Chair for Computer Science 10 (Media Computing and Human-Computer Interaction)

RWTH AACHEN UNIVERSITY

Rotating Objects: Implementation and Evaluation of Rotation Techniques for the ARPen System

Bachelor's Thesis
submitted to the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Donna Klamma

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof.Dr.Torsten Wolfgang Kuhlen

Registration date: 07.06.2019
Submission date: 29.08.2019

Eidesstattliche Versicherung

Klamma, Donna Marie	355381
Name, Vorname	Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

Rotation von Objekten: Implementierung und Evaluation von Rotationsmethoden für das ARPen System / Rotating Objects: Implementation and Evaluation of Rotation Techniques for the ARPen System

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Aachen, den 29.08.2019	
Ort, Datum	Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Aachen, den 29.08.2019	
Ort, Datum	Unterschrift

Contents

List of Figures

List of Tables

Abstract

Augmented Reality (AR) is on the rise with a focus in development on creating applications for smartphones which fulfill practical purposes. One of these purposes is 3D modeling. Within tools for 3D modeling, there are several basic operations which are utilized to transform and interact with virtual objects. Some of those are selection, translation, scaling and rotation.

In this Bachelor's thesis, we explored five rotation techniques for the ARPen system which strives to make mid-air 3D modeling within a bimanual, handheld AR system possible through employing a smartphone in conjunction with the 3D printed ARPen. The chosen techniques were derived from research into related endeavors. As a result, the five rotation techniques we defined and implemented into our application were (1) *touchscreen rotation*, (2) *direct device rotation*, (3) *device rotation with "pedal" effect*, (4) *direct pen rotation* and (5) *pen rotation with "pedal" effect*.

We conducted a user study evaluating the techniques on quantitative factors such as speed and precision, as well as qualitative measures received from the participants in form of a rating in seven categories and a final ranking of all techniques. *Direct device rotation* has crystallized itself as the technique best at meeting our requirements. Besides an overall positive feedback from participants, it also excelled in the measurements for speed and precision. Additionally, we noticed good results from *touchscreen* and *direct pen rotation* which could be further improved through implementing suggestions brought up during the study. Only the techniques involving a "pedal" effect were most often considered negatively during the qualitative remarks by participants.

Überblick

Augmented Reality (AR) ist auf dem Vormarsch und konzentriert sich in der Entwicklung auf die Kreation von Smartphone-Anwendungen, welche praktische Zwecke erfüllen. Einer dieser Zwecke ist die 3D-Modellierung. Innerhalb von Anwendungen für 3D-Modellierung gibt es mehrere grundlegende Operationen, die zur Transformation und Interaktion mit virtuellen Objekten verwendet werden. Einige davon sind Auswahl, Translation, Skalierung und Rotation.

In dieser Bachelorarbeit haben wir fünf Rotationstechniken für das ARPen-System untersucht, welches darauf abzielt, die 3D-Modellierung im Raum innerhalb eines bimanuellen, tragbaren AR-Systems durch den Einsatz eines Smartphones in Verbindung mit dem 3D gedruckten ARPen zu ermöglichen. Die gewählten Techniken stammen aus Nachforschungen zu ähnlichen Projekten. Die fünf Rotationstechniken, die wir definiert und in unserer Anwendung implementiert haben, waren (1) *Touchscreenrotation*, (2) *direkte Handyrotation*, (3) *Handyrotation mit "Pedal"effekt*, (4) *direkte Stiftrotation* und (5) *Stiftrotation mit "Pedal"effekt*.

Wir haben eine Benutzerstudie durchgeführt, in der die Techniken zu quantitativen Faktoren wie Geschwindigkeit und Präzision sowie durch qualitative Messungen in Form einer Punktevergabe durch die Teilnehmer in sieben Kategorien und einer abschließenden Rangliste aller Techniken bewertet wurden. *Direkte Handyrotation* hat sich als die Technik herauskristallisiert, welche unsere Anforderungen am besten erfüllt. Neben einem insgesamt positiven Feedback der Teilnehmer überzeugte sie auch bei den Messungen für Geschwindigkeit und Präzision. Zusätzlich haben wir gute Ergebnisse bei *Touchscreen* und *direkter Stiftrotation* festgestellt, welche durch die Umsetzung von Vorschlägen, die während der Studie gemacht wurden, weiter verbessert werden können. Nur die Techniken mit "Pedal"effekt wurden bei den qualitativen Bemerkungen der Teilnehmer meist negativ bewertet.

Acknowledgements

I would like to thank the Media Computing Group of the RWTH Aachen for hosting, as well as Prof. Dr. Borchers and Prof. Dr. Kuhlen for examining, this thesis.

Furthermore, I want to thank my supervisor Philipp Wacker for giving me the necessary advice for the execution of this project.

I'm also grateful for all of my family and friends who supported me throughout the last couple of month.

At last, I want to thank everybody who was so kind to participate in the user study and give me an hour of their precious time.

Conventions

Throughout this thesis we use the following conventions.

Text conventions

Source code and implementation symbols are written in typewriter-style text.

`myClass`

The whole thesis is written in American English.

Chapter 1

Introduction

The term Augmented Reality (AR) was originated in the 1950s, coined by Thomas P. Caudell in 1989 and first applied to a mobile application by Bruce Thomas in 2000 (Heilig [2019]). While there are ambitions to achieve almost unnoticeable AR integration through e.g. contact lens projections in the future, nowadays the hardware placed between user and information still holds a key role in the discussion around AR. (Alkhamisi and Monowar [2013]) Head-Mounted Displays (HMDs) for Virtual Reality (VR) applications such as the HTC Vive[1] and the Oculus Rift[2] have strengthened the development and numbers of software in their field, whereas HMDs for AR generally termed ″smartglasses″ are only now on the rise (Sumra [2019]).

Development in AR focuses majorly on smartphone applications.

Through the results of a 2018 survey (Perkins Coie LLP [2018]) we determined that while cost is a concerning factor for VR due to the high price tags attached to HMDs, and might turn away less tech-savvy consumers, a lack of content is currently a bigger issue for AR. A focus on development of AR applications for smartphones (82%) and a stronger leaning in both VR and AR away from the to this point prominent gaming industry towards more practical applications is highlighted. Despite HMDs advanced tracking and graphic options,

---

[1] `https://www.vive.com/eu/` *(Accessed: 28.08.2019)*

[2] `https://www.oculus.com/rift/` *(Accessed: 28.08.2019)*

only about 35 million units are expected to be sold in 2019[3] with numbers for earlier years being lower. In comparison, 2.71 billion unique smartphone users worldwide[4] benefit if AR is made available to them through their already owned hardware.

Our project is the ARPen system which strives to make mid-air 3D modeling accessible through being a handheld AR system using an iPhone and a 3D printed pen as well as an open-source iOS application. Our concern with mid-air 3D modeling and personal fabrications as a utilization of the system fit into the trend of practically orientated mobile AR applications. Since the beginning of development for the ARPen system in 2018 (Wehnert [2018]), we have been investigating how to integrate basic 3D modeling operations found in popular Computer Graphics (CG) and 3D modeling applications. Some of these essential operations have already been developed for the ARPen system while others are still missing.

We want to find an intuitive and effective rotation technique for the ARPen system.

After studies on the selection and translation of virtual objects have already been conducted (Wacker et al. [2019]), this Bachelor's thesis focused on developing methods for rotating objects mid-air using the various inputs provided by our bimanual system. *Bimanual* in this context describes the coordination of smartphone and ARPen for interaction by the same user. The goal was to produce intuitive methods to let even untrained users efficiently solve 3D modeling tasks. For this we identified approaches that have been successful in similar endeavors, implemented them fitting for our system and evaluated them through a user study.

The first task was to determine the ARPen orientation from the tracked marker system. Afterwards, we developed five rotation techniques based on the three input methods we defined within the system: the touchscreen, the device motion and the ARPen. Special interest was put into pitching touchscreen against mid-air interaction, and whether the

---

[3] `https://www.statista.com/statistics/509154/head-mounted-displays-worldwide-shipments/` *(Accessed: 24.08.2019)*

[4] `https://www.bankmycell.com/blog/how-many-phones-are-in-the-world` *(Accessed: 24.08.2019)*

bimanual nature of the ARPen system could be as beneficial in this case as it was for selection and translation tasks where techniques involving the ARPen directly performed very well. (Wacker et al. [2019])

Chapter 2

Background

This chapter provides the necessary background information to put this thesis into context. We first introduce the idea of Augmented Reality and then move forward to describe the ARPen application as the system we are working with, including a brief discussion of the frameworks used.

2.1 Augmented Reality

Augmented Reality places virtual objects into real world space with which the user can interact in real time.

In 1997, Azuma [1997] has presented the often cited definition of Augmented Reality (AR) as a system that (1) combines real and virtual elements with which the user can not only (2) interact in real time, but which is also (3) registered in 3D space. AR requires a real world space in which virtual objects can be placed. This could seem like a restriction, but it gives the user unique enhancements for the tasks they are trying to solve and strives to provide "natural and intuitive means for users to navigate or work efficiently in the real world" (Shin et al. [2005]).

AR provides an improved spatial understanding. The ability to look at an object from a new point of view enhances the efficiency in which tasks based in 3D space are completed (Shin et al. [2005]). An AR application needs to be able to capture a scene, identify the important information within it, and process the scene, so it can place the virtual content at the desired places (Alkhamisi and Monowar [2013]).

Technological advances let AR become more beneficial for practical applications.

The technology has already established itself in the gaming and entertainment industry, which is evident through the rise of mobile applications such as PokemonGo [1]. A high "fun factor" with a lower degree of accuracy (Seidinger and Grubert [2016]) has limited the practical use of such applications in the past. But as the technology advances and the data processing speed needed for real-time transfer of information between user and hardware becomes accessible (Li et al. [2017]), AR is used for education in classrooms (e.g. Brown [2015]) and museums (e.g. Billock [2017]) as well as in maintenance (e.g. Potter [2019]) and other technical applications where a super imposed view can provide another layer of information, guide the user in their tasks and drive their creativity.

Mid-air 3D modeling saves time by combining steps in the working process.

With a multitude of possible purposes for AR (Chatzopoulos et al. [2017]), geometry modeling being one of them, the demand for further research in the area is given. AR could provide an accessible tool for mid-air 3D modeling as instead of measuring out a real object and constructing a CAD (*computer aided design*) model as a fitting part afterwards, the user can directly trace a model in real time. As a result of this, the user saves several steps in the work process by combining them. For this purpose, the Media Computing Group of the RWTH Aachen[2] has developed a handheld AR system to explore interaction techniques within 3D mid-air modeling and provide a tool for personal fabrication.

2.2 The ARPen System

The ARPen system[3] has been developed as an open-source iOS application which tracks the 3D printed ARPen

---

[1] `https://www.pokemongo.com/en-us/` (Accessed: 24.08.2019)
[2] `https://hci.rwth-aachen.de/` *(Accessed: 27.08.2019)*
[3] `https://github.com/i10/ARPen` (Accessed: 24.08.2019)

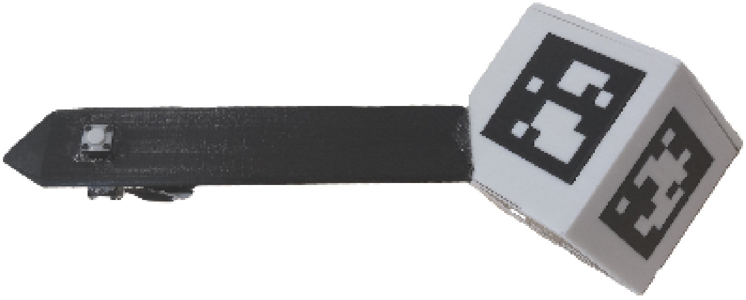


Figure 2.1: The ARPen as an interactive stylus has three buttons connected to a Bluetooth chip which transfers the signals to the device. The box at one end features six trackable arUco markers.

through an iPhone's camera. At one end, the ARPen as seen in Figure 2.1 is equipped with a cube of six arUco[4] markers which makes tracking it from all points of view possible. The markers each resemble a black frame around a non-symmetrical 6x6 pattern of black and white squares whose contrasting colors make tracking easier. They were taken from the marker dictionary *ARUCO_MIP_36h12* and encode the ids "1" to "6" which enables the application to calculate the position of the `penTip` node used for tasks involving the ARPen like e.g. drawing lines. A Bluetooth chip transmits the state of the buttons on the ARPen for further interaction.

A cube of six trackable arUco markers on the ARPen is used to calculate the position of the `penTip` node.

The framework used to make AR possible using an iPhone is Apple's ARKit[5] with the help of SceneKit[6] for 3D rendering. For the ARPen system, as is usual with SceneKit, the scene is organized through `SCNNodes`. After the creation of the scene at the start of the application, the origin of the world is set at a certain point in the real world. In relation to that origin other points like for example the position of the node representing the camera, which is pro-

ARKit and SceneKit are employed for the application.

---

[4] `https://sourceforge.net/projects/aruco/` (Accessed: 24.08.2019)

[5] `https://developer.apple.com/documentation/arkit` (Accessed: 24.08.2019)

[6] `https://developer.apple.com/documentation/scenekit` (Accessed: 24.08.2019)

vided by ARKit, can be calculated. Furthermore, in relation to the camera node, the marker positions are detected and the globally available `penTip` node, which is relevant for some of the rotation tasks, is derived from it Nowak [2019]. The `penTip` node is then attached as a child node to the camera node.

The selection method and how to hold the device were determined in earlier studies.

Already researched and evaluated have been the basic operations of selecting and translating virtual objects within the ARPen system. The selection method of *penRay* selection, which is further described in a later chapter, has performed as one of the best techniques within the respective user study and was suggested to be integrated into the ARPen system. As a grasp when using the left hand, holding the phone with the camera on the right by supporting it with the pinkie finger was evaluated best and chosen to be employed in further studies. (Wacker et al. [2019])

A majority of the code was written in Swift[7] using Xcode. Additionally, within the ARPen system Objective-C and Objective-C++ were used to import C++ frameworks such as arUco.

---

[7] `https://swift.org/` (Accessed: 24.08.2019)

Chapter 3

Related Work

With the goal of identifying viable rotation methods which can then be evaluated for the ARPen system in mind, we take a look at related work in the subject area. First, as a primary goal of the ARPen project is to create a tool with which mid-air 3D modeling becomes accessible, the following section of this chapter will be utilized to take a look at what kind of tools have already been developed in the fields of and surrounding AR, and how the ARPen could offer new ideas. After this general look at the related work which includes applications with commercial use, a discussion of rotation methods which have already been evaluated for similar research projects will follow.

3.1 Applications and Demand for Mid-air 3D Modeling

HMDs are popular for VR applications, but are also expensive.

Virtual Reality (VR) portrays a completely virtual scene to the user that takes little to no input from the real world except for the user's movements and commands. While there are other ways to present VR applications such as the aixCAVE[1] which surrounds the user with on screen projec-

---

[1] `http://www.itc.rwth-aachen.de/cms/IT-Center/Forschung-Projekte/Virtuelle-Realitaet/Infrastruktur/~fgqa` (Accessed: 24.08.2019)

tions of the virtual scene in all directions, HMDs such as the Oculus Rift[2] are the most popular and accessible ones. Even though their prices are expected to sink in the coming years (Boland [2018]), HMDs still struggle through an expensive price tag reminiscing that of a gaming console. This could be a factor leading to a hard time attracting a magnitude of users.

In the field of 3D modeling VR has already come very far and is rapidly evolving. After the release of Google's Tilt Brush[3] in 2016 which lets the user draw in virtual 3D space using handheld controllers and a HMD, nowadays software like Gravity Sketch [4] allows the user to create complete models with many tools reminiscent of popular desktop based 3D modeling software such as Autodesk Maya[5] or Blender[6].

AR/VR software developers value 3D modeling and personal fabrication as applications.

In the development of the collaborative 3D modeling VR software Unbound[7], special interest was put into producing models which can then be 3D printed. This indicates the interest in the field of Personal Fabrication as an application of the technology.

Grib[8] is an AR modeling software that has the aim to utilize a phone for an intuitive user experience where experience in 3D modeling is not as important, and proficiency in applying the software can be gained faster. This implies that the market is not only looking to make AR 3D modeling accessible through the use of smartphones, but to also create an application appealing even to users who are not willing to put a lot of time and effort into learning the required tools.

Other research projects also use trackable markers on a pen for interaction.

In addition, there are already AR applications using a trackable pen to interact with the scene same as the system does. Seidinger and Grubert [2016] use the different markers not

---

[2] `https://www.oculus.com/rift/` (Accessed: 24.08.2019)

[3] `https://www.tiltbrush.com/` (Accessed: 24.08.2019)

[4] `https://www.gravitysketch.com/` (Accessed: 24.08.2019)

[5] `https://www.autodesk.com/products/maya/overview` (Accessed: 24.08.2019)

[6] `https://www.blender.org/` (Accessed: 24.08.2019)

[7] `http://unbound.io/` (Accessed: 24.08.2019)

[8] `https://grib3d.com/` (Accessed: 24.08.2019)

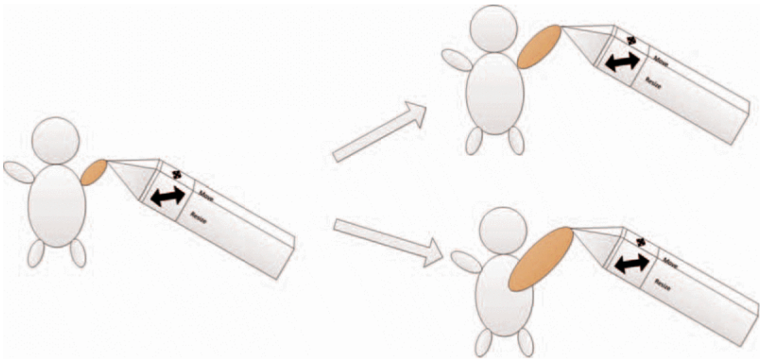


Figure 3.1: Seidinger and Grubert [2016] used markers on a passive stylus for detecting the type of interaction a user wants to perform. Displayed is the scaling of a character's arm.

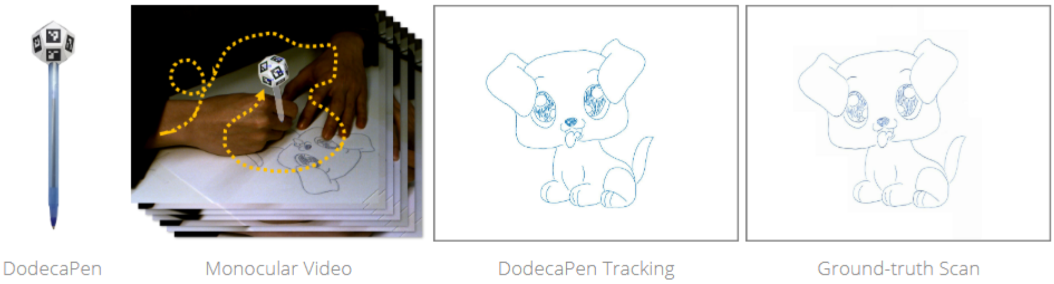


Figure 3.2: The DodecaPen also uses markers for real-time position tracking, but only consists of a passive stylus. Image taken from Wu et al. [2017]

to calculate the pen's position and orientation as is the case with the ARPen, but to signal which kind of interaction (e.g. translation, rotation, etc.) should be performed as seen in Figure 3.1. For this system users suggested that they would rather have hardware buttons on the pen to press than having to interact with the touchscreen. As these are integrated into the design of the ARPen, it is a desire we were able to fulfill and put to test in our user study.

Similarly, the DodecaPen (Figure 3.2) designed by Wu et al. [2017] features a marker system comparable to the ARPen, but also only provides a passive pen to interact with the scene. In order to make it possible to enter a "rotation mode" which lets the user differentiate the action from other tasks such as translating objects, having some kind of button interaction – whether on the touchscreen, through a virtual panel or on the pen itself – would be necessary. Utilizing a button on the pen itself could aid in maintaining the faster workflow AR strives to achieve as the user is not required to change their hand positioning mid-task.

Making 3D modeling in AR accessible is the goal of several independent projects.

The ARPen system follows a trend of creating AR experiences with an emphasis on 3D modeling and personal fabrication made accessible by utilizing hardware which is less expensive or which most potential users already own. As an enhancement of the existing system the basic operation of rotating virtual objects must be integrated. For this we looked at other research projects which are comparable to the ARPen system in one way or another and investigated how they have resolved the integration of the operation.

3.2 Rotation Techniques in Related Work

The first task in the development of rotation techniques for the ARPen system was defining the goals of the application. In order to define which inputs to use and how to evaluate them in the later user study, we had to decide whether we wanted to try to provide a tool which could be operated by users with little to no experience in 3D modeling to achieve results without having to learn a complex tool first, or if we also wanted to provide a portable tool for more experienced users. Therefore, to start off we looked into literature in order to find out how AR applications employing mid-air object manipulation performed in comparison with professional desktop-based software.

Mid-air gestures can save time during rotation tasks.

Wang et al. [2011] (Figure 3.3) have concluded that eliminating the adjustment of the camera view as an individually selectable task when employing AR, and therefore creating fewer transitions between tasks, using mid-air gestures for the rotation of objects in 3D space can be a significant time saver. In comparison with a traditional CAD software used in conjunction with a desktop pc and mouse, those results hold true and let an expert user save up to 40% of their time. When compared with an expert user working

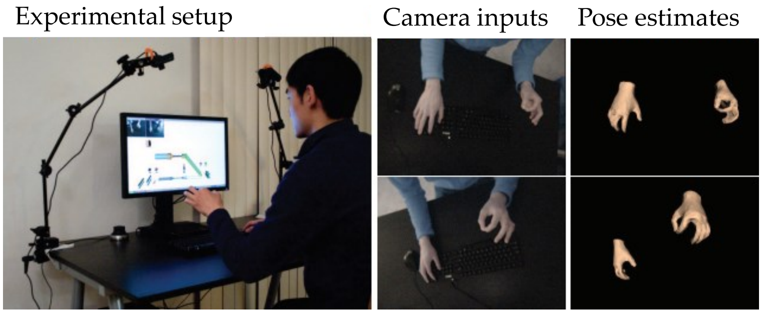


Figure 3.3: The image taken from Wang et al. [2011] shows their markerless tracking system which they used to assemble CAD models. They utilized both hands to control six degrees of freedom.

with a 3D mouse though, the AR application fell behind.

In an early presentation for the topic of this thesis, special interest in using the device motion as the trigger for object rotation was expressed. Harviainen et al. [2009] have shown that users can intuitively pick up on how to rotate objects through device motion and do not need explicit instructions. The phone movement simply mirrors the desired rotation closer than the motions performed on a 2D screen.

Using device motion for rotation tasks is intuitive.

When device motion is used as an initiator of rotation, another aspect resulting from the bimanuality of the system might prove important. As the users will be asked to hold the device in their left hand in order to reserve the right hand for the ARPen, most users will operate the device motion tasks with their non-dominant hand. Guiard's kinematic chain model suggests that the dominant hand works on a finer spatial-temporal scale (Edge and Blackwell [2009]) and would therefore provide higher accuracy. Effects where users struggle to achieve precise results while using the non-dominant hand for rotation might come into play.

Performing tasks with the non-dominant hand might affect accuracy.

Despite that, there are also advantages in utilizing the device motion as describe by (Shin et al. [2005]). It has been shown that humans have a better understanding of the

Figure 3.4: Harviainen et al. [2009] used only camera-based input for their application. Even with earlier technology, device motion as a technique was appreciated by users.

scene when their viewpoint is changed opposed to a scene simply being rotated. Especially physical movement towards a new observation point enhances that understanding. Therefore, rotation methods where the object is glued to the viewpoint or moves relatively to physical motions of the user might be easier to work with then a method where parts of the scene rotate despite a constant viewpoint.

Tilting the viewpoint away is not a significant problem.

Henrysson et al. [2007] demonstrated that in their application device motion performed similarly well as keyboard input despite the inability to rotate around the z-axis pointing through the camera. They concluded that after the removal of this problem resulting from limitations in technology, device motion could perform significantly better. Within the framework used for the ARPen, all degrees of freedom are accessible from the device motion input and users are able to utilize the twisting motion as they have intuitively tried in the mentioned above study. The study also investigated how tilting the screen can obscure the view from the user, but noted that this problem did not arise for most users in their evaluation.

Comparing users who only have experience using touchscreens opposed to users who already have experience with mid-air interactions in virtual 3D environments might show whether a switch to mid-air interaction methods could provide a better user experience in the future. It was shown that users with little to no 3D modeling experience think of hand gesture interaction – which would be comparable to using the ARPen in the ARPen system – for rotation as more intuitive than using only the 2D screen (Bai et al. [2014]).

Most rotation techniques interacting solely with the touchscreen also require the user to apply a different gesture for rotating around the z-axis (mostly a two-finger touching rotation versus a one-finger touching approach). This can confuse some participants and is generally not intuitive but requires explanations at first. (Bai et al. [2014])

Touchscreen rotation only involving one way to rotate virtual objects is more intuitive.

3DTouch as described by Mossel et al. [2013] only uses horizontal and vertical swipes across the screen for rotation tasks in x- and y-direction respectively. In order to achieve mobility around all possible axes the user is required to change the viewpoint and therefore move the device around. This only calls for a single gesture during the rotation task and could consequently be judged as more intuitive.

While mid-air gestures opposed to touchscreen interactions are generally considered more intuitive and fun, participants of studies have also reported physical stress in handheld AR (Bai et al. [2014]). Mid-air gestures in a bimanual system such as with the ARPen require the user to hold the phone at a certain distance to the hand performing the gesture. Dependent on the interaction method chosen, the user must also move the phone around frequently. These unfamiliar situations can lead to muscle aches even after a short time period. The strain on the body using different techniques would therefore be a factor that is to be judged in the user study. Common touchscreen interaction will in

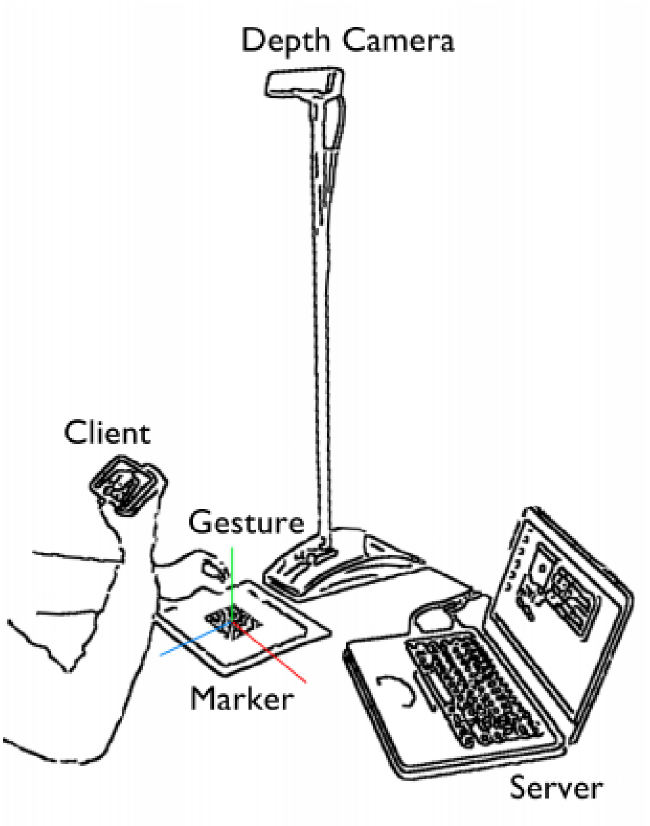


Figure 3.5: The setup by Bai et al. [2014] for detecting free-hand gestures for interaction involves a depth-camera. Despite being slower, users found 3D gesture inputs to be intuitive and fun, and gave them higher ratings.

this aspect most likely perform better as most users are already familiar and therefore physically trained in those.

Mid-air gestures have a high “fun factor”.

Multiple studies have concluded that mid-air gestures are better used for entertainment purposes than for practical use as they in most cases cannot provide the same amount of accuracy (Bai et al. [2014], Polvi et al. [2016]). Mid-air gestures perform especially well when assessed on the level of joy experienced during usage (Seidinger and Grubert [2016]) and have as a result potential to perform really well with participants in the user study.

Wacker et al. [2018] have evaluated how much the accuracy in 3D drawings of models improves if the user is working with a virtual or real object to trace around. The results showed that especially concave edges on real objects were getting good results from the users. It was also clearly stated that humans still have a problem with depth detection in virtual environments which could hypothetically get better with further training from working in those. Potentially, this could also act as a negative factor in rotating objects precisely.

Precision differences between touchscreen and mid-air rotations are interesting.

As rotation tasks work with existent virtual objects and are less dependent on depth detection than translation or selection tasks, it is to be evaluated how much accuracy differs between mid-air and touchscreen interactions for those.

The results of this thesis depict the effects the advanced technology as well as the ARPen as an interactive stylus have on the techniques already explored in research, and will enhance the ARPen as a system for 3D modeling.

Input for rotation	Source	Arguments
Device motion	Harviainen et al. [2009]	+ Intuitive
	Shin et al. [2005]	+ Change in viewpoint and physical movement benefit understanding of scene
	Henrysson et al. [2007]	+ Device rotation already performed well with fewer degrees of freedom than accessible to us + Obscuring the view by tilting the device was not a significant issue
	Edge and Blackwell [2009]	- Guiard's kinematic chain model: non-dominant hand has lower accuracy
Pen motion	Bai et al. [2014]	+ Hand gesture interaction more intuitive than being restricted to 2D screen
General mid-air interaction	Seidinger and Grubert [2016]	+ Joy experienced during usage
	Bai et al. [2014]	- Physical Strain
	Bai et al. [2014], Polvi et al. [2016]	- Does not provide the same amount of accuracy
Touchscreen	Mossel et al. [2013]	+ Only using swipes for rotations around x- and y-axis, and changing the viewpoint for further degrees of freedom was evaluated well
	Bai et al. [2014]	- Using both one-finger and two-finger touching in the same technique for rotation tasks is not intuitive

Table 3.1: Table summarizing the results collected in section 3.2 about rotation techniques in related work.

Chapter 4

Description of Rotation Techniques

In this chapter we discuss the techniques chosen for evaluation within the ARPen system based on the research presented. The ARPen application as a bimanual AR system is using the ARPen in combination with a smartphone and therefore offers several input methods. Our goal with this thesis was to find out how to use those to rotate objects midair in the most intuitive and effective way. Therefore, we are taking a closer look at what the ARPen system has to offer first. Afterwards, we define all techniques implemented and evaluated for this thesis.

4.0.1 Input Methods of the ARPen

We initially defined three parts in the ARPen system which can be used either for rotation tasks individually or in different combinations with one another.

The first and most familiar is the touchscreen of the smartphone. This far it has been used to switch modes or adjust settings and will also provide those methods for our purposes. Furthermore, the touchscreen offers a familiar haptic interaction which might prove to be intuitive for the rotation task. (Wacker et al. [2019]) Especially users not familiar with AR/VR systems might work better with touchscreen interactions.

Another input method is provided through the ARPen itself. Due to the tracking of the six markers attached to the marker cube, the orientation of the `penTip` node can be calculated. Through that we can transfer the rotation of the ARPen towards a selected virtual object. In addition, the ARPen provides buttons which can be employed to signal interaction from the user to the system.

We defined three parts of the ARPen system the user can use for rotation tasks: the touchscreen, the ARPen and the device.

At last we have the smartphone as the device. Its motions can also be tracked and translated towards an object. ARKit already provides the properties necessary to get the information about the device orientation from the view.

Considering the ARPen as a handheld stylus delivers a rare aspect in the field of AR applications, and an advancement in tracking technology provides enhanced options, both employing the motion of the ARPen as well as the device motion as interaction methods for rotation tasks and pitching them against the familiar touchscreen interaction is an interesting premise. This also delivers the question of how important the increased spatial awareness given by the ARPen and device motion interaction methods is in regard to rotation tasks or whether the familiarity of the touchscreen suffices to triumph over the other options.

4.0.2 Definition of Rotation Techniques

Considering the key feature of the system is the ARPen itself and mid-air gestures have been found to be intuitive, a technique involving a direct translation from ARPen rotation onto the selected object made sense. In addition, the positive feedback device motion has gotten led to it being chosen as a technique as well.

Touchscreen interaction only involving swiping gestures across the screen has been suggested as familiar and intuitive. The comparison of its results with mid-air interaction techniques is certainly interesting, and was therefore another technique to be explored in this thesis.

Shortly touched upon was the problem of device motion occluding the view of the selected virtual object. In addition, for mid-air interaction in general limited wrist mobility might be a negative factor.

We introduce a "pedal" effect to counteract limitations in mobility.

Both of these problems despite the first being judged less important in an earlier example, would result in the user either having to move around a lot or clutch and grab an object repeatedly during rotation tasks (Hürst and van Wezel [2012]). Therefore, introducing a "pedal" effect may prove more successful. With this method the direction of movement of the device or the ARPen simply suggests the direction of rotation. The angle the device or pen is held at determines the speed of rotation. Whether this method is considered significantly less intuitive than the direct translation or if it can take physical strain away from the user, needed to be evaluated. As a technique not significantly explored in research before, this method was interesting to explore.

In the completed rotation method designs, which were evaluated in the user study, a way to select the objects which the user wants to rotate had to be included. Studies about different selection methods using the ARPen system had already been conducted.

Selection methods which utilize the touchscreen have been evaluated by the users as easier to perform than selecting objects mid-air. Overall though, also considering measured factors like success rate and selection time, a selection via *penRay* was chosen for the ARPen system. For the *penRay* method a ray is cast from the view through the `penTip` node and the first object hit is selected. As depth detection has been a problem for users, this method was the more successful than a mid-air selection where the tip of the ARPen must be within the virtual object. (Wacker et al. [2019])

For the rotation techniques evaluated in this thesis, the selection methods fitting the type of interaction were chosen. When the ARPen is used for rotation tasks, it is a sensible decision to also utilize a pen-based selection method. As the *penRay* selection technique performed best within its respective study, it was adopted for the pen-based methods evaluated for rotation tasks.

We chose *penRay* selection and selection via touchscreen for our purposes.

A rotation performed solely through the touchscreen called for a selection technique via touch which was rated highly for ease of use. This could reduce performance time significantly as the user is not required to switch between pen-based and touchscreen interaction during the task.

For device motion interaction both pen-based selection and selection via touchscreen are not directly linked to the device motion and require an additional action from the user. Therefore, *penRay* selection, which was overall the preferred choice in its study, was picked for rotation tasks involving device motion.

Basic definition of rotation techniques evaluated in this thesis.

Overall, we defined five different rotation techniques that were implemented and evaluated for the ARPen system: **(1)** *touchscreen rotation* with selection via touchscreen, **(2)** *direct device rotation* with selection via *penRay*, **(3)** *device rotation with "pedal" effect* with selection via *penRay*, **(4)** *direct pen rotation* with selection via *penRay*, and **(5)** *pen rotation with "pedal" effect* with selection via *penRay*.

Chapter 5

Implementation

In this chapter we will describe how we implemented both the fixed local coordinate system of the ARPen as well as the five different rotation methods to be evaluated in our user study. All code for the ARPen system can be found on GitHub[1] and encourages developers to design their own features for the ARPen system through the help of plugins. We utilized those to implement the earlier defined rotation techniques, so we can easily switch between them in the user study.

As we dealt with how to implement rotation in the ARPen system, it was necessary to first consider the common mathematical methods with which rotations and orientations are managed in CG.

5.1 Mathematical methods

We compared three mathematical methods for rotation.

When deciding on a suitable mathematical method for the applications in our techniques, we first discussed the advantages and disadvantages of three popular options: Euler angles, rotation matrices and quaternions.

Euler angles are from now on used to refer to the defini-

---

[1] `https://github.com/i10/ARPen` *(Accessed: 25.08.2019)*

tion of Tait-Bryan angles. This definition describes a rotation around all three axes (e.g. z-y-x) instead of the proper definition of Euler angles describing a rotation around only two axes (e.g. z-y-z). This is the way Scenekit defines them and is therefore applicable in this project.[2]

Euler Angles are very intuitive and easy to handle. Comparable to three-dimensional vectors used for translation in 3D space, with Euler angles the developer must only define the angles by which a rotation should be applied around each of the three axes.

When working with Euler angles the programmer has to mind that they are always applied in a defined order. In the case of Apple's SceneKit that order is roll (rotation around the z-axis), yaw (rotation around the y-axis) and then pitch (rotation around the x-axis)[3]. A different order would yield a different resulting orientation for the rotated vector. Due to that limitation, rotation operations may become more complex or involve several different Euler angle rotations.

Euler angles are intuitive to work with, but are not singular and the issue of gimbal lock can arise.

One problem often brought up when talking about Euler angles is gimbal lock. If one axis is rotated by 90 degrees, it is rotated onto one of the other two axes and cannot be separated from it anymore. Therefore, a loss of a degree of freedom results which prohibits certain rotations. Furthermore, Euler angles are not singular and different rotations can lead to the same result. [4]

A rotation matrix uses a combination of sine and cosine functions to rotate a vector by a given angle within a defined coordinate system. Euler's rotation theorem states that any rotation can be expressed through only three components (or as seen with Euler angles through rotation around three axes). Therefore, an orthonormal 3x3-matrix is sufficient for describing any rotation.[5]

---

[2] `https://developer.apple.com/documentation/scenekit/scnnode/1407980-eulerangles` *(Accessed: 25.08.2019)*

[3] refer to footnote 2

[4] `http://www.informatikseite.de/animation/node16.php` *(Accessed: 25.08.2019)*

[5] `http://mathworld.wolfram.com/RotationMatrix.html` *(Accessed: 25.08.2019)*

While the math involved with applying rotation matrices is as simple as a matrix-vector-product, composing them is unintuitive[6]. Also, in the process of rotating objects, round-off errors are prone to arise during the frequent matrix-vector-multiplications (Parent [2001]) and the necessary orthonormalization of the matrix is computationally expensive (Koch [2016]). For the GPU (*graphics processing unit*) all rotations must be expressed through matrices[7]. The ARPen system though was developed on a higher level with the functions provided by SceneKit which makes this less of a priority.

Rotation matrices make for easy calculations, but are prone to round-off errors and are not intuitive to construct.

Quaternions are represented though a four-dimensional vector. They are hyper-complex as three of the vector entries are imaginary numbers. Quaternions extend the general definition of complex numbers into three dimensions and any quaternion with an absolute value of 1 describes a rotation.[8]

SceneKit provides an intuitive way to implement quaternions by letting them be defined in an axis-angle-representation. This as represented by its name is a definition through a three-dimensional vector as the rotation axis and a floating point parameter as the angle defining the amount by which should be rotated.

Quaternions take less computational capacity during calculations then rotation matrices and avoid the error of gimbal lock often met during the usage of Euler Angles (Groÿekatthöfer and Yoon [2012]). In addition, the rounding effects occurring when working with quaternions are less significant than those arising with rotation matrices.

With quaternions we can create smooth rotations which face less significant round-off errors. Their complexity in implementation is taken away through functions provided by SceneKit.

Invented by William Rowan Hamilton in 1843 (Koch [2016]), Quaternions are now a common way for CG programmers to apply rotations on vectors in 3D space. Due to the interpolation techniques available for quaternions such as SLERP (*spherical linear interpolation*) smooth rotation ef-

---

[6] `https://imada.sdu.dk/~rolf/Edu/DM567/E18/rotationRepresentations_v3.pdf` *(Accessed: 25.08.2019)*

[7] refer to footnote 6

[8] `http://www.iti.fh-flensburg.de/lang/algorithmen/grundlagen/quat.htm` *(Accessed: 25.08.2019)*

fects at constant velocity can be achieved. Therefore, using quaternions in CG is visually pleasing. (**?**)

Within Scenekit a scene is organized through nodes which can represent virtual objects. Nodes hold different properties defining their orientation in space using all three mathematical methods presented. At any time each property could be called and manipulated, having an effect on all other properties.[9] Therefore, switching between different mathematical methods during the duration of the program was possible.

The mathematical methods which is at the same time intuitive to work with, but also able to tackle complex rotations while presenting them smoothly to the user is the method of quaternions, and was therefore chosen for most of our application.

5.1.1 Quaternions

This section provides further information on quaternions.

In the following section we wanted to further present quaternions as our majorly used mathematical method to express orientations and rotations within the ARPen system. We are first going to give a brief introduction to complex numbers and how they are extended into three-dimensional space, and then describe the most important quaternion operation needed for the rotations tasks at hand.

Complex numbers are a combination of a real and an imaginary part and can be visualized through a two dimensional graph with one axis representing each. Quaternions extend the definition of the complex numbers into three-dimensional space and can be imagined through a hypersphere (or more accurately the 3-sphere whose surface is essentially the space of all unit quaternions) (Connellan [2014]). Through the inclusion of imaginary numbers, quaternions tend to be less intuitive than their Euler angle counter parts. This is less significant within this the-

---

[9] `https://developer.apple.com/documentation/scenekit/scnnode/1408048-orientation` *(Accessed: 25.08.2019)*

sis as intuitive functions for implementing quaternions are already part of the framework.

The most important calculation for our purposes is finding the difference between two quaternions which determines how to get from one orientiation to the other by multiplication. The order of multiplications is important as quaternions are not commutative (Moti Ben-Ari [2014]).

A rotation q can be applied to an orientation p though

$$p' = q * p * q^{-1}$$

where p' is the orientation after the rotation and q^{-1} is the inverse. This and further information on quaternions and their math can be found online[10].

We utilized quaternions to record and manipulate orientations in the ARPen system as we describe in the following sections.

5.2 Detecting the ARPen Orientation

We define a fixed local coordinate system for the `penTip` node which rotates with the orientation of the ARPen.

In order to use the ARPen itself for the purpose of rotating virtual objects in the AR scene, the `penTip` node of the ARPen must have a clearly defined orientation and local coordinate system (LCS). For the already developed drawing or translation methods only the position of the `penTip` node was calculated from the currently visible markers. Each of these markers itself also has a defined LCS that gets detected through the arUco framework. As the `penTip` node was derived from the markers, the `penTip` node orientation also originates from them. Due to the multitude of markers, that LCS of the `penTip` node flickers around to fit the most prominent detected marker.

To solve this problem we wanted to rotate the LCS of the markers to the same defined orientation. As a result, the

---

[10] `http://mathworld.wolfram.com/Quaternion.html` *(Accessed: 25.08.2019)*

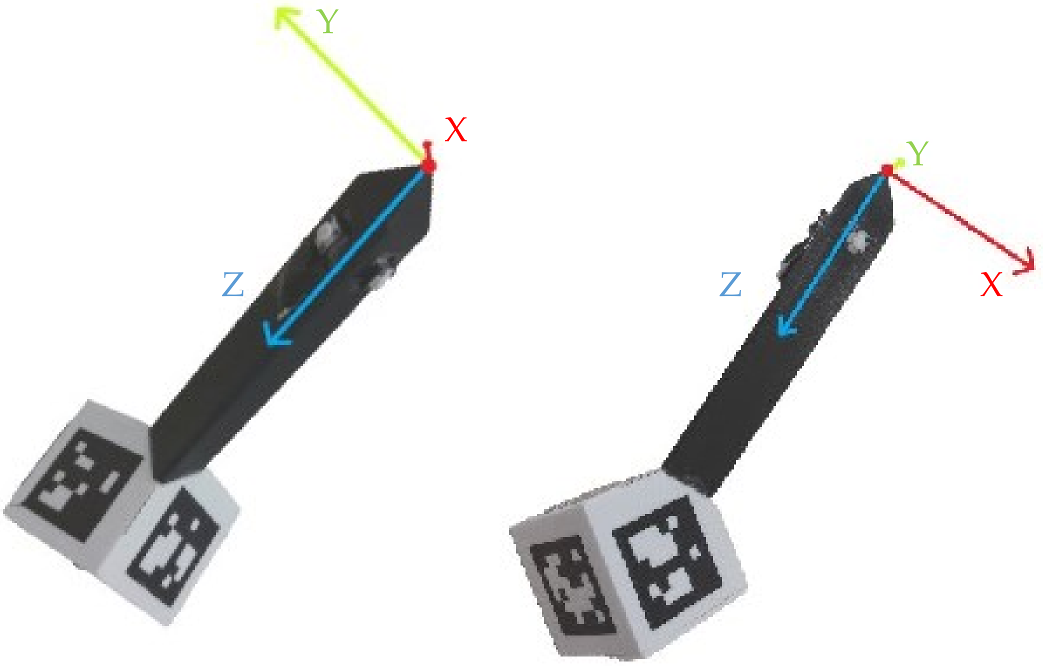


Figure 5.1: Qualitative visualization of the fixed local coordinate system of the *penTip* node. The z-Axis points through the shaft of the pen. The other axes are derived from the marker with the id=2.

LCS of the `penTip` node can rotate in space in accordance to the pen motion, but will keep steady in relation to the ARPen. We chose a LCS whose z-Axis is pointing along the shaft of the pen towards the middle of the cube as seen in Figure 5.1.

For this purpose, we took one marker's LCS as a reference. The first step, therefore, was rotating each marker's LCS to fit that of the marker with the label "top". Afterwards, the same rotation could be applied to all markers' LCSs by rotating those to the earlier defined orientation. As a result, the `penTip` node always has the same fixed orientation in relation to the ARPen independent of the markers detected.

5.3 Implementing the Rotation Techniques

Plug-ins were used to implement the different techniques.

For each of the five rotation techniques chosen to be evaluated in the user study, we implemented a plugin containing first and foremost the rotation method, but also the selection of objects as well as the user study task. In this segment we want to present how the different rotation methods are realized for the ARPen system. For each technique, rotating the virtual object is only possible while it is selected.

During the calculations in the plugins, we majorly work with quaternions and vectors of the type `simd`. This provides us with a greater range of mathematical operations for the complex rotations.[11]

Here begins the detailed description of the rotation techniques.

Direct pen rotation. For this technique, we record the rotation of the ARPen between updates received through `didUpdateFrame()` and apply that same rotation onto the selected object. Quaternions are used to calculate the change of orientation of the ARPen between frames.

At the first selection of an object, `startPenOrientation` saves the orientation of the `penTip` node. During each update of the frame, an `updatedPenOrientation` is recorded as well. At the end of each update the `startPenOrientation` is set to the current `updatedPenOrientation`, so it can be compared with the `updatedPenOrientation` in the next recorded frame.

At this point we want to compare the two orientations and find a quaternion which transforms the orientation of the `penTip` node at the start into the updated orientation. The calculated quaternion can then be applied to the object orientation. This requires a quaternion multiplication where the order is significant. As we want a rotation from the `startPenOrientation` towards the `updatedPenOrientation`, the `updatedPenOrientation` is multiplied by the inverse of

---

[11] `https://developer.apple.com/documentation/accelerate/simd` *(Accessed: 25.08.2019)*

the `startPenOrientation`:

```
startPenOrientation =
     updatedPenOrientation
     * startPenOrientation^-1
```

If this quaternion is applied to the orientation of the object, the same rotation between a different start and end points is achieved. For this the `simdLocalRotate(by: simd_quatf)` function is used.

For the proper rotation, the rotation axis of the calculated quaternion should be transformed from world coordinate space to the object's LCS. Additionally, we normalize the calculated quaternion to prevent an unwanted scaling of the rotated object.

Pen rotation with pedal effect. For this method we also use the recorded change in orientation of the ARPen, but apply it to the object indirectly.

While the ARPen is rotated away from its initial orientation measured at the selection of an object, the object rotates in a continuous motion until the direction of the pen rotation is changed. The rotation axis is translated from the ARPen to the object, so the ARPen's change in orientation can be mirrored. The speed of rotation is defined through the angle at which the ARpen is rotated away from its initial orientation. We defined a number of intervals for the rotation angle of the ARPen in which the orientation of the object is changed by a defined amount of degrees in every updated frame. The first interval of up to three degrees does not cause any rotation for the object to prevent unwanted movement immediately after selection. A bigger angle leads to faster rotations.

In the case of *pen rotation with pedal effect* the quaternion from the object orientation to the `penTip` node orientation is measured as a starting value in `startBoxToPenOrientation` right after selection. This value is only updated if the button for selecting an object is newly pressed as a constant value is essential for calculating the rotation axis and angle for the pedal effect. The `updatedBoxToPenOrientation` is used for calculating the difference as it was done in the *direct pen rotation* technique. The quaternion describing the difference provides the axis and determines the angle for the rotation with pedal effect.

Direct device rotation. For this method, the device motion is recorded between each updated frame and applied directly onto the selected object. Device orientation is derived from the SCNView instance of our scene. The orientation of this `SCNView.pointOfView` is taken as the orientation of the `penTip` node was in the case of *direct pen rotation* and is applied in the same way.

Device rotation with pedal effect. Equivalent to the pen rotation with pedal effect, the device motion is applied to the object indirectly.

Touchscreen rotation. In this case rotation is only possible around the x- and y-Axis as they are defined by the camera's LCS. A swipe across the screen in horizontal direction for the x-Axis and in vertical direction for the y-Axis is translated to an angle by which the object is rotated around the respective axis. The dominant direction gives the direction of rotation for the object. As a result the direction an object can be rotated in is to an extent restricted for touchscreen rotation unlike with the other methods. In order to rotate around any other axis, the camera view has to be changed and therefore the device to be moved.

A more detailed explanation of the selection methods used.

Selection is done via *penRay* for all rotation techniques except *touchscreen rotation*. For *penRay* selection a `hitTest` is performed on the press of a button on the ARPen to evaluate which objects lie in the same line of sight of the view as the `penTip` node. For the task we asked the participants to perform in the user study, we filtered out any possible result from the `hitTest` except the object that is supposed to be selected. This was done to prevent errors in the selection process from affecting the data we wanted to collect on the rotation methods. As selection has already been evaluated for the ARPen, it simply provided a defined mode in which rotation is enabled within in our study. For *touchscreen rotation* the same `hitTest` is not performed through the *penRay* method, but by tapping the object on the screen.

The object is deselected by either letting go of the button in the case of *penRay* selection or tapping the object once more in the *touchscreen rotation* method. On selection and deselection the object changes color to provide a visual cue for the user.

All virtual objects visible in the scene are created and positioned at the activation of the plugin during the `activatePlugin()` function. Objects do not change scale or position during the duration of the plugin, and are only there to be selected and rotated.

Chapter 6

Evaluation

For the evaluation of the implemented rotation techniques, we conducted a user study. Each participant was asked to individually test out the five methods and offer their opinion through comments during and a questionnaire after each task.

This chapter features a description of the task and group composition, as well as the quantitative and qualitative results collected during the study.

6.1 Task and Study Design

We evaluated all rotation methods through the same task. The participants were asked to rotate a virtual object into a translucent model of it. For this, they only needed to select and rotate the object while no other interaction with the object was possible. If the participants decided that they were happy with the result, they pressed a check mark on the left of the screen in order to move on to the next randomly oriented model. After each iteration the object returned to its original orientation.

Following the introduction of one rotation method, the participants were able to test the technique until they felt com-

	Technique 1	**Technique 2**	**Technique 3**	**Technique 4**	**Technique 5**
User 1	Touchscreen	Direct Pen	Pen "Pedal"	Direct Device	Device "Pedal"
User 2	Device "Pedal"	Touchscreen	Direct Pen	Pen "Pedal"	Direct Device
User 3	Direct Device	Device "Pedal"	Touchscreen	Direct Pen	Pen "Pedal"
User 4	Pen "Pedal"	Direct Device	Device "Pedal"	Touchscreen	Direct Pen
User 5	Direct Pen	Pen "Pedal"	Direct Device	Device "Pedal"	Touchscreen

Figure 6.1: The Latin square was used to determine the order of rotation techniques within the user study. The order rotates with each participant. As we have evaluated five techniques, the Latin square repeated itself four times throughout the twenty participants.

Participants went through two phases with each technique: a self regulated testing period and a recorded phase of six iterations.

fortable enough to move on to the recorded phase. In the recorded phase, which was used for the quantitative evaluation, the participants had to perform the described rotation task six times. After they completed all iterations for one technique, we asked them to answer a series of question of the NASA TLX model type. This process was repeated for each rotation method. We determined the order of the methods through a Latin square as seen in Figure 6.1.

The original object used was a cuboid whose length was significantly bigger than its height or width. For orientation the sides were marked by different colors. During the first of two pilot studies we conducted to verify our study procedure, the subject struggled to regain information about the orientation of the object and model quickly enough as for this they had to look at the targets from different viewpoints. Furthermore, the colors were hard to grasp on the translucent model. For a more intuitive task design we chose the model by John Marstall[1] as pictured in Figure 6.2 which had several reference point for an easier understanding of the required rotation.

We used reference images for ARKit to root the scene in.

During implementation and the first pilot study, we also noticed how occasionally the virtual object tended to fly away from its fixed position. In order to reduce that issue, pictures were glued across the designated space to provide ARKit with reference points to root the scene in.

---

[1] `https://www.dropbox.com/s/u0tifsdbzwrt0e1/ARKit.zip?dl=0` *(Accessed: 28.08.2019)*

Figure 6.2: The scene shown to participants in the user study. The virtual object, which is to be rotated, is fully colored while the model, which provides the target orientation, is translucent.

We required participants to sit in front of a table facing a wall. Furthermore, we asked everybody, including left-handed participants, to hold the iPhone in their left and the ARPen in their right hand in the pinkie tray grasp. We also encouraged them to think aloud and allowed them to stand up in order to change their point of view if they desired to do so.

All tasks were performed on an iPhone 6s.

6.2 Group Composition

Overall, 20 individuals participated in the user study with ages raging from 17 to 47 years (M = 24.25, SD = 6.069). Of the six female and fourteen male participants only two identified as predominantly left-handed.

Six participants reported to have no experience in 3D mod-

14 participants already had experience in 3D modeling, but not necessarily in AR.

eling while the same amount of participants also accounted for no experience in AR/VR. Twelve participants had little experience with 3D modeling and two a lot, mostly in tools for CAD applications as well as the open-source software blender[2]. Only four participants communicated to possess a lot of experience with AR/VR. Two participants have worked with the ARPen system before in any way.

The sessions took about one hour with times varying between 40 and nearly 120 minutes. Longer times were usually caused by participants using more time in their training periods or in two cases a need to wait for the phone's battery to charge back up in between techniques.

6.3 Results

The results of the user study are presented in two parts. In the first section, we are taking a look at the quantitative data collected during the study. Afterwards, we present the qualitative personal assessments of the participants through the comments made during the tasks, the questionnaires about the individual methods and the final ranking of all five techniques.

6.3.1 Quantitative Results

Any rotation can be defined through a rotation axis and an angle. Therefore, we measured the angle in degrees between object and model at the beginning and end of the task. In the same style we determined how much the object was rotated overall during one iteration of the task as well as how much the pen/device was rotated while the object was selected.

We considered the angles between object and model recorded at the end and filtered out huge spikes towards higher degrees as we determined these to be a result of

---

[2] `https://www.blender.org/` *(Accessed: 28.08.2019)*

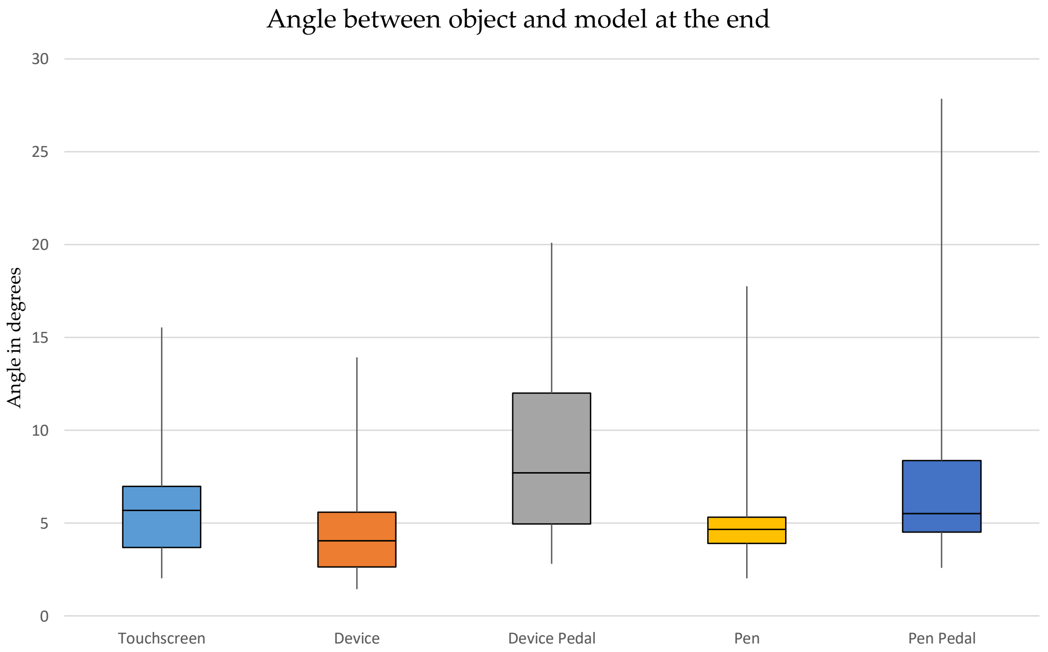


Figure 6.3: A box plot showing the average remaining differences between the object and the model orientation in degrees at the end of an iteration for each technique. Most techniques performed in a similar range while *device rotation with pedal effect* performed noticeably worse. *Direct device rotation* was on average the most precise technique.

visual mistakes by the participants rather than a struggle while working with the rotation methods.

The remaining results depicted in Figure 6.3 show that the differences between the averages of techniques are overall small with *device rotation with pedal effect* being the noticeably less precise method with an average angle of 8.5° (SD=7.55°) at the end of an iteration. *Pen rotation with pedal effect* performed on average slightly better (M=7.16°, SD=7.53°). *Direct device rotation* delivered the most precise performance with an average of 4.79° (SD=5.2°) remaining.

The differences in precision are not significant except for techniques involving a pedal effect which had some of the worst average results.

The time spent on individual tasks was measured between the first selection and the last deselection of the virtual object during one iteration of the task. Despite there being participants who generally took more time for each technique than others and the fact that we observed par-

Time spent on a task does not significantly influence precision.

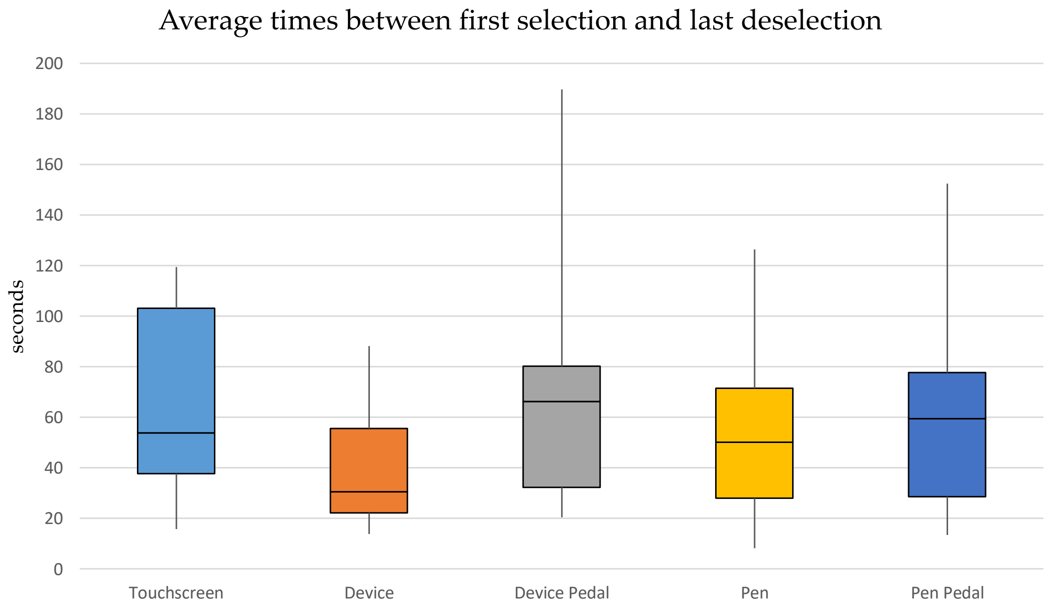


Figure 6.4: Box plot showing the speed of performance throughout the different techniques averaged over the six iterations of each participant. Times are measured in seconds between first selection and last deselection of the object. In general, the times measured are similar with *direct device motion* performing slightly better and *touchscreen rotation* performing slightly worse.

ticipants declaring themselves to be “perfectionists”, we could not find a significant Pearson correlation between time spent on the tasks and precision in the results.

While the times measured were very diverse and we did not take the times for direct pen rotation by participant 19 into consideration as they were extraordinarily large and distorted the results, most of the recorded times for the techniques fell into the same range. Only tasks done with *direct device rotation* were generally performed quicker. Additionally, there were a greater range of participants taking longer with *touchscreen rotation* than with the other techniques as can be seen in Figure 6.4.

Touchscreen rotation had on average the most amount of unnecessary rotation.

We also compared how much an object was rotated in relation to an ideal execution of the task for which the shortest angle between the object and the model at the start of an iteration was considered. For the results presented in Figure 6.5, we averaged the ratios recorded of all the partici-

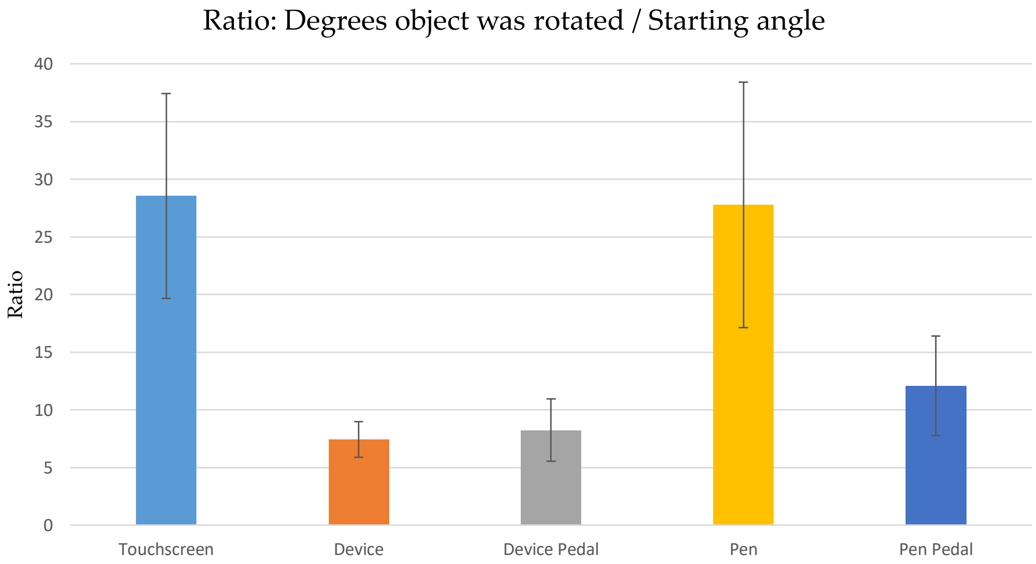


Figure 6.5: Bar chart with confidence interval: Averaged ratio between the amount the object was rotated in degrees and an ideal rotation described through the angle at the beginning of the task. *Touchscreen* and *direct pen rotation* had the highest average ratios. *Direct pen rotation* was affected by a tracking error causing fast changes in orientation which has most likely influenced the data.

pants per technique. For more than half of the twenty participants *touchscreen rotation* had the highest ratio equaling the highest amount of unnecessary rotation with an overall average of 28.56 times (SD=20.79) the amount of rotation. Some participants mentioned applying a technique of quickly rotating the object several times on the touchscreen to end up closer to the model orientation. Also, participants were not restricted by wrist rotation in this case.

Surprising at first is the fact that the average ratio while working with *direct pen rotation* is also very high with a factor of 27.79 times (SD=24.93) the rotation. During rotation with the ARPen, errors in tracking caused a “jittering” effect on the object which was recorded as a change in orientation, and, therefore, had a factor on this ratio. The “jittering” was more or less severe for the individual participants, but did not occur for *pen rotation with pedal effect*. Both techniques involving a pedal effect had smaller ratios (*device rotation with pedal effect*: M=8.25, SD=6.35; *pen*

Direct device rotation had on average the lowest amount of unnecessary rotation.

rotation with pedal effect: M=12.09, SD=10.12) than expected. Participants had the chance to rotate an object around itself several times without much effort (as was the case for *touchscreen rotation*), but elected to rotate it on average very little. The lowest overall average ratio was recorded for *direct device rotation* with 7.42 times (SD=3.62) the actual rotation of the object compared to the ideal rotation.

Participants seemed to utilize the pedal effect more for device rotation than for pen rotation.

For the techniques involving a pedal effect we also calculated the average ratio between the amount of degrees the virtual object was rotated and the rotation the participants performed with the ARPen or device. Any iterations where the object wasn't rotated as the participants were already satisfied with the result were filtered out. As a conclusion, we found that on average participants seemed to utilize the pedal effect when in conjunction with device rotation as they rotated the device 0.47 times (SD=0.22) as much as the object. For *pen rotation with pedal effect* participants recorded an average ratio of 0.98 times the amount with a large deviation (SD=0.83) between results. Of notice is that the change of orientation of the ARPen and device were only recorded while the object was selected. Any rotation between selections to, for example, bring the ARPen or device into a new starting position did not accumulate onto the presented numbers. If that was the case, they would be even larger.

Figure 6.6 shows the average number of selections for each technique. The number of selections performed during the *touchscreen rotation* method was the lowest for all the participants as most selected the object only once or twice. The selection method for this technique did not require the participant to continuously hold down a button. Also, as participants could only manipulate the object's orientation through swipes across the touchscreen, a change of viewpoint or arm position without the purpose of rotating the object did not require a deselection.

In tasks involving the pedal effect, participants used deselection for stopping the rotation, leading to a large average number of selections.

During the techniques involving a pedal effect most participants used deselection to stop a rotation instead of rotating the ARPen or device back to it's starting orientation as was intended. This led to the number of selections for the pedal techniques being the two highest average numbers.

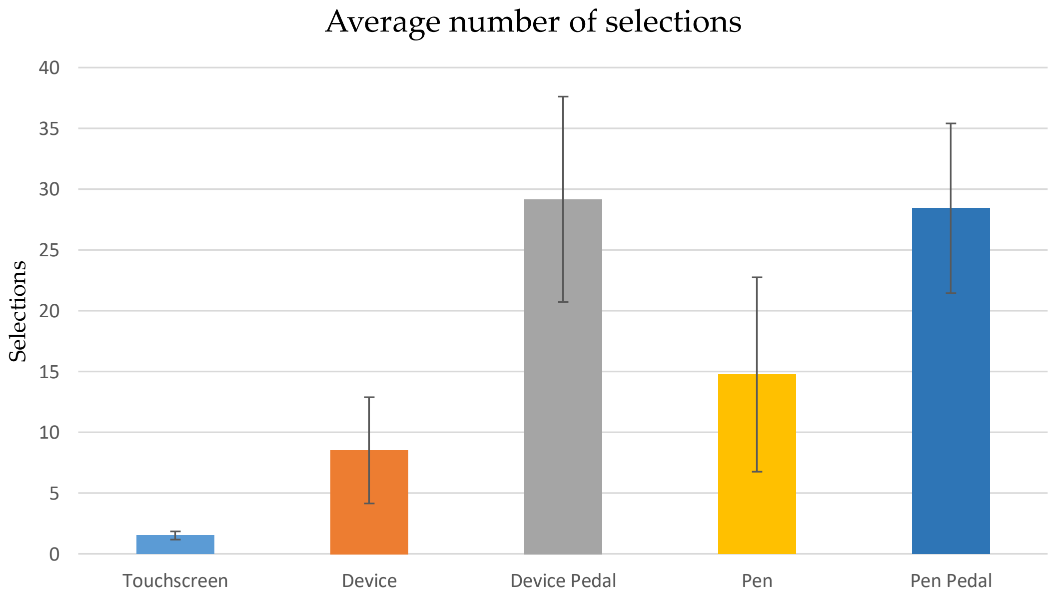


Figure 6.6: Bar chart with confidence interval showing the average number of selections for the techniques. The methods involving a pedal effect had on average the highest amount of selections while *touchscreen rotation* always had the lowest amount.

Direct device rotation performed best in most categories.

Overall, considering the different factors measured, *direct device motion* performed best with on average the least amount of unnecessary rotation, the fastest times and the highest precision. Only in the number of selections category, it was triumphed by touchscreen rotation.

While especially precision is an important factor in 3D modeling and personal fabrication, an application developed for user interaction has to withstand the qualitative evaluation as well.

6.3.2 Qualitative Results

We used a Nasa-TLX questionnaire for a personal assessment from the partcipants.

The questionnaire on a participant's personal evaluation of the techniques followed a Nasa-TLX format and included questions on the mental, physical and temporal demands of the task as well as the participant's own judgment of their performance, effort they had to invest, frustration and precision. The format offered twenty stages valued in equivalent intervals from five to hundred which people could decide between. The higher the rating they gave in this questionnaire, the more positive they evaluated a technique in the respective category.

We asked the participants to rotate an object onto a translucent model until they were satisfied with the results. Therefore, most of them when requested to think aloud spoke about how precisely they were able to work with a certain technique.

Most participants disliked the pedal effect.

Both techniques involving a pedal effect received similar comments. Most participants enjoyed the methods for the initial big rotation towards the desired orientation, but disliked them for precise motions. One big problem participants faced was overshooting their goal and oscillating around the model orientation through several tries until they ended up with a satisfying result. With the same effect, they generally felt like tries to make small adjustments ended up in big unwanted rotations. This led to a number of participants, especially in later iterations of their task, which did not even try for perfect results as they did not think that more time would lead to much more precision. In general, participants expressed their frustration with the techniques and felt demotivated by them.

Four participants wished for a way to control the velocity of rotation. Others also disliked the discrete intervals and found the jumps between velocities to be annoying. Furthermore, the gap at the beginning of the velocity intervals, which was left to have no effect on the object, earned disapproval when participants tried to make precise rotations.

For smaller rotations, participants frequently selected and deselected an object in order to gain control over the rotation. Through this they imitated the effects a direct rotation technique would provide. Especially for the *pen rotation with pedal effect*, a larger amount of pen motion was evident. As already mentioned above, on average participants performed about 98% of the amount of rotation with the pen as they did on the virtual object. For *device rotation with pedal effect* this factor was only about 47%. Though, as there were motions that did not immediately translate to object rotation and a change in direction usually caused the object's rotation to slow down or stop for a second, we can deduct that participants still utilized the pedal effect to some extent.

With *direct device rotation* participants felt in control.

When the device motion was employed without the pedal effect, participants generally felt in control of the rotation and like they could achieve precise results. Four participants expressed that working with the technique was easy. Despite that, one commented to have a harder time keeping a grip on the phone and one mentioned problems getting accurate rotations with their non-dominant hand. Even though big rotations moved the object out of the view, which was solved by a "clutch and grab" approach, most participants picked up on the technique quickly and expressed joy at working with it.

Tracking errors made *direct pen rotation* harder to work with.

Direct pen rotation had the disadvantage that due to noisy tracking data, the described "jittering" of the object occurred for most participants. This led to a lower degree of accuracy and the feeling that luck was responsible for deselecting the object in the right moment for a precise result. Many participants were frustrated by this and two explicitly stated that they would have rated the method better if it was not for this effect.

Due to limited wrist rotation, big rotations required a "clutch and grab" approach as with *direct device motion*. After holding the phone further away from the ARPen, some participants could resolve their problem of keeping the marker cube box within the view during the rotation task. As they were concentrating more on the object than the ARPen, this was a recurring issue. Because the marker cube is essential for tracking, this prolonged the task time when it happened.

Figure 6.7 depicts how participants rated the precision for each method from low (5) to high (100). Most participants agree on *direct device motion* being very precise while both *direct pen* and *touchscreen rotation* got a similar, lower range of scores with values of 15 to 85 with an average around 60. For both these techniques, there were participants who

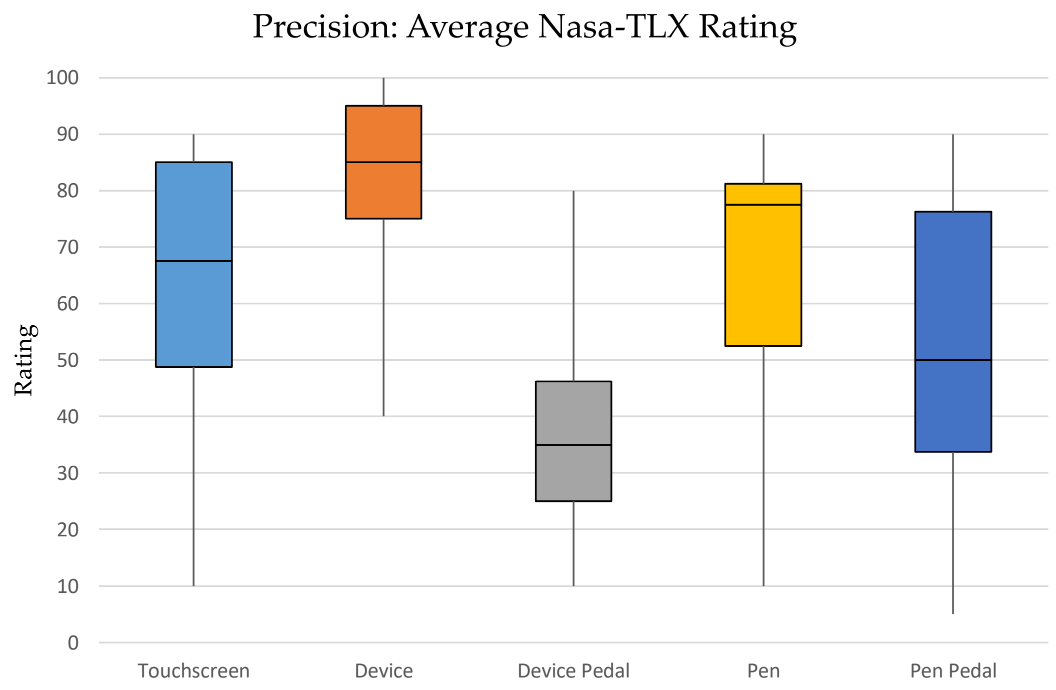


Figure 6.7: Box plot: Results from the Nasa-TLX Questionnaire on Precision. Participants judged on a 20-point scale from 5 to 100. There are significant difference between the overall best rated *direct device rotation* and the worst rated *device rotation with pedal effect*. The results for the other three techniques span the space between those two opposites.

struggled immensely with applying the methods while others felt them to be easy and intuitive.

Despite the very similar comments about struggling to get accurate results from the *pen rotation with pedal effect* and *device rotation with pedal effect*, the former got a significantly larger portion of positive results in this personal assessment of precision.

A factor influencing the amount of rotation participants performed was how at least ten of them struggled to imagine the correct rotation axis and how to transfer their ideas to their physical motions to achieve the desired results. This was to some extent evident for all techniques. Therefore, the wish for some kind of visualization arose. For the *touchscreen rotation* especially, participants wanted to see the rotation axes of their current orientation. In the case of the other techniques, participants would have enjoyed seeing the pivot point. Quite a few first tries before clarification also featured participants selecting an object at the point they wished to be the pivot point. For rotation techniques involving the ARPen, some participants solved this problem by holding the pen within the object or at least in the same orientation as it, so there was no need to translate the desired rotation.

Participants wished for a visualization of the rotation axis or pivot point.

In the case of *touchscreen rotation*, three participants explicitly wished for a way to rotate around the third axis pointing through the camera. For them, using a two-finger touching approach to rotate around it came intuitively and they were frustrated with not being able to employ it in the application.

Translational motions where intuitively used for rotation.

During rotation techniques involving the ARPen at least six participants initially used translational motions where they expected to pull or push a part of the object to the desired orientation. Others also intuitively tried to swipe on the touchscreen starting at a point on the object which they wanted to pull into another position.

Often participants were met with rotations, which fit the model in their current point of view better, but made the errors in three dimensions bigger. Also, due to misjudging the rotation axis, they sometimes ended up with unwanted changes in orientation. As a result, four participants desired a way to fixate the object orientation around one axis, so rotation was only possible in one plane at a time.

Rotation techniques involving the ARPen were the least intuitive, but participants displayed a steep learning curve.

In general, dependent on their experience, participants had different opinions on the intuitiveness of techniques. As two participants stated, rotation methods involving device motion were familiar through mobile games in which virtual objects were manipulated using the same type of motions, while touchscreen interaction was understood quickly by all participants even though some struggled with finding the viewpoint for the desired rotation axes. Only three participants thought pen rotation to be immediately intuitive with one participant expressing that it was just like picking up an object and looking at it in real life. Most participants, though, gained confidence in the unfa-

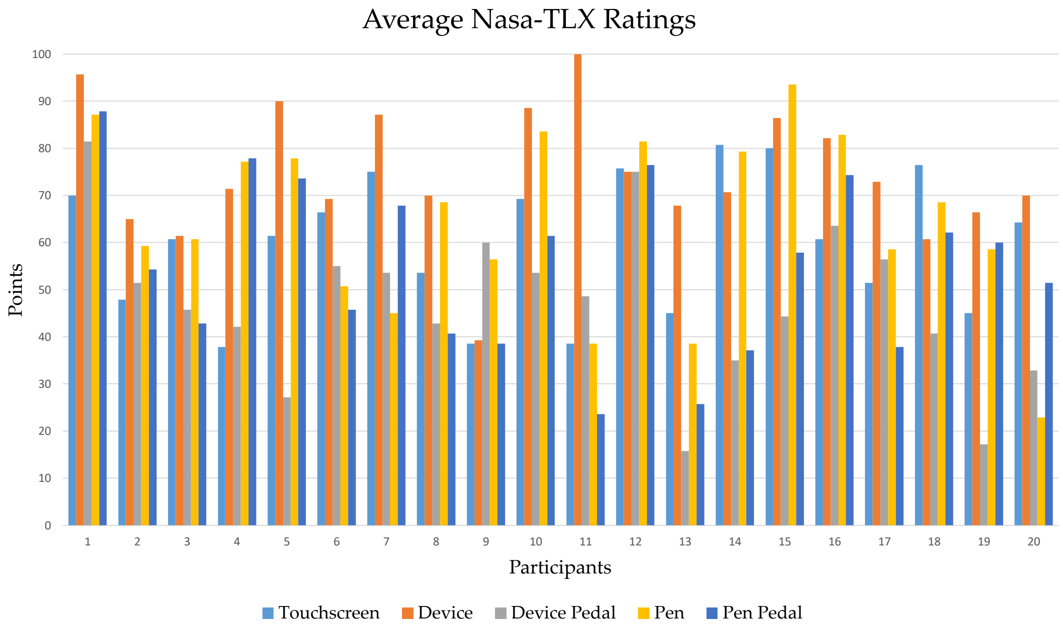


Figure 6.8: Bar chart showing the Nasa-TLX ratings of the individual participants averaging over the ratings for the categories: Mental Demand, Physical Demand, Temporal Demand, Performance, Effort, Frustration and Precision. A low score (min: 5) presents an overall negative score while a high score (max: 100) stands for an overall positive score. There are difference in how participants awarded points with some giving ratings generally lower than others. *Direct device rotation* had the highest average thirteen times while *device rotation with pedal effect* had the lowest average eight times.

miliar pen rotation quickly.

Five participants expressed feeling fatigue and light pain in their arms during methods that required holding both device and ARPen. It came as a coordination challenge to move the device for a change in view and the pen for rotations in the same task.

Participants moved a lot.

In most studies especially touchscreen and device rotations caused the participants to stand up and move around a lot. They did this as it was either necessary for the technique or because they wanted to check their orientation from a different point of view.

Figure 6.8 displays the average overall scores individual

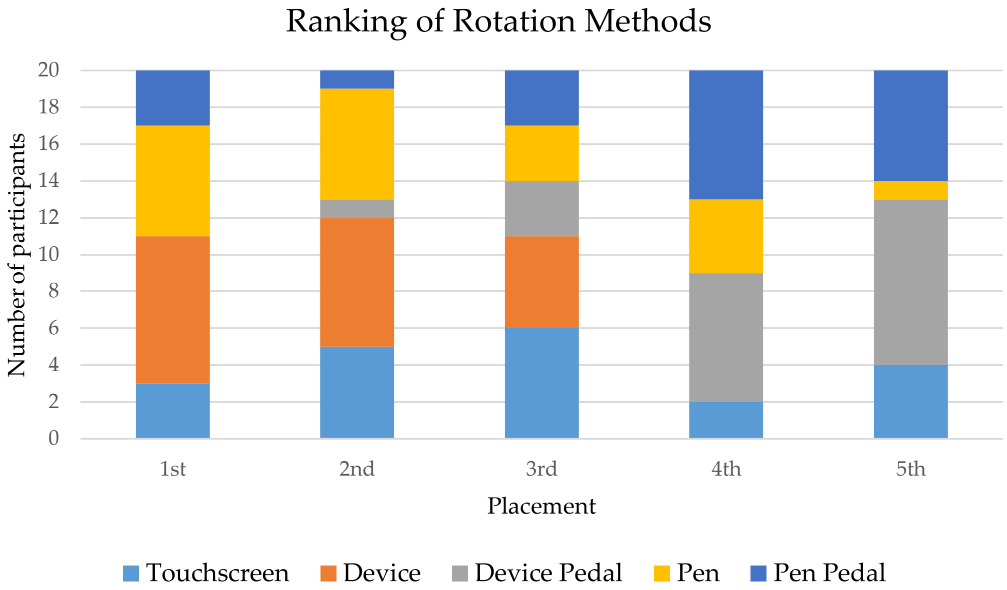


Figure 6.9: Stacked bar chart: Overview of the numbers at which participants ranked all five rotation techniques from 1st to 5th place. *Direct device rotation* never placed lower than third while *device rotation with pedal effect* only placed second once.

participants awarded to the techniques through the seven questions in the questionnaire. We can see that between independent participants the score for the techniques occasionally diverged strongly, but most participants on average did not assign very negative ratings. There were very few scores at the extreme on either end of the scale. *Direct device* and *pen rotation* generally got the higher, more positive scores whereas *device* and *pen rotation with pedal effect* were rated more negative.

The amount of points generally awarded varied between participants, but we can still observe trends in which technique they liked best.

Finally, after all tasks have been completed, participants were asked to rank the methods from their favorite on first place to their least favorite on fifth place. Figure 6.9 shows the results of that ranking. We can see that opinions were very diverse. Only *direct device rotation* never received lower placement than third and *device rotation with pedal effect* only reached as high as second place once. *Touchscreen rotation* was ranked most often in the middle while *pen rotation with pedal effect*, performing better than its device coun-

Direct device rotation was never ranked lower than third place.

terpart, still for the most part took up rankings on the lower places. The opinions on *direct pen rotation* were split with a focus on the middle to upper rankings. This might derive from some participants ignoring the object "jittering" within their evaluation while others took it into account and ranked the technique reportedly lower than they could have.

The ranking overall aligns pretty well with the recorded scores from the questionnaire.

6.4 Discussion of the Results

Throughout the qualitative questionnaire and ranking as well as the quantitative scores, *direct device motion* has most often performed best. A case could be made that *direct pen rotation* would have gained better results if there were no errors in tracking affecting the user experience. Despite that, there were participants who also struggled immensely with pen rotation as a concept. They had trouble rotating the ARPen in a way that would mirror the rotation they desired in the virtual object. Furthermore, trying to keep the marker cube within the frame was a struggle which affected both the time spent on tasks as well as participants' ability to smoothly rotate the object.

We expected the entertainment factor in mid-air rotation.

Overall, though, participants expressed a lot of joy working with the techniques involving mid-air motions (*direct pen and device rotation*) which fulfilled our expectation on this matter. Furthermore, even though only few thought mid-air rotation to be immediately intuitive, the majority recorded a steep learning curve and quickly mastered how to effectively work with at least the direct techniques. Therefore, the initial intuitiveness of *touchscreen rotation* became of little relevance in the discussion around the techniques.

Despite mentioning useful purposes for the pedal effect, participants generally disliked it.

On the other hand, participants disliked techniques involving a pedal effect way more than expected. Whether a slower, continuous rotation, a smaller gap at the beginning of the rotation or a way for participants to adjust the sensitivity would benefit the methods significantly needs to be explored. One participant mentioned that a pedal effect could be useful for larger objects which are harder to rotate through the "clutch and grab" approach. As most participants liked the pedal techniques for the initial rotation, we could even think about creating a combination between the direct methods and those involving the pedal effect.

Worth mentioning is that one participant exclaimed that the selection through *penRay* involved an unneccessary component (the ARPen) during techniques with device rotation. Additionally, due to a bug, the marker cube of the ARPen did not have to be within the point of view for selection after the first one. That happened because the `penTip` node was not erased when the markers temporarily left the scene. This removed some strain on the arm and complexity from the task. It is therefore interesting if techniques involving device rotation are judged worse when paired with a bigger necessity for the ARPen.

Even though *touchscreen rotation* was generally intuitive, some participants struggled with not being able to rotate around all three axes from the same viewpoint.

Participants generally understood quickest how to rotate with *touchscreen rotation* as it was familiar to all. Despite that, there were participants, same as it was the case for methods involving the ARPen, who had huge problems utilizing the touchscreen technique. They struggled to find the correct viewpoint to rotate the object in the desired way. For this problem, an added visualization of the possible rotation axes might improve the technique greatly. Furthermore, three participants intuitively used two fingers to rotate around the third axis while even more participants expressed a wish for a touchscreen rotation involving all three axes. Our assumptions based on the research by Bai et al. [2014] that utilizing both a one-finger touch and two-finger touch approach in the same technique is not intuitive were therefore challenged. This might be a result of how other applications for smartphones realize touchscreen rotations and how this affects what users find to be most intuitive.

Despite some participants' struggle with rotation techniques involving pen motion, we concluded that the added spatial and physical factors of mid-air motions benefited the participants. This was evident through the fact that a significant number of participants had issues identifying the rotation axes and achieving the desired rotations for the *touchscreen rotation* technique.

Direct device rotation was evaluated best.

While there could be made adjustments on some of the methods to achieve improvement, the fact that there were participants who disliked rotations involving the ARPen even without the “jittering” effect, and *direct device rotation* was in contrast never placed below third place in the ranking, we can deduct that *direct device rotation* has potential to stay the most favored technique.

Chapter 7

Conclusion and Outlook

In this final chapter we conclude this thesis by summarizing our work and giving a short outlook on what we believe to be potential next steps for the ARPen system.

7.1 Summary

We evaluated five rotation techniques.

In this Bachelor's thesis we evaluated five different techniques for rotating virtual objects mid-air in an Augmented Reality scene utilizing the ARPen system. We wanted to find the most intuitive and effective method to include as a basic operation in this tool for 3D mid-air modeling.

After introducing the concept of AR and the principles of the ARPen system in chapter 2, we investigated into literature in order to find potential techniques for rotating virtual objects within our application in chapter 3. As a result we decided in chapter 4 on techniques involving the ARPen, the device motion and touchscreen interaction. Furthermore, for the ARPen and device motion we differentiated between a direct transfer of rotation and a "pedal" effect. According to the results of Wacker et al. [2019] we chose *penRay* selection and a *pinkie tray* grasp with the camera on the right in landscape mode as a base for our methods. *Touchscreen rotation* was the only technique involving another selection method, namely selection via touchscreen.

A summary of the results from the user study.

Following the realization of these concepts as described in chapter 5, we conducted a user study involving twenty participants and presented the results in chapter 6. Even though the quantitative results for techniques involving a pedal effect were for the most part similar or only slightly worse than for the other three methods, the frustration and dislike expressed by the participants towards these techniques during the study painted a clearer picture. While the participants also acknowledged their usefulness for big rotations, these methods involving a pedal effect are not ideal on their own. Opinions on both *direct pen rotation* and *touchscreen rotation* were split between participants as some found them to be intuitive while others did not feel like they fully understood the techniques even until the end of their tasks. Participants proposed changes which might improve their effectiveness with the methods. Despite that, *direct device rotation* has proved itself to be overall the most effective, easy to use and precise technique.

Despite potential for improvement in other techniques, *direct device rotation* satisfies our requirements.

Therefore, we propose that *direct device rotation* is integrated as the chosen rotation technique for the ARPen system. Regardless, there should be investigations into whether *direct pen rotation* is a better fit for the application as other operations such as translation and selection are performed with the ARPen. In the end, there are still improvements possible for most of the evaluated methods, but we have found one which fulfills our requirements and was appealing to the participants in the user study.

7.2 Future Work

There are some potential future endeavors we want to suggest for the ARPen system. First and foremost, our studies on rotation techniques could be expanded by applying the suggestions proposed by the participants within our study. Additionally, an improvement in tracking by smoothing the data received for the ARPen orientation could change participants' opinions about *direct pen rotation* and their overall rankings. A combination of direct rotation techniques and those involving a “pedal” effect is also an option that could be explored.

At the same time, research on how to integrate rotation with the other operations already evaluated for the ARPen system is an important and interesting agenda. It could also influence participants’ opinions of a rotation technique if the switch between it and other operations, such as the translation of virtual objects, is more complex than necessary.

Furthermore, in order to create a more versatile 3D modeling application, the basic operation of scaling virtual objects should also be researched in a study similar to the one presented in this thesis.

Overall, while we have found meaningful results to our studies further explorations especially in the context of the whole application and it’s purposes should be conducted.

Appendix A

Questionnaire

On the following pages we displayed the paper questionnaire given to participants to gather anonymous personal information before and a ranking of the rotation techniques after the tasks.

Number:__________

Evaluation of rotation techniques

Gender? ☐ Female ☐ Male ☐ Other

Age? ________________

☐ Right-handed ☐ Left-handed

Experience in 3D modeling? ☐ No experience ☐ Little ☐ Much

Experience in AR/VR? ☐ No experience ☐ Little ☐ Much

Experience with the ARPen? ☐ Yes ☐ No

Fill out at the end:

Ranking of rotation techniques (1 to 5):

- Touchscreen Rotation _______
- Direct Pen Rotation _______
- “Pedal” Pen Rotation _______
- Direct Device Rotation _______
- “Pedal” Device Rotation _______

Do you have further comments about the rotation techniques?

Nummer: __________

Evaluation der Rotationsmethoden

Geschlecht? ☐ Weiblich ☐ Männlich ☐ Anderes

Alter? ________________

☐ Rechtshänder ☐ Linkshänder

Erfahrung im Bereich 3D Modeling? ☐ Keine ☐ Wenig ☐ Viel

Wenn ja, welche Programme? _____________________

Erfahrung in den Bereichen AR/VR? ☐ Keine ☐ Wenig ☐ Viel

Erfahrungen mit dem ARPen? ☐ Ja ☐ Nein

Am Ende auszufüllen:

Ranking der Rotationsmethoden (1 bis 5):

- Touchscreen Rotation _______
- Direkte Pen Rotation _______
- „Pedal“ Pen Rotation _______
- Direkte Device Rotation _______
- „Pedal“ Device Rotation _______

Haben Sie weitere Kommentare zu den Rotationsmethoden?

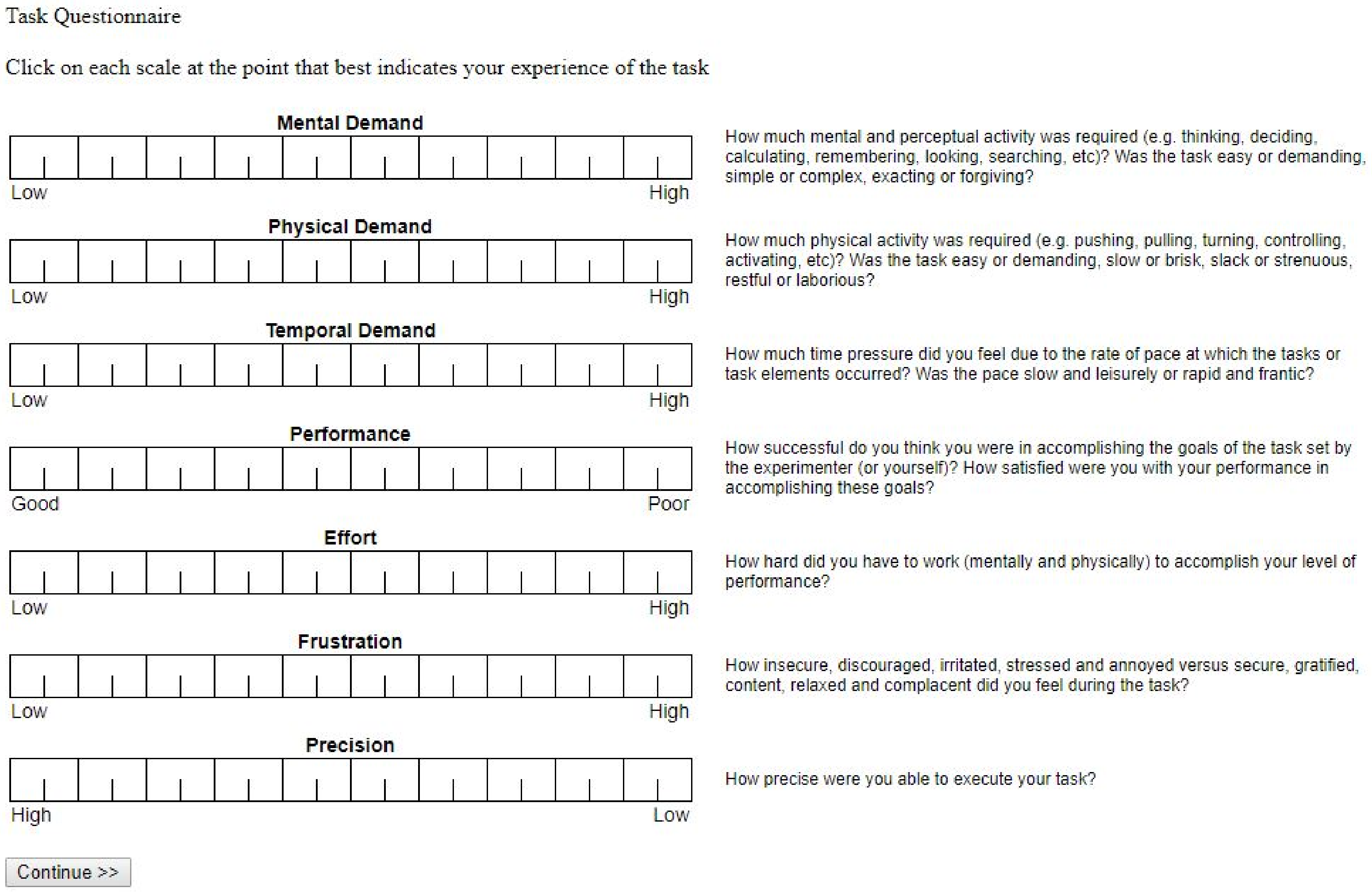
Appendix B

Nasa-TLX Questionnaire

Below are the questionnaires of the Nasa-TLX format which we used for ratings of the individual techniques by the participants. It was filled out on a computer by clicking into one of the twenty boxes on the scale where the more positive factor was on the left and the more negative one on the right. The questionnaires are shown in both english and german. The JavaScript code used is a modification of the template by Keith Vertanen[1].

---

[1] `https://www.keithv.com/software/nasatlx/` *(Accessed: 25.08.2019)*



Task Questionnaire

Click on each scale at the point that best indicates your experience of the task

Mental Demand

How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving?

Physical Demand

How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

Temporal Demand

How much time pressure did you feel due to the rate of pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

Performance

How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

Effort

How hard did you have to work (mentally and physically) to accomplish your level of performance?

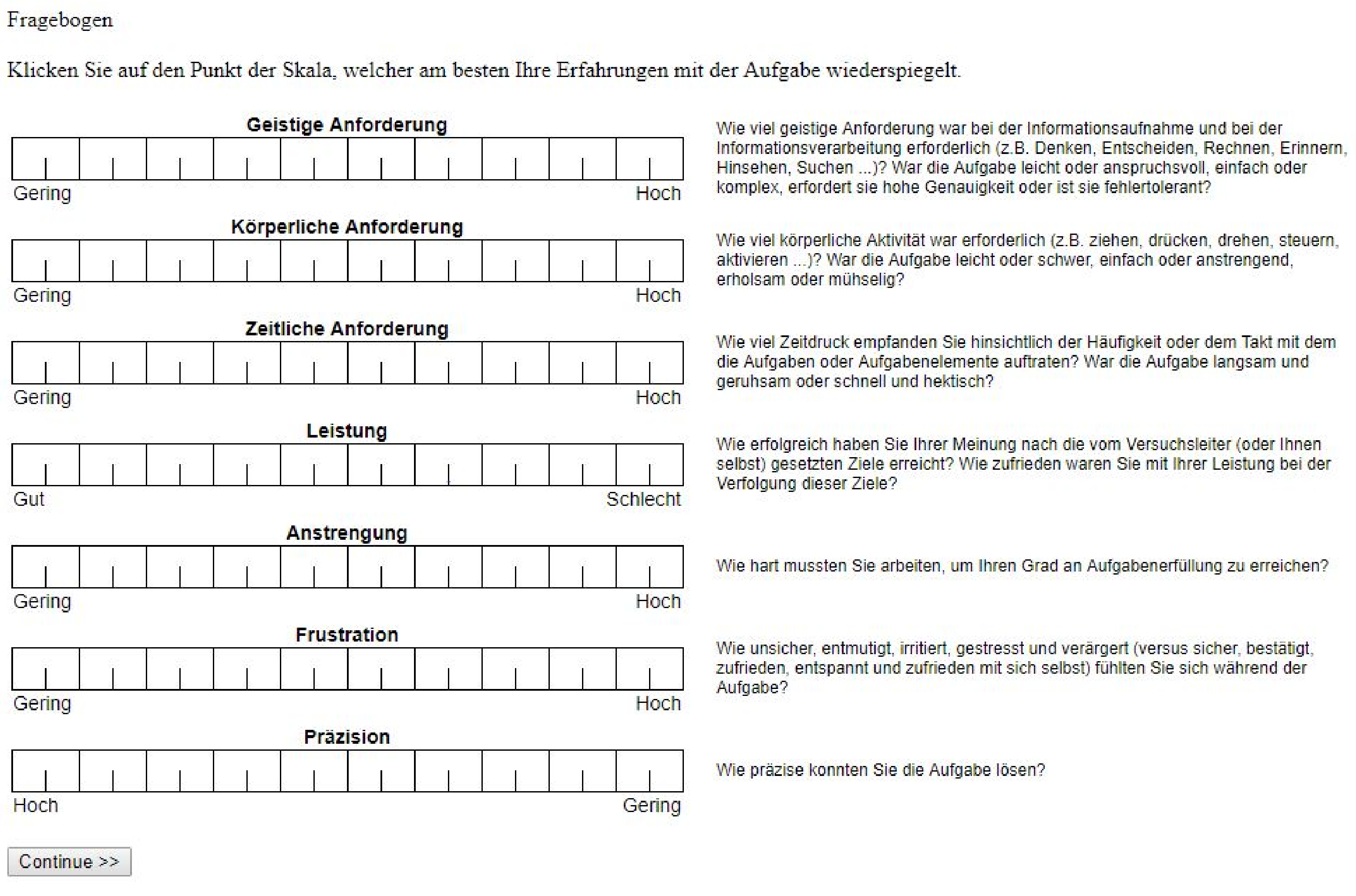
Frustration

How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Precision

How precise were you able to execute your task?

Figure B.1: Screenshot of Nasa-TLX questionnaire in english.



Fragebogen

Klicken Sie auf den Punkt der Skala, welcher am besten Ihre Erfahrungen mit der Aufgabe wiederspiegelt.

Geistige Anforderung

Gering — Hoch

Wie viel geistige Anforderung war bei der Informationsaufnahme und bei der Informationsverarbeitung erforderlich (z.B. Denken, Entscheiden, Rechnen, Erinnern, Hinsehen, Suchen ...)? War die Aufgabe leicht oder anspruchsvoll, einfach oder komplex, erfordert sie hohe Genauigkeit oder ist sie fehlertolerant?

Körperliche Anforderung

Gering — Hoch

Wie viel körperliche Aktivität war erforderlich (z.B. ziehen, drücken, drehen, steuern, aktivieren ...)? War die Aufgabe leicht oder schwer, einfach oder anstrengend, erholsam oder mühselig?

Zeitliche Anforderung

Gering — Hoch

Wie viel Zeitdruck empfanden Sie hinsichtlich der Häufigkeit oder dem Takt mit dem die Aufgaben oder Aufgabenelemente auftraten? War die Aufgabe langsam und geruhsam oder schnell und hektisch?

Leistung

Gut — Schlecht

Wie erfolgreich haben Sie Ihrer Meinung nach die vom Versuchsleiter (oder Ihnen selbst) gesetzten Ziele erreicht? Wie zufrieden waren Sie mit Ihrer Leistung bei der Verfolgung dieser Ziele?

Anstrengung

Gering — Hoch

Wie hart mussten Sie arbeiten, um Ihren Grad an Aufgabenerfüllung zu erreichen?

Frustration

Gering — Hoch

Wie unsicher, entmutigt, irritiert, gestresst und verärgert (versus sicher, bestätigt, zufrieden, entspannt und zufrieden mit sich selbst) fühlten Sie sich während der Aufgabe?

Präzision

Hoch — Gering

Wie präzise konnten Sie die Aufgabe lösen?

Figure B.2: Screenshot of Nasa-TLX questionnaire in german.

Bibliography

Abrar Omar Alkhamisi and Muhammad Mostafa Monowar. Rise of augmented reality: Current and future application areas. *International Journal of Internet and Distributed Systems*, 01(04):25–34, 2013. ISSN 2327-7157. doi: 10.4236/ijids.2013.14005. URL `https://www.researchgate.net/publication/276494825`.

Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoper. Virtual Environ.*, 6(4):355–385, August 1997. ISSN 1054-7460. doi: 10.1162/pres.1997.6.4.355. URL `http://dx.doi.org/10.1162/pres.1997.6.4.355`.

Huidong Bai, Gun A. Lee, Mukundan Ramakrishnan, and Mark Billinghurst. 3d gesture interaction for handheld augmented reality. In *SIGGRAPH Asia 2014 Mobile Graphics and Interactive Applications*, SA '14, pages 7:1–7:6, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-1891-4. doi: 10.1145/2669062.2669073. URL `http://doi.acm.org/10.1145/2669062.2669073`.

Jennifer Billock. Five augmented reality experiences that bring museum exhibits to life: Ar features allow visitors to explore historical spaces and artifacts in new ways, 2017. URL `https://www.smithsonianmag.com/travel/expanding-exhibits-augmented-reality-180963810/`. *(Accessed: 03.08.2019).*

Mike Boland. How much are consumers willing to pay for vr?, 2018. URL `https://arinsider.co/2018/08/22/how-much-are-consumers-willing-to-pay-for-vr/`. *(Accessed: 28.07.2019).*

Patricia Brown. How to transform your classroom with augmented reality, 2015. URL `https://www.edsurge.com/news/2015-11-02-how-to-transform-your-classroom-with-augmented-reality`. *(Accessed: 03.08.2019).*

D. Chatzopoulos, C. Bermejo, Z. Huang, and P. Hui. Mobile augmented reality survey: From where we are to where we go. *IEEE Access*, 5:6917–6950, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2698164. URL `https://ieeexplore.ieee.org/document/7912316`.

Lloyd Connellan. Spheres, hyperspheres and quaternions. 2014. URL `https://warwick.ac.uk/fac/sci/masdoc/people/studentpages/students2014/connellan/spheres.pdf`.

Darren Edge and Alan F. Blackwell. Bimanual tangible interaction with mobile phones. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, TEI '09, pages 131–136, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-493-5. doi: 10.1145/1517664.1517697. URL `http://doi.acm.org/10.1145/1517664.1517697`.

Groÿekatthöfer and Yoon. Introduction into quaternions for spacecraft attitude representation. 2012. URL `http://www.tu-berlin.de/fileadmin/fg169/miscellaneous/Quaternions.pdf`.

Tatu Harviainen, Otto Korkalo, and Charles Woodward. Camera-based interactions for augmented reality. In *Proceedings of the International Conference on Advances in Computer Enterntainment Technology*, ACE '09, pages 307–310, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-864-3. doi: 10.1145/1690388.1690444. URL `http://doi.acm.org/10.1145/1690388.1690444`.

Morton Heilig. Augmented reality – the past, the present and the future, 2019. URL `https://www.interaction-design.org/literature/article/augmented-reality-the-past-the-present-and-the-future`. *(Accessed: 20.08.2019).*

Anders Henrysson, Joe Marshall, and Mark Billinghurst. Experiments in 3d interaction for mobile phone ar. In *Proceedings of the 5th International Conference on Computer*

Graphics and Interactive Techniques in Australia and Southeast Asia, GRAPHITE '07, pages 187–194, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-912-8. doi: 10.1145/1321261.1321295. URL `http://doi.acm.org/10.1145/1321261.1321295`.

Wolfgang Hürst and Casper van Wezel. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications*, 62, 01 2012. doi: 10.1007/s11042-011-0983-y. URL `https://www.researchgate.net/publication/257627138`.

Valentin Koch. Rotations in 3d graphics and the gimbal lock. 2016. URL `http://www.okanagan.ieee.ca/wp-content/uploads/2016/02/GimbalLockPresentationJan2016.pdf`.

Wenkai Li, Andrew Nee, and S K. Ong. A state-of-the-art review of augmented reality in engineering analysis and simulation. *Multimodal Technologies and Interaction*, 1:17, 09 2017. doi: 10.3390/mti1030017. URL `https://www.researchgate.net/publication/319494915`.

Annette Mossel, Benjamin Venditti, and Hannes Kaufmann. 3dtouch and homer-s: Intuitive manipulation techniques for one-handed handheld augmented reality. In *Proceedings of the Virtual Reality International Conference: Laval Virtual*, VRIC '13, pages 12:1–12:10, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1875-4. doi: 10.1145/2466816.2466829. URL `http://doi.acm.org/10.1145/2466816.2466829`.

Moti Ben-Ari. A tutorial on euler angles and quaternions. 2014. URL `http://www.weizmann.ac.il/sci-tea/benari/sites/sci-tea.benari/files/uploads/softwareAndLearningMaterials/quaternion-tutorial-2-0-1.pdf`.

Oliver Nowak. Armenus: An evaluation of menu interfaces in pen-based ar applications on smartphones. 2019. URL `https://hci.rwth-aachen.de/publications/nowak2019a.pdf`.

Rick Parent. *Computer Animation: Algorithms and Techniques*. The Morgan Kaufmann Series in Computer Graphics. Elsevier professional, s.l., 1. aufl. edition, 2001. ISBN

1558605797. URL `http://gbv.eblib.com/patron/FullRecord.aspx?p=413851`.

Perkins Coie LLP. 2018-vr-ar-survey-digital. 2018. URL `https://www.perkinscoie.com/images/content/1/8/v2/187785/2018-VR-AR-Survey-Digital.pdf`. *(Accessed: 20.08.2019).*

Jarkko Polvi, Takafumi Taketomi, Goshiro Yamamoto, Arindam Dey, Christian Sandor, and Hirokazu Kato. Slidar. *Comput. Graph.*, 55(C):33–43, April 2016. ISSN 0097-8493. doi: 10.1016/j.cag.2015.10.013. URL `https://doi.org/10.1016/j.cag.2015.10.013`.

Kelly Potter. Augmented reality becoming a focus in maintenance technology, 2019. URL `https://www.geospatialworld.net/blogs/augmented-reality-becoming-a-focus-in-maintenance-technology/`. *(Accessed: 03.08.2019).*

I. Seidinger and J. Grubert. 3d character customization for non-professional users in handheld augmented reality. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 129–134, Sep. 2016. doi: 10.1109/ISMAR-Adjunct.2016.0059. URL `https://ieeexplore.ieee.org/document/7836480`.

Do Hyoung Shin, Phillip S. Dunston, and Xiangyu Wang. View changes in augmented reality computer-aided-drawing. *ACM Trans. Appl. Percept.*, 2(1):1–14, January 2005. ISSN 1544-3558. doi: 10.1145/1048687.1048688. URL `http://doi.acm.org/10.1145/1048687.1048688`.

Husain Sumra. The best augmented reality glasses 2019: Snap, vuzix, microsoft, north & more, 2019. URL `https://www.wareable.com/ar/the-best-smartglasses-google-glass-and-the-rest`. *(Accessed: 20.08.2019).*

Philipp Wacker, Adrian Wagner, Simon Voelker, and Jan Borchers. Physical guides: An analysis of 3d sketching performance on physical objects in augmented reality. In *Proceedings of the Symposium on Spatial User Interaction*, SUI '18, pages 25–35, New York, NY, USA,

2018. ACM. ISBN 978-1-4503-5708-1. doi: 10.1145/3267782.3267788. URL `http://doi.acm.org/10.1145/3267782.3267788`.

Philipp Wacker, Oliver Nowak, Simon Voelker, and Jan Borchers. Arpen: Mid-air object manipulation techniques for a bimanual ar system with pen & smartphone. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 619:1–619:12, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-5970-2. doi: 10.1145/3290605.3300849. URL `http://doi.acm.org/10.1145/3290605.3300849`.

Robert Wang, Sylvain Paris, and Jovan Popović. 6d hands: Markerless hand-tracking for computer aided design. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 549–558, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. doi: 10.1145/2047196.2047269. URL `http://doi.acm.org/10.1145/2047196.2047269`.

Felix Wehnert. Pen-based drawing in augmented reality on mobile phones. 2018. URL `https://hci.rwth-aachen.de/publications/wehnert2018a.pdf`.

Po-Chen Wu, Robert Wang, Kenrick Kin, Christopher Twigg, Shangchen Han, Ming-Hsuan Yang, and Shao-Yi Chien. Dodecapen: Accurate 6dof tracking of a passive stylus. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 365–374, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126664. URL `http://doi.acm.org/10.1145/3126594.3126664`.

Index

Typeset August 28, 2019