theme emerged early as almost all of the test subjects identified a difficulty to release the object at the desired point using the Leap controller. When the user seemed content with its positioning and started releasing the object it would follow the index finger on its way out until the distance threshold was met. The result of this was an increased difficulty to place the object and therefor reduced the final result, as many of the subjects remarked.

Figure 18: Leap test: Jostein

> "Moving the objects around goes fine. but releasing it is hard." - Jostein

> "Seems to follow the index finger wen released." - Tom-Robert

Selecting objects using the Leap also presented some minor issues. The distance from the generated pickup point between the users fingers to the origin of the closest object which where positioned in the center of the object could be hard to manage when the objects overlapped. This resulted in grabbing the sphere when trying to manipulate the cube. The collision detection approach whit the Hydra did not seem to have the same problem as the user could more easily latch on to parts of the object that protruded outside the obscuring object.

Some of the test subjects noted that two-handed manipulation was intriguing, but most of the users did reverted back to only utilizing one of their hands to perform interactions. Some of the users that tried using both hands commented on complications occurring due to their hands colliding in the real world when trying to overlap objects. The Hydra cursors inside the simulation protruded a bit from their real world counterparts, and as such did not exhibit the same problem.

When transitioning from the Leap to the Hydra controller most of the users seemed relieved and more impressed with both responsiveness and ease of use. Even with such a favorable impression some issues also emerged during the hydra portion of the test. The Hydra controller required the user to position the controllers relative to the base and initiate a calibration. This slightly increased the adoption difficulty, and could result in unfavorable working postures for some users. The Hydra base also seemed to produce some erratic behavior when the user accidentally came to close to it during use. The Leap controller on the other hand where devoid of this problem as it operated in a fixed area in front of the user. In a none virtual setting the user can easily locate the base and understand its reach, but locating this area required some of the users to wave their hands in order to establish a connection with the device.

6 Discussion

In cooperation with the institute for energy technology (IFE), This study was an attempt to try and identify the inherent advantages with some of the new and promising interaction technologies available on the market. The field of camera based tracking had been explored internally at IFE using the Microsoft Kinnect, and interest in the Leap Motion controller where high. Even tho both technologies incorporate the similar kinds of tracking technologies, the Leap emphasise on a smaller working area and is arguably more suited for small and precise tracking than that of the Kinnect's larger full body scanning capabilities. In order to complement the Leap with a worthy opponent the Razor Hydra was included as a tangible 6 degrees of freedom input device. When deciding on what to focus on to compare these two technologies precision and efficiency where chosen as the point of interest when it came to detailed interaction tasks, and the differences in tactile feedback provided by the devices which inspired the first sub-question.

6.1 SQ1

- How does the inherent change in haptic feedback from each technology influence the result?

The Hydra controller offered a plethora of buttons to map actions to, as well as two trigger-buttons with a travel closely resembling the movement needed to release the object with the Leap API gesture. Even tho the grab on the Hydra controller was mapped to these trigger-buttons to complement the gesture travel of the users fingers, having a tangible controller to base the movement from apparently improved the result of the grab and release action. The tactile resistance feedback provide a granular control not present in a free hand gesture as the camera based tracking technology offers. Several of the subjects commented negatively on the complications of releasing the object where intended.

> "Object seems to jump when releasing." - Magnus

The difference in feedback between the technologies in this study indicates that the haptic pressure provided by the Hydra controller was a one of the prominent factor in improving the results of the task. Pressure actuators applied to the users fingertips as proposed by Stefano Scheggi [17] may improve the Leap controllers release accuracy aspect in detailed interactions by releasing the pressure on the given fingers corresponding to the distance variable in the LEAP API.

6.2 SQ2

- SQ2: How prevalent is the traditional depth-perception issue in conjunction with these new types of 6-DOF input devices?

Related work in the field of computer interactions indicated significant issues with depth-perception when interacting with objects in 3-Dimensional space. To try and examine if this issue would also be prevalent in a virtual environment task the second sub question was formed to put an emphasise on this its contribution to the results.

The result of this study did however not uncover any noticeable disparity in the Z-Axis results pertaining to the final distance. The overall result where marginally better with the Hydra controller averaging from 0.016 to 0.027 across the axis, while the Leap controller varied from 0.032 to 0.038. The only disparity in accuracy across the axis was provided by the Hydra controller where results show a small decrease in precision on the Y-axis. The findings in this study pertaining to the depth-perception issues are not decisive enough to make any substantial observation on the subject other than concluding that in this setup depth-perception probably did not have much influence on the test results. Comparing this study to the work of Christine J. Ziemer [20] there was some fundamental differences in the task that may have been a prominent factor in explaining the different results. This study focus entirely on interactions inside a virtual environment, while Ziemer‘s results where gathered from a virtual to real-world scenario. Secondly the ranges compared in Ziemer‘s work ranged from 20 to 70 ft in one configuration and 20 to 120 ft in the second configuration. The range of the task performed in this study however only covered the extent of the users reach.

6.3 RQ

- RQ: What benefits and drawbacks are prevalent in the selected input technologies for performing positioning tasks in a virtual environment, with a focus on precision and efficiency?

Both technologies provided precise and fluid tracking when working as intended, some issues occurred as expected from technology currently under development, but they where minor. Prior to the start of this project hopes where high for both of the selected input technologies, but during development of the testbed most of the issues discussed later in this section started to emerge. When comparing the results with a focus on precision the mean final distance almost doubled with the Leap controller, while averaging 38.6 seconds longer per task than its competitor. Whit a 24.9 seconds higher average interaction-time while simultaneously requiring 26 more interactions per task, the Hydra also proved superior when it came to efficiency.

6.3.1 Pinch Gesture Drawback

When examination the results of the testbed logs in conjunction with observations of the user sessions two prevailing drawbacks of the Leap controller emerged. The first problem seems to reside in the software portion of the device, more precisely in the definition of the pinch gesture. The Leap API calculates the distance between the thumb and index fingers and initiates a pinch when the distance decrease under a set amount. When pinched the object is anchored to a point between the thumb and index fingers and so far this approach works well. The problem occurs when the user tries to release the object as a majority of the users noted during the think-aloud process and the post testing interview. As the anchor point fluctuates when the user starts to move their index finger away from the thumb the object follows suit. Even tho most of the users managed to identify the problem, correcting it appeared a lot harder and made obtaining the same level of precision as the Hydra controller difficult. This pinch gesture part of the software is not present in the Hydra controller as it is able to map the most needed actions to the supplied buttons and circumvent this particular problem.

6.3.2 Benefit of Haptic Feedback

Secondly the missing haptic feedback component as expected with the first sub-question appear to be a contributing factor to the Leap controller gesture problem, and including this feature of haptic feedback in conjunction with the visual feedback as a proponent for multimodal feedback Adam Faeth [16] suggests would most likely further improve task results. These two vastly different types of input technologies provide a very different approach to virtual interactions. The leap focuses on the intuitive and untethered input approach confined to a smaller area of operation, it offers the user a natural way of interacting with objects similar to the real world equivalent of grasp and object. This intuitive nature seemed beneficial to the technology as it appeared easy to understand and adopt, as most of the user had no problems understanding how to perform a pinch gesture. But the drawbacks of this controller started to emerge when precision and efficiency where taken into consideration.

Moser‘s work whit physics-based games coupled with the Leap controller [25] show that even with a system adapted for interactions using mid-air gestures, in this case the tablet game Cut the Rope, show that a lack of haptic feedback left a lot more importance on the visual feedback and suffered for it. The double interaction loop proposed by Hiroshi Ishii [10] when working with tangible input controllers separate the passive haptic feedback and the corresponding digital feedback provided by the system. In the haptic feedback loop users are not dependant on the computer to process and provide feedback. The user can perform its intended actions where as the digital loop is more reliant on the response of the computer to continue an action and therefor prolonging it. The findings of this study corroborates Ishii‘s proposed advantage of immediate tactile feedback as shown with the total lack of it in the Leap controller. When working as intended the reaction time of the Leap controller was fluid and did not induce much jitter that would negatively influence the digital feedback loop to unacceptable limits. But the total lack of a passive haptic feedback loop coupled with the logic of the pinch gesture from the Leap API seemed to negatively influenced the accuracy and therefor also the efficiency obtained by the users. This contributed to the users disdain for the Leap controller as reflected in the overwhelmingly disappointing SUS scores averaging in the low 40s, compared to the Hydras 82.5. The methods of issuing commands to the computer that the Leap utilizes, referring to the predefined gestures that seems lacking as it is currently implemented.

The Hydra on the other hand incorporates this passive feedback loop in the trigger action as the user gets pressure feedback from the spring that enables the user to more precisely estimate when an action occurs. Mapping the action to a physical button also eliminates the need for more complex logic in the grab and release phases of the interaction, completely avoiding one of its competitors main weaknesses. This tactile feedback provided by the Hydra‘s trigger button appeared to be a major benefit and gave the users enough resistance during the release of an object which greatly improved both the precision and efficiency aspect of the task.

6.3.3 Compounding Issues

Even tho the results greatly favored the tangible Hydra controller in regards to the criteria set for this study, the Leap controller out shined its competitor when it came to usability. In hindsight more emphasis could have been put on this aspect of the technologies as it can be important when selecting the correct input technology to use in future projects. These benefits and drawbacks does however not directly influence the precision and efficiency that where the focus point of this study.

The Hydra controller consist of a base station tethered to the computer with two cord connected controllers. This required more of a complex setup procedure then that of the Leap controller, as it only required one connection to get up and running with a much smaller footprint on the table. Once setup the Hydra requires the user to manually calibrate it by holding the controllers in a fixed position. This was completely avoided by the Leap controller as it was ready to use once plugged in and placed on the table.

Locating the Leap controllers field of view on the other hand did result in the users waving their hands in order to locate it. Once the user established a connection to the devise they where mostly up and running. Moving outside this area would result in loosing the virtual hand representation as the Leap was configured to only show hands visible to the camera. The user would then have to locate the field of view and reestablish control. The cursor representations of the hydra controller on the other hand where always visible to the user and corresponded to its physical counterpart keeping the connection with the user intact and always responding to user input.

As both technologies are still in development some issues occurred during testing that appeared out of the controls of the testbed design. The Hydra controller utilizes magnetic fields to track its controllers, and interference is a possibility. During testing erratic behavior only occurred once and the reason for this is hard to pinpoint. A reset of the scene and software was needed to remedy the problem. The Leap controller on the other hand seems to require huge bandwidth in order to get the data from the controller and into the computer.

The shadowing problems where surprisingly not that influential to the result, as the users quickly understood how to avoid the issue and work correctly so as to not be overly affected by it.

7 Conclusion

At first glance the results of this study greatly favors the Hydra technology pertaining to the aspects emphasized in this study. The difference in final distance might not seem so dominating, but combined with other factors like interaction count and duration we start to see that the users spend a lot more time and effort obtaining their goals.

Both technologies offer very precise and fluid tracking when working as intended, but the drawbacks present in the software layer of the Leap controller combined with its lack of haptic feedback pose a problem for the Leap controller that seems inherent to the camera-based tracking input technologies. The wast amount of data needed to transfer from the controller to the computer could pose some limitation as the test computer was far superior to the normal workstation computers on the market. The gestures defined in the Leap API posed severe limitations when it came to precision which also negatively effected the efficiency as these two variables are closely linked.

8 Future Work

Compared to a tangible controller with customizable buttons, a system comprised only of camera-based tracking seems required to implement other solutions for simple actions like a grab or a click. As the users of this study did not effectively utilize the benefits of two-handed interactions it appears more suitable to utilize the none dominant hand to issue commands directly to the system by pairing the Leap with some other kinds of input technology.

9 Acknowledgements

The author would like to thank Harald Holone at Østfold University Collage for his guidance and support throughout this project, and IFE for supplying the required devices as well as technical assistance whenever needed.

SUS was developed as part of the usability engineering programme in integrated office systems development at Digital Equipment Co Ltd., Reading, United Kingdom.

References

[1] *Leap Motion*. URL: https://www.leapmotion.com/.

[2] *Microsoft Kinect*. URL: https://www.microsoft.com/en-us/kinectforwindows/.

[3] *Nintendo Wii*. URL: http://wii.com/.

[4] KEN HINCKLEY et al. "Passive Real-World Interface Props for Neurosurgical Visualization". In: *Association for Computing Machinery* (1994).

[5] David Navarre et al. "A Formal Description of Multimodal Interaction Techniques for Immersive Virtual Reality Applications". In: *Association for Computing Machinery* (2005).

[6] Marc Erich Latoschik. "A User Interface Framework for Multimodal VR Interactions". In: *Association for Computing Machinery* (2005).

[7] Frank Weichert et al. *Analysis of the Accuracy and Robustness of the Leap Motion Controller*. URL: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3690061/.

[8] Stuart K. Card, Thomas P. Moran, and Allen Newell. *The Psychology of Human-Computer Interaction*. 1983.

[9] Richard A. Bolt. "Put-That-There: Voice and Gesture at the Graphics Interface". In: *Association for Computing Machinery* (1980).

[10] Hiroshi Ishii. "Tangible Bits: Beyond Pixels". In: *Association for Computing Machinery* (2008).

[11] Hiroshi Ishii. "THE TANGIBLE USER INTERFACE AND ITS EVOLUTION". In: *Association for Computing Machinery* (2008).

[12] George W. Fitzmaurice, Hiroshi Ishii, and William Buxton. "Bricks: Laying the Foundations for Graspable User Interfaces". In: *Association for Computing Machinery* (1995).

[13] Marie-Luce Bourguet. *Designing and Prototyping Multimodal Commands*. URL: http://www.researchgate.net/publication/221054248_Designing_and_Prototyping_Multimodal_Commands.

[14] Ying Zhang, Reza Sotudeh, and Terrence Fernando. "The Use of Visual and Auditory Feedback for Assembly Task Performance in a Virtual Environment". In: *Association for Computing Machinery* (2005).

[15] Oviatt S.L. "Multimodal interactive maps: Designing for human performance. In: Human-Computer Interaction". In: *Association for Computing Machinery* (1997).

[16] Adam Faeth and Chris Harding. "Effects of Modality on Virtual Button Motion and Performance". In: *Association for Computing Machinery* (2012).

[17] Stefano Scheggi et al. "Touch the virtual reality: using the Leap Motion controller for hand tracking and wearable tactile devices for immersive haptic rendering". In: *Association for Computing Machinery* (2015).

[18] Mounia Ziat et al. "Enhancing Virtual Immersion through Tactile Feedback". In: *Association for Computing Machinery* (2014).

[19] Stefano Scheggi et al. "Wearable Haptics and Hand Tracking via an RGB-D Camera for Immersive Tactile Experiences". In: *Association for Computing Machinery* (2014).

[20] Christine J. Ziemer et al. *Estimating distance in real and virtual environments: Does order make a difference?* URL: `http://psychology.uiowa.edu/files/psychology/groups/hank/files/DistancePerceptAPP.pdf`.

[21] Atsushi Sugiura, Masahiro Toyoura, and Xiaoyang Mao. "A Natural Click Interface for AR Systems with a Single Camera". In: *Association for Computing Machinery* (2014).

[22] William Buxton and Brad A. Myers. "A STUDY IN TWO-HANDED INPUT". In: *Association for Computing Machinery* (1986).

[23] George W. Fitzmaurice and William Buxton. "An Empirical Evaluation of Graspable User Interfaces: towards specialized, space-multiplexed input". In: *Association for Computing Machinery* (1997).

[24] Masaki Oshita. "Character Motion Control Interface with Hand Manipulation Inspired by Puppet Mechanism". In: *Association for Computing Machinery* (2013), pp. 134–136.

[25] Christiane Moser. "Physics-based Gaming: Exploring Touch vs. Mid-Air Gesture Input". In: *Association for Computing Machinery* (2015).

[26] Sandor G.N. and Erdman A.G. *Advanced Mechanism Design, vol.I and II*. Prentice-Hall Inc, 1984, p. 16.

[27] *Sixence*. URL: `http://sixense.com/`.

[28] Tuukka M. Takala. "RUIS – A Toolkit for Developing Virtual Reality Applications with Spatial Interaction". In: *Association for Computing Machinery* (2014).

[29] Norman G. Vinson et al. "Cybersickness Induced by Desktop Virtual Reality". In: *Association for Computing Machinery* (2012).

[30] Mary K. Young et al. "A Comparison of Two Cost-Differentiated Virtual Reality Systems for Perception and Action Tasks". In: *Association for Computing Machinery* (2014).

[31] Doug A. Bowman et al. *3D User Interfaces: Theory and Practice*. Addison-Wesley, 2004, pp. 31–52. ISBN: 078-5342758672.

[32] *Oculus Rift*. URL: `http://www.oculus.com/dk2/`.

[33] *Unity Engine*. URL: `http://unity3d.com`.

[34] *MonoDevelop*. URL: `http://www.monodevelop.com`.

[35] *MonoBehaviour Documentation*. URL: `http://docs.unity3d.com/ScriptReference/MonoBehaviour.html`.

[36] Doug A. Bowman et al. *User Interfaces Theory and Practice*. Pearson Education, Inc., 2004, p. 151.

[37] *Leap Motion*. URL: `https://developer.leapmotion.com/getting-started/javascript/developer-guide#units`.

[38] Doug A. Bowman, Donald B. Johnson, and Larry F. Hodges. "Testbed Evaluation of Virtual Environment Interaction Techniques". In: *Association for Computing Machinery* (1999).

[39] John Brooke. *SUS - A quick and dirty usability scale*. URL: `http://hell.meiert.org/core/pdf/sus.pdf`.

[40] Aaron Bangora, Philip T. Kortumb, and James T. Miller. *An Empirical Evaluation of the System Usability Scale*. URL: `http://www.tandfonline.com/doi/full/10.1080/10447310802205776#abstract`.

[41] Erica L. Olmsted-Hawala et al. "Think-Aloud Protocols: A Comparison of Three Think-Aloud Protocols for use in Testing Data-Dissemination Web Sites for Usability". In: *Association for Computing Machinery* (2010).

A
Think-Aloud, interview, Observations

- Andre
  - Think-Aloud (Leap)
    - Does not release when i want to.
    - Feels like it goes a little better on the second attempt.
    - Hard to select the object when your are on top of each other.
    - The Object moves right after release.
  - Think-Aloud (Hydra)
    - Feels a lot more in control now, compared to the leap controller.
  - Interview
    - Subject felt he had to try multiple times to initiate a grab.
    - Did not experience significant fatigue during testing, but could see it becoming an issue during prolonged sessions.
    - Rotation and movements was a lot easier on the Hydra. Compared to some of the jittering on the Leap due to poor tracking outside its preferred field of view.
  - Observations
    - Tend to place the objects high above the field of view for the leap controller.
    - Working primarily with the dominant hand.
- Gina
  - Think-Aloud (Leap)
    - No problems grabbing an object, but hard to place it.
  - Interview
    - The user preferred to use her non dominant hand due to moving the object to the right.
    - The Hydra was easier to use due to problems when releasing an object with the Leap controller.
  - Observations
    - Commits due to lack of confidence that she can obtain a better result with the Leap controller.
    - Subject seems to primarily be using the left hand (None dominant).
    - Moves out of the field of view of the Leap controller.

- Jostein
  - Think-Aloud (Leap)
    - Feels imprecise.
    - The fingers are holding on to the object longer than what was preferred.
    - Feels more in control during subsequent tries.
    - Not really smooth.
    - Hard to find the magical distance where the pinch gesture ends.
    - Moving the objects around goes fine. but releasing it is hard.
    - Object jumps into the simulated fingers.
    - The user likes the fact that he can see his own hands inside the simulation.
  - Interview
    - Did not enjoy the pinch gesture of the Leap controller at all.
  - Observations
    - Commits due to lack of confidence that he can obtain a better result with the Leap controller.
    - Tends to move a bit out of the field of view of the Leap controller.
    - Didn't say anything during the Hydra test, because it didn't present any big issues and he could easily focus on the task at hand.
- Magnus
  - Think-Aloud (Leap)
    - Very responsive.
    - Experiencing fatigue when not able to rest the arm on something.
    - Object seems to jump when releasing.
    - Can unintentionally grab the wrong object.
    - No problems lining up the objects, but releasing is a problem.
    - Releasing slowly seems to help.
  - Think-Aloud (Hydra)
    - A lot more responsive, and precise.
    - Very easy to use, precision is ...wow.
  - Interview
    - Leap offered a little inconsistency.
    - The Hydra felt more intuitive to pick up.
    - the mapping of the cursors to the Hydra controller enabled a lot more precision with pinpoint accuracy
  - Observations

    * Did learn a way to compensate for the release problem by slowly releasing the object.
    * Feels ready to start with little training.
    * Speeds through the Hydra testing.

- Ole
  - Think-Aloud (Leap)
    * Late to release.
    * Grabbing the wrong object.
    * Fun to use both hands, but feels a loss in the finer movement.
    * Not so hard to see where the ads are relative to the other objects when you get a feel for it.
  - Think-Aloud (Hydra)
    * Jittering issues on the coursers.
    * Easier to move the sphere than the square.
  - Interview
    * The user felt like big movements where needed to do manipulation on the Leap controller.
  - Observations
    * The subjects is having a hard time locating the Leap controllers field of view due to a far back chair position.
    * Problems recalibrating the Hydra controller. This led to a high working position.
    * Had to reset the Hydra controller due to jittering with the cursors. Likely a disturbance with the magnetic tracking.

- Tom-Robert
  - Think-Aloud (Leap)
    * Hard to distinguish what object your selecting when their close in proximity.
    * Seems to follow the index finger wen released.
    * Movement in the release of an objects makes placement difficult.
    * The ability to use both hands is an advantage.
  - Think-Aloud (Hydra)
    * More buttons to keep track of.
    * Feels more precise.
    * Missing granular axes control as with the standard mouse interactions.
    * Tracing distorts when close to the base.

    - lesser motion sickness than normal for the user, possibly from a static task.
  - Interview
    - High precision tasks don't seem suited for this technology.
    - Did experience fatigue.
    - Enjoyed using it, just difficult.
    - Commented on the limitations of camera based tracking i.e shadowing
    - Was intuitive and easy to use.
- Thomas
  - Think-Aloud (Leap)
    - "Cool" to see your hands inside the simulation.
    - What object to select can become an issue.
    - You can faster crudely position the objects. The problems seems to be in the finger tracking.
    - More intuitive.
    - Experienced problems in tracking closer to the base, then overcompensated and moved the objects to far from the base.
    - Releasing the objects moves it.
    - Experiencing fatigue in the arms after a relatively short time period.
    - Hands crash if trying to manipulate both objects in close proximity.
    - High precision tasks don't seem suited for this technology.
    - Did experience fatigue.
  - Think-Aloud (Hydra)
    - Easier to release the object.
    - More control
  - Interview
    - High precision tasks don't seem suited for this technology.
    - Did experience fatigue.
    - Enjoyed using it, just difficult.
    - The limitations of camera based tracking i.e shadowing and
    - Was intuitive and easy to use.
- Svein Tore
  - Think-Aloud (Leap)
    - Not to easy to know what your grabbing.
    - Jumped a bit after release.

    - Observations
        * Difficulties finding the correct buttons on the Hydra controller.
        * Forgot to follow calibration instructions on the Hydra controller.
    - Interview
        * When asked about the leap pickup issue, the subject wanted more feedback for when the object was picked up.

- Link
    - Think-Aloud (Leap)
        * Object don't want to be standing where intended.
        * Picks up object B when releasing object A.
        * Want a nudge feature for a more precise control.
        * Object moves from the spot when releasing.
    - Think-Aloud (Hydra)
        *
    - Interview
        *
    - Observations
        * Problems staying in the field of view of the leap controller, resulting in a lot of jittering and unintentional interactions.
        * Frustration led to being content with a poor result on the leap controller.

B
Form of Consent

C
Subject Information

D
System Usability Scale