



ulm university universität
ulm

Ulm University | 89069 Ulm | Germany

**Faculty of Engineering,
Computer Science and Psychology**
Institute of Media Informatics
Human-Computer-Interaction Group

An In-Depth Analysis of and Compensation for the Heisenberg Effect

Bachelor Thesis in Media Informatics at Ulm University

Presented by:

Marco Combosch
marco.combosch@uni-ulm.de

Examiner:

Prof. Dr. Enrico Rukzio

Advisor:

Dennis Wolf

2019

Last updated May 29, 2019

© 2019 Marco Combosch

This work is licensed under the Creative Commons
Attribution-NonCommercial-ShareAlike 3.0 Unported License. To view a copy of
this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/>.

Typesetting: PDF-L^AT_EX 2_<

Abstract

In modern virtual reality (VR) systems, distal interaction with a controller on canvas is the standard input model. In games and applications in VR, distal pointing is used as the main interface to interact with the virtual environment. Some of them need precise selections to work accurately. Predictable and accurate pointing can also reduce frustration when working and/or playing in VR. Some scientific papers suggest that the “Heisenberg Effect”, an effect resulting in selecting a target by clicking or pressing a button, has an impact on the performance, accuracy and usability of VR applications. This thesis takes a look at this “Heisenberg Effect”, its underlying mechanism and on possible actions to compensate the effect. The foundation of this work is a study which was implemented to analyze if the “Heisenberg Effect” can be detected and what its effect is.

The evaluation of the collected data suggests, that the “Heisenberg Effect” exists and has an impact on accuracy and performance on distal pointing tasks. On the basis of this statement, a short overview of ideas is given, on how to compensate this effect. These ideas give a basis for future work on this topic.

Contents

1	Introduction	1
1.1	Pointing in VR	1
1.2	Objective	1
1.3	Structure	2
2	Related Work	3
2.1	Defining the Heisenberg Effect	3
2.1.1	First reference by Bowman et al.	3
2.1.2	Description and link to Fitt's Law by Kopper et al.	4
2.1.3	Own definition	4
2.2	Background & Related Work on pointing tasks	4
2.2.1	Distal pointing tasks	4
2.2.2	Fitts' Law	5
2.2.3	Extended Fitt's Law	6
2.2.4	Related work on distal pointing	7
3	Content	9
3.1	Implementation	9
3.1.1	Setup	9
3.1.2	First attempts	9
3.1.3	Fitts' Law based circle pointing task	11
3.1.4	Study implementation	13
3.1.5	Design	14
3.1.6	Tracking device jitter	14
3.1.7	Source Code	14
3.2	Experiment	15
3.2.1	Participants	15
3.2.2	Procedure	15
3.3	Collected Data	16
4	Results	19
4.1	Clearing the data from disturbing effects	19
4.1.1	Calculation offset	19

CONTENTS

4.1.2 Device-caused jitter	20
4.1.3 Hand tremor	20
4.2 Indications for the Heisenberg Effect	21
5 Conclusions and future work	23
5.1 The “Heisenberg Effect”	23
5.2 Compensation for the “Heisenberg Effect” and Implications for future work	23
5.3 Acknowledgements	24

1 Introduction

1.1 Pointing in VR

In normal desktop applications, the mouse is the standard device for input. Mobile devices use mostly touchscreens and sometimes gyroscopes as input devices, where the mouse and the touchscreen or -pad work on a 2D plane.

In contrast to that, virtual reality (VR) uses pointing in three dimensions to present the applications and to receive inputs. Most modern VR devices use a controller with three (3DOF) or six (6DOF) degrees of freedom to navigate and select in the virtual environment (VE).

The actions are based on the procedure of the user pointing the controller to the desired target and selecting the target by clicking or pressing a button on the controller. Sometimes this pointing is indicated in VR by a raycast. This raycast shows the direction and the aim of the position of the controller, by spawning a ray out of the top of the device. The user has then the opportunity to directly see, where he's pointing with the controller. Because the controller is tracked in at least three dimensions, the action of selecting something (by clicking or pressing) has an impact on the position of the controller itself, and therefore on the resulting position of the selection on the canvas. This unintentional change of the selected position by selecting something is described in some scientific papers as the "Heisenberg Effect".

This paper gives an in-depth view on this "Heisenberg Effect", shows its impact and what actions are required to compensate it. To show the "Heisenberg Effect", it's required to isolate other effects, caused by the three dimensional (3D) architecture of the VR controller and the virtual environment.

Based on this it is necessary to understand these effects and to take note of these effects in investigating the "Heisenberg Effect".

1.2 Objective

Applications in VR rely on a dependable interaction with the user. To gain immersion, it is necessary, that a user has the sense, the virtual environment correctly responds to his actions. If selections and clicks are not registered by the system, caused by

1 INTRODUCTION

different effects, immersion suffers from this experience. One of these effects could be the so-called “Heisenberg Effect”. Because it is only referenced in a small amount of scientific papers, more research on this topic is needed. To understand this “Heisenberg Effect” better and even show its impact is incentive for this thesis. Finding a way to compensate the effect is also one of the motives for this work because reliable selections are the basis for good and efficient user experience.

1.3 Structure

To have a good insight into the topic of distal pointing and the associated effects, this work takes at first a look at related work in the field of distal pointing. Used procedures for study design and to analyze these effects are also described in the second chapter (Related Work - 2). To have a good basis on describing and characterizing the “Heisenberg Effect” a short view on the already done work will be a part of the second chapter.

To explore more aspects of this effect, a study was carried out. The third chapter (Content - 3) takes a look on the way of implementing a useful study design, and the used resources and algorithms. The experiment is explained in detail and a short look on the resulted data is given.

The collected data out of the experiment is analyzed and discussed in the fourth chapter (Results - 4). Also, an explanation of the “Heisenberg Effect” based on this data is given. The last chapter (Conclusions and future work - 5) takes a look on the now newly described “Heisenberg Effect” and shows some steps that could compensate this effect. Incentives and lookouts for future work on this topic are also presented.

2 Related Work

The “Heisenberg Effect” is described only in a few scientific papers. First mentioned in the paper by Bowman et al. [1], and picked up by Kopper et al [4], a real definition and evidence for the “Heisenberg Effect” were never given. In this section, a general definition, based on the previous works mentioned before, is drawn up and explained. To analyze and describe the Heisenberg Effect it is necessary to take a look at different researches on pointing tasks and the used procedures. Therefore, a detailed look at related work on 2D and 3D pointing tasks, spatial jitter, and resulting effects is given. Based on these papers this chapter is concluded by taking a look at underlying laws and algorithms which are used in the later study design to show and analyze the “Heisenberg Effect”.

2.1 Defining the Heisenberg Effect

2.1.1 First reference by Bowman et al.

The first reference of the “Heisenberg Effect” was in the paper “Using Pinch GlovesTM for both Natural and Abstract Interaction Techniques in Virtual Environments” by Bowman et al. [1]. The paper shows the use of cloth gloves (called after the brand name “Pinch GlovesTM”) as input devices in virtual environments. They also describe different techniques to interact in VR with these gloves. One problem they mentioned in their paper, was the difficulty to point at an object and simultaneously select it, without disturbing the previous pointing. This effect was called “the Heisenberg Effect of spatial interaction”. To avoid this “Heisenberg Effect” while still using the selected gloves, they used one hand to point to the desired target and then the other hand to select it. More pieces of information or hints are not given by the mentioned paper. The meaning of the name “Heisenberg Effect” could be derived by “Heisenberg’s uncertainty principle” in physics, declare it is only possible to either point on something with certainty or to select something with certainty. But not both at the same time.

2.1.2 Description and link to Fitt's Law by Kopper et al.

The term “Heisenberg Effect” was also used in the work of Kopper et al. [4] called “A human motor behavior model for distal pointing tasks”. This paper takes a focus on distal pointing tasks and gives an overview of different models to precisely measure movement time, by extending Fitt’s Law. They conclude, that the natural hand jitter and the movement made while pressing down a button to select a target, add to an increase in movement time and difficulty. They call this effect also “Heisenberg Effect” and refer to the work of Bowman et al. [1].

2.1.3 Own definition

Based on the previous work by Bowman and Kopper, the “Heisenberg Effect” in this thesis is defined as follows:

The “Heisenberg Effect in distal pointing” describes the difference between the position of a mental selection of a target and the real tracked position, distinguished by the disturbance of the selecting action.

To show the “Heisenberg Effect” in an example, a look at Figure 2.1 shows a scenario, where a user is pointing with a controller via raycast on a target. The line P1 describes the acquired direction and aim of the user right before he is selecting the target by pressing or clicking a button. Through the action of selecting the target, the raycast gets disturbed which is shown exemplarily by line P2. The now arising difference between “wanted position” and “resulted position” on the canvas is what was described previously and called the “Heisenberg Effect” (labeled as H1).

Based on this definition, this thesis wants to show, that the “Heisenberg Effect” exists, can be measured and therefore compensated.

2.2 Background & Related Work on pointing tasks

2.2.1 Distal pointing tasks

To measure and analyze accuracy in distal pointing, many studies use distal pointing tasks. In Figure 3.3 a simple setup of such a task is shown. In VR the user handles with a controller a visible ray cast. This ray cast is a ray emitted from the tip of the controller in the given direction controlled by the user. If this ray cast hits a canvas the user can see, on which target on the canvas he points his controller (the ray cast is in Figure 3.3 the red line P1). With a button on the controller, the user can select now a target on the canvas on which he is pointing at the moment he presses or clicks the

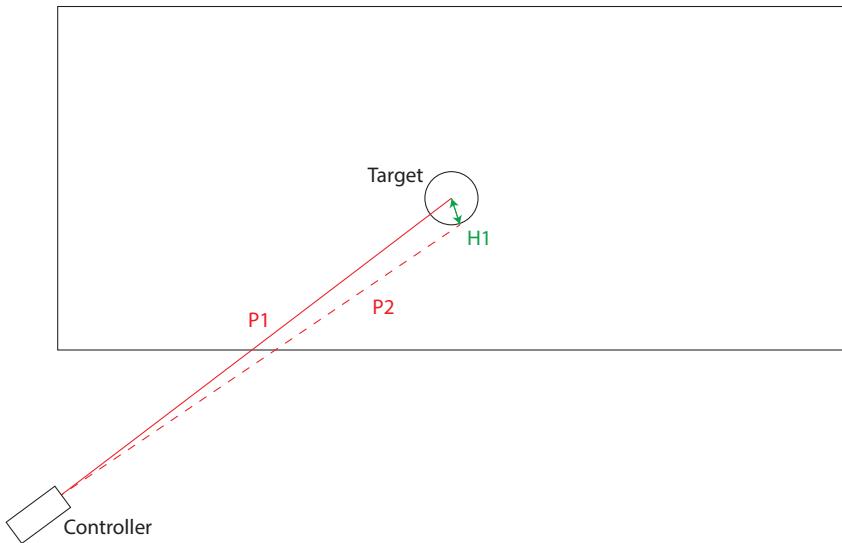


Figure 2.1: Describing the Heisenberg Effect on a simple distal pointing task example

button. In a distal pointing task, target after target is shown to the user and he has to move his controller in the direction of the target and select it with a press/click on the button. These movements can be analyzed and described by Fitt's Law:

2.2.2 Fitts' Law

Fitts' Law is a model to calculate movement times and the difficulties of movements between a start point and a target or between two targets. The difficulty of such movements, called Index of Difficulty (*ID*) and measured in bits, is described in one-dimension as:

$$ID = \log_2(2 \cdot A/W)$$

where *A* is the distance (or amplitude) between two targets and *W* is the size of the target. In this work we use for *ID* the so called “Shannon formulation” as suggested by MacKenzie and Buxton [6], to scale on two-dimensions, fit better the observation and correct negative values for *ID*:

$$ID = \log_2(A/W + 1)$$

2 RELATED WORK

Based on this Index of Difficulty, Fitts' Law describes the Movement Time (*MT*) as follows:

$$MT = a + b \cdot ID$$

a and b are empirically determined constant values based on the given input devices for the specific pointing task. Fitt's Law argues by these equations, that correct movements to targets get more difficult if the targets get smaller. With movement time (*MT*) and *ID*, we can calculate the throughput of an input device in bits per second:

$$TP = ID / MT$$

2.2.3 Extended Fitt's Law

In the study design for this work an extended model of the Fitt's Law, proposed by MacKenzie [5] and described on pointing tasks by Kong and Ren [3], is used. MacKenzie claims in his paper, that the traditional Fitt's Law doesn't account the actual performance of a user when trying to hit small targets. For the extended Fitt's Law used here, the values for Distance and Width (now called *Effective Distance* and *Effective Width*) are calculated for the set of collected data. By using the standard deviation, the extended Fitt's Law can filter out different strategies used by the user to accomplish the tasks. MacKenzie described the *Effective Width* as follows:

$$W_e = 4.133 \cdot \sigma$$

where σ is the standard deviation defined as:

$$\sigma = \frac{\sum_{i=1}^n (x_i - \bar{x})}{\sqrt{n-1}}$$

where n is the number of trials and the sum describes the sum of deviations. As in the standard Fitt's Law the *Effective Throughput* is defined here as:

$$TP_e = \frac{ID_e}{MT_{mean}}$$

where

$$ID_e = \log_2 \left(\frac{A_e}{W_e} + 1 \right)$$

2.2.4 Related work on distal pointing

The extended Fitt's Law is used in the work done by alternately Pavlovych, Stuerzlinger, MacKenzie and Teather [7] [9] [10] to analyze the effects of spatial jitter and lag on the performance in distal pointing tasks. Spatial jitter is described by Pavlovych and Stuerzlinger [7] as “a combination of hand tremor and noise in the device signal”. Their work gave a good foundation for the study design of this paper, mainly because of three reasons:

- The study design in all three papers used the ISO 9241-9 specification [2] which gives clear conditions for this study design. Although the specification was not fully applied.
- To isolate the “Heisenberg Effect”, it was important to clear our data from disturbances like the effects of hand tremor and noises of the device signals, described in their work.
- To give further works a base on how the “Heisenberg Effect” may affect performance in distal pointing tasks, by taking a look on the *Effective Throughput* proposed by MacKenzie.

3 Content

To show the “Heisenberg Effect” a study design based on the work done by Pavlovych and Stuerzlinger [7] was set up. The foundation of the study was a simple distal pointing task. This section gives an insight into the underlying implementation and considerations for the used study design. The experiment and the procedure are also described in detail.

3.1 Implementation

3.1.1 Setup

For the study, it was used the HTC Vive PRO and Unity with Steam VR to create the study setup. The controller for the HTC Vive PRO has a trigger at the bottom to click and select targets. To stay near the real devices, this trigger was used as the main button. The implementation is described in detail in the next sections. Based on the environment with Unity, C# was used as the main programming language. In the room in which the study was performed, it was built up two tracking sensors, to capture most of the room. In the center of the room stood a chair, fixed on the ground. An HTC Vive Pro headset and one controller were used in the demonstration. The user stood on the predefined spot in front of the chair or sit on it.

3.1.2 First attempts

The goal of the study was to find possible effects and evidence for the “Heisenberg Effect”. As the effect described in a previous section (Subsection 2.1.2), it was necessary to create a point and aim application for measuring the overall movements and hits, performed by the study subjects. Even at the beginning of implementation, it was clear to collect data of clicks out of the movement and in a static position. The aim of these two scenarios was to find and show differences in clicks out of motion and resting clicks (the clicks out of motion were discarded because of too many disturbances). Resulting from this specification, it was created a procedure for a single target task:

3 CONTENT

1. Show one target
2. Wait for first click (not necessarily on target) - arising out of the movement
3. Show counter if the user is on target and wait for the counter to finish
4. Wait for second click (also not necessarily on target)
5. Hide the target and begin again with the next one

With this specification, it was possible to observe differences in static and motion-based clicks. This specification was transferred into the final implementation.

After solving the problem of the procedure, it was necessary to clarify how to arrange the targets on the canvas. The first attempt in implementation was to show a grid-based target layout (as shown in Figure 3.1 (a)).

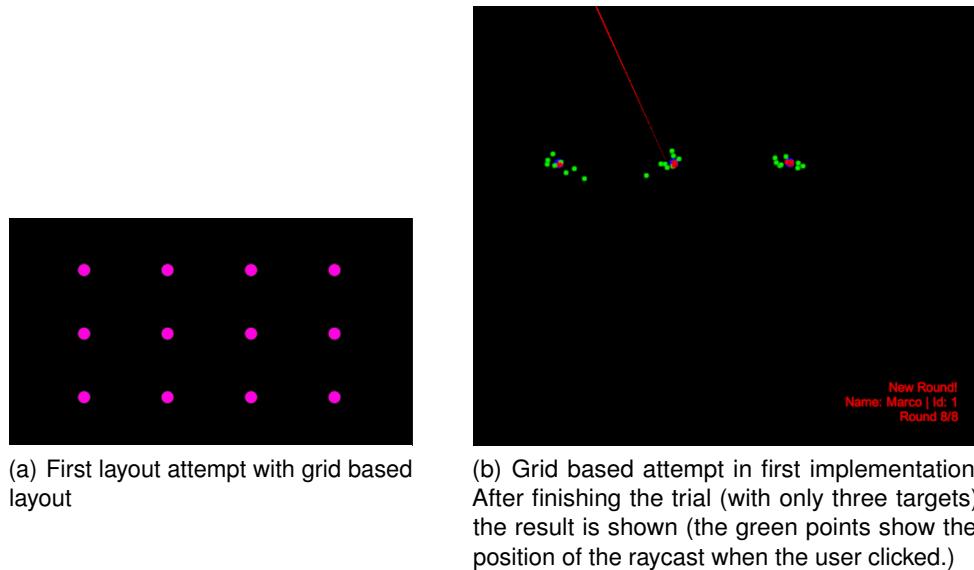


Figure 3.1: Different Layout attempts

The targets were displayed, one after another, on a three-by-four grid (three in height and four in width). Depending on the grid layout, there were three possible ways to display the targets one after another:

1. Show them in a static order
2. Show them in a random order

Width	Amplitude	ID
15	150	3.46
15	350	4.60
30	150	2.58
30	350	3.66
50	150	2.00
50	350	3.00

Table 3.1: Used Index of Difficulty

3. Create different static orders and rotate through them randomly

The first layouts and experiments were based on this grid. After some testing (one of the results of the test runs is shown in Figure 3.1 (b)), there were some issues with this type of grid layout: The distances between the targets were different, caused by the three-by-four grid. Also, no other studies used such a grid layout for testing accuracy in pointing tasks. After some research, the paper by Pavlovych and Stuerzlinger [7] showed some usable target layout. They used a Fitts' Law based circle task to test accuracy in their pointing tasks, which was adopted to this setup (see 3.1.3).

Based on the used setup (3.1.1), the software should also handle most of the functionality the controller and the system supported. The controller supported six degrees of freedom (free position in three dimensions, plus three-dimensional angles). To portray also results for controllers with three degrees of freedom (fixed position, but three-dimensional angles), a mode for switching between these variations was implemented.

The controller uses a trigger to measure clicks. These clicks are not binary, meaning the strength of a press on the trigger can be measured by decimal values between 0.0 (trigger not touched) and 1.0 (trigger fully pressed). In the implementation, these values get also tracked to see, when a user initialized the click by slightly pressing the trigger, and when the click is really registered as a click by the implementation.

Besides the mode for degrees of freedom, the implementation handles two different modes, to create more variety in the study setup. The position of the user's arm (stretched out or applied to the body) and the position of the user itself (standing or sitting) is also tracked.

3.1.3 Fitts' Law based circle pointing task

Based on the experience made with a grid-base pointing task (described in section above: 3.1.2), it was clear to use a Fitts' Law based circle pointing task as described by Soukoreff and MacKenzie [8]. To get a wide range of difficulties in the study, the *ID*

3 CONTENT

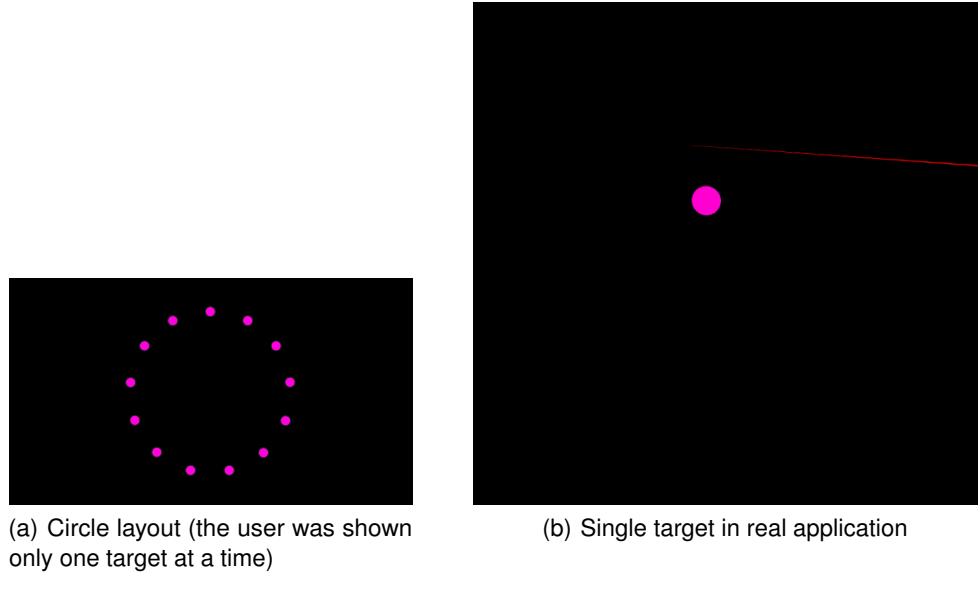


Figure 3.2: Final implementation

was calculated with different numbers of *Amplitude* and *Width* to get diverse *ID*'s as described in Table 3.1.

On the basis of the *Width* and *Amplitude* values, described in Table 3.1 for each *ID* value was created an own Fitts' Law based circle pointing task. In consideration of the study setup by Pavlovych and Stuerzlinger [7] each circle was created with 13 targets (in consideration, that the first target cannot be evaluated, thanks to the unknown starting point). To calculate each point on the circle as a vector, the software iterates through the points from 0 to 12 (as i) and uses the following formula:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (Amp/2) \cdot \sin(i \cdot (360/13)) \cdot (\pi/180) \\ (Amp/2) \cdot \cos(i \cdot (360/13)) \cdot (\pi/180) \\ 0 \end{bmatrix} \quad (3.1)$$

The size (*Amp*) of the target was implemented later when each point was shown on the canvas. A sketch of the circle is shown in Figure 3.3. The Figure 3.2 shows the circle layout in implementation (a) and a single target with a raycast in the application.

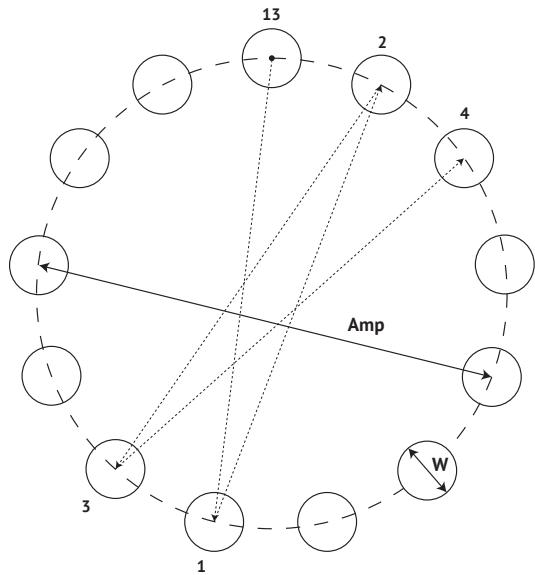


Figure 3.3: Multidirectional pointing & clicking task

3.1.4 Study implementation

On the basis of the previous discussions, the software was implemented to run the study with participants. There are 8 rounds (2 body positions, 2 arm positions, 2 degrees of freedom = 8 possibilities) for each participant, which differ in the following modes:

arm position	stretched out	applied to the body
participants position	standing	sitting
controller degree of freedom	3	6

Table 3.2: Task modes

Each of these rounds contained six Fitts' Law circles, based on the six different *ID* values for width and amplitude calculated in Table 3.1. The modes mentioned in 3.2 and the previously mentioned circles were combined with the help of Latin Squares. These Latin Squares were used to give each participant a different sequence of modes and of circles. The implementation calculated the two of these Latin Squares. One to iterate through the different possible combinations of arm position, body position and degrees of freedom. The other one was used to iterate through the possible amplitudes and

3 CONTENT

sizes of the Fitts' Law circle task. Both are based on a unique user id, so the iterations are steady.

3.1.5 Design

Resulting from the previous research and testing, the experiment had five different and independent variables in a $2 \times 3 \times 2 \times 2 \times 2$ arrangement (makes a total of 48 combinations):

- Target amplitude: 350 and 150;
- Target width: 15, 30 and 50;
- Arm position: stretched out and applied to body;
- User position: standing and sitting;
- Degrees of freedom: 6 and 3;

Based on these numbers, each participant had to complete 48 Fitts' Law circles with each 13 recorded targets and two clicks each target. Given the 16 subject participating of this study, it results in a total number of $16 \cdot 48 \cdot 13 \cdot 2 = 19,968$ trials.

3.1.6 Tracking device jitter

To eliminated device jitter, the position of the controller and headset were tracked in a total static position (headset on a table and controller on the floor). The controller and headset showed fluctuations in position, even when in a total static position. The results were used in Subsection 4.1.2 to clear the data from the device-caused jitter.

3.1.7 Source Code

The full source code and all assets can be found in a GitHub repository¹. Under *assets/HeisenbergScene/Scripts* all relevant scripts can be found. The script *Processing.cs* is the entry point for the application.

¹<https://github.com/mc17uulm/Heisenberg>

3.2 Experiment

3.2.1 Participants

All of the sixteen participating subjects were actual or former students of the Ulm University. The ages of the participants reached from 20 to 30 years (by an average age of 24.5 years). The group of participants consisted of 8 male, 7 female, and one other. 15 out of 16 subjects were right-handed. 7 out of 16 had visual aids.

Most of them had already experience with virtual reality: 13 of 16 participants used VR at least once, 11 out of 16 had even more than one month of experience in VR. Half of the subjects used VR frequently. The subject also got asked if they ever used games in VR, where pointing and shooting is the main task. Half of the subjects reported so. When to judge their skills, the subjects reported an average value of 2.5 (by a scale from 1 to 5; 1 is basic and 5 is skilled).

The duration of the study was 40 minutes to one hour, mostly based on the pointing skills and strategies of the participants.

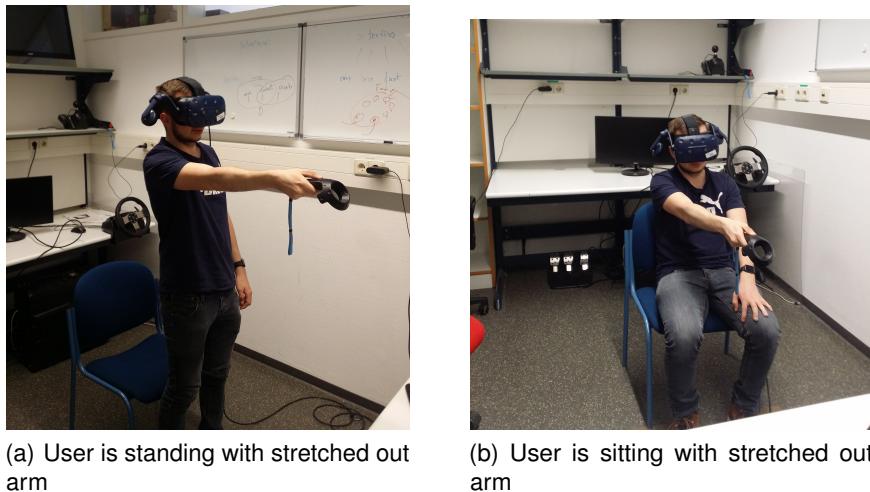


Figure 3.4: Body positions in study

3.2.2 Procedure

The user held the controller in his dominant hand and was able to interact via a click on the trigger. After the user read the task (Task description illustrated by Figure 3.5 (c)) and changed his arm and/or body position he could start the round by clicking the trigger. By starting a round, all controller positions, rotations, and clicks were tracked

3 CONTENT

and logged. Between these rounds, the user had the chance to have a small break and rest his arm.

Each of these eight rounds showed one target at a time. The user had to direct his controller to the target (the user could see where he was pointing by a raycast) and click for the first time. This click was called the “ballistic click” because it resulted out of the motion. It wasn’t necessary for the click to be on target. The data should show where the user aimed to and where he finally ended when clicking the trigger. For the second click, the user had to be on target for at least 500 milliseconds (illustrated by a timer turning from red to green - example in Figure 3.5 a & b). This second click was called “static click”. The click in this position was necessary to compare the differences between a click out of motion and one out of a static pointing on the target.

All targets were placed on a circle, which implemented a standard Fitts’ task. But only the actual target was shown to the user. After the two clicks on one target, the next target was shown.

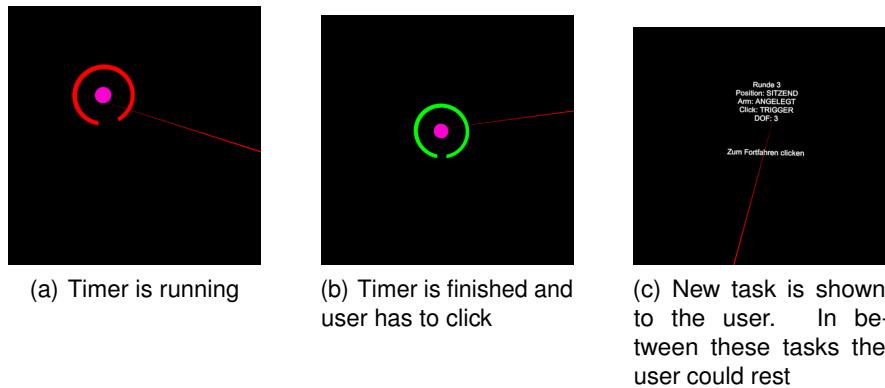


Figure 3.5: Procedure in real application

3.3 Collected Data

The software saved all tracked data in csv files (**character separated values**) for each user. Following values get tracked:

- Identification of the user (only an id to match with the demographic data)
- Standard information about the task (arm and body position; degrees of freedom; a timestamp)
- Events and system states (pressed trigger, click, etc.)

- Strength of pressure on the trigger (value between 0 and 1)
- Position and size of the target
- Position and rotation of the controller
- Position of pointer on canvas

These values were tracked and saved for each frame. To ease the evaluation of the data (all of the tracked data for one participant ranged from 20 megabytes (MB) to 45 MB.), the software produced two extra log files. In one of them, the point of first press and click were selected and the difference between them calculated and saved. In the other one, the software saved the calculated *Throughput* and the values (*Effective Width*, *Effective Distance*, *Effective ID*, *Effective Throughput* and *Mean Movement Time*) based on the model of the Extended Fitt's Law, explained in Section 2.2.3. So each trial with one participant ended with three anonymous log files.

4 Results

To find indications for the “Heisenberg Effect” in the collected data, it was important to first take a look at other effects that may occur and had an influence on the accuracy of the pointing tasks. The paper done by Pavlovych and Stuerzlinger [7] suggest to take a deeper look on hand tremor and device jitter to precisely analyze effects in distal pointing tasks.

4.1 Clearing the data from disturbing effects

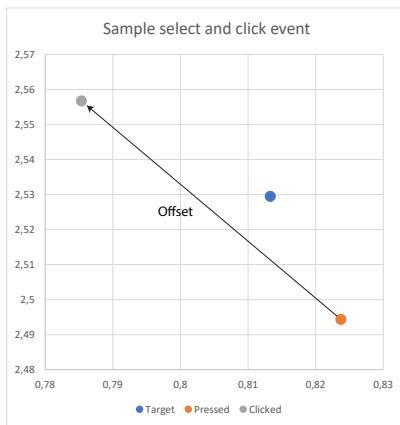


Figure 4.1: A single sample from the data.

4.1.1 Calculation offset

To clear the data from disturbing effects, it was important to analyze the overall offset at first. With offset, it is meant the distance between the position of the cursor, where the user first slightly pressed the trigger (a slight press is not activating a click event. The trigger has to be fully pressed in), and the position of the cursor, when the user clicked and hit the target. It was to suspect, that in this offset, the “Heisenberg Effect” can be found. For illustration, Figure 4.1 shows in orange the position when the user

4 RESULTS

first pressed the trigger. The blue dot marks the actual target and the grey dot marks the position where the controller pointed at when the user fully clicked the trigger. The distance between the pressed position and the clicked position is the described offset.

4.1.2 Device-caused jitter

As mentioned in Subsection 3.1.6, the jitter caused by the controller and the VR headset were tracked while both devices weren't moved at all. The controller was placed on the floor and the headset on a table. The tracked positions showed a small fluctuation in the position reporting of the devices. With the saved positions a maximum and minimum of these fluctuations were calculated and set in perspective with the other effects. In Figure 4.2 is shown how minimal the effect of the jitter based on the devices is, compared to the overall offset. The offset is in this figure, the difference between the point the user is first pressing the button to select a target, and the position the click event is tracked.

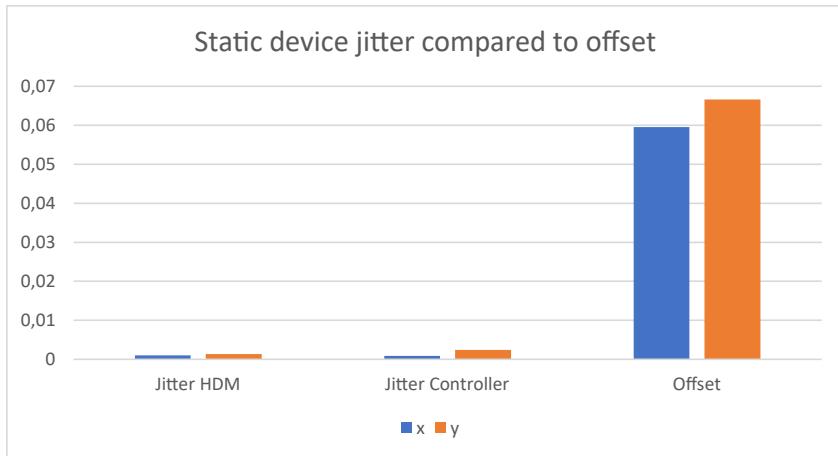


Figure 4.2: Static jitter of the devices compared to the actual offset between intention to click and result hit on the target

4.1.3 Hand tremor

To determine the effects of hand tremor to the data, the changes in position were tracked, when users had to hold still in the target for finishing the timer. These fluctuations were plotted over all tasks. Tasks are here the different stages of the procedure: to stand

up/sit down, to stretch the arm out/apply it to the body and the degrees of freedom given to the controller. The data for the offset was also distributed over these tasks and set in comparison to the data calculated as the hand jitter. The result is shown in Figure 4.3. The figure shows clearly, that hand tremor has an impact on the accuracy of the pointing but can't be declared as the main reason for the offset. It is also shown, that with the arm stretched out hand tremor increases as expected.

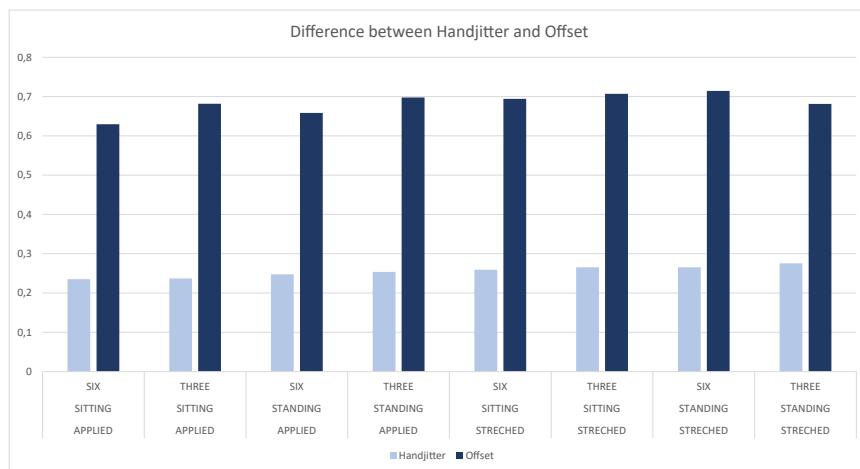


Figure 4.3: Difference between offset by hand tremor and overall offset plotted over all different tasks

4.2 Indications for the Heisenberg Effect

After clearing the data and setting it in comparison with other effects, it is clear, that hand tremor and device jitter aren't the main reasons for the offset between aimed position and hit position in the collected data. The results suggest, that the, first by Bowman [1] mentioned, "Heisenberg Effect" is the reason for this offset.

If all data is plotted together, the effect gets even clearer. A look at Figure 4.4 shows, that the offset has one direction, upwards. The blue dots symbolize the targets on the Fitt's Law based circle (as explained in detail in subsection 3.1.3), and the green and red dots symbolize the final position of the hit, when the click was tracked (green are the positions with 3DOF and red are the positions with 6DOF). Hand tremor and device jitter would have caused balanced "hit fields" around the targets. The collected data suggest, that the effects of hand tremor and jitter result in a normal distribution around

4 RESULTS

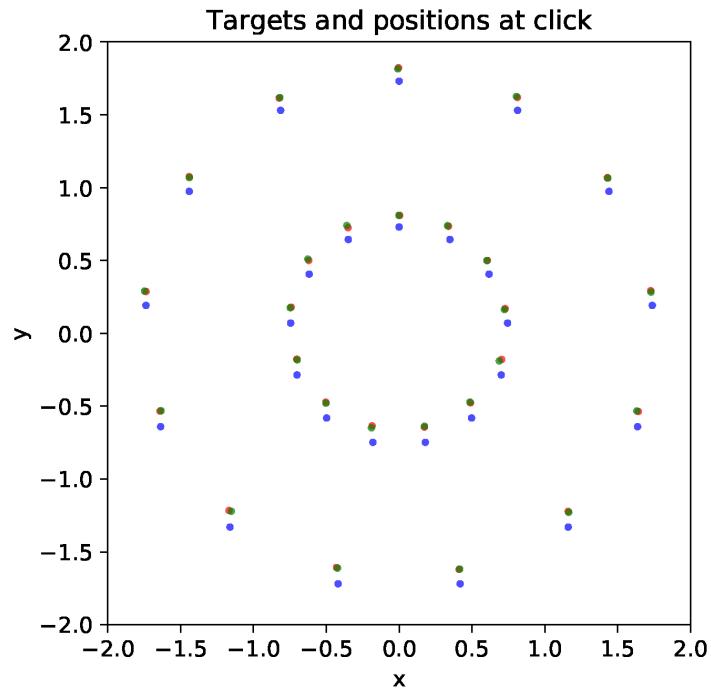


Figure 4.4: Combination of all targets and results

the targets. But as the Figure shows, all hits are above the target. This suggests the conclusion, that the up going movement of the click, is resulting in this offset.

5 Conclusions and future work

5.1 The “Heisenberg Effect”

The analysis of the collected data in section 4.2 shows that there is an effect, which cannot be explained by hand tremor or device jitter, resulting in a worse pointing accuracy and maybe even a worse throughput. This effect shows many similarities to the described “Heisenberg Effect” by Bowman [1] and Kopper et al. [4], like the occurred offset in direction of the pressed trigger, demonstrated in Figure 4.4.

The incentive to this work was to show evidence for the “Heisenberg Effect”, analyze its effects and give a good basis for compensating it. The results which are shown in this thesis leap to the conclusion, that the “Heisenberg Effect” exists and has a bigger impact on pointing accuracy and throughput as previously thought.

5.2 Compensation for the “Heisenberg Effect” and Implications for future work

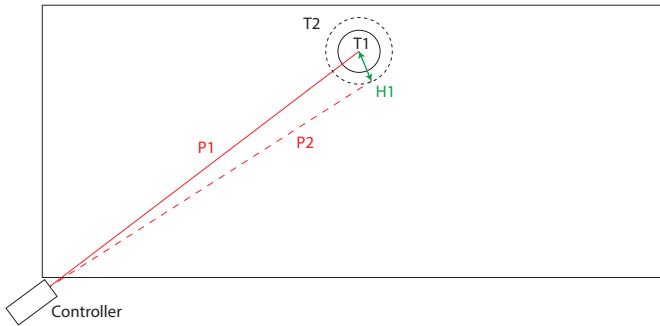


Figure 5.1: Possible Compensation for the Heisenberg Effect

Because the “Heisenberg Effect” shows a similar range of effect on different modalities (like shown in Figure 4.3), there is a chance, this effect can be predicted and measured. If this is, in an efficient algorithm and time possible, the effect could be canceled out in real life applications. For future work, it would be interesting to analyze the “Heisenberg

5 CONCLUSIONS AND FUTURE WORK

Effect" in more real-life applications. A possible way to implement this, was already given by MacKenzie in his work on an extended Fitt's Law (described in subsection 2.2.3) [6]: With enough data, it is possible to calculate if the user **wanted** to hit the target, more than if he really did. The software could now count a target as clicked on or hit, even if the user didn't hit it directly. Future work would have to take in consideration, that the created lag, caused by this additional computing time, could also have an effect on the performance of the user, like suggested by Pavlovych and Stuerzlinger [7].

The Figure 5.1 shows such a possible situation. The user wants to click on the target (T1) and aims the controller on this target (Line P1). Based on the disturbance in pointing made through the click (Line P2), the system would not detect this click, as one in the target. Because the "Heisenberg Effect" can be plotted over time, the system could now detect, that the user wanted to hit a target, a little bigger than the actual target (bigger target labeled as T2). Now the click is in the expanded target and the click would count.

One problem with the "Heisenberg Effect" wasn't cleared already and could be a topic for future work: The decrease of pointing accuracy could have an effect on the throughput. This aspect of the "Heisenberg Effect" wasn't discussed in this work. To sum up this work: it is clear that the effect needs to be farther researched, but a good base is given here.

5.3 Acknowledgements

Special thanks to Dennis Wolf for his help and advice in creating this thesis.

List of Figures

2.1	Describing the Heisenberg Effect on a simple distal pointing task example	5
3.1	Different Layout attempts	10
3.2	Final implementation	12
3.3	Multidirectional pointing & clicking task	13
3.4	Body positions in study	15
3.5	Procedure in real application	16
4.1	A single sample from the data.	19
4.2	Static jitter of the devices compared to the actual offset between intention to click and result hit on the target	20
4.3	Difference between offset by hand tremor and overall offset plotted over all different tasks	21
4.4	Combination of all targets and results	22
5.1	Possible Compensation for the Heisenberg Effect	23

List of Tables

3.1 Used Index of Difficulty	11
3.2 Task modes	13

Bibliography

- [1] D. Bowman, C. Wingrave, J. Campbell, and V. Ly, "Using Pinch Gloves(TM) for both Natural and Abstract Interaction Techniques in Virtual Environments," Jan. 2001. [Online]. Available: <http://eprints.cs.vt.edu/archive/00000547/>
- [2] S. A. Douglas, A. E. Kirkpatrick, and I. S. MacKenzie, "Testing Pointing Device Performance and User Assessment with the ISO 9241, Part 9 Standard," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '99. New York, NY, USA: ACM, 1999, pp. 215–222. [Online]. Available: <http://doi.acm.org/10.1145/302979.303042>
- [3] J. Kong and X. Ren, "Calculation of Effective Target Width and Its Effects on Pointing Tasks," *Information and Media Technologies*, vol. 1, no. 2, pp. 1057–1059, 2006. [Online]. Available: https://www.jstage.jst.go.jp/article/imt/1/2/1_2_1057/_article/-char/ja/
- [4] R. Kopper, D. A. Bowman, M. G. Silva, and R. P. McMahan, "A human motor behavior model for distal pointing tasks," *International Journal of Human-Computer Studies*, vol. 68, no. 10, pp. 603–615, Oct. 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1071581910000637>
- [5] I. S. MacKenzie, "Fitts' law as a research and design tool in human-computer interaction," *Human Computer Interaction*, vol. 7, no. 1, pp. 91–139, 1992. [Online]. Available: https://doi.org/10.1207/s15327051hci0701_3
- [6] I. S. MacKenzie and W. Buxton, "Extending Fitts' Law to Two-dimensional Tasks," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '92. New York, NY, USA: ACM, 1992, pp. 219–226. [Online]. Available: <http://doi.acm.org/10.1145/142750.142794>
- [7] A. Pavlovych and W. Stuerzlinger, "The Tradeoff Between Spatial Jitter and Latency in Pointing Tasks," in *Proceedings of the 1st ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, ser. EICS '09. New York, NY, USA: ACM, 2009, pp. 187–196. [Online]. Available: <http://doi.acm.org/10.1145/1570433.1570469>

BIBLIOGRAPHY

- [8] R. W. Soukoreff and I. S. MacKenzie, "Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI," *International Journal of Human-Computer Studies*, vol. 61, no. 6, pp. 751–789, Dec. 2004. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1071581904001016>
- [9] R. J. Teather, A. Pavlovych, W. Stuerzlinger, and I. S. MacKenzie, "Effects of tracking technology, latency, and spatial jitter on object movement," in *2009 IEEE Symposium on 3D User Interfaces*, Mar. 2009, pp. 43–50.
- [10] R. J. Teather and W. Stuerzlinger, "Pointing at 3d targets in a stereo head-tracked virtual environment," in *2011 IEEE Symposium on 3D User Interfaces (3DUI)*, Mar. 2011, pp. 87–94.

Declaration

I hereby declare that this thesis titled:

An In-Depth Analysis of and Compensation for the Heisenberg Effect

is the product of my own independent work and that I have used no sources or materials other than those specified. The passages taken from other works, either verbatim or paraphrased in the spirit of the original quote, are identified in each individual case by indicating the source. I further declare that all my academic work was written in line with the principles of proper academic research according to the official "Satzung der Universität Ulm zur Sicherung guter wissenschaftlicher Praxis" (University Statute for the Safeguarding of Proper Academic Practice).

Ulm,

Marco Combosch