A Virtual Reality Implementation of the Trier Social Stress Test Using Head-Mounted Displays

Interaction with Physical Objects In Virtual Reality

Master Thesis, June 2016

Andreas Møller Thomsen
amth11@student.aau.dk

Jens Stokholm Høngaard
jhoeng09@student.aau.dk

Patrick Mark Houmann Christensen
pmhc11@student.aau.dk

Supervisor:
Martin Kraus
martin@create.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Department of Architecture,
Design and Media Technology
Rendsburggade 14
9000, Aalborg
Telefon +45 99 40 87 93
http://create.aau.dk

Medialogy 10th Semester

Title:
A Virtual Reality Implementation of the Trier Social Stress Test Using Head-Mounted Displays

Theme:
Master Thesis

Project Period:
February - May 2016

Project Group:
MTA161038

Members:
Andreas Møller Thomsen

Jens Stokholm Høngaard

Patrick Mark Houmann Christensen

Supervisor:
Martin Kraus

Circulation: 2

Pages: 44 + 1 DVD

Finished on May $25th$ 2016

Abstract:

A Trier Social Stress Test (TSST) is designed to induce stress in participants by having them give a presentation to three unknown people, whom act as a committee, as well as other exercises. A previous study using a virtual reality TSST in a CAVE was a success, but an easier way of conducting the test in virtual reality was deemed necessary. Using a CAVE is expensive, and as such, a virtual reality system using head-mounted displays (HMDs) was requested.
A problem with HMDs is that people are not able to see their own limbs while wearing it. This problem makes it hard to measure the participants stress level during the TSST. The way of measuring stress levels in this case is through cortisol in saliva samples. Taking these saliva samples requires the participants to chew on a cotton swab for at least 10 seconds, and doing this without seeing the real world could prove difficult.
An experiment with comparisons of three different methods of taking a saliva sample through cotton swabs, or in our case pieces of similar shaped candy, while wearing a head mounted display was conducted.
All our results point toward a method without facilitator assistance yielding the best outcome.

Contents

Chapter 1

Introduction

In 2014, Fich et al. conducted a study to see how architectural design can alter the physiological reaction to psychosocial stress. They used a virtual version of a Trier Social Stress Test, or TSST, to do this, in which the space is computer generated and presented in a CAVE, instead of constructing a new physical environment [1]. A TSST is conceived to induce stress in participants by;

1. Recording of a physiological baseline (Heart Rate, Heart Rate Variability, T-Wave Amplitude, Cortisol)
2. The participant faces an unknown committee, who instructs in the following task, which requires the participant to give a speech to the committee.
3. The participant is allowed 5 minutes to prepare the tasked speech.
4. After 5 minutes, the participant again faces the committee and presents the speech.
5. After giving the speech, the participant is instructed by the committee to count backwards from 1687 in steps of 13. If making an error, the participant is prompted to start over.
6. After completing all tasks, the participant recovers during a period which lasts approximately 40 minutes.

The result of their study was a success, but an easier way of conducting the test in virtual reality was deemed necessary. The CAVE is a very expensive equipment to have running, and with newer technology available, a virtual reality system using head mounted displays was requested. Here two new primary problems present themselves;

1. Users cannot see their own hands. This could pose a problem, especially in the task where they have to give the presentation since many people uses their hands to gesticulate. This would require hand tracking and possibly full body tracking.

2. Users will have to give a saliva sample. Taking these saliva samples requires the participants to chew on a cotton swab for about 1 minute, and doing this without seeing the real world could prove difficult. This means that the system has to facilitate not only hand tracking but also finger tracking and tracking of physical objects, i.e. the saliva sample swab, as close to 1:1 as possible in a virtual reality. Problems therein could lie in the calibration and interaction of the system, e.g. size, placement, and orientation of the objects. Height of users can also vary, resulting in different lengths of limbs.

This present project will serve as a preliminary study, investigating how to create this virtual reality system that satisfies the criteria mentioned above. As such, we will not conduct an actual TSST but instead make sure that a TSST could be conducted using this virtual reality system.

We discuss previous research done in the area in the following section, investigating concepts and topics as distance estimation and presence in virtual reality. Chapters 3 goes through the various iterations of technology used to make this project and Chapter 4 explains the system created with the purpose of being able to perform a TSST in virtual reality. Chapter 5 describes the experiment conducted in this project, while Chapter 6 shows and discusses the results of said experiment. We conclude the project in Chapter 7, and end with a short discussion surrounding a potential further development of the system in Chapter 8.

Chapter 2

Related Works

In the following chapter, we will identify the key fields of interest associated with the present project. These will be studied, compared, and their relevance to the project will be explained. Distance estimation is important to take into account due to the nature of the project. Besides this, the notion of presence will be investigated as the project revolves around creating a virtual reality experience. This experience should mimic a real life experience, and as such, achieve the highest level of presence possible.

2.1 Distance Estimation

According to Steinicke et al., distances are perceived as significantly compressed in virtual environments when compared to how distances are perceived in the real world, in some cases up to 50% [8]. For the present project, if the participants' perceived egocentric distance to the saliva sample tube in virtual reality is up to half of what it is in the real world, this could pose a problem when they have to reach out for it. Steinicke et al. suggested that factors such as a limited field of view of a HMD compared to the field of view of humans may have contributed to this compression [8]. However, the technology behind VR has progressed rapidly since their study, including larger field of view, therefore it is unknown if that is still a factor. Other than speculations, the real reasons remain unknown.

Multiple studies have investigated how to align the perceived distance in a virtual environment with the actual distance in the real world. A study by Interrante et al. found that if the virtual environment is a replicate of the real environment they are physically occupying, the compressions of the perceived distances is not as strong [2]. However, immersing test participants in different environment is the purpose of this present study and system. Steinicke et al. found that you could transition from a virtual replicate of the real world environment to a virtual artificial environment, and still maintain the more accurate distance perception. They created a transitional environment that represented the real world environment, when ready, the user could press a button that brought up a virtual portal. When the user walked through it, the portal disappeared and they were in the artificial environment [8].

Mohler et al. found that users showed improved distance estimation skills when able to see a fully-articulated avatar that represented their own motions, by performing blind-walking task with and without the avatar [4]. A follow-up study by Mohler et al. found that even a non-animated avatar also improved distance estimation compared to no avatar present, however, to a lesser extent than a fully-animated avatar [5].

According to Thompson et al., graphic quality does not have an impact on how well people preserves distance in a virtual environment, on a distance of five to fifteen meters [10].

2.2 Presence In Virtual Reality

Virtual reality is the method of creating a digitally rendered world which people will experience as if real. However, this virtual reality must fulfil a series of criteria in order to feel real and thus make the user forget about the virtual aspect of said reality. If these criteria are met, then one may label the user of said virtual reality as being present in said reality. Sanchez-Vives and Slater states that the notion of presence refers to the behaviour one elicits while immersed in a virtual environment [8]. If the behaviour of the individual in question mimicks the natural behaviour of said individual in real world environment, then the individual experiences presence. As such, presence as a term originates with the growing focus on virtual realities and accompanying environments during the mid-1990's [3].

Presence is a key topic while developing a virtual reality experience in which individuals ought to find themselves in a seemingly authentic situation. The present goal is to obtain physiological responses of optimal quality, which renders the authenticity of the experience a vital objective.

Sanchez-Vives and Slater describe an important factor for generating presence within a virtual environment; the correlation between human movement and virtual movement. If individuals move their heads in real life, the manner of which the movement is translated into the virtual environment must be equally authentic. This authenticity is furthered by implementation of a virtual body for the participants. Whether detailed or crude in design, said virtual body results in an increased sense of presence compared to a simple cursor-based pointing system [7], which supports the findings of Mohler et al.. In order to measure presence, Usoh et al. develop a questionnaire containing six questions. This questionnaire is used for measuring presence in virtual reality.

2.3 Trier Social Stress Test

A method of inducing and measuring stress in individuals may be done by conducting a Trier Social Stress Test (TSST). A TSST forces the participant to complete a series of theoretically stressing tasks in front of a committee of unknown individuals to the participant [1]. The structure of this type of test usually consists of the following steps, divided into two separate rooms, i.e. preparation room and experiment room. During the whole TSST, physiological data, including heart rate (HR), high-frequency heart rate variability (HF-HRV), T-Wave amplitude (TWA), and cortisol, is collected from the participants.

1. **Baseline**:
   Recording of a physiological **baseline** is completed before commencing with the stressful tasks. This is often done as the participant arrive to the experiment room, being situated in the preparation room.

2. **Anticipation**:
   A requirement restricted upon the participant, **Anticipation** moves the participant from the preparation room to the experimental room, where he/she faces the aforementioned committee for the first time. As such, the leading committee member instructs the participant of the tasks to come, with the first task being preparing a speech as if applying to the participant's dream job. The second task is not specified here. After this, the participant is transferred back to the preparation room.

3. **Preparation**:
   In the preparation room, the participant is allowed 5 minutes to **prepare** the tasked speech. In a traditional TSST, the participant is allowed to write the speech down as notes, but not allowed to bring the notes to the speech itself.

4. **Speech**:
   Once the allocated 5 minutes has passed, the participant is brought back into the experiment room, presenting the prepared **speech** for the committee members. This tasks usually lasts 5 minutes itself.

5. **Counting Backwards**:
   After the speech, the participant is tasked with the seconds task, which is **counting backwards** from 1687 in steps of 13, in front of the committee. If making any mistakes, the participant is promptly halted, and told to start from the beginning. Again, 5 minutes is allocated for this specific task.

6. **Recovery**:
   When all tasks are completed, the participant is returned to the preparation room and **recovers** from the test itself. This recovery period lasts approximately 40 minutes.

The TSST structure listed above is similar to the structure used by Fich et al., which developed a virtual TSST. The strength of a virtual TSST (V-TSST) is ease of alteration of environments. This is due to all visual aspects of said V-TSST being digitally rendered, and as such, may be switched with other digitally rendered visuals. The underlying theory of Fich et al. is that the human nature unconsciously seeks potential escape routes, if the fight-or-flight response is activated within the body. If situated in a closed room with no method of escape, the stress levels within an individual may be increased. As such, if situated in a room with methods of escape (windows, doors, etc.) an individual may become less stressed as a result.

From a physiological point of view, early indicators of stress in the human body is shown in the saliva produced in the given situation. This is in form of the hormone cortisol, produced within the adrenal gland of a human [6].

Cortisol is produced when the human body experiences stress, especially physical stress [6]. This physical stress can come in all shapes and forms, as hunger, dehydration, and fever amongst others all act as stressors for the human body. When cortisol is produced within the adrenal glands a series of events is triggered in the human body. Cortisol increases the blood sugar by decomposing protein. Once decomposed, the protein is converted to glucose with the help of the hormone glucagon. This is extremely unhealthy for humans as the protein decomposed and converted into glucose is not derived from food, but rather from tissue, namely muscles, bones and soft tissue. As such, stress is unhealthy for the human body as it essentially converts the body into sugar. Another event triggered by cortisol is the decreased ability of the cells to absorb the sugar, glucose, which supplies the energy needed for cells to function properly. Cortisol is, in terms of TSSTs, measured through saliva samples. Cortisol levels in saliva is not affected by saliva flow rates or the enzymes found in saliva. As such, saliva gives a reliable measure of cortisol levels in the human body. These saliva samples are gathered through cotton swabs delivered in small test tubes. The swab is placed in the mouth for a certain amount of time, where said swab is returned to the test tube afterwards.

As with many physiological responses, the increase in cortisol in the saliva of an individual is not enough to conclude on what actually caused this increase. Fich et al. triangulated several physiological responses to firmly conclude whether or not the participants of their experiment experienced stress or not. These additional physiological responses included HR, HF-HRV, and TWA. Both of these responses are connected to the beat of the heart of an individual.

For the present project, HR, HF-HRV, and TWA are not of importance. However, the method of collecting the much needed cortisol samples is at the center of this project. As such, the following chapters of this report document the process of creating a virtual system for creating a TSST, not through a CAVE, but rather, by use of HMDs.

Chapter 3

Integrating The System Technologies

The present project revolves around creating a virtual reality system which may be used to conduct a TSST, or more specifically, allow participants to deliver a saliva sample without breaking presence. As such, said system must use a method of motion tracking of humans, in addition to tracking of static, physical objects. These physical objects may be interacted with, while immersed in the virtual reality system. For motion tracking of a human body, the motion capture suit the Perception Neuron (IMU-sensor based) was planned to be used, while tracking of static objects is done by using the OptiTrack system (Infrared Camera based). Together with an Oculus Rift DK 2, we strive to create an optimal system for conducting a TSST. However, the tracking of the body was also done by use of the OptiTrack system in the end of the project.

3.1 Perception Neuron

The Perception Neuron is a motion-capture suit developed to be adaptive and affordable. Comprised of a carefully placed Inertial Measurement Unit sensors (IMU-sensors), the suit tracks the movement of the user through gyroscopes, accelerometers and magnetometers, all of which are 3-axis. As such, the dynamic range of each individual IMU-sensor is 360°.

The suit used for this particular project is comprised of 31 IMU-sensors called Neurons, which allows for full-body motion capture, see Figure 3.1. There is an optional 32^{nd} Neuron allocated for external objects, such as, a sword. The Neurons are placed on key features of the human body (such as the head, the upper back, lower back, shoulders, hands etc.), and are mapped to a rough rendition of a human body digitally. This is achieved through the proprietary software Axis Neuron, which records the tracked motions of the user in either 60 frames-per-second (19-32 Neurons connected) or 120 frames-per-second(18 Neurons or less connected), see Figure 3.1.

The entire Perception Neuron suit will take approximately 10 minutes to equip, and is comprised of a series of velcro-closed straps. On these straps are sockets designed for encapsulation of the Neurons. Before using the Perception Neuron suit, calibration must be done in order to reach optimal motion tracking. If the suit is not calibrated, the Neurons will not know their relative distance to each other, along with their own relative roll, pitch, and yaw. Said calibration is done in physical poses, where the user is;

A) ... sitting completely still by a desk,

B) ... standing up in an A-pose,

C) ... standing up in a T-pose,

D) ... standing in a S-pose.

One thing to note is that, even though there are 8 Neurons allocated for each hand, there is no dedicated hand calibration. Once the suit is calibrated, the user is able to capture true-to-life motion and record it for usage digitally. This could be as rough animation for video games, or motion capture for computer-generated imagery elements (CGI) in film making. Recording the motion of the user is possible through either a USB 2.0 cable, WiFi, or an SD-card connected to the Connection Hub of the Perception Neuron.

The developers of the Perception Neuron created a plug-in library for the game engine Unity. This plug-in allows third-party developers to use the data gathered through Axis Neuron within Unity to control a virtual human body. This essentially is a mirror of the virtual body shown within Axis Neuron, see Figure 3.1. This method opens up the possibility for real-time usage of the Perception Neuron suit within a virtual environment, rather than recording motion for post-processing. As such, the incorporation of Unity when using the Perception Neuron is seamless. The latency between the two pieces of software (Axis Neuron and Unity) is low enough for the user not to notice. The only requirements are that the two pieces of software run on the same computer and that Unity is of version 4.x or higher. The latency of the Perception Neuron itself is 10-13 ms on-board the Neurons, and 3-5 ms within Axis Neuron [11].

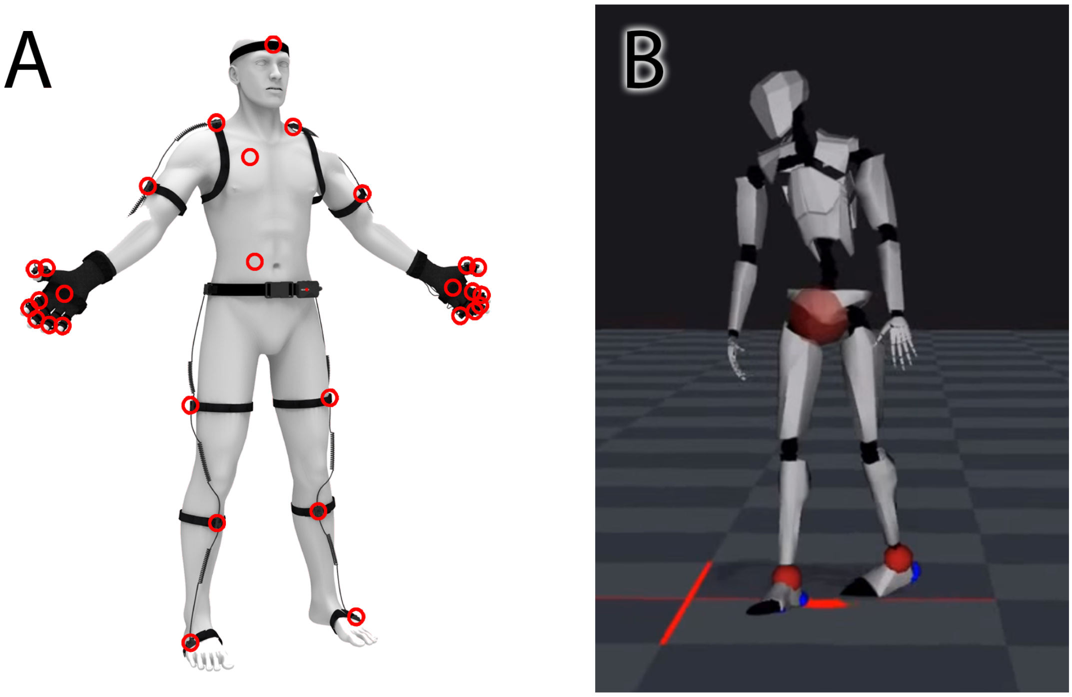


Figure 3.1: (A) Placement of the 31 Neuron Sensors. (B) The digital skeleton in Axis Neuron.

3.1.1 Inertial Measurement Unit

The IMU-sensors used on the Perception Neuron suit are as mentioned comprised of three unique sensor types; gyroscope, accelerometer, and magnetometer. By triangulating the data sent by these three types of sensors, the IMU-sensor is able to calculate the relative position and movement of itself.

In order to estimate the pitch, yaw, and roll of itself, the IMU-sensor uses the gyroscope. The gyroscope detects any rotation of the sensor, see Figure 3.2. Essentially, the three terms describe the rotations around the x, y - and z-axis of an object.

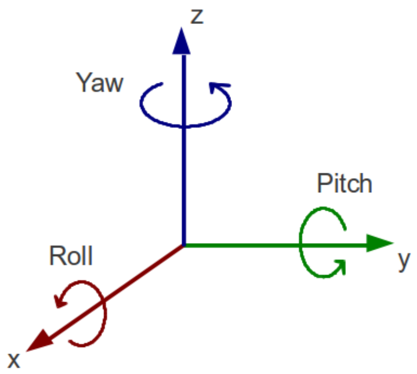


Figure 3.2: Pitch, Yaw & Roll

The accelerometer of the IMU-sensor detects the movement of the sensor. As such, the detected acceleration of the IMU-sensor, when calibrated, calculates the relative distance between the IMU-sensors connected to the suit. This allows for a graphical representation digitally through Axis Neuron.

The incorporation of the magnetometer controls the potential drift of the IMU-sensor. If exposed to electric fields, the IMU-sensors might experience a slight drift in position and rotation. As such, the magnetometer creates a magnetic field around the IMU-sensor.

The IMU-sensors can become demagnetised, this can happen if the IMU-sensor gets too close to electronic equipment or strong magnets. As a result, they would start to drift. At this point, a recalibration is needed through the use of a program called Neuron Doctor supplied by the developers themselves. The process recalibrating the IMU-sensors involved taking all the sensors one by one and spinning them around in a circle for around 30 seconds for the sensors to get enough data to re-calibrate itself.

3.2 OptiTrack System

The OptiTrack system is a full body tracking system that uses 20 Flex3 cameras, developed by OptiTrack. These cameras have a resolution of 640x480 with a refresh rate of 100 frames a second and a latency of 10 ms from the camera to the software. These cameras are placed in an even distribution along a lattice rig. These cameras have a ring of infrared LEDs around the infrared camera to bathe the area in infrared light to light up the reflective spheres users equip. Said spheres are linked in specific triangular patterns that can be seen and identified in the proprietary OptiTrack software Motive, which are then linked to specific objects e.g. hands, legs, body. In our case, the cameras are connected to the desktop PC by USB 2.0 to a USB hub with up to 6 cameras in each hub and then connected to the main computer. There it goes through a program called Motive that uses the camera data to find the triangular patterns, and then send the position and rotation of these patterns to Unity.

Throughout the present project, the OptiTrack system will be used for tracking of physical objects in the virtual environment. The objects we track through the OptiTrack system is described in Section 3.6.1.

3.3 Custom Calibration of the Perception Neuron

One of the main problems we encountered working with the Perception Neuron is that the standard calibration is not sufficient enough for precise hand and finger tracking. This resulted in the user's hand and finger not lining up inside and outside of virtual reality to a satisfying degree. Figure 3.3 highlights this miss-match. At times the virtual fingers would only move fractions of how much the real finger moved, e.g., when you rotated your thumb 60 degrees in real life, the corresponding virtual thumb would only move approximately 10 degrees.

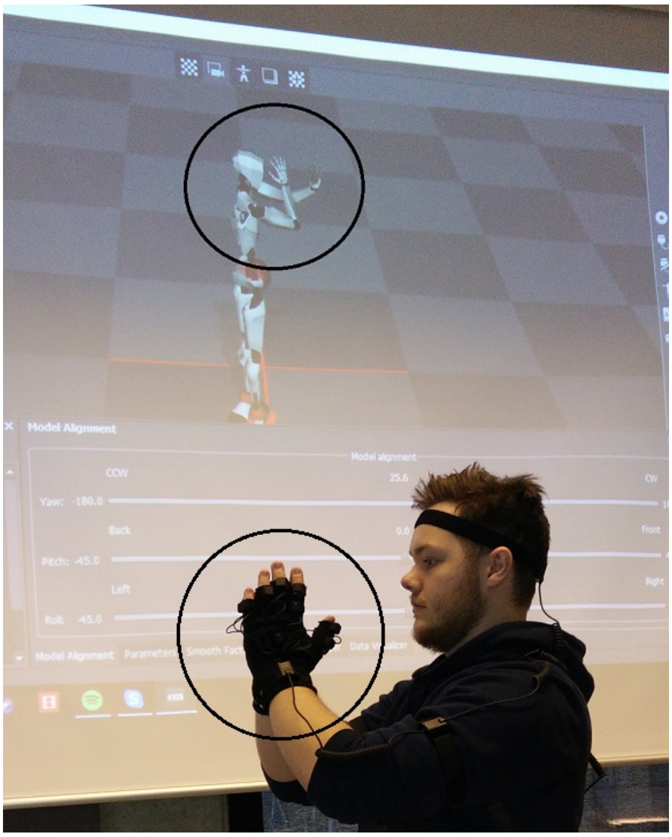


Figure 3.3: Difference in real hands position and Axis Neuron's predicted position.

Due to this problem of the Perception Neuron not being capable of fine finger and hand control, we found solutions to both aforementioned issues independently.

To fix the calibration of the fingers, we took separate models of each hand and moved their fingers into a certain gesture that we decided. This resulted in a *"pinch"*-gesture as seen in Figure 3.4, these hands were dubbed the *calibration hands*. The Euler angles for every digit of every finger of the *calibration hands* was then saved in a script. To calibrate the system the user had to mirror the same gesture with their own hands. When hitting a designated button, the current Euler angles for every digit of every finger of the Perception Neuron hands were saved in a script as well. Continuously from this point on, all *calibration hands*' saved Euler angles were divided by all of the Perception Neuron hands' saved Euler angles, which was then multiplied by the Perception Neuron's current Euler angles. See Equation 3.1.

$$\frac{savedCalibrationHandRotation}{savedPerceptionNeuronRotation + \varepsilon} * currentNeuronRotation \tag{3.1}$$

The newly calibrated rotation is then applied to the geometry of the hand model. A problem with this approach is that if the Perception Neuron's saved Euler angles are close to zero, and the calibration hand's saved Euler angles are not zero, moving the hands will result in an exaggerated movement. As an example, stretching your thumb could make it spin around itself a number of times instead of just stretching itself.

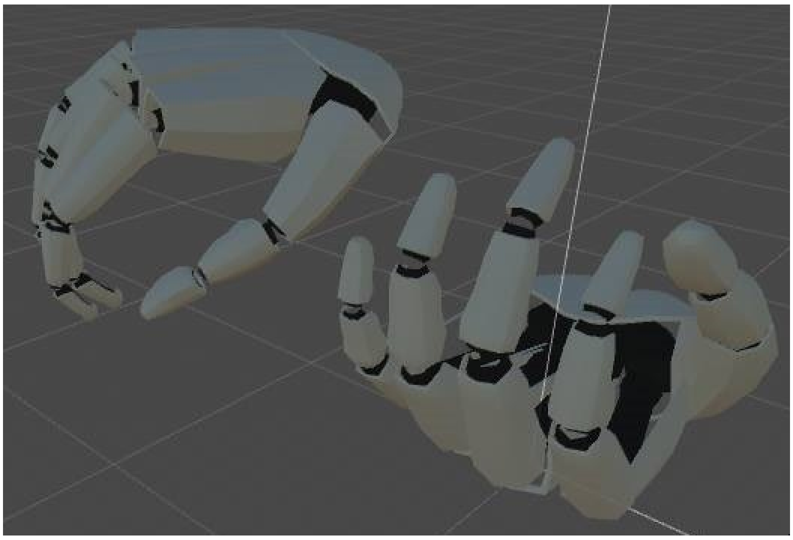


Figure 3.4: Presentation of the pinched hands.

To fix the problem concerning the position of the hands, we used the OptiTrack system. Instead of the Perception Neuron controlling the position of the hands, we attached a OptiTrack tracker to each hand and used their position as the virtual hand's position. Because of this, we removed the arms of the model, resulting in two floating hands which were controlled by the OptiTrack system and fingers that were controlled by the Perception Neuron.

3.4 Oculus Rift

The purpose of this project from the start, was to create a virtual reality environment that could be used in a TSST. As such, a head-mounted display (HMD) is needed. For the present project, the Oculus Rift DK2 is used. The Oculus DK2 features one organic light-emitting diode (OLED) screen with a resolution of 1920 × 1080, that is split in two for a respective resolutions of 960 × 1080 for each eye. Like with the Neurons of the Perception Neuron, the Oculus Rift DK2 features a combination of a 3-axis gyroscope, accelerometer, and magnetometer. These are used to calculate the relative position of the Oculus Rift DK2 within physical space, while avoiding drifting within the virtual space. The surface of the Oculus Rift DK2 have built-in infrared emitters. These are placed in a distinct pattern which infrared cameras are able to detect and differentiate between. This may also be used to calculate the relative position of the Oculus DK2 within the physical space.

The Oculus Rift used for the present project, together with the OptiTrack system, is locked to a specific desktop computer located at Aalborg University Create Campus. The desktop computer features an Intel i7-4770 3.4 GHz processor, 32 GB memory and a Nvidia GTX 980 Ti graphics card.

The manufacturers of the Oculus Rift DK2 provide developers with plug-ins for various pieces of third party software. Amongst these are Unity, which allows us to use the Oculus Rift DK2 in our project, together with the Perception Neuron and the OptiTrack system.

3.5 Shift in Tracking Technology

As development progressed with the system as a whole, issues became apparent with the Perception Neuron. The Neuron suit promises easy motion capture, where the users just plug the suit into a computer and thus are able to record movements. However, motion capture is not optimized for precision. Rather, the Perception Neuron suit is designed to give rough and quick recordings and not precise and fine recordings. For the present project, this proved to be a large issue. The end-goal of the system is to be able to track fingers with precision which is not obtainable with the Perception Neuron suit. Furthermore, after prolonged use we experienced that the Perception Neuron would start to lose precision due to drifting. At some point, arms would even start to bend backwards. This still occurred to some degree after recalibrating all Neurons using the Neuron Doctor software as well. The TSST takes about 60 minutes, as such, this imprecision is unacceptable.

We contacted the manufacturer of the Perception Neuron explaining our problems, they informed us to the actual designed use of the Neuron; that the Neuron suit is not meant for precise finger tracking. This is also reflected in the lack of proprietary finger calibration within the Axis Neuron software provided from the manufacturer. Rather, this particular software only accounts for calibration of arms, legs, and body.

This meant that a lot of our assumptions of the Perception Neuron were wrong and we could not use it as a way for reliable finger tracking. Therefore, we needed a better solution to the problem. Our solution to the problem was to discard the Perception Neuron and only go with the OptiTrack system. This solution came with its own limitations in that it would induce bigger delay and that it was not possible to get finger tracking, and therefore, the hand would either be in an *"open palm"*-gesture or a *"pinched"*-gesture depending on what was needed. Since the OptiTrack system would not drift the overall gain would be better than having a combination of the two.

3.6 Rapid Prototyping

During the present project we relied a great deal on rapid prototyping, in order to achieve the optimal quality, performance, and usability of the completed system. This rapid prototyping mainly solidified in 3D-printing various parts to use for the OptiTrack system. Multiple iterations were made of these parts. The following section will explain the various iterations in-depth.

3.6.1 Custom OptiTrack Trackers

The OptiTrack system located in the AVA-Lab at Aalborg University utilizes tracking-objects that can be attached to any given body part or object, and assigned to corresponding virtual objects. These tracking-objects, or trackers, consists of three reflective spheres that form a triangle. Any given tracker has to be calibrated to the system so that it is recognized by the system, so it can be detected by the software. All trackers are entirely unique by having the three reflective spheres form unique triangles. This allows the system to distinguish them from each other. Figure 3.5 shows one of these trackers.

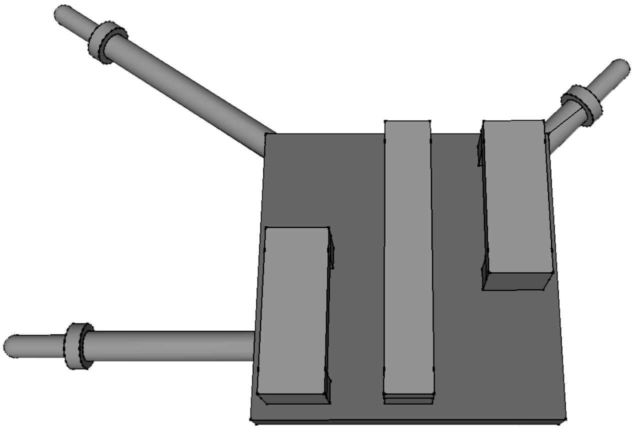


Figure 3.5: The original tracker. It has a base and three struts for reflective spheres.

A tracker consists of a base and three struts, with the base dimensions of $40mm \times 40mm \times 5mm$. The length of the three struts vary in order to form unique triangles. The reflective spheres are attached one to each strut. The base is customized with bridges, that allows elastic straps to be run through. This allow the tracker to be easily attached around body limbs or other objects. However, for the present project, we intend to use these bridges for another purpose, which will be explained through the coming iterations of the custom trackers.

Early in the process of the project, we discussed using a 0.5 liter beverage can as the trackable object for initial testing. As such, we designed a tracker, akin to the original OptiTrack tracker. However, the main difference here is found in the ability of the new tracker to hold a beverage can upright, while still containing three struts for the reflective spheres, see Figure 3.6.

After designing the 3D-model for the first iteration of the Beverage Tracker, it became apparent that it would not yield the intended functionality. This was largely due to difficulty of 3D-printing this specific design as well as possibly becoming too large and unstable. As a result of this, this design never got printed.

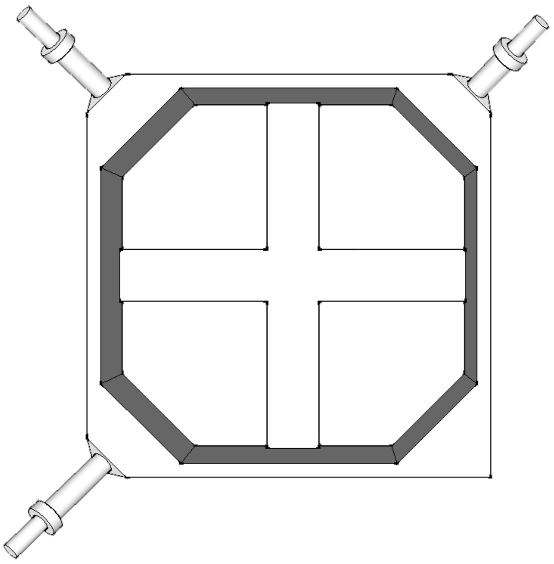


Figure 3.6: First iteration of a custom tracker. Intended to hold a 0.5 liter beverage can.

We still wanted to use a 0.5 liter beverage can for initial testing, so the beverage tracker was redesigned. We came up with a new method of tracking said beverage can, which entailed using the original OptiTrack trackers. The new iteration of the beverage tracker acts as an add-on that can be clicked onto to the original tracker's base. The stand for the beverage can would then be glued onto the second custom base, see Figure 3.7.

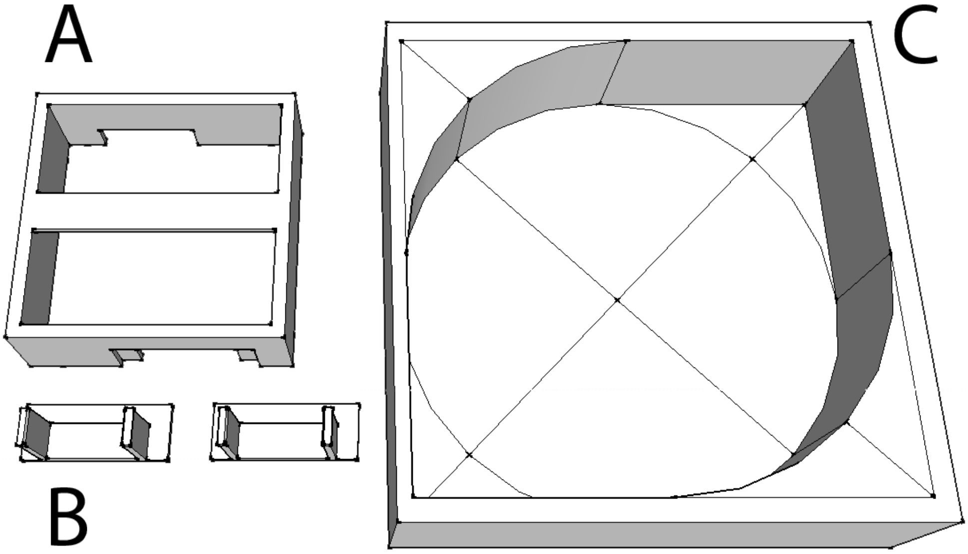


Figure 3.7: (A) A base that can click onto the original tracker's base. (B) Clips to hold the base in place. (C) The beverage stand, intended to be glued onto the base.

This custom base would, as mentioned, be clicked onto the preexisting original tracker's base. To obtain this effect, small clips with hooks in the end would be slided into two of the sides of the custom base, attaching themselves to the bridges on the original tracker's base. However, due to rushing the design, this idea would not work. Although the idea of clicking the custom base unto the original tracker's base is a fitting idea, it does not secure the custom base as it can simply be pulled upwards, due to lacking closing on the custom base itself, see Figure 3.7.

After we printed this design, we realised that using a 0.5 liter beverage can might not yield the best result for initial testing of the system, as it might end up being too large of an object compared to the tasks needed to be completed in an eventual test of said system.

Since we did not have the intended test tube to be used for an actual TSST, we made a simple rod to grab hold of instead. As such we designed a new custom base, much akin to the one shown on Figure 3.8. However, for the new iteration, we designed the closing mechanism to grab hold of the Original Tracker's base and raise the entire tracker, acting as 'legs'. The dimensions of this new Custom Base are $40mm \times 40mm \times 25mm$. A hole is present in the center of the custom tracker's base. This particular hole is designed to support the rod or other accessories if needed, onto which the user grabs, see Figure 3.8. This dimensions of the rod are $6mm \times 6mm \times 100mm$, and is slided into the hole.

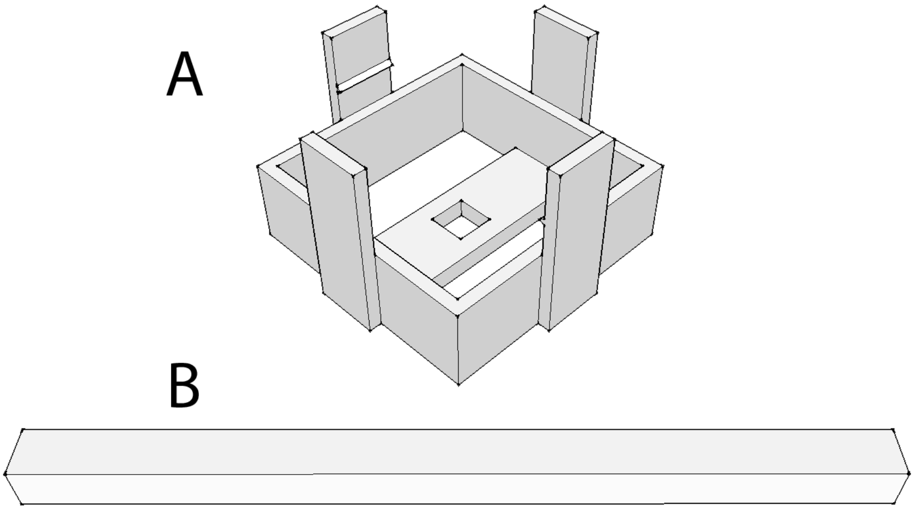


Figure 3.8: (A) Third and final iteration of the custom tracker. (B) The holding rod.

3.6.2 Custom Hand Trackers

Since we do not use the Perception Neuron to locate and track the hands of the user, we resorted to using the OptiTrack system for hand tracking as well. This meant that we needed a method of fixing original tracker to each hand. As such, we decided on using a set of spare gloves from the Perception Neuron set. These gloves only feature a single Neuron socket (i.e. there is no sockets on the fingers; only on the top part of the hand itself). We designed two custom bases to place upon the single Neuron socket on each hand, see Figure 3.9. The dimensions of the Custom Hand Trackers are $40mm \times 40mm \times 5mm$, with supporting legs of $15mm$ in length. Once placed upon the socket, an original tracker may be clicked onto the custom base.

However, while placed upon the Neuron socket, the custom hand tracker base can very easily fall off. As such, we designed the custom base to feature two slits into which Velcro-style fabric is slided through. This allows us to fasten the custom base onto the glove, since the socket on this is also fastened with Velcro-style fabric. As such, there were extra room for the custom hand tracker to be fastened as well.

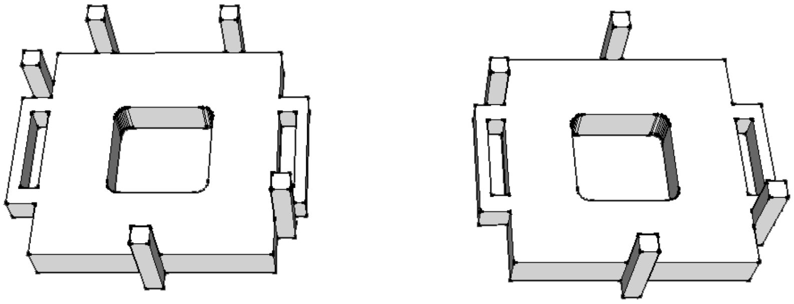


Figure 3.9: Custom Hand Tracker base for both hands.

3.6.3 Oculus Cover

The Oculus Rift DK2 features an array of infrared emitters on the front of the HMD. These infrared emitters allows for tracking of the Oculus Rift DK2 through e.g. the OptiTrack system if need be. However, for the present project this poses a problem due to the interference this creates when trying to track other objects precisely. The Rift's infrared emitters are not arranged in the triangle-patterns needed for precise tracking by the OptiTrack system, resulting in the OptiTrack system often mistaking the trackers with the Oculus Rift itself.

As such, we needed a method of removing this interference, since it hindered the functionality of our system. The Oculus at our disposal is used by many different students and professors, so simply taping the entire front of the Oculus Rift DK2 was not a viable solution. Neither was disabling the emitters since this cannot be done without 'hacking' the Oculus. The proprietary software accompanying the development kits of the Oculus does not feature any such ability to disable the infrared emitters. As such, we decided to 3D-print a custom cover for the Oculus Rift DK2, see Figure 3.10, that allows for easy attachment and removal.

This custom cover for the Oculus occludes the entire front, top, bottom, and sides of the HMD. This encapsulates the Oculus Rift DK2 and ensures that the OptiTrack system cannot register any of the infrared emitters located onto said HMD.

The biggest of the various 3D-prints made for this project, this custom cover for the Rift measures $192mm \times 112mm$, and is $105mm$ deep. On top of the cover are two rods pointing towards the wearer, which allows us to slide one of the aforementioned original trackers into it. As such, we use the original trackers for tracking the user head position within the virtual reality instead of the built-in infrared emitters featured on the Rift itself.

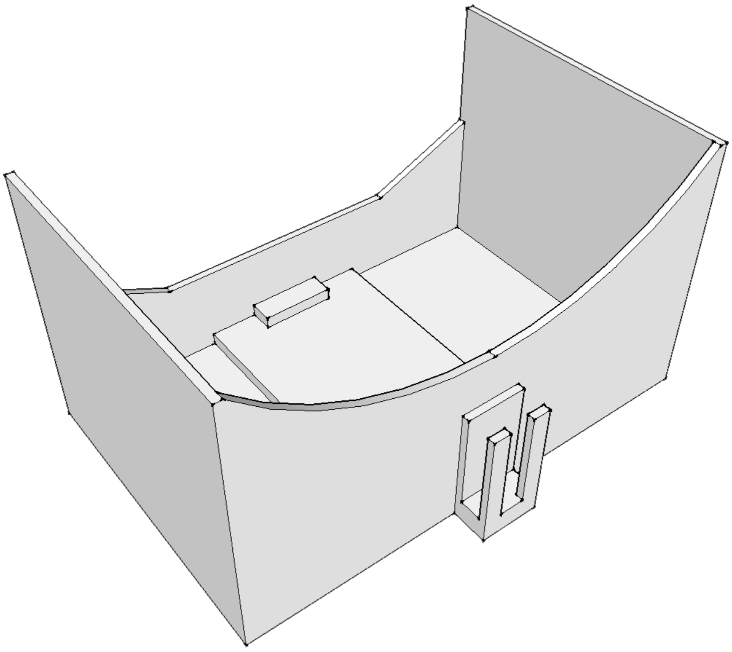


Figure 3.10: Custom cover for the Oculus Rift DK2.

Chapter 4

The System

This chapter will explain the software part of the virtual reality system, which was created with the purpose of being able to perform a TSST in virtual reality.

Since the purpose of this TSST is to see how different environments affect a person differently, the system needs to make the process of changing environments possible in the easiest way, this will come in the form of a Scene Manager. We will also show how to make an environment. As part of the TSST, participants need to move between a experiment room and a preparation room twice. A committee also needs to be present during the TSST and able to give instructions and respond to the participants. To this purpose, a digital committee will be created with a complementing system that allows these committee members to give instructions and respond.

4.1 Example Environment: AVA-Lab

This section describes how an example environment is constructed. This example takes an existing room and replicates it in Unity for use in this system. This environment will also be the one used in the experiment for this project.

The construction of the environment started by measuring the dimensions of the entire room, as well as some of the objects inside the room such as the tables, stairs and lattice pillars. The environment is $7.5m \times 7.5m \times 4.5m$. The floor is lowered from ground level, thus stairs are placed at each door leading to the floor. In addition to this, the room has three tables and metal lattice pillars at each corner of the room, with more lattice beams from corner to corner close to the ceiling. This lattice structure is used to mount the OptiTrack system.

The environment's four walls, ceiling and floor is created by using Unity's quad mesh. A quad is four vertices facing inwards, towards the center of the room. Each quad is assigned a corresponding texture depending on its position. The textures is created to match the real environment as close as possible. The objects in the room was modeled in the same scale as their real world counterparts, textured with matching colors, and then placed in the room in the correct locations. The textures and models used in the environment can be seen in Figure 4.1.

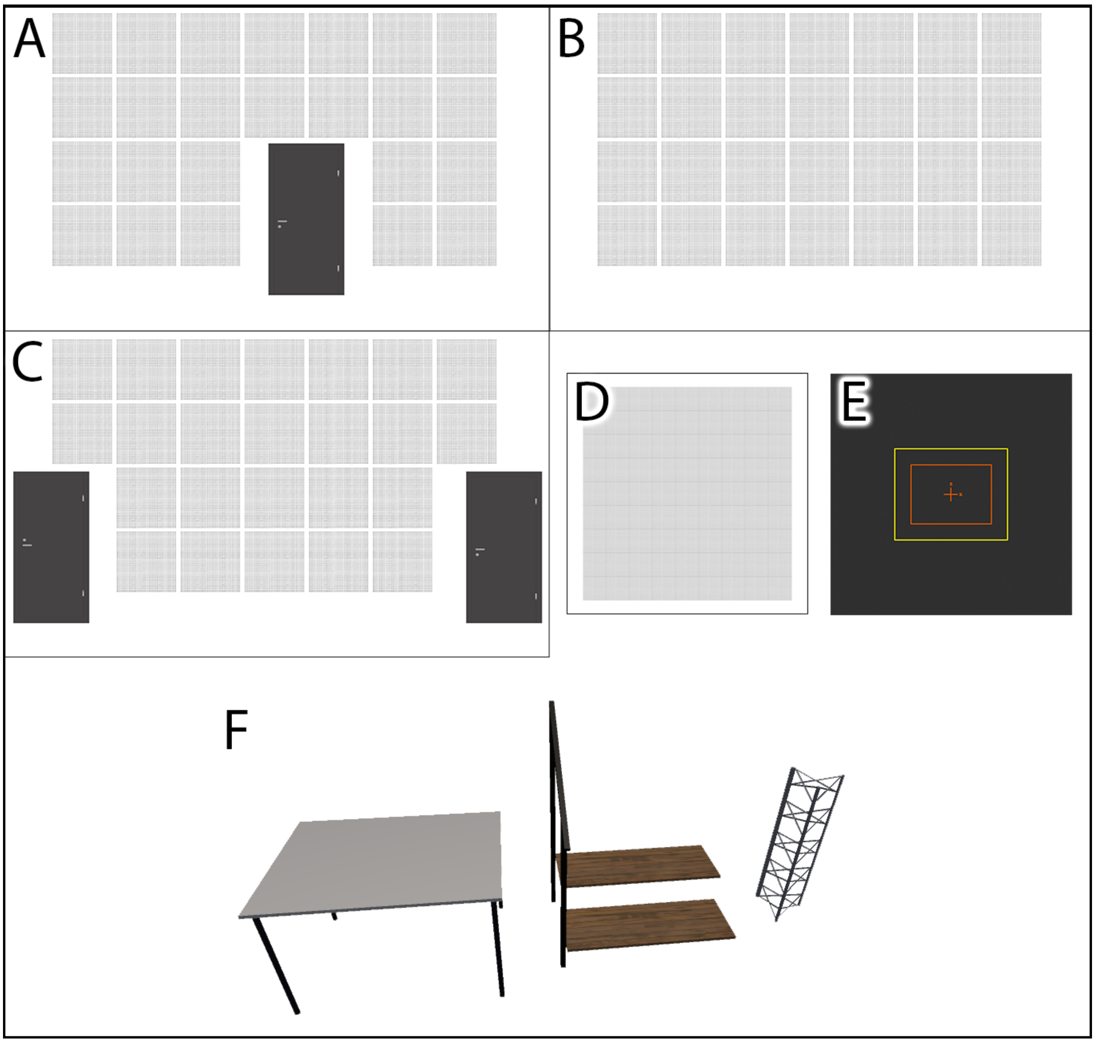


Figure 4.1: Texture used for the (A) front facing wall, (B) side walls, (C) back wall, (D) the ceiling, and (E) the floor. (F) Table, stairs and a single lattice pillar used to furnish the room. The lattice pillars were stacked on top of each other to create a bigger structure.

Next thing to consider was lighting. The physical room is evenly lit with soft white fluorescent light. As such, a large spotlight above the room was shining directly down into the room, hitting all the walls and floor with a soft white fluorescent color, resulting in the finished environment seen in Figure 4.2.

4.2 Scene Manager

Scene Manager is the name given to the interface we created that easily allows the facilitator to switch one environment with another. Figure 4.3 shows the interface in Unity's inspector.

The environments are required to be built around the use of Unity's prefab system [9]. In our case, prefabs act as a single object containing the virtual environment with its texturing and lighting. As an example, the environment depicted in Section 4.1 is a single prefab called *AVA-Lab*, containing various other game objects as children that in total makes up the entire virtual environment. The following list shows the *AVA-Lab* prefab and its contents.

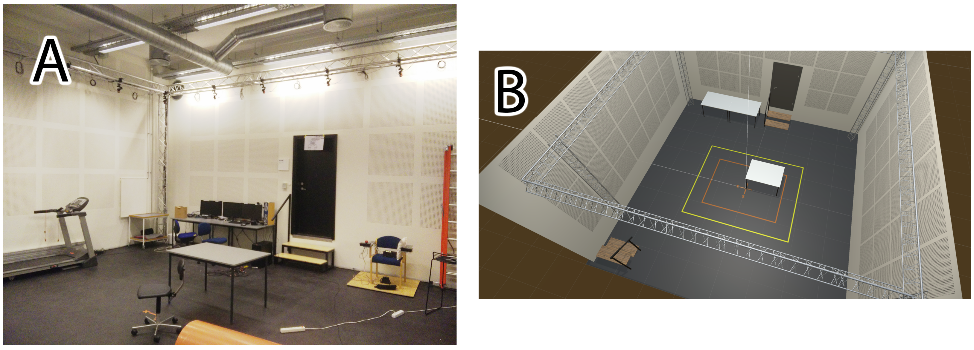
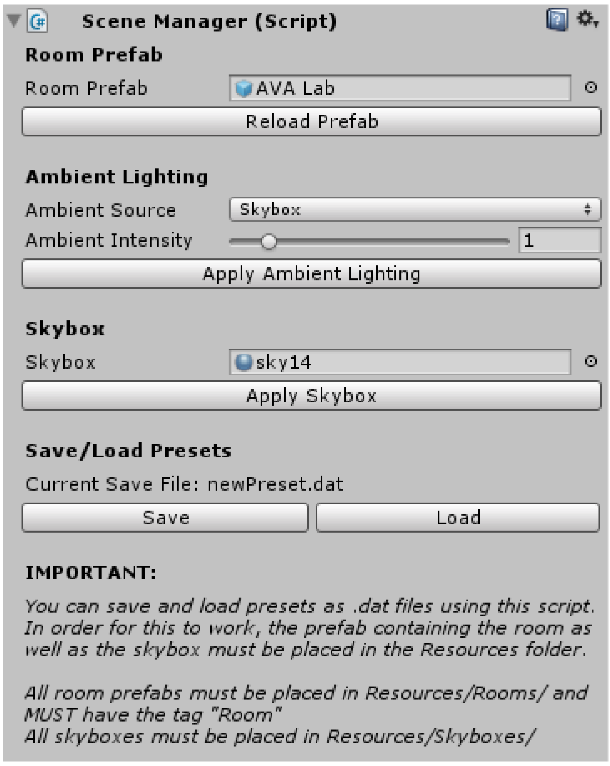


Figure 4.2: (A) A picture of the example environment. (B) The finished virtual rendition of the environment.



Scene Manager (Script)

Room Prefab

Room Prefab AVA Lab

Ambient Lighting

Ambient Source Skybox

Ambient Intensity 1

Skybox

Skybox sky14

Save/Load Presets

Current Save File: newPreset.dat

IMPORTANT:

You can save and load presets as .dat files using this script. In order for this to work, the prefab containing the room as well as the skybox must be placed in the Resources folder.

All room prefabs must be placed in Resources/Rooms/ and MUST have the tag "Room"
All skyboxes must be placed in Resources/Skyboxes/

Figure 4.3: The Scene Manager's Inspector UI within the Unity Editor.

- AVA-Lab
  - Spot Light
  - Walls
  - Ground
  - Ceiling
  - Stairs
  - Tables
  - Lattice Pillars

However, not everything making up an environment can be saved in a prefab. If you want an outdoor environment, or have the possibility of seeing the outside world through windows or similar, you will want to have the possibility of changing the world's skybox. In Unity, skyboxes are added to the scene through the *Lighting*-menu, and therefore have no relation to a prefab. The same thing is true for a scene's ambient lighting, which can either be sourced from the skybox itself or a specified color value. Therefore, these two are added separately in the UI, independent of the prefab.

As seen in bottom half of Figure 4.3, the interface also allows you to save and load presets for every different environment crafted, meaning that once the user made one particular environment, he can save that preset and load it again at another time. This allows the users of the system to create multiple environments and swap between them at will. Because the load function makes use of Unity's *Resources* class, all assets used in the preset (prefabs, skyboxes, etc.) must be placed in the Resources folder in the Unity project.

4.3 Preparation Room

As part of the TSST, the participant needs to move into a separate room at two separate occasions. We came up with three different methods of transitioning between the experiment room and this preparation room.

Method 1: The test participant physically walk from one location to another. The VR system will track the movement of the body.

Method 2: At the press of a button, the participant's avatar is interpolated over time from one location to another. This is most similar to what Fich et al. did as well.

Method 3: The participant closes their eyes and the HMD fades to black. At the press of a button, the participant's avatar is moved to another location instantaneously.

Method 1 would be the preferred solution here, however, due to the limitations of technology this is not a good solution because of latency which could induce motion sickness. For the same reasons we also chose not to do Method 2. As such, we implemented the third solutions; hitting a shortcut initiates the move from the preparation room to the experiment room. Hitting it again reverses the action.

4.4 The Committee

The system by Fich et al. used very crude static 3D-models, see the left image of Figure 4.4. We were tasked with creating a system that used video recordings of actors instead. We did not have the actual recordings that were going to be used for the project since they had not been recorded yet, so we assumed that they would be recorded against a green screen. Our system is then capable of removing the green screen through a chroma-keying shader. This feature ensures that if chroma-keying was not done in post-processing through an external application prior to being imported into Unity, is may be done within Unity itself. This results in something similar to what is seen in the right image of Figure 4.4, which displays three looping video clips of a person (although digitally animated) in a standing idle pose. Said idle pose features slight movement mimicking the movement done by a human standing in a similar idle pose.

The video recordings are projected onto a plane, but since the participant immersed within the virtual reality system would be seeing a 3D stereoscopic world, this plane is always rotating to face the camera, creating a billboard effect and giving more depth to the committee.

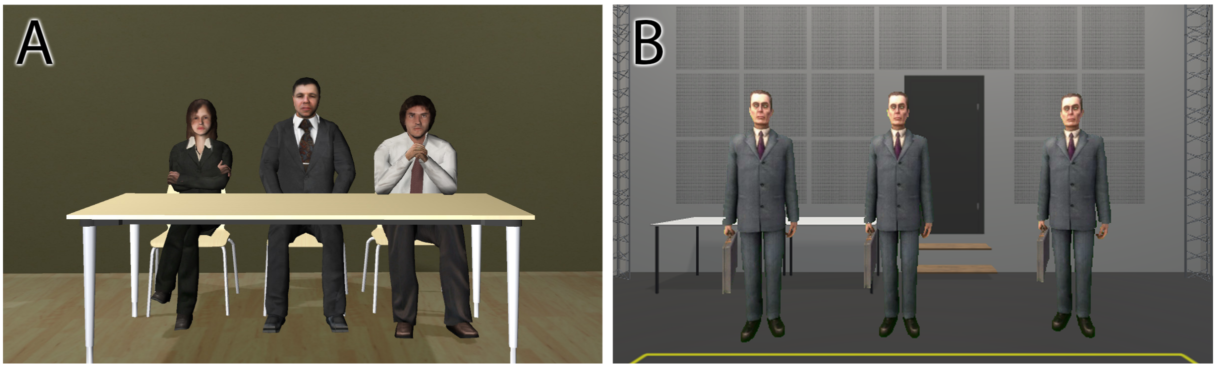


Figure 4.4: (A) The 3D-modelled committee used in Fich et al.'s experiment. (B) A temporary committee made from a looping video clip used as an idle pose.

4.4.1 Dialogue Soundboard

As mentioned previously, certain committee members must be able to either give instructions or respond to the participant. As the committee is now made up off video recordings, every line and response is its own small video clip. A certain video clip, selected by a facilitator presumably based on the situation, can be played at any time over the idle video clip. Once the selected video clip is over, the committee will go back to the looping idle video clip.

We came up with two possible methods for playing these video clips; a keyboard-shortcut to play a video clip or a soundboard-like UI displaying every video clip, and then selecting a video clip by clicking on it with a mouse. We chose to go with the soundboard-like UI because we felt it would be the easiest to expand upon, in case more video clips had to be added later in the process. Also, we received a preliminary list containing 46 responses, so finding enough combinations of keyboard-shortcuts to fit all responses would quickly become too complicated.

The dialogue soundboard is a collection of multiple buttons, as seen in Figure 4.5. The inspector UI for the dialogue soundboard can be seen in Figure 4.6. The figure shows all the properties for one button; The *Label*-field describes the button's label on the UI, so it's easier to identify. The *Clip*-field contains the actual video clip that will be played when pressed. Finally, the *Actor*-field, dictates which of the three committee members that will play the video clip.

Every button is sorted in an expandable list, in case you want more buttons you simply add more. If you make any change, you have to hit the "*Generate DialogueUI*"-button in the inspector, in order for the changes to take effect.

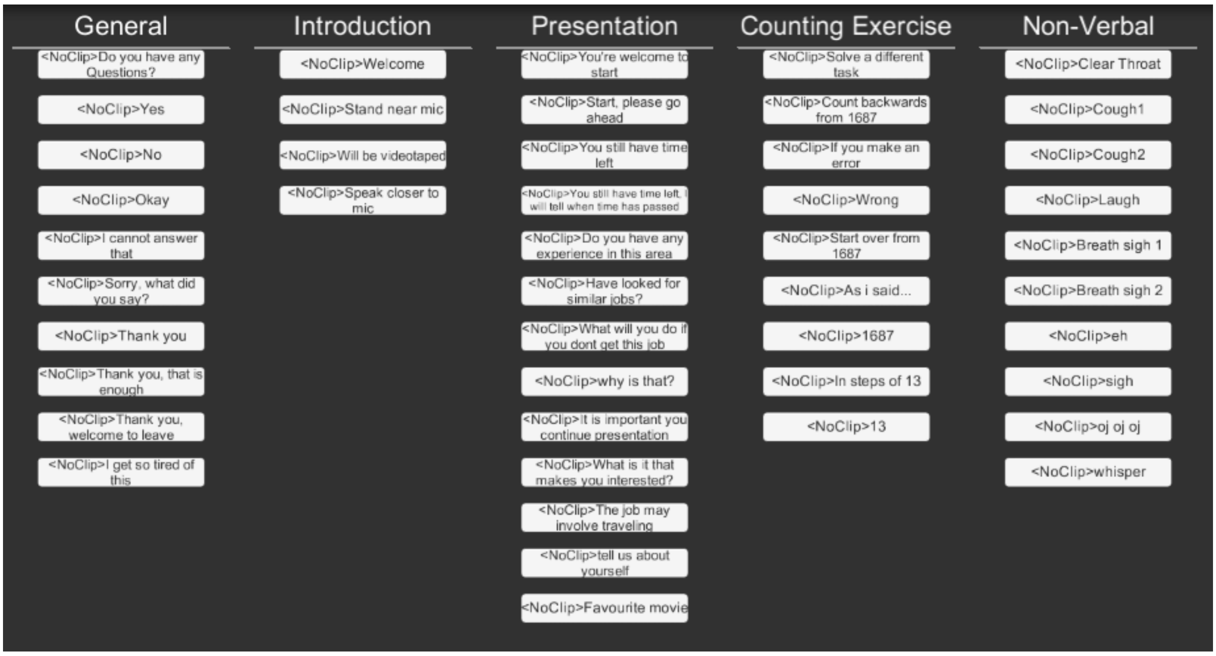
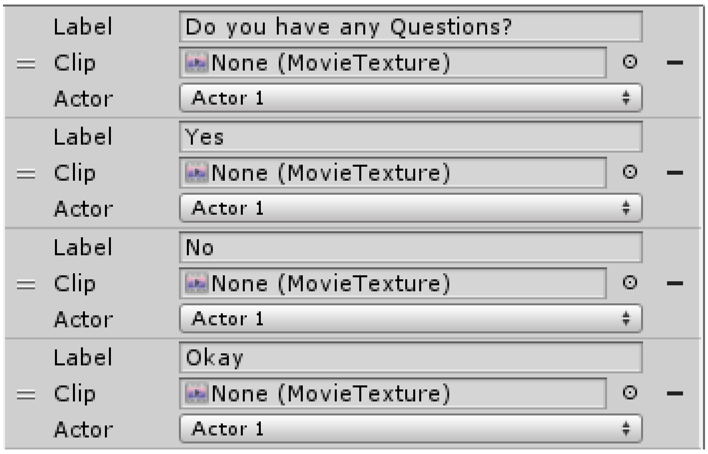


Figure 4.5: The dialogue UI in its current state. Note that all button labels contain the *"<NoClip>"* keyword, indicating that no actual video clip is assigned to the button so nothing will happen if pushed.



Label Do you have any Questions?
Clip None (MovieTexture)
Actor Actor 1

Label Yes
Clip None (MovieTexture)
Actor Actor 1

Label No
Clip None (MovieTexture)
Actor Actor 1

Label Okay
Clip None (MovieTexture)
Actor Actor 1

Figure 4.6: Shows how the four buttons (*Do you have any questions?*, *Yes*, *No*, and *Okay*) are setup in the inspector. Note that none of them currently have a video clip (MovieTexture) attached to it.

In its current state, the soundboard consists of the categories *General*, *Introduction*, *Presentation*, *Counting Exercise* and *Non-Verbal*, this is simply to make it easier for the facilitator to see which video clips might be relevant during the different phases of the TSST experiment. Also, the soundboard is only rendered on the monitor running the system, and not the HMD equipped by the eventual participant of the TSST.

Chapter 5

Experiment Design

Since we cannot compare our system to the previous system used by Fich et al., we assume that the system created through this project in itself works to a satisfying degree. As mentioned previously, using a HMD instead of a CAVE complicates the process of giving a saliva sample, because the participant is not able to see the physical world. Therefore, an experiment is conducted to find the best interaction method of giving a saliva sample while using our system. For the actual TSST experiment, cotton swabs will be used. The cotton swabs will collect saliva from which cortisol levels can be extracted from. However, due to miscommuniction with the company delivering the specialized cotton swabs regarding supplies, pieces of candy similar in size and shape will be used in their place.

We came up with three unique methods of achieving this goal. Participants perform tasks based on these methods and answer questions to gather results. See the tasks below.

Task 1: In order to place the piece of candy in the mouth, the participant grabs the tracked cup in front of themselves, takes the piece of candy out of the cup and places it in their mouth. After approximately 10 seconds, the participant again takes the cup and spits the piece of candy out into said cup.

Task 2: Instead of the participant taking the piece of candy out of the cup by themselves, the facilitator asks the participant to extend their hand. Once this is done, the facilitator places a piece of candy in the hand of the participant, who may place it in their mouth by themselves. After approximately 10 seconds, the facilitator asks the participant to extend their hand again, in which the facilitator places the tracked cup. The participant then places the cup by their mouth and spits the piece of candy into the cup.

Task 3: Instead of the facilitator placing the candy in the hand of the participant, the facilitator asks the participant to close their eyes and open their mouth. Once done, the facilitator places a piece of candy within the mouth of the participant. After approximately 10 seconds, the facilitator asks the participant to again open their mouth to which the facilitator hold the cup. The participant then spits the piece of candy out into the cup.

Task 1 would be the preferred method, as no involvement from a facilitator is needed here. The least amount of involvement from a facilitator is preferred as otherwise we hypothesize this would detract from the participant's eventual presence within the virtual environment, and thus tamper with the physiological data. As such, Task 2 would be the second most preferred method and Task 3 the least preferred.

Each participant experienced the three conditions in fixed-random orders, allowing for unbiased results, see Table 5.1.

Table 5.1: The Factorial Design for Task Rotation.

Participant	**Task Rotation**
1, 7, 13	T1 → T2 → T3
2, 8, 14	T1 → T3 → T2
3, 9, 15	T2 → T1 → T3
4, 10, 16	T2 → T3 → T1
5, 11	T3 → T1 → T2
6, 12	T3 → T2 → T1

After the participants completed the tasks in our virtual reality, all tasks were completed again in real life. This anchors the results, allowing us to compare behaviours during the various tasks.

5.1 Questionnaires

Throughout the experiment, participants is asked to answer a number of questionnaires. These questionnaires are vital for our data analysis as these questionnaires forms the foundation of our evaluation. We did not log any data from the software-side of the system.

5.1.1 Presence in Virtual Reality

As stated previously, a high level of presence is desired for a TSST. A widely acknowledged presence questionnaire, by Slater, Usoh & Steed (SUS), is used to obtain presence values for the three different tasks [11]. It was modified slightly to fit our experiment and location better, see the questions in the list below. Note that all 6 questions had three Likert-scale ratings, one for each task.

1. During the time of the tasks, please rate your sense of being in the virtual AVA-Lab, on the following scale from 1 to 7, where 7 represents your normal experience of being in a place.
   I had a sense of 'being there' in the virtual AVA-Lab:
   1. Not at all ... 7. Very much.
2. During the time of the tasks, to what extent were there times during the experience when the virtual AVA-Lab was the reality for you.
   There were times during the experience when the virtual AVA-Lab was the reality for me...
   1. At no time ... 7. Almost all the time.
3. During the time of the tasks, when you think back about your experience, do you think of the virtual AVA-Lab more as images that you saw, or more as somewhere that you visited?
   The virtual AVA-Lab seems to me to be more like...
   1. Images that I saw ... 7. Somewhere that I visited.

4. During the time of the tasks, which was strongest on the whole, your sense of being in the virtual AVA-Lab, or of being elsewhere?
   I had a stronger sense of...
   1. Being elsewhere ... 7. Being in the virtual AVA-Lab.

5. During the time of the tasks, consider your memory of being in the virtual AVA-Lab. How similar in terms of the structure of the memory is this to the structure of the memory of other places you have been today? By ‘structure of the memory’ consider things like the extent to which you have a visual memory of the virtual AVA-Lab, whether that memory is in colour, the extent to which the memory seems vivid or realistic, its size, location in your imagination, the extent to which it is panoramic in your imagination, and other such structural elements.
   I think of the virtual AVA-Lab as a place in a way similar to other places I’ve been today...
   1. Not at all ... 7. Very much so.

6. During the time of the tasks, did you often think to yourself that you were actually in the virtual AVA-Lab?
   During the experience I often thought that I was really sitting the virtual AVA-Lab...
   1. Not at all ... 7. Very much so.

5.1.2 Independent Task Comparison

To further compare the different tasks with each other, this questionnaire is used to see the difference between a task in VR and the same task outside VR, i.e. without using the HMD.

1. How comfortable did you find the corresponding task in VR compared to just now?
   1. Not at all ... 7. Very much so.

2. How intrusive did you find the corresponding task in VR compared to just now?
   1. Very Intrusive ... 7. Not Intrusive.

3. How disruptive was the corresponding task in VR compared to just now?
   1. Very Disruptive ... 7. Not Disruptive.

5.1.3 Task Preference

The last questionnaire is a simple preference question, to see which task the participants preferred.

1. Which method of ’eating’ candy did you prefer?
   Task 1, Task 2, or Task3.

The questionnaire ends with an open question: *"Do you have any thoughts you want to add about the three tasks?"*.

5.2 Apparatus

The computer used for this experiment is running Windows 7, has an Intel i7-4770 3.4 GHz CPU, 32 GB memory and a Nvidia GTX 980-Ti GPU. This computer runs the Oculus Rift DK2 and the OptiTrack system through the OptiTrack Motive software. We use the OptiTrack system in our test to track four objects, which are both of the participant's hands, the head of the participant and the Cup Tracker.

The Custom Hand Trackers are located on the back of the hand to make it easier for the OptiTrack Flex 3 cameras to register said trackers and not interfere with the participants' fingers. The tracker placed on the head is located on top of the custom cover for the Oculus Rift. See Section 3.6.1, for more info on the trackers used in the experiment. One laptop was also used by the participants to answer the questionnaires.

5.3 Experimental Procedure

When the participant entered the room they were asked to sit on a chair in the middle of the room, see Figure 5.1. A table was placed in front of them and to their right. The Cup Tracker was placed on the table in front of them, and a laptop was placed on the table to their right, which they used to answer the questionnaires.

The first facilitator of three stood in front of the participant. This facilitator gave the participants instructions and was also the one who handled the candy. The second facilitator handled the computer with the system running on it. The third facilitator observed the experiment and took notes. A video camera was located between the second and third facilitator looking at the test participant.

When the participant had taken their seat, they were presented with a consent form giving us access to video record the experiment and collect and use data for the writing of this thesis. After they signed the consent form they were given a brief introduction to the overall experiment orally by the facilitator, and were then prompted to answer the first part of the questionnaire containing demographic questions. The participant was then instructed to put on the tracking gloves and place their hands on the table in order to calibrate them.

Once the hands had been calibrated, the participants were instructed with a minor task in order for them to get used to interacting with a physical object while being in virtual reality. To do this, we placed the tracker at five prespecified locations on the table in front of them, see Figure 5.2. We asked them to equip the HMD and then reach out for it, lift it up, and then place it down again in the same spot. The tracker would then be moved to the next location and the procedure would repeat.

An introduction was then given to the next part of the experiment by the facilitator. Here the three tasks were explained in the order the test participant would experience them, the order can be seen in Table 5.1. After explaining the tasks to the participant, the facilitator prompted the participant to start. This was done either by the facilitator giving the participant a piece of candy or by the participant grabbing the plastic cup in front of them, depending on the task. Once done with the task, the participant were asked to answer the next part of the questionnaire, see Section 5.1.1. At any time the facilitator handled candy, he wore disposable rubber gloves.

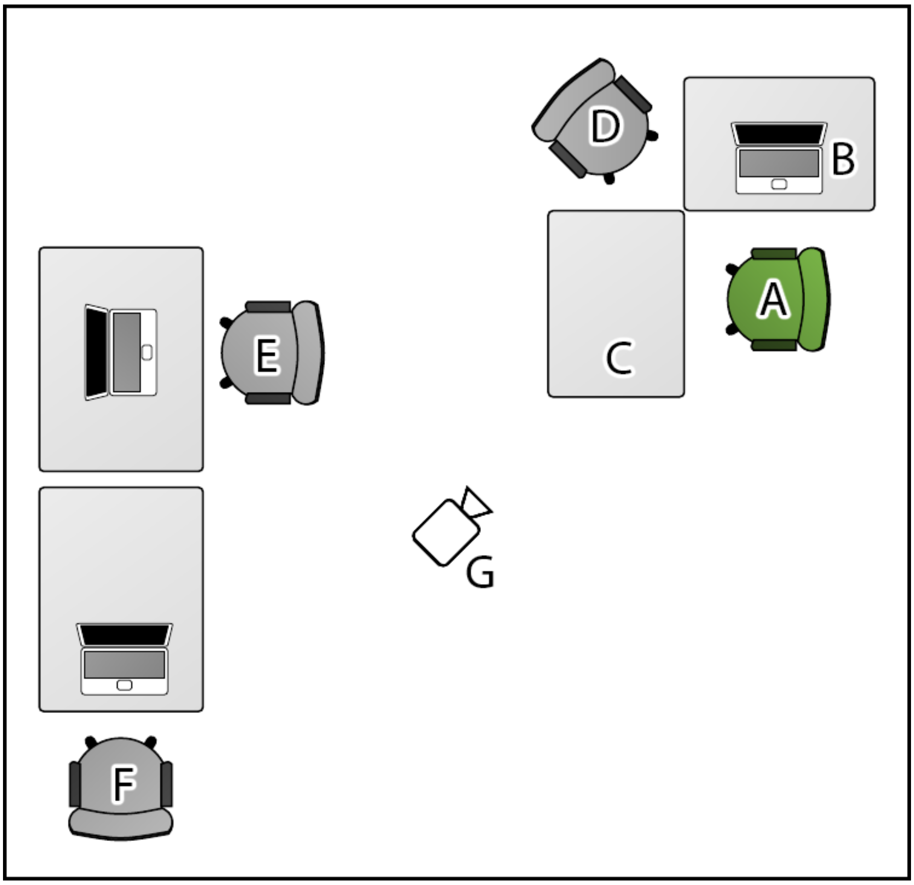


Figure 5.1: (A) The chair the test participants were asked to sit during the experiment. (B) The laptop participants used to answer the questionnaire. (C) The main table used in the experiment for calibration and during the all tasks. (D) The location of the main facilitator. (E) The chair where the second facilitator sat, and the computer running the system. (F) The third facilitator who observed the experiment and took notes. (G) The location and orientation of the camera.

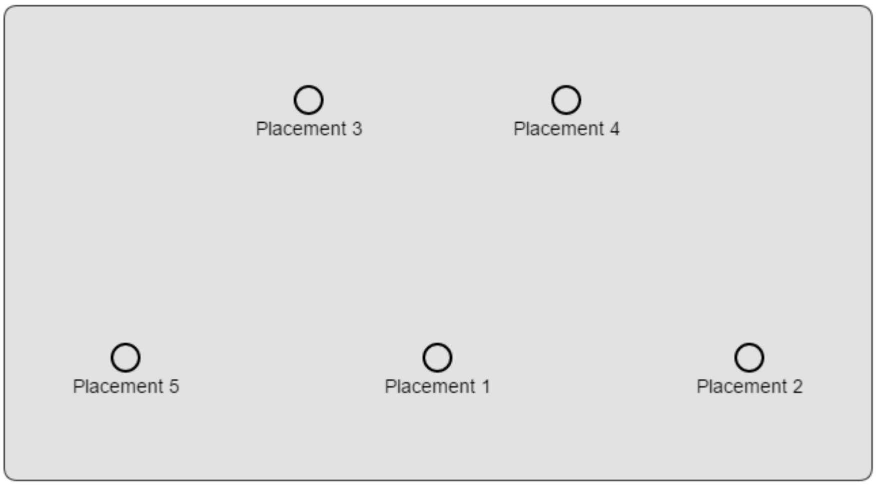


Figure 5.2: An illustration approximating the 5 placements of the Tracker.

Next, the participants had to do all three tasks again, this time outside of the virtual environment, meaning that they did not wear the HMD. In between each task, they answered the independent task comparison questionnaire, see Section 5.1.2, where they had to compare the task they just completed with the corresponding task in VR. They did this process for all three tasks, and then they answered a final questionnaire, summing up the experiment, see Section 5.1.3. Finally, they were thanked for their help and giving a small reward and they could now leave.

Chapter 6

Results & Discussion

Through the experiment, we gained data regarding presence of the participants, along with their own preference of the various tasks. To re-iterate, the three tasks used in the aforementioned experiment are as follows.

Task 1: In order to place the piece of candy in the mouth, the participant grabs the tracked cup in front of themselves, takes the piece of candy out of the cup and places it in their mouth. After approximately 10 seconds, the participant again takes the cup and spits the piece of candy out into said cup.

Task 2: Instead of the participant taking the piece of candy out of the cup by themselves, the facilitator asks the participant to extend their hand. Once this is done, the facilitator places a piece of candy in the hand of the participant, who may place it in their mouth by themselves. After approximately 10 seconds, the facilitator asks the participant to extend their hand again, in which the facilitator places the tracked cup. The participant then places the cup by their mouth and spits the piece of candy into the cup.

Task 3: Instead of the facilitator placing the candy in the hand of the participant, the facilitator asks the participant to close their eyes and open their mouth. Once done, the facilitator places a piece of candy within the mouth of the participant. After approximately 10 seconds, the facilitator asks the participant to again open their mouth to which the facilitator hold the cup. The participant then spits the piece of candy out into the cup.

6.1 Participants

16 participants were recruited for the experiment. All were students of Medialogy at Aalborg University. 4 out of 16 (25%) reported to being prone to motion sickness, yet no one reported any symptoms during the experiment. 13 out of 16 (81.3%) had prior experience with an Oculus Rift or any other virtual reality device. Most people said they only tried it a few times for small periods of times and a few had worked with it.

Ideally, we would want a larger sample size. However, we argue that due to the significant trends we are able to observe from our relatively small sample size, the general assumptions we are about to present may be true even for a larger sample size.

6.2 Presence Questionnaire

Due to the fact that we have used a standardized presence questionnaire, used by Usoh et al., the mean values achieved for each question is of importance. See Table 6.1 for mean - and standard deviation values.

Table 6.1: Mean and Standard Deviaton of Presence Questionnaire Scores.

Group	Q1	Q2	Q3	Q4	Q5	Q6
Task 1	4.94 ± *1.18*	4.31 ± *1.66*	3.87 ± *1.93*	5.00 ± *1.59*	4.50 ± *0.89*	4.31 ± *1.89*
Task 2	4.37 ± *1.58*	4.06 ± *1.65*	3.69 ± *1.96*	4.94 ± *1.48*	4.00 ± *1.09*	4.56 ± *1.55*
Task 3	3.00 ± *1.75*	2.56 ± *1.79*	3.37 ± *1.71*	3.44 ± *2.19*	3.25 ± *1.73*	3.19 ± *2.04*

By analyzing the mean values, it is very noticeable that for all questions, Task 3 scored the lowest, being the task where the facilitator placed the piece candy inside the mouth of the participant. Compared to the other two tasks, one may not find it surprising that the condition with the least control for the participant scored the lowest presence score. When feeling another person, who is not represented within the virtual environment, the feel of presence may fade. This is the trend that we are able to observe from our data.

If observing the Standard Deviation (SD) for each question, it is noticeable that Task 3 also have the only two SDs above 2.00 (which are for $Q4, mean = 3.44, SD \pm 2.19$ and $Q6, mean = 3.19, SD \pm 2.04$.) This indicates, that participants are more divided on their opinion of this task. Some might not feel that bad about having another person put the piece of candy placed into their mouth, while others certainly do.

The Likert scale scores gained through the Presence Questionnaire may show if there is any significant difference in the objective success of the three tasks. As such, the scores gathered from each question were subjected to a Kruskal–Wallis one-way analysis of variance, see Table 6.2. The Kruskal-Wallis is a non-parametric statistical method. This method is used to compare two or more independent samples. In our case, we have samples of equal size, since we compare the the three tasks used in the experiment.

Table 6.2: Kruskal-Wallis p-values for the Six Presence Questions.

Group	Q1	Q2	Q3	Q4	Q5	Q6
p-value	0.0065	0.0145	0.7285	0.0775	0.0861	0.1062

A post hoc Wilcoxon signed rank test was performed for Q1 and Q2, since the Kruskal–Wallis one-way analysis of variance shows that at least one task from Q1 and Q2 stochastically dominates the other two ($p < 0.05$). The Kruskal-Wallis method can only tell as such, which is why we need post hoc analysis. For each question, we made three post hoc comparisons; $Task1 \times Task2$, $Task1 \times Task3$, and $Task2 \times Task3$, see Table 6.3.

The Wilcoxon signed rank test does not assume the presented data sample pools yield normal distribution. As with the Kruskal-Wallis one-way analysis of variance, this is due to the non-parametric nature of this specific statistical method. The data sample pools are paired, i.e. sample one in Task 1 and sample one in Task 2 comes from the same participant. This is another assumption which the Wilcoxon signed rank test makes.

Table 6.3: The Wilcoxon Signed Rank Test p-values for the Presence Questionnaire.

Group	Q1	Q2
Task 1 × Task 2	0.0898	0.5859
Task 1 × Task 3	0.0032	0.0134
Task 2 × Task 3	0.0103	0.0205

To account for the multiple comparisons problem, the Bonferroni correction were used, meaning that to gain a significant difference, the p-value has to be lower than $0.05/3 = 0.01667$.

An interesting observation by reviewing Table 6.3 is that all comparisons including Task 3, except $Task2 \times Task3$ for Q2, yields a significant p-value ($p < 0.01667$). This indicates that Task 3 is the least successful of the three in achieving presence of the three tasks.

6.3 Independent Task Comparison

Besides the presence questionnaire, the participants were told to answer a series of questions regarding comfortability, intrusiveness, and disruptiveness (we abbreviate these to CID) when comparing the virtual and real versions of the three tasks. As with the presence questionnaire, these CID questions were also answered on a Likert scale. The following answers are as such interpreted as the relative comfortability, intrusiveness, and disruptiveness of the *virtual condition*. We can see the mean values from all tasks and questions are around average ($mean \approx 4.5$), except for Task 3's comfortability rating ($mean = 3.81$) and intrusiveness rating ($mean = 3.19$), see Table 6.4.

Table 6.4: Mean and Standard Deviaton of Task Comparison Questionnaire Scores.

Group	Comfortability	Intrusiveness	Disruptiveness
Task 1	4.19 ± *1.52*	4.50 ± *1.50*	4.81 ± *1.33*
Task 2	4.25 ± *1.07*	4.56 ± *1.55*	4.62 ± *1.20*
Task 3	3.81 ± *1.97*	3.19 ± *1.60*	4.06 ± *1.69*

The scores gathered from each question were subjected to a Kruskal–Wallis one-way analysis of variance, see Table 6.5.

Table 6.5: Kruskal-Wallis p-values for the Task Comparison Questionnaire.

Group	Comfortability	Intrusiveness	Disruptiveness
p-value	0.4719	0.5835	0.3844

The Kruskal–Wallis one-way analysis of variance shows no significant difference for any of the questions in the questionnaire. For comfortability, this may be due to the level of control entrusted to the participant in these specific tasks. We can, however, observe a trend in that the mean-values decrease from Task 1 to Task 2, and then again to Task 3. This indicates that Task 3 is the least comfortable for the participants. A reason for the above high comfortability mean-values for all tasks might be found the the types of participants recruited for the project. All participants were gathered from the same study as the facilitators. As such, none of the participants were total strangers. Had this not been the fact, we might get other results for the comfortability.

As a whole, we get the lowest mean for Task 3 for all three questions, presumably as a result of control being taken away from the participant or because the participants personal space is in conflict.

6.4 Task Preference

When asked the question *"Which method of 'eating' candy did you prefer?"*, 15 out of 16 participants preferred Task 1, the task where they had to do everything themselves. The one participants who did not prefer Task 1, preferred Task 2 instead and wrote the following as a possible explanation (direct quote):

> *. . . task 2 seemed to me best because i did not have to think about getting hold of the candy. maybe i would prefer task 1 if i the mapping between what i see and where my hand goes was exact.*

We read this comment as if the tracking between the real-world and the virtual hand had been 1:1, or closer to it at least, this particular participant would have preferred Task 1 instead. The state of the system used during the experiment did not account for participants' hand size or fine adjustment of the position of the tracker on the hand, meaning that the position might have been a bit off as well. This could have resulted in the tracking not being 1:1. During the experiment, other participants also commented on the not entirely accurate tracking of the hands, but these participants still preferred Task 1, leading us to believe that totally precise tracking is not necessary.

Going in to this experiment, we as facilitators expected that the hierarchy of task preference equaled Task 1 → Task 2 → Task 3. Since all but one participant preferred Task 1, we can confidently assume that this task is the most preferred. As mentioned previously, Task 3 scored the absolute lowest of the three tasks for both questionnaires, indicating that Task 3 indeed was the least preferred.

6.5 Observations

In the first part of the experiment, participants reached for the cup in order to prime them with the interaction modality of our virtual reality. The following data from this test is purely observational, as we did not log any data during the test itself. Rather, we video recorded the test and analyzed upon this based on the interactions we saw. When the participants reached for the cup, their confidence levels varied quite a lot. Some were very careful, inching their hand forward until they touched the cup, while other were faster. This is also reflected in their accuracy when reaching for the cup. We classified the accuracy of each participant when they reach for the cup, as well as the visual confidence shown by them using the classifications show in the list below and in Figure 6.1. If participants grabbed the cup perfectly in the first try, we classified the reach as neutral.

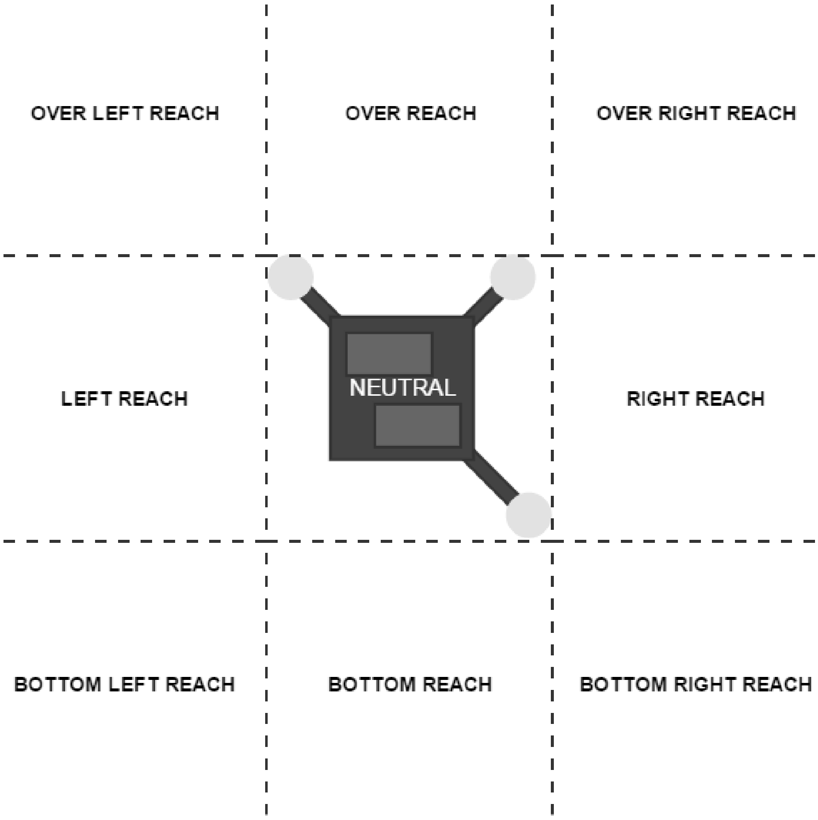


Figure 6.1: Reach Classification.

N = Neutral

O = Over Reach

R = Right Reach

B = Bottom Reach

L = Left Reach

ON = Over Neutral Reach

OR = Over Right Reach

OL = Over Left Reach

BN = Bottom Neutral Reach

BR = Bottom Right Reach

BL = Bottom Left Reach

LN = Left Neutral Reach

RN = Right Neutral Reach

When listing the various reach of the participants, as in Table 6.6, we are able to observe that the placement with the most occurrences of **Neutral Reach** is the last of the five placements (denoted as *P5* in the table). Here, 50% of all participants are denoted as Neutral Reach, compared to the other placements P1 = 31.25%, P2 = 18.75%, P3 = 31.25%, and P4 = 25.00%. This may indicate that there is a learning curve with the virtual reality and the method of interaction within it. The participants seemingly learned how to interact with the cup as time progressed, which was the desired effect of this priming test. Also, 9 out of 16 participants (56.25%) showed signs of being unconfident while performing the priming test.

Table 6.6: Reach of the Participants During Distance Priming Tasks.

PID	P1	P2	P3	P4	P5	CONFIDENCE
1	N	N	N	N	O	CON
2	L	BL	N	BL	N	UN
3	BL	BL	BL	BL	L	CON
4	BL	BL	B	B	R	UN
5	N	BL	N	BR	RN	CON
6	N	BR	B	B	N	CON
7	BR	BL	BL	B	N	UN
8	B	BL	BL	B	BL	UN
9	B	L	B	N	N	CON
10	B	BL	B	B	N	UN
11	B	N	N	N	N	CON
12	N	BL	N	L	N	CON
13	B	BL	B	N	R	UN
14	N	B	B	B	N	UN
15	BL	N	BL	O	L	UN
16	OL	L	BL	BL	BL	UN

Steinicke et al. found that users of virtual reality system may decompress their perceived distances with as much as 50%. If this is the case, we should be able to observe participants reaching shorter than they should in our results, indicating a large sample size of **Bottom Reach** classifications, and its variations. Classifications **Bottom Left** (*BL*) and **Bottom** (*B*) both have the second and third most frequent occurrences respectively, see Table 6.7. This correlated with the findings of Steinicke et al., indicating that participants tended to underestimate the egocentric distance from themselves to the cup. Many of the participants whom showed being unconfident when completing the priming tasks also tended to reach the for the variations of Bottom, see Table 6.6.

Participant 8 was arguably the most unconfident of all participants, and when reviewing the classified reach for said participant, all of them are variations of Bottom. Participant 4 also showed being unconfident during the priming tasks, and when reviewing the classified reach for this participants, 4 out of 5 of the tasks showed variations of Bottom as the classified reach. This trend is also observable for other participants. This indicates that the system created may skew the perceived distance for the participants, influencing their reach for the cup during the tasks. This may be a result of the rather static hand gestures the system ended up with. Participants either saw a long, flat hand or a pinched hand depending on where their hand was in physical space (controlled by a second facilitator during the experiment). If the virtual hand was larger than the hand of the participant, who may have a slightly curled up hand in physical space, this would create a dissonance between the hand of the participant and the virtual hand. This dissonance may be quite large, resulting in the underestimation of the participant's perceived distance, see Figure 6.2.

Another factor we are able to observe from our experiment, which supports the findings of Steinicke et al. is that very few participants overreached when grabbing the cup. Referring back to Figure 5.2, Placement 2 is located to the right of the participant. When reviewing the reach for P2 in Table 6.7, no participants overreached to the right of this specific placement. However, many participants reached to either the **Left** (12.50 %) or **Bottom Left** (56.25 %) of Placement 2, indicating that distance estimation may also

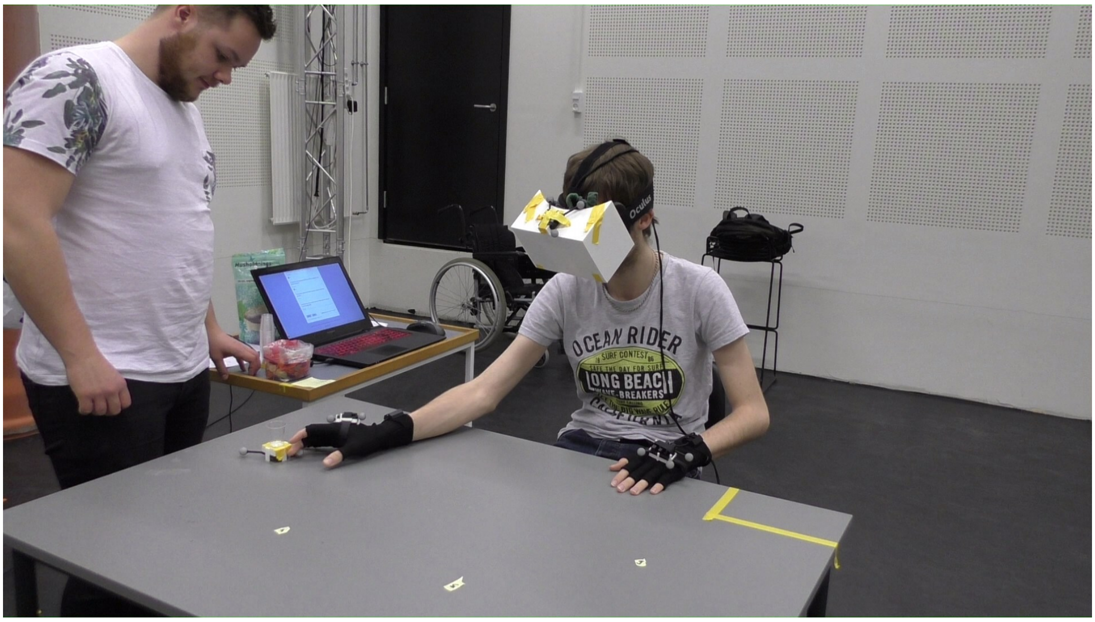


Figure 6.2: Faulty distance estimation by Participant 10.

Table 6.7: Percentage of Frequency of Reach Classifications. * Without Occurrence. ** Slight Occurrences.

Reach	**P1**	**P2**	**P3**	**P4**	**P5**
N	31.25 %	18.75 %	31.25 %	25 %	50 %
O**	0 %	0 %	0 %	6.25 %	6.25 %
R**	0 %	0 %	0 %	0 %	12.50 %
B	31.25 %	6.25 %	37.5 %	37.5 %	0 %
L	6.25 %	12.50 %	0 %	6.25 %	12.50 %
ON*	0 %	0 %	0 %	0 %	0 %
OR*	0 %	0 %	0 %	0 %	0 %
OL**	6.25 %	0 %	0 %	0 %	0 %
BN*	0 %	0 %	0 %	0 %	0 %
BR**	6.25 %	6.25 %	0 %	6.25 %	0 %
BL	18.75 %	56.25 %	31.25 %	18.75 %	12.50 %
LN*	0 %	0 %	0 %	0 %	0 %
RN**	0 %	0 %	0 %	0 %	6.25 %

be compressed in arc around the participant. this is supported by the reach for Placement 5, where a couple of participants reached to the right of the placement, showing the same trend of misinterpreting the placement of objects in arc around the participants.

6.5.1 Other Observations

The system used during the experiment did not allow for finger tracking, therefore, we had two hand gestures for each hand (open palm and a pinch) that the facilitator could switch between at will, ideally to match what the participant did in the real world. We observed that one participant seemed to only have his hands in the gestures of the virtual hands, where he matched his own hands to the virtual, instead of us matching the virtual hands to his own hands. It is a rare occurrence caused by not having finger-tracking, but might impact results for some participants. We could not see any difference on his results compared to others however.

When observing the behaviour of the participants, we noted that a large group of participants reacted either humorously or reservedly to the information given about Task 3. The idea of having a facilitator placing an object within the mouth of the participant is relatively unknown at the facility of testing, i.e. our university. The facilitator explained thoroughly the upcoming procedure with Task 3, as well as Tasks 1 and 2, in order to be sure participants all knew the tasks. During Task 3, most participants reacted calmly to the task. They knew what was going to happen, and were prepared. However, a couple of participants found said task very odd, and thus, found it hard to remain calm. This indicated to us that this specific task would not work in a TSST, due to the fact that a high level of presence is needed to conduct said test optimally.

Chapter 7

Conclusion

In this project, we created a system that could be used in conducting a virtual reality Trier Social Stress Test (VR-TSST). Through many iterations of both software and hardware, we ended up using an OptiTrack system for hand tracking, head tracking, and object tracking. An Oculus Rift DK was used as the HMD. The system allows for quick and easy change of different environments, and a committee, all required for the given VR-TSST. One problem presented itself, in that not seeing the physical world could complicate participant's process of delivering a saliva sample. As such, we conducted an experiment, investigating different interaction methods by having participants perform various tasks.

Task 1: The participant grabs the piece of candy and places it in their mouth. After a while the participant spits it out into a cup.

Task 2: The facilitator places a piece of candy in the hand of the participant, who may place it in their mouth. After a while, the facilitator places the cup in the hand of the participant, who then spits the piece of candy out into a cup.

Task 3: The facilitator places the candy in the mouth of the participant. After a while, the facilitator places the cup in front of the mouth of the participant who then spits it out into the cup.

Throughout the experiment, we gathered data through various questionnaire as well as observations through video recordings. One of these questionnaires were regarding the amount of presence experienced during the various tasks. High presence was desired to make the physiological data as valid as possible. Data revealed that Task 3 yielded lowest amount of presence. A possible issue with the data from our presence questionnaire is that the various tasks that the participants were asked to evaluate presence on were perhaps too short in order for them to achieve presence.

Trends can be observed when having participants measure the level of comfortability, intrusiveness and disruptiveness when doing a task in Virtual Reality compared to doing the same task in real life. Results shows that Task 1 and Task 2 are better than Task 3, indicating that leaving some control in the hand of the participant is preferred. Also, Task 3 requires the facilitator to invade the participants personal space, as such, Task 1 and Task 2 is preferred here as well.

All but one participant chose Task 1 as their preferred task. The other two tasks took control away from the participants, which may be the deciding factor in preference. The one participant who did not choose Task 1 as his preferred task stated that his choice was due to inaccuracy of the hand tracking and as such, would probably choose Task 1 if it were more accurate.

Th first time participants were instructed to reach for the Cup Tracker, only a third grabbed it in their first try. However, at the last placement, half of the participants grabbed the cub in their first try. This shows that our priming period helped the participants to better reach the cub tracker, but it also shows that the hand tracking is not 1:1 with the real world. But was close enough for people not to feel that they where totally missing the cup tracker.

Overall, as a result of the Perception Neuron not being suitable for this project, we find the quality of the project below expectations. This is because of the long time spent learning and working with the Perception Neuron, which we ended up discarding. The system we ended up with, however, does seem to work to a satisfyingly degree, as data and observations gathered from the experiment show trends that the system is viable. As for the interaction experiment, all our results point toward a method without facilitator assistance yielding the best outcome. Many of the things that would have improved this project will be elaborated upon in the next chapter.

Chapter 8

Further Development

During this project, a lot of elements ended up being cut from the final system. Many of the decisions we made as a group were hard, as they removed many of the desired core elements initially planned. The biggest element to be cut from the system was the incorporation of the Perception Neuron. Due to all of the technical difficulties endured throughout the project caused by the Perception Neuron ultimately drove the deciding factor.

Further development of this system would include a more optimized calibration of the Perception Neuron than conceived for the present project. As an afterthought we think that using quaternions will maybe yield a better solution to the calibration problems, as quaternion angles can be multiplied together, while the used Euler angles cannot. The equation would look something like this:

$$currentRotation * savedRotation^{-1} * savedCalibrationRotation \tag{8.1}$$

If the hand calibration is perfected, perhaps by use of said quaternion angles, the Perception Neuron may still be of use for the system as a whole. If more time was allocated to this project, this solution might have been possible to implement.

We believe we achieved the best results with the hardware available to us. Using the OptiTrack system for moving the virtual camera position introduced a lot of latency that could potentially induce motion sickness when moving around in the virtual environment. No participants commented on this during the experiment, though this is most likely because they were sitting down the entire time and not moving around. When one of the authors of this thesis walked around the environment he started to feel motion sick due to the latency. He had not felt motion sick when exposed to a virtual environment in previous experiences.

Therefore, using hardware capable of doing room-scale virtual reality, such as the HTC Vive, would have meant that the system did not need the OptiTrack System to track the position of the camera, only the hands. As a result, the latency would have would have been decreased and the potential for motion sickness would also have decreased. Furthermore this allows us to create two separate rooms which the participants could walk to and from. In a real TSST, two rooms are needed, a preparation room and an experiment room. Something like the HTC Vive allows for a $5.0m \times 5.0m$ large virtual room.

The preparation room could then be a small cubicle, in which the participant can sit and prepare the presentation to be given to the committee. As for the committee to better minick the CAVE environment the committee have to sit down behind a table but as we do it now the committee is standing and facing the participant. With the committee facing the participant, a potential problem can arise. If the committee are sitting down, the committee will also follow the participant's orientation. This can be hidden by giving the table a cloth to cover the legs and chairs from the participant, which results in the participant not being able to see the slight movement of the committee's orientation.

We only showed the hands of the digital avatar to the participants. According to Sanchez-Vives and Slater and Mohler et al., even crude and rough animated full-body avatars increases presence and distance estimation, respectively. We do not label the floating hands that the participants are able to see as full-body avatar, which makes this a lacking feature in our developed system. Further development of the present project would see the incorporation of a full-body avatar, moving authentically in accordance with the physical movement of the participants. In initial testing of our current system, we still had the imported Axis Neuron avatar which was a full-body avatar. It moved in accordance with the Perception Neuron's IMU-sensors. This allowed the user of the system to see its own movement in real time within Unity, and would as such yield a higher level of presence.

Besides the lack of a full-body avatar during the experiment, the hands of the participants were semi-static. The hands moved to the relative virtual position of the participant's hands in physical space. However, the hands were always extended as a flat hand, and only pinched together when the participant's hand was close to the Cup Tracker during testing. The pinching motion the hands made was controlled by a button being pressed by a second facilitator. This may also contribute to the lack of presence within our developed system.

Further work on this particular project may also include a more fleshed out version of the facilitator-side of the system. Status quo of the facilitator-side is somewhat superficial, even though the core elements needed for a complete TSST were implemented. As we did not conduct full TSSTs, we may not conclude upon the usability of the system we already have. We can only make assumptions as to how well the facilitator-side of the system functions. The system has to be usable for potential facilitators whom have no prior experience with this specific system.

In terms of the environment itself, Steinicke et al. found in their study that if participants, who are exposed to non-realistic environments, may benefit of entering a digital rendition of the physical space of which they were currently occupying first. Once immersed within this familiar environment, the participants may press a button porting them to the non-realistic setting. This, through Steinicke et al.'s findings, theoretically optimizes the perceived egocentric distance for the participants. Early in the process of developing the system used for the present project, we developed a rough rendition of a portal-style door through which participants could walk. This portal never finished development, due to time being allocated to the Perception Neuron and OptiTrack system instead.

Bibliography

[1] Lars Brorson Fich, Peter Jönsson, Poul Henning Kirkegaard, Mattias Wallergård, Anne Helene Garde, and Åse Hansen. Can architectural design alter the physiological reaction to psychosocial stress? A virtual TSST experiment. *Physiology & behavior*, 135:91–97, 2014.

[2] V. Interrante, B. Ries, and L. Anderson. Distance Perception in Immersive Virtual Environments, Revisited. In *Virtual Reality Conference, 2006*, pages 3–10, March 2006. doi: 10.1109/VR.2006.52.

[3] Charlene Jennett, Anna L. Cox, Paul Cairns, Samira Dhoparee, Andrew Epps, Tim Tijs, and Alison Walton. Measuring and defining the experience of immersion in games. *International journal of human-computer studies*, 66(9):641–661, 2008.

[4] Betty J. Mohler, Heinrich H. Bülthoff, William B. Thompson, and Sarah H. Creem-Regehr. A full-body avatar improves egocentric distance judgments in an immersive virtual environment. In *Proceedings of the 5th symposium on Applied perception in graphics and visualization*, page 194. ACM, 2008.

[5] Betty J. Mohler, Sarah H. Creem-Regehr, William B. Thompson, and Heinrich H. Bülthoff. The effect of viewing a self-avatar on distance judgments in an HMD-based virtual environment. *Presence: Teleoperators and Virtual Environments*, 19(3):230–242, 2010.

[6] Oluf Falkenberg Nielsen. *Anatomi og fysiologi (Hånden på hjertet)*. Gyldendal Akademisk, 2012.

[7] Maria V. Sanchez-Vives and Mel Slater. From presence to consciousness through virtual reality. *Nature Reviews Neuroscience*, 6(4):332–339, 2005.

[8] Frank Steinicke, Gerd Bruder, Klaus Hinrichs, Markus Lappe, Brian Ries, and Victoria Interrante. Transitional environments enhance distance perception in immersive virtual reality systems. In *Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization*, pages 19–26. ACM, 2009.

[9] Unity Technologies. Unity - Manual: Prefabs. URL `http://docs.unity3d.com/Manual/Prefabs.html`.

[10] William B. Thompson, Peter Willemsen, Amy A. Gooch, Sarah H. Creem-Regehr, Jack M. Loomis, and Andrew C. Beall. Does the Quality of the Computer Graphics Matter when Judging Distances

in Visually Immersive Environments? *Presence: Teleoperators and Virtual Environments*, 13(5): 560–571, October 2004. ISSN 1054-7460, 1531-3263. doi: 10.1162/1054746042545292.

[11] Martin Usoh, Ernest Catena, Sima Arman, and Mel Slater. Using Presence Questionnaires in Reality. *Presence: Teleoperators and Virtual Environments*, 9(5):497–503, October 2000. ISSN 1054-7460. doi: 10.1162/105474600566989.