UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Tõnis Kristian Koppel

Reduced Contrast in the Area of the Visual Field Where the Own Hand Is Moving: A Virtual Reality Study

Bachelor's Thesis (9 ECTS)

Supervisor: Jaan Aru, PhD
Supervisor: Madis Vasser, MSc
Supervisor: Kristjan-Julius Laak, MSc

Tartu 2017

Reduced Contrast in the Area of the Visual Field Where the Own Hand Is Moving: A Virtual Reality Study

Abstract: From an evolutionary standpoint, being able to quickly detect motion within the visual field is very important. However, when an agent is moving, a large part of motion consists of moving parts of the agent himself. The results of numerous studies suggest that the brain predicts the results of its movements and attenuates the sensory signals resulting from those movements. Using Oculus Rift virtual reality (VR) headset and a Leap Motion hand tracking device, we can study the aforementioned phenomenon. As a part of this thesis, a program was developed that enables to test whether self generated hand movement affects the perceived contrast in the area of visual field where the hand is moving. We conducted a VR experiment where we monitored the hand positions during movements while the subjects had to perform a two-alternative forced choice task - they answered the side of the stimulus which seemed higher in contrast. The data gathered from 11 participants in the study did not provide enough evidence to confirm that the brain attenuates the apparent contrast of objects in the area of the visual field where the own hand is moving. It was concluded that further studies with slightly modified experimental design and more participants are necessary.

Kontrasti alanemine nägemisvälja osas, kus enda käsi liigub: katse virtuaalse reaalsusega

Lühikokkuvõte: Evolutsioonilisest seisukohast on liikumise kiire tuvastamine vaateväljas väga oluline. Samas, kui indiviid liigub, siis suur osa liikumisest tema vaateväljas on tema enda poolt tekitatud. Mitmete uurimuste tulemused viitavad sellele, et aju ennustab enda liigutuste tulemusi ette ja pidurdab neist liigutustest tekkivaid sensoorseid signaale. Kasutades Oculus Rifti virtuaalreaalsusprille ja Leap Motion käe jälgimise seadet, saame me eelpoolnimetatud fenomeni uurida. Selle bakalaureusetöö raames valmis programm, mille abil saab leida vastust küsimusele, kuidas mõjutab enda käe liikumine tajutavat kontrasti nägemisvälja selles osas, kus käsi liigub. Me viisime läbi VR katsed, kus jälgisime katseisikute käe positsiooni sel ajal, kui nad sooritasid ülesannet, kus pidid raporteerima kahest stiimulist kontrastsemana näiva stiimuli poole. 11lt katseisikult kogutud andmete põhjal ei saa kindlalt väita, et objektid, mis asuvad nägemisvälja selles osas kus enda käsi liigub, tunduksid vähem kontrastsed. Katsete tulemuste põhjal järeldati, et täiendavad uuringud suurema valimi ja veidi modifitseeritud katse disainiga on vajalikud.

Võtmesõnad:
Oculus Rift, Leap Motion Controller, Unity, virtuaalreaalsus, sensoorne allasurumine

CERCS: P175, Informaatika, süsteemiteooria

Contents

Introduction

For animals, quick detection of motion within the visual field is recognized as an evolutionarily adaptive trait. Successfully identifying potential dangers in an environment can be a matter of life and death. However, when moving, a large part of motion in the visual field consists of moving parts of the agent itself. Moreover, focusing too much attention on own movements can hinder the agent from noticing something important.

Results of numerous studies have shown that the brain predicts the results of its movements and attenuates the sensory signals resulting from those movements [1, 2, 3]. Furthermore, it is suggested that the brain withdraws attention from the area of the visual field where own hand is currently moving [3]. It has also been noted that attention alters the appearance of objects, notably boosting the contrast of the attended stimulus [4].

Combining the two notions - that attention alters appearance and that attention is withdrawn from the area of the visual field where own hand is moving - this thesis helps to seek answer to the question: does the withdrawn attention in the area of visual field where self-generated hand movement is taking place lead to alteration of appearance of objects in that area?

The purpose of this thesis is to develop a program that enables to test whether self generated hand movement affects the perceived contrast in the area of visual field where the hand is moving. The program is written using Unity game engine and is partially based on work done by Kristjan-Julius Laak, who developed a VR environment to study reaction times in the area of visual field where the hand is moving [3]. The main idea of the experiment is as follows: while wearing a VR headset and unable to see their hands, the participants are asked to raise their hand in front of their eyes. While doing that, the hand position is being tracked by a sensor. If the hand reaches a certain spot two stimuli appear on the screen, one of them being behind the (unseen) hand. The participant is then asked questions about the contrast of the stimuli.

The structure of this thesis is organized as follows. Section 1 consists of descriptions of the hardware and software used to program the experiment. Generating the stimuli as well as adjusting and measuring their contrast is covered in Section 2. The experiment setup and the functionalities of the program are covered in Section 3. The pilot study is described in Section 4.

1   Technologies used

The application was developed using Oculus Rift VR headset, Leap Motion hand tracking device, and Unity game engine software. This chapter gives a brief overview of these technologies and specifications which are important in the context of this study.

1.1   Unity engine

For developing the application Unity Engine version 5.5.2f1 was used. Unity is one of the most used game engines on the market today. It allows the user to make 2D or 3D games for a lot of different environments like web, console, mobile and desktop. Unity has built-in support for multiple VR devices, including Oculus Rift. When a headset is connected and VR is enabled, any in-game camera's view can be rendered for both eyes of the VR device. Optimizations, such as applying image effects to eyes at once, are automatically made by Unity engine to make rendering for both eyes faster [5].

Unity projects consist of scenes - unique scenes can be menu screens and different levels. One important concept in the Unity Editor is a GameObject [6]. Every object in a scene is a GameObject - it can be a character, special effect or an environment. GameObjects can be thought of as a containers where different components, such as scripts or Unity's in-built components, can be added. Each GameObject placed in a scene has a Transform component where its scale, rotation and position can be adjusted. Unity's unit of measurement is called a unit. 1 unit in the game world corresponds to 1 meter in real world. This is an important aspect to consider when developing a VR application. Unity supports two scripting languages: C# and UnityScript - a language modelled after JavaScript designed to be used with Unity [7].

1.2   Oculus Rift

For the experiment we use Oculus Rift Consumer Version 1 virtual reality headset. The Rift contains a 2160x1200 OLED display with 1080x1200 resolution per eye and a refresh rate of 90 Hz [8]. The headset uses HDMI and USB cables for transferring data. According to the Oculus Rift website, the horizontal field of view (FOV) of the Rift is 100 degrees, but it can vary depending on the distance from eyes to the lenses inside the head-mounted display [9]. A small test conducted at our lab showed that the average perceived FOV of 6 subjects was about 90 degrees. For our study the hue and vibrance of the head-mounted display (HMD) screen were set to the default settings.

Rift's rotation (yaw, pitch, and roll movements) are tracked by built-in gyroscope, accelerometer and magnetometer. There are numerous infrared micro-LEDs integrated into the headset which are used to track the position of the head-mounted display using an infrared camera. The infrared camera is usually stationary on the table and connected to the computer via USB [10].

1.3   Leap Motion sensor

For hand tracking we used Leap Motion sensor. It is a small, 8cm x 3cm hand tracking sensor that uses 3 infrared LEDs and 2 infrared cameras to track hand and finger velocity, position, and accuracy. In our study the camera was attached to the front of the Oculus HMD but it can also be used on desktop or built into devices, for example HP produces laptops and keyboards with built-in Leap Motion devices [11].

In addition to the hardware device the Leap also consists of a software component running on host computer - we are using Leap Motion Orion Beta version 3.2 and Leap Motion Unity SKD version 4.1.6. Software component's main tasks are analyzing the images from hardware device and sending the tracking data to other applications. The Leap Motion Unity plug-in connects to this software service to get the data. The plug-in automatically transforms the tracking data from Leap-native right-handed convention to the left-handed coordinate system Unity uses. It also scales distance values to Unity's default units [12].

The Leap has a field of view of 150 degrees with the most effective tracking range being from 0.03 to 0.6 meters in front of the device (see figure 1). When attached to the front of the HMD, the tracking algorithms are optimized to track hands from the HMD perspective. The sensor works best when it has a high-contrast view of the hands [12]. The overall average accuracy of the Leap Controller has been shown to be 0.7 mm [13].
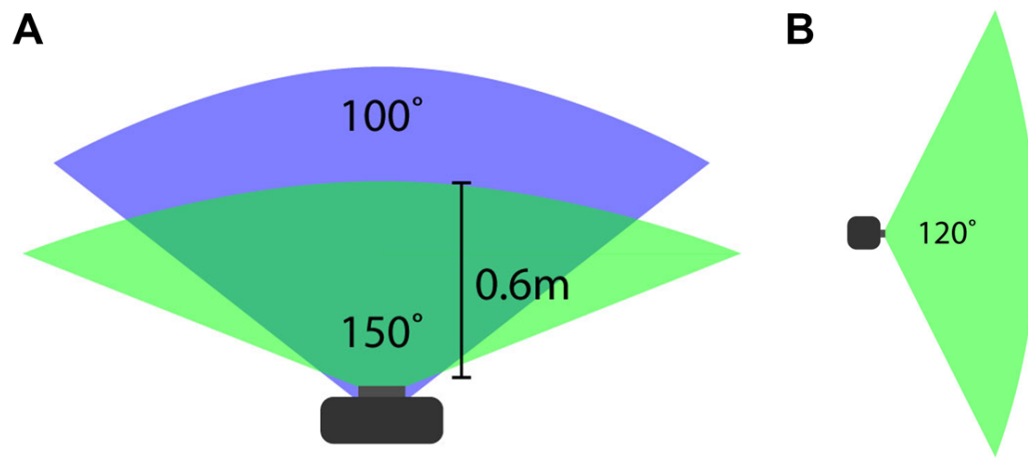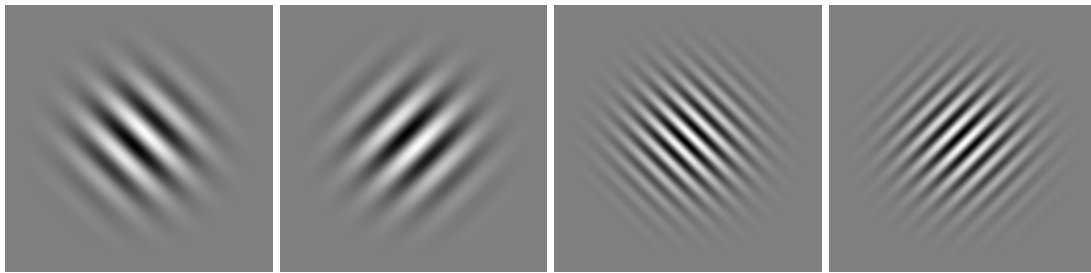
Figure 1. The official horizontal FOV of Oculus Rift is drawn in blue (A). The horizontal (A) and vertical (B) FOV of the Leap Motion Controller is shown in green. Figure from Kristjan-Julius Laak's Master's thesis [14]

2 Stimuli and contrast

Through the experiment the test subjects were shown stimuli and asked questions about their contrast. This chapter gives an overview of stimuli used and how they were generated. Contrast - how it's measured and adjusted - is also covered.

2.1 Stimuli

For stimuli we used Gabor patches. Gabor patches, named after electrical engineer and physicist Dennis Gabor, are sinusoidal gratings with a Gaussian envelope. They are often used in various psychological experiments. Gabor wavelets and filters have a lot of applications in computer image processing and analyzing neurons of the human visual system. The Gabor patch's grating can have different orientation and wavelength. The frequency of the stimulus can be measured in cycles per degree (CPD) meaning how many cycles does the patch have in one FOV degree. The stimuli used in this experiment are shown in figure 2.

(a) 2 cpd stimulus with left orientation (b) 2 cpd stimulus with right orientation (c) 4 cpd stimulus with left orientation (d) 4 cpd stimulus with right orientation

Figure 2. Gabor patches used as stimuli in the study. Note that the perceived spatial frequency (CPD) of the stimulus depends on the size and distance of the stimulus. The numbers indicating the CPD of stimuli reference to how the stimuli are used in the experiment (e.g. stimulus in 2a is used in the experiment as a 2 CPD stimulus.)

2.2 Generating the stimuli

Since stimuli with different parameters had to be tested throughout the development of the experiment, a script was written in Unity using C# that allows user to generate Gabor patches during runtime. A user interface was implemented to enable the user to

run the program without using the Unity editor (figure 3). The program also allows to modify the contrast of the generated stimulus and save generated stimuli as .PNG files. The stimuli used in this study were generated using this program.
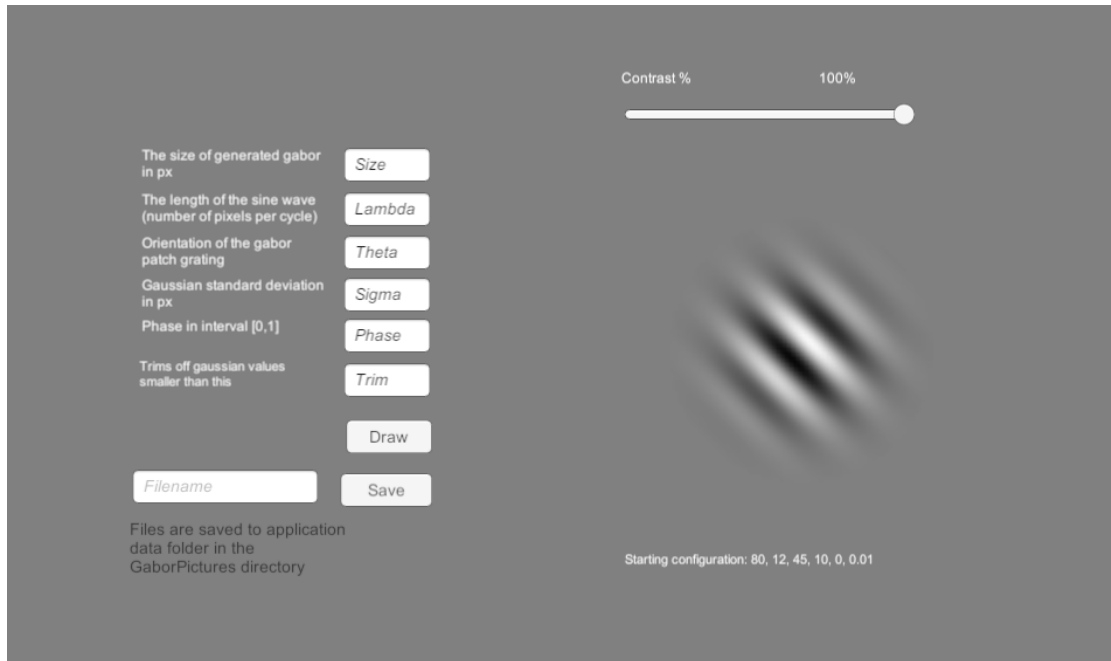

Figure 3. A program for generating Gabor patches

When generating the Gabor patches, the user can specify the size of the generated Gabor patch in pixels, the length of the sine wave (pixels per cycle), orientation of the grating, the size of Gaussian standard deviation, phase of the sinusoidal grating and a trim value, used to trim off pixels whose value is smaller than specified.

2.3   Contrast

A large part of this study consists of measuring and adjusting contrast of stimuli. For this study the particular contrast that interests us is called luminance contrast - the difference in luminance that makes an object or its representation distinguishable.

2.3.1 Luminance

In physics, luminance is the apparent brightness of an object or a surface, measured in candelas per square meter. In computer graphics using relative luminance is more practical. Relative luminance means that luminance values are normalized to 1 or 100 according to a reference white. The luminance value of a color represented in HSL is just its L (lightness) component. Because the human eyes perceive the lightness of different colors differently, the L value of an RBG color can be calculated as a sum of RGB components which have different weights [15]. The exact weights depend on the RGB primaries used. In context of this study, luminance is measured using the following sRGB to luminance formula

$$L = 0.2126R + 0.7152G + 0.0722B. \tag{1}$$

2.3.2 Luminance contrast

When determining the visibility of an object, the absolute luminance of an object is not as important as its luminance relative to background and other areas of the display. The concept of luminance contrast was developed as an engineering concept that enables to describe a stimulus' ability to stimulate the differential mechanisms of the visual system [16]. There are multiple ways of measuring luminance contrast. Which method should be used depends on the stimulus pattern and background of the object.

The main idea of measuring contrast is the same for the different methods. The numerator that describes change between the luminances in question is divided by a denominator that describes the average luminance to which the eye is adapted to.

$$Luminance\ Contrast = \frac{Luminance\ Change}{Adaption\ Descriptor} \tag{2}$$

When working with a small stimulus on uniform background the background luminance affects the adaptation more than the stimulus luminance. When this is the case, a contrast statistic like **Weber contrast** may be used. Weber contrast is one of the oldest measures of luminance contrast and is calculated using the formula

$$C_W = \frac{L - L_b}{L_b} \tag{3}$$

where L represents the luminance of stimulus and L_b the background luminance [16].

Michelson contrast is best for periodic patterns when there is no large uniform area that dominates the brightness adaptation of the viewer. Since Gabor patches fall to this category, this study uses Michelson contrast as a measure. The equation for calculating Michelson contrast is

$$C_M = \frac{L_{max} - L_{min}}{L_{max} - L_{min}} \tag{4}$$

where L_{min} is the lowest luminance of the stimulus and L_{max} the maximum luminance of the stimulus [16].

3 The virtual environment

This chapter gives an overview of the VR environment developed as a part of this thesis. The first section of this chapter focuses on Kristjan-Julius Laak's Master's thesis, because the program made as a part of this thesis is based on a similar program written by him. Secondly, an overview of the experiment is given. Finally, this chapter is going to cover the main functionalities of the program.

3.1 Previous implementation

As a part of his master's thesis [14], Kristjan-Julius Laak developed a program that enables to test people's reaction times to moving stimuli. In his experiment, people were told to raise their hand in front of their eyes while wearing Oculus Rift headset. Test subject wearing the HMD could not see their hands but the hand location was tracked using Leap Motion sensor that was attached to the front of the headset. In the VR environment the test subjects were shown a grid of horizontally moving objects. If the hand was detected to be in a certain area in front of the subject's face, a stimulus was shown. Subject's reaction time to the stimulus was measured. The results of experiments showed that reaction times were slower when the stimulus appeared behind the test subject's hand.

The program developed as a part of this thesis is using some scripts and ideas from Kristjan-Julius Laak's work. Some of the design choices were also done along the lines of Kristjan-Julius's design, so the used components were easier to add to the program. It uses assets that are needed by both programs, for example data logging and hand tracking scripts. Most scripts could not be used as they were, but needed modifications in order to work properly in the context of our program. All used scripts were written in C#.

Since both programs need hand tracking functionalities, the HandController, LeapHandVisibilityController and SwipeController scripts were used. The HandController script is mainly used for tracking the hand position and velocity. LeapHandVisibilityController can be used to toggle the visible hand models - it was used during the development phase for testing the application. The SwipeController script is used to get global notifications about the movement of the hand (for exapmle if the hand is moving up or down, if the hand has turned at the top etc).

For controlling the textual notifications during the experiment UIController, UIFader and WarningController scripts were used. The UIController allows the developer to

display or hide Canvas type GameObjects that contain the Canvas Group and UIFader components. UIFader allows the developer to fade Canvas objects in or out and adjust their fade speed. The WarningController is used to fade UI elements in and out so they are shown for a specific time.

For data logging, SubjectDataController and IdControl scripts are used. SubjectData-Controller is used for saving the test subject's data and the data of experiment trials into CSV-files. IdControl script is used to automatically increment the id of the test subject from test-to-test.

3.2 An overview of the experiment

Before the study, 5 subjects participated in pilot experiments. Different experiment setups and parameters were tested during these experiments to test the program and to refine the design of the experiment.
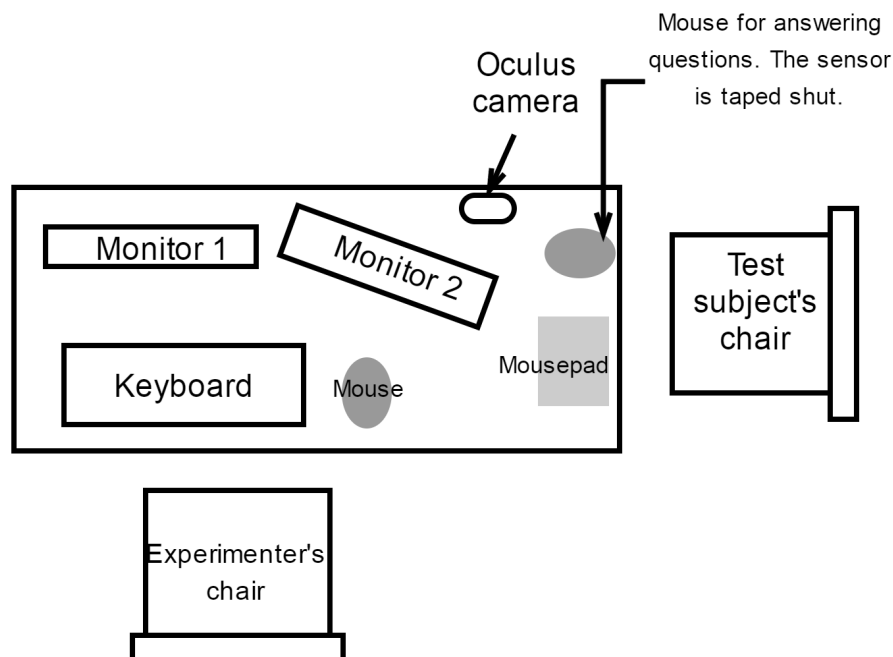
3.2.1 Physical setup

Figure 4. Room setup for the experiment

Before the test subject arrived, the experimenter had to make sure the room was setup as it is in figure 4, with the Oculus headset on the table in front of the test subject's chair and the computer under the table near the test subject's chair. The Oculus screen and the Leap motion sensor were both cleaned with a microfiber screen cleaning cloth and a label was placed on the door that said "Quiet please, experiment in progress." The test subject had to remove jewelry from his/her left hand and the left hand sleeve had to be rolled up to the elbow. Subjects wearing glasses tried to wear them also when using the headset, but if the test subject's glasses did not fit in the headset or the subject refused to wear the glasses with headset, the corresponding option was chosen when in the menu.

3.2.2 Hand training

Before the test subject could put on the VR headset, the hand movement had to be explained to them and practiced. The movement was done with left hand, it had to move straight up and down without stopping at the top (see figure 5). The palm of the hand had to reach a bit higher than test subject's eyes with the fingers pointing up and the back of the hand facing the test subject. The participants were instructed to move the hand at a comfortable pace so that they would not have to focus attention to the hand movement.
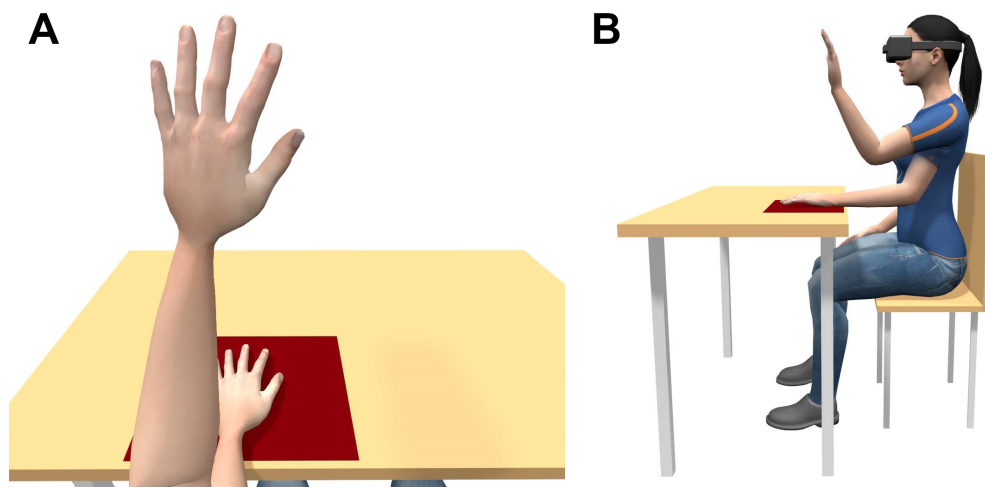

Figure 5. An example of the hand movement with the resting position between movements also shown. Note: in our experiment the hand should touch the **left** side of the mousepad in the resting position. Figure from Kristjan-Julius Laak's Master's thesis [14].

After the hand training, the headset was put on. Subjects were instructed to adjust the headset so that they felt comfortable wearing it and could see the screen clearly. The headset wires had to run from the right side of the test subject so that they would not distract the hand movement.

3.2.3 Vocabulary

The first part of the experiment scene was designed to familiarize the test subject with the terminology and the basics of the experiment. First part of this section was recentering the VR tracking - while the test subject looked straight ahead, screen had to be recentered using 'r' key on the keyboard. The test subjects could center the screen by clicking the mouse. Screen had to be centered so that the fixation point was at the center of the screen when looking ahead. After that the test subject was introduced to the elements of the experiment one after another - the fixation point on which (s)he had to focus during the experiment, the signal to start the hand movement, the stimuli and how to answer to questions were covered.

3.2.4 Practice

In the practice stage the subject could practice performing the experiment. The whole cycle of the experiment trial could be practiced. One experiment trial consisted of following steps:

1. While the fixation point was at the center of the field of view, test subject focused on the fixation point.

2. The fixation point flashing red was the signal to start the hand movement.

3. When the hand movement was correct, the stimuli were shown.

4. After the stimuli were shown, questions were asked about the contrast of the stimuli.

5. After answering the questions, the headset screen faded to black for 500ms. This was done because the visual system adapts to unchanging low contrast stimuli very rapidly and this could result in fading of the visual field.

The number of trials for practice could be set at the menu. Once the specified number of trials had been completed, the experimenter could decide if the test subject should continue to the experiment part or do another set of practice trials.

3.2.5 Experiment

The final part of the program was the main experiment. The experiment trials were the same as in the practice stage, the only difference was that in this stage the answers and other trial info was saved. The exact data saved is specified further below in section 3.3.5. There were 10 second breaks every 20 trials so the test subject stayed sharp trough the experiment. The final part of the experiment was a debriefing stage where test subject was asked questions about the experiment.

3.3 Functionalities

3.3.1 Virtual setup of the experiment

In the description of the experiment, the positions of elements used are described in terms of FOV degrees. In order to correctly place and scale objects in the scene, the positions have to be converted from FOV degrees to Unity units. Using sine law, we find that 1 degree of FOV at 100 unit distance is approximately 1.7454 units. The experiment plane is placed 100 units from the test subject to minimize the effect test subject's head movements have on the relative distance of the experiment plane. The experiment plane is curved 180 degrees, its radius is 100 units and its center is at the (0,0,0) point in the Unity coordinate system (see figure 6). The center point is also where the test subject is located in the virtual space so that every raycast parallel to the x and z axis has a distance of 100 units.

The fixation point is placed at the point (0,0,100), facing the camera rig. The stimuli appear from 3 to 9 degrees either side from the fixation point.
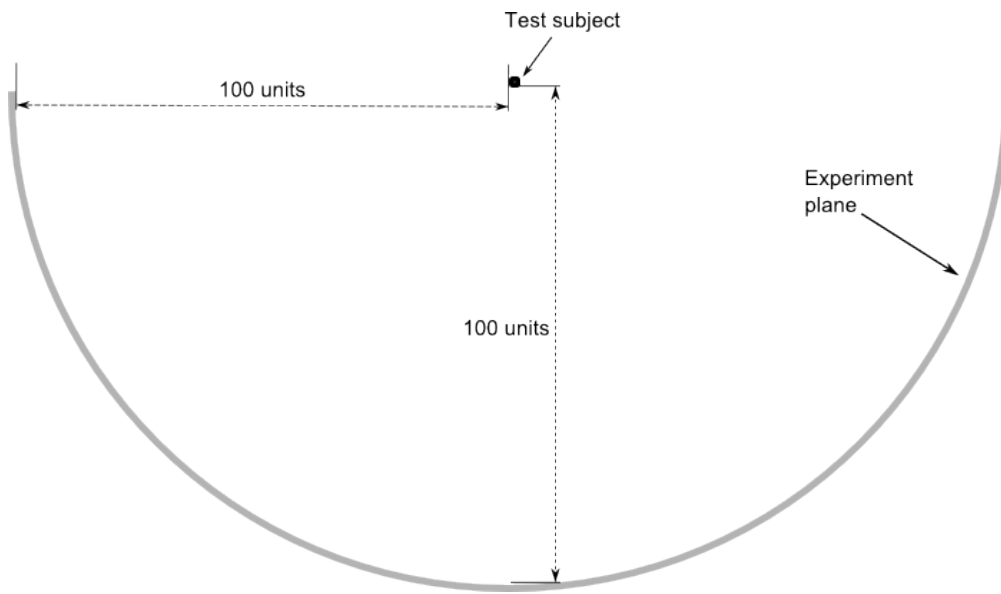
Figure 6. The placement of the test subject's camera rig and the experiment plane in the experiment scene

3.3.2 Multi-display setup

While conducting the experiment, it is important for the experimenter to get feedback on the test subject's actions. Multi-display support was implemented so that the experimenter can see a more detailed view of the experiment while the test subject only sees things (s)he is supposed to see. When running the experiment with two monitors and a HMD, the experimenter can see both the test subject's view and the detailed view on separate monitors. In order to separate the detailed view and experiment view in Unity, GameObjects are divided into different layers and camera culling masks are used to render only specific layers. The experimenter view can be seen on figures 7 and 9, the view of the test subject can be seen on figure 8.
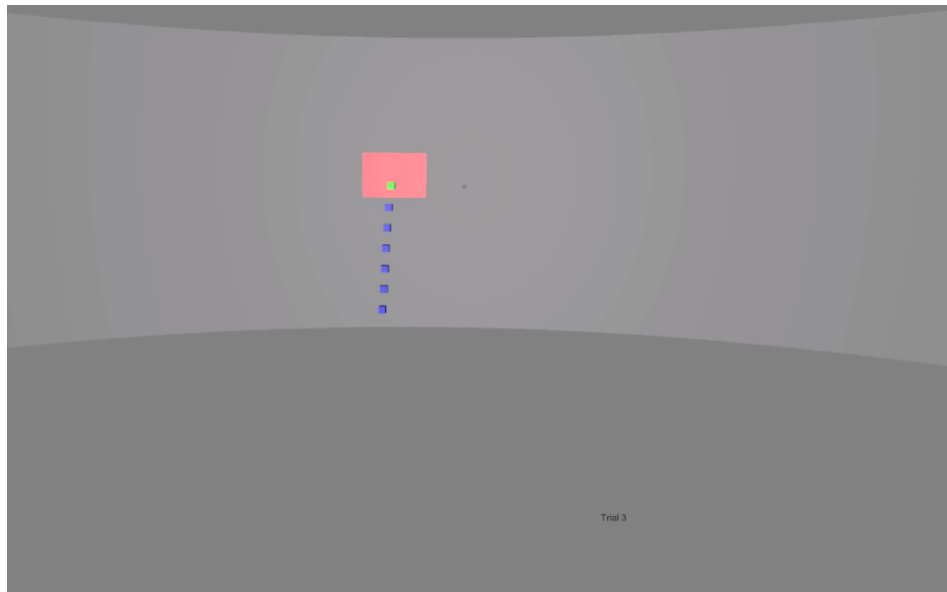
Figure 7. Experimenter view. When the hand moves over the red box, stimuli are shown for a certain amount of time. The blue and green squares mark the movement path of the hand. These points also mark the frames when the data for the hand movement speed is logged.

Figure 8. View from the Oculus headset - 4 cpd stimuli shown. The participant only sees the fixation point, stimuli and the questions about stimuli.
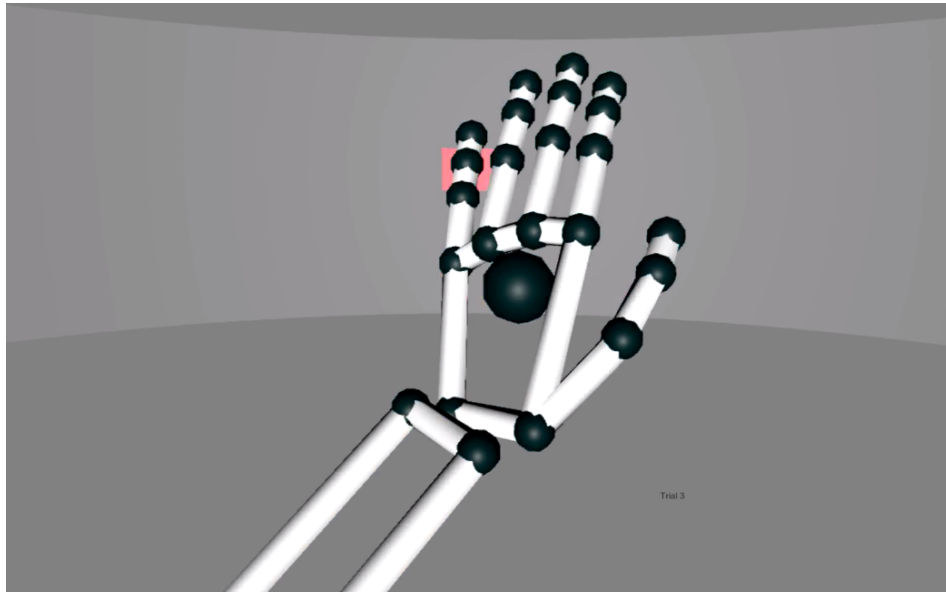
Figure 9. The extra details in the experimenter's view help the experimenter to conduct the experiment. Experimenter can see the hand models, the experiment plane, some textual notifications and instructions.

3.3.3   Distribution of trial parameters

When preparing trials for the tutorial or the main experiment, first the distribution of attributes between trials is calculated. For each parameter, the number of times each unique value is used is equal across values but the order of values is shuffled. For each trial the following information will be calculated:

- Which side the test/standard stimulus will be on?

- Which orientations will the stimuli have?

- What will be the test stimulus' contrast?

- What frequency stimuli will be shown?

3.3.4 Hand tracking

Leap's Unity API is used for hand tracking. The program tracks the position of the intermediate phalanx of the test subject's left hand middle finger. A raycast [17] is made from the 'center eye' (midpoint between two eyes) component of the Unity camera rig towards the center point of the intermediate phalanx of the middle finger (see figure 10). If the raycast hits the tracking plane, the hand velocity is saved and a marker is left at the raycast hitpoint to visualize the hand movement path. If the raycast hits a specific area on the experiment plane, the stimuli are shown. The decision to raycast from the middle finger instead of the palm like in Kristjan-Julius's program was made to maximize the time the left stimulus is behind the (invisible) hand.
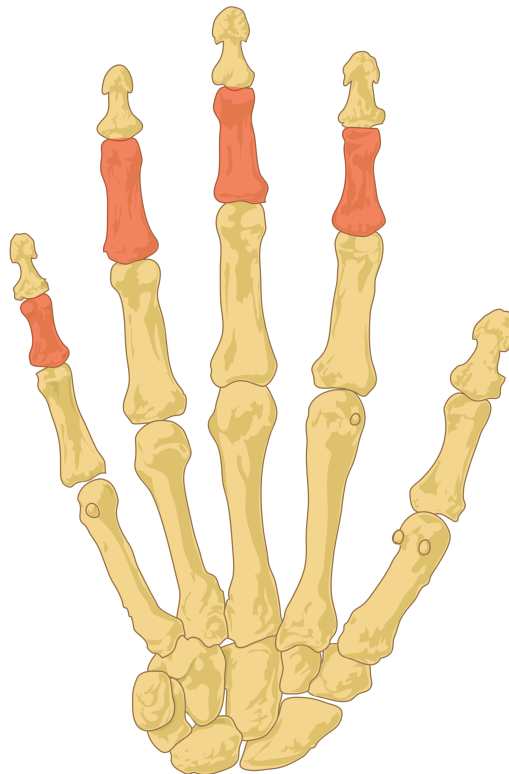

Figure 10. Intermediate phalanges. A raycast is made from the 'center eye' component of Unity camera rig towards the center point of the intermediate phalanx of the middle finger. Figure from the Internet [18].

3.3.5 Data logging

Kristjan-Julius Laak's data collection script was used to save the experiment data in CSV-format. The scripts were modified so that the data of one trial will be saved right after the trial has taken place as opposed to all at once at the end of the experiment. This ensures that a minimal amount of data gets lost if the experiment is not finished.

For each experiment the following data was saved:

- Test subject's id

- Date and time of the experiment

- Test subject's age

- Test subject's sex

- Test subject's handedness

- Test subject's vision

- Number of trials

- Standard contrast value

- Upper bound of test contrast

- Lower bound of test contrast

- Number of logarithmic increments the test contrast is divided into

- Time between trials

- How many trials were between rest

For each trial the following data was saved

- Trial id

- Left stimulus contrast

- Right stimulus contrast

- Left stimulus orientation. Right stimulus orientation can be derived from that

- Did test subject move his/her hand?

- Were the stimuli shown?

- Average hand velocity a couple frames before showing the stimuli

- Left stimulus x,y and z coordinates in the Unity coordinate system. Right stimulus position can be derived from that.

- What frequency stimuli were shown

- Left stimulus distance to center eye component of HMD in Unity units

- Answer to the question

- Was an extra question asked?

- Answer to the extra question

3.3.6 Performance

In order for the VR experience to feel smooth and not cause simulator sickness to the user, the application has to run steadily at a minimum of 90 frames per second (FPS). The developed VR program is not very cumbersome on the GPU since it is graphically quite minimalistic and most of the program logic is calculated on CPU. The most taxing parts of the program are calculating the distribution of the experiment parameters and changing the contrast of the stimulus - during these calculations the framerate can drop as low as 60 FPS for one or two frames with the main CPU cycle taking an average of 13.6 ms to complete. Performance of the program could be improved by changing the contrast of the stimulus using the GPU, but since the framerate drop is not significant enough to be noticed by the user, the implementation using CPU is good enough in the context of this study. The computer used for conducting the experiments and benchmarking had an Intel i7-6700 3.4GHz CPU, 16 GB RAM, NVIDIA GeForce GTX 980 Ti GPU and uses 64-bit Windows 10 operating system. While benchmarking the program was run inside the Unity editor, when compiled the performance should be better.

3.3.7 User Interface

In order for the multi-display setup to work properly the program has to be compiled. It means that experiment parameters cannot be read and written by the experimenter using Unity editor during runtime. In order to allow the user to set the parameters before an experiment, a menu scene was created (see figure 11). It is possible to toggle different parts of the experiment, set the number of iterations for tutorial and main experiment. It is also possible to adjust the time the stimuli will be shown, time between test iterations, and how many trials there are. The test stimuli contrast distribution and a standard contrast can be chosen. The test contrast distribution is logarithmically distributed, the number of log increments can be chosen by the experimenter. There is an input form for the test subject's data - age, sex, handedness and vision. If wearing a VR headset during the main menu, the test subject will see a white environment with text "Adjust your headset until you can see this text clearly."

Figure 11. A screenshot of the menu view.

4 The experiment

4.1 Methods

4.1.1 Participants and apparatus

12 naive observers participated in the study. The experiments were conducted by Madis Vasser. No participant was aware of the purpose of the study, all participants had normal or corrected-to-normal vision. Participants signed an informed consent sheet before the experiment.

They viewed the stimuli through the Oculus Rift CV 1 HMD. A Leap Motion hand tracking device was attached to the front of the HMD and a Logitech optical mouse was used for answering questions.

4.1.2 Stimuli

A black dot ($0.7°$ x $0.7°$) was shown at the center of the screen in front of a gray (rgb values (0.5, 0.5, 0.5)) background for 500 ms. The dot served as a fixation point that the participants were instructed to focus on. As a cue to start the hand movement, the fixation point flashed red for 300 ms and then turned to dark grey (rgb values (0.39, 0.39, 0.39)). The stimuli used were Gabor patches ($4°$ x $4°$) presented to the left and right of the fixation point. The stimuli were presented between 3 and 9 degrees from fixation point, depending on the hand movement. The stimulus on left appeared in the area of the visual field where the hand was currently moving. The stimulus on the right appeared at the right side of the fixation point, mirroring the position of the left stimulus. The contrast, spatial frequency and orientation of stimuli randomly varied with equal probability from trial to trial. One of the Gabor patches was always presented at fixed contrast of 24.5 %, while the other one was randomly sampled from a set of Michelson contrast in 5 log increments from 15% to 40% (see figure 12). The stimuli were visible for 250 ms.
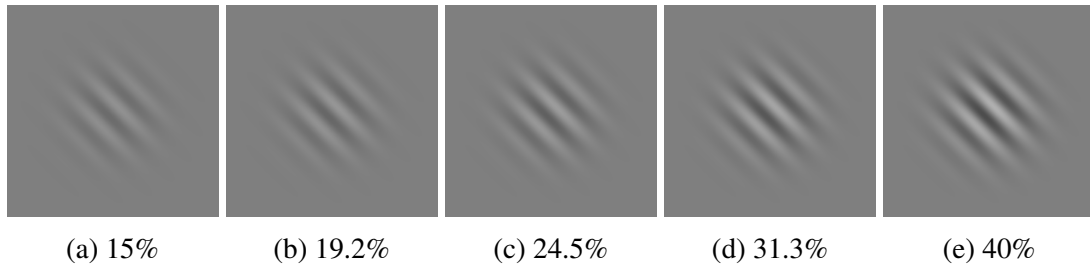
(a) 15%   (b) 19.2%   (c) 24.5%   (d) 31.3%   (e) 40%

Figure 12. The stimuli with contrast distribution used in the experiment.

4.1.3 Procedure

Each observant participated in 200 trials. They were instructed to focus on the fixation point through the experiment. The hand movement initiated the display of the stimuli. The hand movement was done with the left hand.

The subjects had to perform a two-alternative forced choice task - they answered the side of the stimulus which seemed higher in contrast. 30% of the trials they were also asked to indicate which orientation did the stimulus that seemed higher in contrast have. Every 20 trials there was a rest period of 10 s.

4.2 Results

A total of 12 test subjects took part in the study (8 male 4 female, age 18-42, mean age 24, median 22.5). One test subject's result were excluded from analysis because she reported the VR headset to be uncomfortable and distracting during the experiment, not letting her to focus on the task. Excluding this one test subject, a total of 1523 trials were collected where participants saw the stimuli and answered the main question. A total of 463 trials the test subject's were asked to indicate the orientation of the stimulus higher in contrast.

To measure the effect that self-generated hand movement has on the contrast of the stimulus, we analyzed the answers of the test subjects. Answers to both questions ("At which side is the stimulus that is higher in contrast?" and "Which orientation does the stimulus that is higher in contrast have?") were analyzed. The results showed that on average the stimulus behind the hand was perceived as having a lower contrast than the mirrored stimulus. The extra question, "Which orientation does the stimulus that is higher in contrast have?" had more distinguishable differences between the two con-

ditions (behind hand and mirrored) than the main question. The results plotted using python can be seen on figures 13 and 14. A two-way ANOVA with factors contrast and stimulus side was conducted. A dataset of only 2 CPD stimuli, a dataset of only 4 CPD stimuli and a dataset containing stimuli of both spatial resolutions were all analyzed for both questions separately. The ANOVA results show that in 5 cases out of 6, the p-value of the contrast factor was smaller than 0.01, the exception being the dataset for the question about orientation that contained only trials with 4 CPD stimuli. The analysis did not show any statistical significance of the hand side factor. The full summary of the ANOVA results can be seen in the appendix.
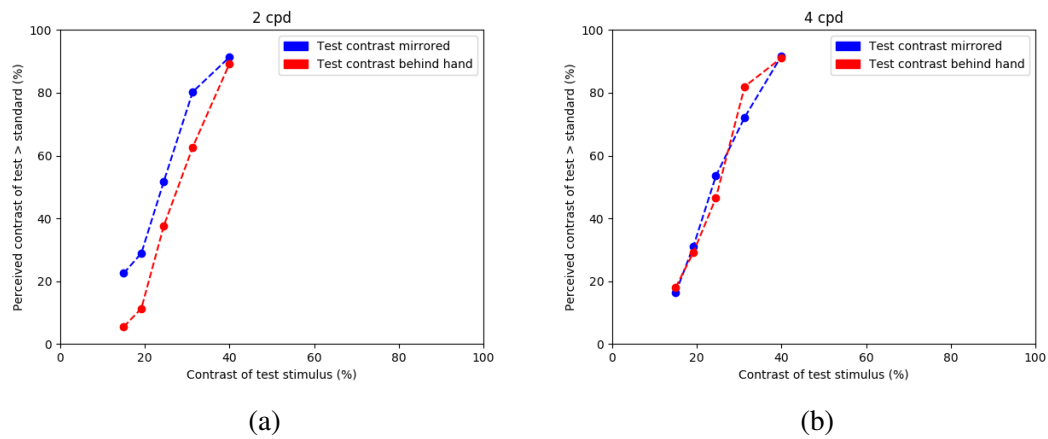

(a)

(b)

Figure 13. Answers to the question "At which side is the stimulus that is higher in contrast?" Percentage of responses in which the participants reported the contrast of test stimulus to be higher than the standard stimulus plotted as a function of the test stimulus contrast.
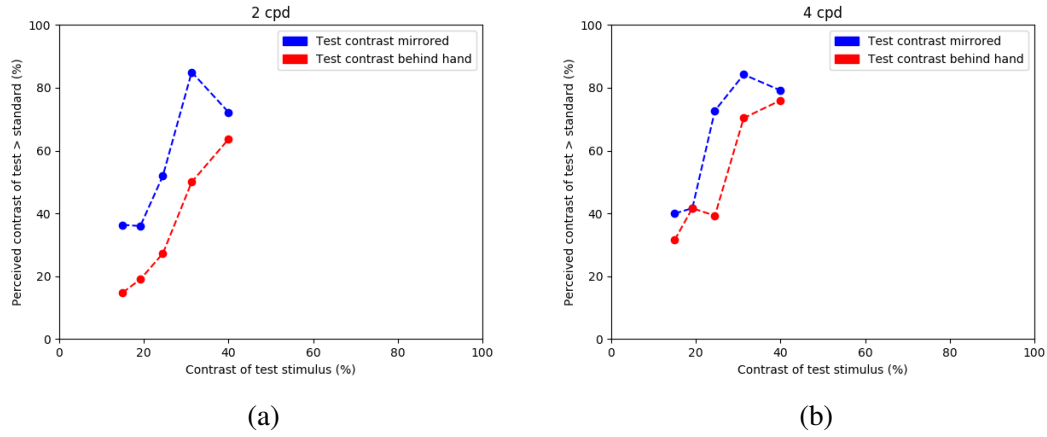
Figure 14. Answers to the question "Which orientation does the stimulus that is higher in contrast have?" Percentage of responses in which the participants reported the contrast of test stimulus to be higher than the standard stimulus plotted as a function of the test stimulus contrast.

4.3 Discussion

Through evolution, humans have developed an effective way of processing visual information, enabling them to quickly visually detect movement in the environment. It is generally accepted that the human brain predicts the consequences of own movements and attenuates the signals resulting from those movements. Having a better understanding on how humans process visual information could help us to design new artificial intelligence and computer vision algorithms. In order to examine the process of visual attenuation, we designed an experiment to assess whether people withdraw their attention from the area of their visual field where own hand is moving.

A study conducted by Carrasco et al. [4] showed that visual attention has an affect on the stimulus, notably that objects under attention are perceived as having a higher contrast. Experiments using VR and hand tracking done by Laak et al. [3] suggested that attention is withdrawn from the area of visual field where own hand is moving. Based on these two notions we hypothesized that the stimuli appearing behind hand would be perceived as having a lower contrast.

Out of 2200 trials, the stimuli were shown and questions were asked in only 1523 of them. This means that the correct hand movement is made in only 69% of the trials. The low accuracy is caused by the relatively small area in front of which the hand has to move in order for the stimuli to be shown (see Figure 7). The accuracy could be

increased by making the area where the stimuli can appear larger, but if the area is too large, the test subjects will not be able to notice the stimuli, their contrast and orientation during the short time they are visible. To increase the number of legit trials, the number of total trials could be increased as well but the 200 trials per participant we used right now was already quite challenging to some participants.

It was observed that when the stimuli were not shown as a result of incorrect hand movement, the test subjects began to lose motivation. This can be improved in future experiments by always showing the stimuli to the test subject - if the subject does not make a correct hand movement, neither stimulus will appear behind the hand. The time stimuli were visible was set to 250 ms. To ensure that the subjects perform the experiment under the conditions of covert attention, they were instructed to focus on the fixation point trough the experiment. In future experiments the time stimuli are shown may be set to as low as 100 ms so that the test subject will not have time to move his/her eyes. To make the stimuli more distinguishable during the short time they are visible, the overall contrast distribution can be higher in contrast.

The results of this study indicate that the perceived contrast may be lower in the area of the visual field where own hand is moving but further studies are needed.

Conclusion

As a part of this thesis, a VR environment was developed that enables to test whether self generated hand movement affects the perceived contrast in the area of the visual field where the own hand is moving. In addition to the experiment program, a program to generate Gabor patches was developed and the main flow of the experiment was designed. Data processing scripts were written for analyzing the data from the program.

An experiment with 12 observers was conducted. The study did not provide enough info to confirm the hypothesis that self generated hand movement affects the perceived contrast in the area of the visual field where own hand is moving, but helped to refine the experiment design and increase the accuracy of future experiments trials.

References

[1] Paul M Bays, J. Randall Flanagan, and Daniel M Wolpert. Attenuation of self-generated tactile sensations is predictive, not postdictive. *PLOS Biology*, 4(2), 01 2006.

[2] Sarah-J Blakemore, Daniel M. Wolpert, and Chris D. Frith. Central cancellation of self-produced tickle sensation. *Nat Neurosci*, 1(7):635–640, Nov 1998.

[3] Kristjan-Julius Laak, Madis Vasser, Oliver Jared Uibopuu, and Jaan Aru. Attention is withdrawn from the area of the visual field where the own hand is currently moving. *Neuroscience of Consciousness*, 3(1):niw025, 2017.

[4] Marisa Carrasco, Sam Ling, and Sarah Read. Attention alters appearance. *Nat Neurosci*, 7(3):308–313, Mar 2004.

[5] Unity vr overview. `https://unity3d.com/learn/tutorials/topics/virtual-reality/vr-overview?playlist=22946`. (30.04.2017).

[6] Unity manual - gameobjects. `https://docs.unity3d.com/Manual/GameObjects.html`. (30.04.2017).

[7] Unity - creating and using scripts. `https://docs.unity3d.com/Manual/CreatingAndUsingScripts.html`. (30.04.2017).

[8] Oculus rift - rendering techniques. `https://developer3.oculus.com/documentation/intro-vr/latest/concepts/bp_app_rendering/`. (30.04.2017).

[9] Oculus rift - field of view and scale. `https://developer3.oculus.com/documentation/intro-vr/latest/concepts/bp_app_fov_scale/`. (30.04.2017).

[10] Oculus rift - tracking. `https://developer3.oculus.com/documentation/intro-vr/latest/concepts/bp_app_tracking/`. (30.04.2017).

[11] Hp leap motion notebook, desktop | hp® official site. `http://www8.hp.com/us/en/ads/envy-leap-motion/overview.html?jumpid=va_r11260_go_leapmotion`. (30.04.2017).

[12] Leap motion unity overview — leap motion unity sdk v3.2 beta documentation. `https://developer.leapmotion.com/documentation/unity/unity/Unity_Overview.html`. (30.04.2017).

[13] Frank Weichert, Daniel Bachmann, Bartholomäus Rudak, and Denis Fisseler. Analysis of the accuracy and robustness of the leap motion controller. *Sensors*, 13(5):6380–6393, 2013.

[14] Kristjan-Julius Laak. From the brain to intelligent systems: The attenuation of sensation of self-generated movement. Master's thesis, University of Tartu, 2016.

[15] Luminance. `http://www.songho.ca/dsp/luminance/luminance.html`. (30.04.2017).

[16] Luminance contrast. `https://colorusage.arc.nasa.gov/luminance_cont.php`. (30.04.2017).

[17] Unity - scripting api: Physics.raycast. `https://docs.unity3d.com/ScriptReference/Physics.Raycast.html`. (01.05.2017).

[18] Mariana Ruiz Villarreal. Intermediate phalanges. `https://commons.wikimedia.org/wiki/File%3AIntermediate_phalanges.svg`, 2007. (10.05.2017).

Appendix

I: Glossary

camera In Unity engine, cameras are the devices that capture and display the world to the player. You can have an unlimited number of cameras in a scene. They can be set to render in any order, at any place on the screen, or only certain parts of the screen. 7, 19

CPD cycles per degree. 10, 28

FOV field of view. 7, 9, 10, 18

HMD head-mounted display. 7, 8, 14, 19, 24, 26

raycast In Unity engine, raycasting allows the user to cast a ray from origin point in a specified direction of specified length against all colliders in the scene. The raycast gives info if it hit anything in its way and at what point the raycast hit occured. 18

render Rendering or image synthesis is the process of generating an image from a 2D or 3D model (or models in what collectively could be called a scene file) by means of computer programs. Also, the results of such a model can be called a rendering. 7

scene In Unity, scenes contain the objects of your game. They can be used to create a main menu, individual levels, and anything else. Think of each unique Scene file as a unique level. In each Scene, you will place your environments, obstacles, and decorations, essentially designing and building your game in pieces. 7, 18

VR virtual reality. 2, 6, 7, 14, 24, 31

II: ANOVA results

Following are the results of Analysis of Variance (ANOVA) done on different datasets using R's aov function. The dataset and script are provided in appendix IV. Each dataset has columns for test subject id, test contrast value, test contrast side and a percentage of times the test subject answered the test contrast to be higher in contrast.

Dataset containing trials with 2 CPD stimuli and answers to the question "On which side was the stimululs that seemed higher in contrast?"

```
Error: SUBJECT_ID
        Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.3864 0.03864
Error: SUBJECT_ID:CONTRAST
        Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST  4  9.247 2.3117 46.57 1.55e-14 ***
Residuals 40 1.985 0.0496
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Error: SUBJECT_ID:TEST_SIDE
        Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE 1  0.5332 0.5332 2.854 0.122
Residuals 10 1.8679 0.1868
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                    Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE   4  0.1740 0.0435 1.827 0.143
Residuals           40 0.9522 0.0238
```

Dataset containing trials with 4 CPD stimuli and answers to the question "On which side was the stimululs that seemed higher in contrast?"

```
Error: SUBJECT_ID
           Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.1748 0.01748
Error: SUBJECT_ID:CONTRAST
        Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST  4 8.282 2.0705 63.54 <2e-16 ***
Residuals 40 1.303 0.0326
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Error: SUBJECT_ID:TEST_SIDE
        Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE 1 0.000 0.00004 0 0.986
Residuals 10 1.132 0.11322
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                    Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE   4 0.0926 0.02316 0.791 0.538
Residuals           40 1.1716 0.02929
```

Dataset containing trials with both stimuli and answers to the question "On which side was the stimululs that seemed higher in contrast?"

```
Error: SUBJECT_ID
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.09044 0.009044
Error: SUBJECT_ID:CONTRAST
          Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST  4 9.228 2.3071 110.3 <2e-16 ***
Residuals 40 0.836 0.0209
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Error: SUBJECT_ID:TEST_SIDE
          Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE 1 0.1627 0.1627 1.185 0.302
Residuals 10 1.3735 0.1373
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                    Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE   4 0.0495 0.01238 0.841 0.508
Residuals           40 0.5889 0.01472
```

Dataset containing trials with 2 CPD stimuli and answers to the question "Which orientation does the stimulus that is higher in contrast have?"

```
Error: SUBJECT_ID
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.1213 0.01213
Error: SUBJECT_ID:CONTRAST
          Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST  4 0.2992 0.0748 3.837 0.00989 **
Residuals 40 0.7798 0.0195
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Error: SUBJECT_ID:TEST_SIDE
          Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE 1 0.0330 0.03296 0.722 0.415
Residuals 10 0.4563 0.04563
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                    Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE   4 0.1254 0.03136 1.826 0.143
Residuals           40 0.6867 0.01717
```

Dataset containing trials with 4 CPD stimuli and answers to the question "Which orientation does the stimulus that is higher in contrast have?"

```
Error: SUBJECT_ID
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.1511 0.01511
Error: SUBJECT_ID:CONTRAST
          Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST   4 0.1375 0.03438  1.723  0.164
Residuals 40 0.7979 0.01995
Error: SUBJECT_ID:TEST_SIDE
          Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE  1 0.0258 0.02580  0.837  0.382
Residuals 10 0.3081 0.03081
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                      Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE     4 0.0642 0.01605  0.743  0.569
Residuals             40 0.8644 0.02161
```

Dataset containing trials with both stimuli and answers to the question "Which orientation does the stimulus that is higher in contrast have?"

```
Error: SUBJECT_ID
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 10 0.04987 0.004987
Error: SUBJECT_ID:CONTRAST
          Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST   4 0.2386 0.05965  4.434  0.00464 **
Residuals 40 0.5382 0.01345
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Error: SUBJECT_ID:TEST_SIDE
          Df Sum Sq Mean Sq F value Pr(>F)
TEST_SIDE  1 0.02177 0.02177  0.988  0.344
Residuals 10 0.22036 0.02204
Error: SUBJECT_ID:CONTRAST:TEST_SIDE
                      Df Sum Sq Mean Sq F value Pr(>F)
CONTRAST:TEST_SIDE     4 0.0522 0.013053 1.359  0.265
Residuals             40 0.3842 0.009605
```

III: Code and data

The Unity project, experiment data, data processing scripts and compiled programs can be found on github: https://github.com/t6niskoppel/BachelorThesis

Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Tõnis Kristian Koppel**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

    1.1 reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

    1.2 make available to the public via the university's web environment, including via the DSpace digital archives, as of 10.05.2018 until expiry of the term of validity of the copyright,

    of my thesis

    "Reduced Contrast in the Area of the Visual Field Where the Own Hand Is Moving: A Virtual Reality Study"

    supervised by Jaan Aru, Kristjan-Julius Laak and Madis Vasser

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 11.05.2017