INF FAKULTÄT FÜR INFORMATIK

Institut für Simulation und Graphik

Masterarbeit

Freihandgesten zur Vermessung medizinischer Strukturen während der Planung und Durchführung von Interventionen

Dominique Kasper

23. August 2016

Erstgutachter Jun.-Prof. Dr. Christian Hansen
Zweitgutachter Prof. Dr.-Ing. Bernhard Preim
Betreuer Patrick Saalfeld, M.Sc.

Kurzfassung

Während der Vermessung anatomischer und pathologischer Strukturen anhand von medizinischen Bilddaten in sterilen Umgebungen wie dem Operationssaal ist die Erhaltung der Sterilität essenziell. Aktuell wird dies entweder durch die Abdeckung der benötigten Eingabegeräte mit Plastikfolie oder mithilfe eines Systems in einem externen Kontrollraum gewährleistet. Die Steuerung von Letzterem erfolgt dabei indirekt durch die Weitergabe von Anweisungen an die Assistenten im Kontrollraum oder direkt durch den operierenden Arzt, indem dieser den Operationssaal verlässt und sich nach der Messung erneut steril macht. Diese Vorgehensweisen sind jedoch ineffizient und fehleranfällig. Daher beschäftigt sich die aktuelle Arbeit mit der Vermessung medizinischer Strukturen über Freihandgesten, die mithilfe berührungsloser Interaktion die Erhaltung der Sterilität sicherstellen. Im Rahmen dieser Untersuchungen wird zunächst eine Anforderungsanalyse anhand von Beobachtungen im klinischen Umfeld und entsprechenden Recherchen durchgeführt. Gemäß dieser erfolgt die Entwicklung der grundlegenden Interaktionskonzepte zur Handhabung der Benutzungsoberfläche und der Navigation in den medizinischen 2D- und 3D-Bilddaten sowie die Diskussion verschiedener Interaktionstechniken zur Vermessung von Distanzen, Winkeln, Flächen und Volumen. Anschließend findet die Umsetzung der entwickelten Konzepte in einem Prototyp zur Messung von Distanzen statt. Schließlich wird eine komparative Nutzerstudie durchgeführt, die sowohl dem Vergleich der Freihandgestensteuerung mit der Steuerung über Maus und Tastatur als auch der Überprüfung der Eignung der entwickelten gestenbasierten Interaktionskonzepte dient. Dabei stellte sich heraus, dass die implementierte Gestensteuerung der Handhabung per Maus und Tastatur in den Punkten Genauigkeit und Effizienz unterlegen ist. Als Gründe hierfür lassen sich vorrangig die mangelnde Robustheit der Gestenerkennung sowie Schwachstellen in den Mechanismen zur Erhöhung der Eingabegenauigkeit identifizieren. Daher sind entsprechende Verbesserungen notwendig, damit die Steuerung über Freihandgesten als Alternative zu den bisherigen Vorgehensweisen zur Vermessung medizinischer Strukturen während der Planung und Durchführung von Interventionen genutzt werden kann.

Abstract

The preservation of the sterility during the measurement of anatomical and pathological structures based on medical images in sterile environments like the operating theatre is essential. Currently this is done either by covering the input devices with sterile foil or with the help of a system in an external control room. The handling of the latter takes place indirectly via the passing on of instructions to the assistants in the control room or directly through the operating physician who therefore has to leave the operation theatre and get sterile again after the measurement. Such practices are inefficient and error-prone. Hence the current work addresses the measurement of medical structures via freehand gestures which ensure the preservation of the sterility through touchless interaction. Within the context of this investigation a requirement analysis based on observations in a clinical environment and corresponding research is conducted. According to this the design and development of the essential interactions for the handling of the interface and the navigation in the 2D and 3D medical images plus the discussion of various interaction techniques for the measurement of distances, angles, areas and volumes takes place. Subsequently the developed concepts get implemented in a prototype for the measurement of distances. Eventually a comparative user study is conducted which serves as a comparison between the freehand gesture control and the control via mouse and keyboard as well as an examination of the applicability of the designed gesture based interaction concepts. In this process it became apparent that the implemented gesture control is inferior to the control via mouse and keyboard when it comes to precision and efficiency. The poor robustness of the gesture detection and weak points in the mechanisms for the enhancement of the input precision could be identified as the primary reasons for this. Thus appropriate improvements are required in order to use the control with freehand gestures as an alternative for present approaches for the measurement of medical structures during the planning and execution of surgical operations.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Masterarbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Stellen sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Dominique Kasper
Magdeburg, 23. August 2016

Inhaltsverzeichnis

Abkürzungsverzeichnis **XIII**

1 Einleitung **1**
1.1 Zielsetzung und Fragestellung . . . 2
1.2 Überblick über die Arbeit . . . 3

2 Grundlagen und Verwandte Arbeiten **5**
2.1 Grundlagen Freihandgesten . . . 5
2.1.1 Begriffsbestimmung . . . 5
2.1.2 Taxonomie . . . 6
2.1.3 Gestenphasen . . . 6
2.1.4 Eingabegeräte für Freihandgesten . . . 7
2.2 Verwandte Arbeiten . . . 10
2.2.1 Freihandgesten . . . 10
2.2.2 Vermessung . . . 12
2.2.3 Einordnung der Arbeit . . . 13
2.3 Zusammenfassung . . . 13

3 Anforderungsanalyse **15**
3.1 Beobachtungen . . . 15
3.1.1 Planung und Durchführung . . . 16
3.1.2 Beobachtete Messungen . . . 16
3.1.3 Interaktionsumfeld . . . 18
3.1.4 Ablauf der Vermessung während einer Intervention . . . 19
3.1.5 Probleme . . . 19
3.1.6 Wichtige Aspekte während der Vermessung . . . 21
3.1.7 Voraussetzungen für Freihandgesten . . . 22
3.2 Nicht-funktionale Anforderungen . . . 22
3.3 Funktionale Anforderungen . . . 26
3.4 Zusammenfassung . . . 31

4 Konzept **35**
4.1 Allgemeine Entwurfsgrundsätze . . . 35
4.1.1 Vermeiden einer Kommandosprache . . . 35
4.1.2 Vermeiden statischer Gesten . . . 36

4.1.3 Bevorzugung bekannter Interaktionskonzepte . . . 36
4.1.4 Ausführung der Gesten aus egozentrischer Perspektive . . . 36
4.1.5 Bevorzugung beidhändiger Interaktion . . . 36
4.1.6 Einhaltung ergonomischer Kriterien . . . 37
4.2 Interface-Konzept . . . 38
4.2.1 Farbschema für Markierungen in den Bilddaten und im Interface . 39
4.2.2 Radialmenüs . . . 40
4.2.3 Laden der Bilddaten . . . 40
4.3 Visuelles Feedback . . . 41
4.3.1 Gesten-Guide . . . 42
4.3.2 Repräsentation der Gesten . . . 42
4.3.3 Stopper . . . 43
4.4 Interaktionstechniken . . . 44
4.4.1 Zeigen . . . 44
4.4.2 Pinch-to-Click . . . 45
4.4.3 Wait-to-Click . . . 45
4.4.4 Steuerung der Radialmenüs . . . 46
4.4.5 Manipulation von Objekten . . . 47
4.4.6 Navigation durch die 2D-Schichtdaten . . . 50
4.4.7 Messungen erstellen . . . 50
4.4.8 Bearbeiten und Löschen von Messungen . . . 59
4.4.9 Rückgängig machen und Wiederholen von Aktionen . . . 60
4.5 Nutzerunterstützung . . . 61
4.5.1 Exponentielle Glättung . . . 61
4.5.2 Snapping . . . 62
4.6 Auswahl des Eingabegerätes . . . 63
4.7 Zusammenfassung . . . 64

5 Implementierung 65
5.1 Entwicklungsumgebung . . . 65
5.2 Gestenerkennung . . . 66
5.2.1 Zeigegeste . . . 66
5.2.2 Pinch-Geste . . . 67
5.2.3 Kamerasteuerung . . . 67
5.2.4 Wischgeste . . . 69
5.2.5 Circle-Geste . . . 70
5.3 Mechanismen zur Nutzerunterstützung . . . 71
5.3.1 Filterung der Handpositionen . . . 71
5.3.2 Snapping . . . 72
5.4 Umsetzung der Steuerung über Maus und Tastatur . . . 77
5.5 Zusammenfassung . . . 78

6 Evaluierung 81
6.1 Planung . . . 81

6.1.1 Bewertungskriterien 81
6.1.2 Auswahl der Probanden 82
6.2 Durchführung 83
6.2.1 Versuchsumgebung 83
6.2.2 Einführung in die Handhabung 84
6.2.3 Praktische Anwendung 84
6.2.4 Befragung 85
6.3 Ergebnisse 85
6.3.1 Zusammensetzung der Probanden 85
6.3.2 Quantitative Ergebnisse 86
6.3.3 Qualitative Ergebnisse 91
6.4 Auswertung 95
6.4.1 Stärken 95
6.4.2 Schwächen 96
6.4.3 Verbesserungsmöglichkeiten 96
6.5 Diskussion 101

7 Zusammenfassung **103**
7.1 Fazit 104
7.2 Ausblick 105

Anhang **107**
Anhang A: Fragebogen für die Interviews am Universitätsklinikum Magdeburg . 107
Anhang B: Fragebogen für die Nutzerstudie 109

Literaturverzeichnis **111**

Abkürzungsverzeichnis

ANOVA Varianzanalyse

C/D-Wert Control-to-Display Ratio

CT Computertomographie

HMD Head-mounted Display

MCI Mensch-Computer-Interaktion

MRT Magnetresonanztomographie

ROI Region of Interest

SDK Software Development Kit

SUS System Usability Scale

1 Einleitung

Sowohl für die Planung als auch für die Durchführung einer medizinischen Intervention spielt die Beurteilung anatomischer und pathologischer Strukturen eine wichtige Rolle. Diese erfolgt anhand entsprechender 2D- und 3D-Bilddaten, für deren Aufnahme zahlreiche bildgebende Verfahren wie beispielsweise die Computertomographie (CT) und die Magnetresonanztomographie (MRT) zur Verfügung stehen. Dabei ist eine rein qualitative visuelle Einschätzung nicht immer ausreichend, wodurch zusätzlich quantitative Angaben benötigt werden [1, 2]. So hängt beispielsweise die Therapieplanung bzw. die Therapierbarkeit eines Tumors von seinem Abstand zu anderem, gesunden Gewebe sowie von seinem Volumen bzw. seiner maximalen Ausdehnung ab [3]. Darüber hinaus werden während eines operativen Eingriffs ebenfalls Informationen zu den räumlichen Verhältnissen der medizinischen Strukturen benötigt. Dies ist unter anderem während der Behandlung eines Aneurysmas der Fall, welches es mit Hilfe von Platindrähten (sogenannten *Coils*) zu verschließen gilt. Die Abmaße der Coils werden dabei anhand des Durchmessers des Aneurysmas geschätzt. Des Weiteren müssen beim operativen Einsetzen von Prothesen die Winkel zu anderen Strukturen stimmen, damit später keine Probleme auftreten [2]. Außerdem führen auch Notfälle dazu, dass Messungen während einer Operation vorgenommen werden müssen, die andernfalls im Laufe der Planung durchzuführen wären.

Vor allem in sterilen Umgebungen wie dem Operationssaal ist auch während der Interaktion mit den Bilddaten darauf zu achten, dass die Sterilität erhalten bleibt [4]. Um diese Vorgabe zu erfüllen, haben sich bisher einige Vorgehensweisen entwickelt, welche sich jedoch als nicht ideal erwiesen. So besteht eine Möglichkeit darin, dass die Vermessung an einen außerhalb des Operationssaals befindlichen Assistenten delegiert wird. Allerdings ist diese Art und Weise des indirekten Kontakts mit einer umständlichen Kommunikation zwischen Operateur und Assistent verbunden, wodurch es neben dem personellen und zeitlichen Mehraufwand zu einer höheren Fehleranfälligkeit kommt. Letzteres kann vermieden werden, indem der Operateur selbst den sterilen Raum verlässt, um die Messungen im externen Kontrollraum durchzuführen. Aufwendig ist diese Lösung dennoch, da der Operateur sich anschließend erneut steril machen muss, um die Operation fortführen zu können. Im Gegensatz zu diesen beiden Vorgehensweisen wäre die direkte Interaktion des Operateurs mit den Bilddaten wünschenswert, wird bisher jedoch nur durch das Verpacken von nicht sterilen Eingabegeräten in sterile Folie und zu Lasten der Benutzungsfreundlichkeit erreicht. Dabei ist ein Reißen der Folie außerdem

nicht gänzlich auszuschließen [5, 6].

Eine mögliche Lösung des Problems zeichnet sich in Form berührungsloser Interaktion mittels Freihandgesten ab, da hier auf den physischen Kontakt mit dem Eingabegerät verzichtet und somit die Erhaltung der Sterilität gewährleistet wird [7, 8]. Auf diese Weise ist eine direkte Interaktion mit den Bilddaten zur Vermessung in sterilen Umgebungen möglich. Darüber hinaus handelt es sich bei den Freihandgesten um eine sehr natürliche Form der Kommunikation, die der Mensch meist schon vor der Sprache beherrscht und die mehr Freiheitsgrade als herkömmliche Eingabegeräte mit sich bringt. Daher bietet sich hier die Chance auf eine Vielzahl neuer, einfacher und intuitiver Interaktionsmöglichkeiten [5, 6, 9].
Gleichzeitig birgt die Steuerung mit Freihandgesten auch einige Herausforderungen. So ist die erreichbare Präzision bisher eher gering [10] und insbesondere wiederholte und längere Interaktion mithilfe von komplexen bzw. aufwendigen Gesten kann schnell zu Ermüdungen der betreffenden Muskelgruppen führen. Des Weiteren erweisen sich unterbewusste Handbewegung als problematisch, wenn sie fälschlicherweise als bewusste Eingabe interpretiert werden. Auch sind die möglichen Gesteninteraktionen nicht so leicht sichtbar wie die Bedienelemente einer herkömmlichen Benutzungsoberfläche, was einen entsprechenden Ausgleich erforderlich macht [11].

1.1 Zielsetzung und Fragestellung

Im Zuge dieser Arbeit soll die Frage beantwortet werden, inwiefern die berührungslose Interaktion mit Freihandgesten geeignete Interaktionskonzepte für die Vermessung medizinischer Strukturen während der Planung und Durchführung einer Intervention bieten kann. Dabei müssen diese nicht nur die Sterilität bei der direkten Interaktion mit den Bilddaten erhalten, sondern auch Kriterien in Bezug auf das Anwendungsgebiet sowie die Benutzungsfreundlichkeit erfüllen. Außerdem sind die zuvor erwähnten Herausforderungen einer Freihandgestensteuerung entsprechend zu behandeln.
Das Ziel der Untersuchungen besteht somit darin, herauszufinden, wie sich die gestenbasierten Interaktionen so gestalten lassen, dass diese Anforderungen erfüllt werden.

Zu diesem Zweck wird zunächst eine auf Literaturrecherchen und Beobachtungen im klinischen Umfeld basierende Anforderungsanalyse durchgeführt. Anhand dieser sollen verschiedene Interaktionskonzepte entwickelt und gegeneinander abgewogen werden, um ein möglichst breites Spektrum an Möglichkeiten der Interaktion mit Freihandgesten zur Vermessung medizinischer Strukturen abzudecken und in die Untersuchungen einzubeziehen. Auf diese Weise soll ein Konzept erarbeitet werden, welches bezüglich der gestellten Anforderungen am besten geeignet ist. Anschließend erfolgt dessen Umsetzung in einem

Prototypen, um anhand einer abschließenden, komparativen Evaluierung die Eignung der entwickelten Interaktionstechniken überprüfen zu können. Dabei wird die Freihandgestensteuerung mit der Steuerung des Prototyps über Maus und Tastatur verglichen, so dass eine Bewertungsreferenz für Erstere entsteht.

1.2 Überblick über die Arbeit

Nachfolgend soll ein Überblick über die einzelnen Inhalte der Arbeit gegeben werden. So behandelt **Kapitel 2** zunächst die Grundlagen zum Thema Freihandgesten. Dabei erfolgt die Bestimmung des Begriffes „Freihandgeste“ sowie die Vorstellung einer entsprechenden Taxonomie. Anschließend werden die Phasen der Gestenausführung beschrieben und mögliche Eingabegeräte für Freihandgesten dargestellt. Außerdem findet eine Auseinandersetzung mit den bereits existierenden Arbeiten zu den Themen Freihandgesten und Vermessung statt, die mit der Einordnung der aktuellen Arbeit in den momentanen Stand der Wissenschaft endet.

In **Kapitel 3** wird eine Anforderungsanalyse als Grundlage der Entwicklung der verschiedenen Interaktionskonzepte zur Vermessung in medizinischen Bilddaten durchgeführt. Diese basiert sowohl auf Hintergrundrecherchen im Bereich der Medizin und der interaktiven Systeme als auch auf der Hospitation am Institut für Neuroradiologie des Universitätsklinikums Magdeburg. Die während der Hospitation gemachten Beobachtungen werden dabei ausführlich erläutert. Im Anschluss daran erfolgt die Schilderung der funktionalen und nicht-funktionalen Anforderungen.

Nachdem die Anforderungen an die zu entwickelnden Interaktionstechniken ermittelt wurden, findet in **Kapitel 4** die Konzeption selbiger statt. Dabei werden zunächst einige, allgemeine Entwurfsgrundsätze festgelegt, die die Basis der gestenbasierten Interaktionskonzepte bilden. Anschließend wird das Konzept der Benutzungsoberfläche erläutert und das visuelle Feedback zur Unterstützung des Nutzers bei der Interaktion mit den Freihandgesten vorgestellt. Des Weiteren erfolgt die Schilderung der für die Vermessung notwendigen Interaktionen. Den Abschluss des Kapitels bilden die Mechanismen zur Nutzerunterstützung, die die Eingabegenauigkeit erhöhen sollen, sowie die Auswahl eines geeigneten Eingabegerätes zur Umsetzung der entwickelten Konzepte.

In **Kapitel 5** wird schließlich die Implementierung eines Prototyps anhand der zuvor entwickelten Interaktionstechniken beschrieben. So erfolgt zuerst die Darstellung der verwendeten Entwicklungsumgebung, um danach zu den Algorithmen der Gestenerkennung und der Umsetzung der Mechanismen zur Nutzerunterstützung übergehen zu können.

Zusätzlich wird die Implementierung einer mit Maus und Tastatur gesteuerten Version des Prototyps erläutert. Sie ist notwendig, um die abschließende, vergleichende Evaluierung zu ermöglichen.

Kapitel 6 widmet sich der komparativen Evaluierung der beiden Prototypen anhand einer Nutzerstudie. Dabei erfolgt anfangs die Schilderung der Auswahl geeigneter Bewertungskriterien und Probanden. An diese schließen sich Erläuterungen zum Ablauf der Studie sowie zu den erzielten Ergebnissen an. Danach findet die Auswertung Letzterer sowie deren Zusammenfassung zu Stärken und Schwächen des entwickelten Systems statt. Außerdem werden die aus den Beobachtungen abgeleiteten Verbesserungsmöglichkeiten beschrieben. Das Ende des Kapitels bildet eine Diskussion der Resultate der Evaluierung.

Abschließend erfolgt in **Kapitel 7** die Zusammenfassung der Arbeit sowie der gewonnenen Erkenntnisse und erzielten Ergebnisse. In diesem Rahmen wird auch ein Ausblick auf die Aufgaben weiterführender Untersuchungen zum Thema der Vermessung mit Freihandgesten während der Planung und Durchführung von Interventionen gegeben.

2 Grundlagen und Verwandte Arbeiten

In Vorbereitung auf die weiteren Schritte der Arbeit soll sich das aktuelle Kapitel sowohl mit den Grundlagen, die in den folgenden Abschnitten vorausgesetzt werden, als auch mit früheren Untersuchungen zu verwandten Themen im Bereich der Freihandgesten und der Vermessung beschäftigen. Die Auseinandersetzung mit verwandten Arbeiten ist dabei wichtig, um sowohl die dort erzielten Fortschritte und gemachten Erfahrungen nutzen zu können, als auch, um die aktuelle Arbeit in den Kontext des momentanen Standes der Wissenschaft einzuordnen.
Daher sollen im Folgenden zuerst die Grundlagen im Bereich der Freihandgesten näher beleuchtet werden und anschließend die Beschäftigung mit den bereits existierenden Arbeiten zu verwandten Themen erfolgen. Den Abschluss bildet schließlich die Einordnung der aktuellen Arbeit in den derzeitigen Stand der Technik.

2.1 Grundlagen Freihandgesten

Da die Konzeption und Umsetzung von freihandgestenbasierten Interaktionstechniken einen großen Teil der Arbeit ausmacht, wird nachfolgend ein kurzer Einblick in die Thematik gegeben.
Den Einstieg stellt dabei die Erläuterung des Begriffes der Freihandgesten sowie einer entsprechenden Taxonomie dar. Im Anschluss werden die Phasen der Ausführung einer Geste beschrieben und abschließend ein Überblick über die möglichen Eingabegeräte zur Erkennung von Freihandgesten gegeben.

2.1.1 Begriffsbestimmung

Bevor sich die Bezeichnung der „Freihandgeste“ klar definieren lässt, ist zunächst die allgemeine Begrifflichkeit der „Geste“ zu klären. Diese wird laut Preim und Dachselt [11] im Rahmen der Mensch-Computer-Interaktion (MCI) wie folgt definiert:

> „*Eine* Geste *ist die Bewegung von Fingern, Händen und Armen - oder auch weiterer Körperteile, wie Kopf, Augen und Lippen - aufgrund einer kommunikativen Absicht. Damit enthält die Bewegung als solche signifikante Informationen, die an den Computer übermittelt werden sollen.*“

Ausgehend von dieser Definition sind unter Freihandgesten solche Gesten zu verstehen, die „mit einer oder beiden Händen frei im Raum ausgeführt werden“ [11]. Sie können dabei sowohl statisch als auch dynamisch verlaufen [11].

2.1.2 Taxonomie

Gesten, und somit auch Freihandgesten, lassen sich zusätzlich hinsichtlich ihres Stils unterscheiden. Eine entsprechende Taxonomie wurde dabei von Preim und Dachselt [11] vorgestellt und beinhaltet folgende Gestenarten:

- **Deiktische Gesten** zum Zeigen und Verweisen
- **Manipulative Gesten**, die der Handhabung und Manipulation von Objekt dienen, wobei die Bewegungen der Hand eng mit dem zu verändernden Objekt verbunden sind
- **Symbolische Gesten**, die Zeichen bzw. Symbole zum Auslösen von Befehlen repräsentieren und dabei statisch oder dynamisch sein können
- **Gesten einer Zeichen- und Gebärdensprache**, die nur im Zusammenhang mit anderen Gesten derselben Sprache eine Bedeutung erlangen
- **Sprachbegleitende Gesten**, die parallel zur Sprache erfolgen und in deren Kontext zu deuten sind

Zusätzlich wird zwischen diskreten und kontinuierlichen Gesten unterschieden, wobei Erstere erst nach Abschluss ihrer Ausführung detektiert werden, während die Erkennung Letzterer fortlaufend erfolgt [11].

2.1.3 Gestenphasen

Die Ausführung einer Geste lässt sich in verschiedene Phasen unterteilen. Diese werden im Allgemeinen wie folgt abgegrenzt [11]:

- **Ruhephase:** Stabile Positionierung vor Beginn der Geste
- **Vorbereitungsphase:** Bewegung aus der Ruheposition heraus
- **Gestenbewegung:** Kommunikative Bewegung mit dem höchsten Aufwand und dem höchsten Informationsgehalt
- **Halten:** Phase ohne Bewegungen vor oder nach der Durchführung der Geste
- **Rücknahme- bzw. Erholungsphase:** Rückführung der Hände in eine Ruheposition

Dabei ist jedoch zu beachten, dass die Phase der *Gestenbewegung* als einzige obligatorisch ist, während es sich bei allen anderen um optionale Phasen handelt, die je nach Ablauf der Gesteninteraktion variieren können.

2.1.4 Eingabegeräte für Freihandgesten

Für die Erkennung der Freihandgesten existieren verschiedene Eingabegeräte. Sie sollen im Folgenden, geordnet nach dem jeweiligen Detektionsverfahren, beschrieben werden.

Kamera-basierte Eingabegeräte

Diese Art von Eingabegeräten erfordert grundsätzlich den Sichtkontakt zwischen den Händen und dem Gerät, da sie optisches Tracking zur Erkennung der Gesten nutzt. Dabei kommen folgende Geräte und Hilfsmittel zum Einsatz [11]:

- Marker zur Lokalisation relevanter Punkte
- Kameras zur Bildaufnahme
- Tiefenkameras zur Tiefenbestimmung

Erstere müssen an der Hand des Nutzers angebracht werden. Anschließend werden die Marker von mindestens zwei Kameras erfasst, um mithilfe von trigonometrischen Berechnungen die 3D-Position und Orientierung der Hände bzw. ihrer Gelenke ermitteln zu können. Dabei ist zwischen passiven und aktiven Markern zu unterscheiden. So handelt es sich bei passiven Markern um regelmäßig geformte sowie kontrastreiche Objekte von ausreichender Größe (siehe Abbildung 2.1a), die jedoch selbst keine Signale aussenden,

(a) Passive Marker **(b)** Leap Motion Controller **(c)** Microsoft Kinect V2

Abbildung 2.1: Beispiele für Hilfsmittel und Geräte des optischen Trackings von Freihandgesten. Quellen: [12, 13, 14]

während aktive Marker anhand des von ihnen abgegebenen Lichts lokalisiert werden.
Darüber hinaus ist auch ein markerloses, optisches Tracking möglich. So lassen sich allein anhand der Bilddaten der Kamera charakteristische Merkmale der Hand extrahieren und entsprechend verarbeiten. Dieses Detektionsverfahren ist jedoch weniger robust und abhängig von Einflüssen wie der Beleuchtung und der Hautfarbe des Anwenders.
Zusätzlich kann mithilfe von Tiefenkameras die Entfernung des verfolgten Objektes zur Kamera bestimmt werden. Dies geschieht meist in Zusammenhang mit Infrarotlicht, welches ausgesendet wird, um zu messen, wie weit dessen Weg zur nächsten Oberfläche ist. Basierend auf diesen Daten können dann die Tiefeninformationen berechnet werden. Genutzt wird diese Vorgehensweise unter anderem vom Leap Motion Controller der Firma Leap Motion sowie der Kinect von Microsoft (siehe Abbildung 2.1b und Abbildung 2.1c). Es ist jedoch zu beachten, dass sowohl Qualität als auch Auflösung des durch die Tiefenkameras erzeugten Signals begrenzt sind. Dieses muss daher meist durch Filterung und Bildrekonstruktion verbessert werden. [11]

Datenhandschuhe

Bei dem Datenhandschuh handelt es sich um einen Handschuh, in den, je nach Ausführung, verschiedenen Sensoren zur Erfassung der Bewegungen der Finger und des Handgelenks integriert sind (siehe Abbildung 2.2 links). So werden beispielsweise Dehnmessstreifen genutzt, um die Krümmung der Finger zu ermitteln. Außerdem enthalten einige Datenhandschuhe sogenannte Aktuatoren, die elektrische Signale in mechanische Bewegungen umwandeln und so eine Kraftrückkopplung ermöglichen. Alternativ können auch nur einzelne Sensoren an ausgewählten Fingern getragen werden (siehe Abbildung 2.2 rechts).
Auf diese Weise lassen sich die Bewegungen der Hand bzw. der einzelnen Finger genauer und zuverlässiger erfassen. Der Nachteil besteht allerdings darin, dass das Tragen des Datenhandschuhs wenig komfortabel ist sowie die Bewegung der Hand teilweise einschränkt. Auch lässt er sich nicht schnell an- und ablegen. Des Weiteren muss der Datenhandschuh vor der Nutzung an den jeweiligen Nutzer angepasst werden, da sich die Proportionen

Abbildung 2.2: Datenhandschuh von Peregrine [15] (links) und Eingabegerät für das Fingertracking von Advanced Realtime Tracking [16] (rechts)

der Hände von Mensch zu Mensch deutlich unterscheiden können. [11]

Magnetisches Tracking

Das magnetische Tracking ist eine weit verbreite Vorgehensweise, die auf über die Zeit veränderlichen, elektromagnetischen Feldern basiert und somit keinen direkten Sichtkontakt benötigt. Die elektromagnetischen Felder werden dabei von Spulen erzeugt. In den Drähten innerhalb dieser Felder entsteht außerdem ein Stromfluss, der sich nutzen lässt, um die Position eines beweglichen Empfängers zu bestimmen.
Die für das magnetische Tracking erforderlichen Sensoren sind mittlerweile kleiner und leichter, so dass deren Ergonomie wesentlich verbessert werden konnte. Auch weisen sie eine hohe Genauigkeit und eine geringe Latenz auf. Von Nachteil ist jedoch, dass die verwendeten elektromagnetischen Felder durch metallische Gegenstände und andere Magnetfelder gestört werden. Des Weiteren steht nur ein relativ kleiner Arbeitsraum zur Verfügung. [11]

Myoelektrische Eingabegeräte

Myoelektrische Eingabegeräte nutzen das durch biochemische Vorgänge entstehende, elektrische Potenzial von Muskelzellen, welches über die Haut gemessen werden kann. Das Aktionspotenzial wird dabei durch das Zusammenziehen und Entspannen der Muskelfasern herbeigeführt. Zum Einsatz kommt diese Technik beim Myo-Armband von Thalmic Labs Inc. (siehe Abbildung 2.3), welches am Unterarm getragen wird und dort die Kontraktion der Muskeln auf entsprechende Handbewegungen abbildet. [17]

Abbildung 2.3: Myo Armband von Thalmic Labs Inc. [18]

2.2 Verwandte Arbeiten

Um frühere Untersuchungen aus verwandten Themenbereichen identifizieren und sich mit ihnen auseinandersetzen zu können, wurde im Vorfeld der Arbeit eine systematische Literaturrecherche durchgeführt. In diesem Rahmen erfolgte die Durchsuchung zahlreicher digitaler Bibliotheken und Online-Datenbanken. Zu diesen gehören:

- ACM Digital Library
- ScienceDirect
- PubMed
- Springer Link
- ResearchGate
- Google Scholar

Im Folgenden werden die dabei ermittelten, verwandten Arbeiten dargestellt. Außerdem soll erläutert werden, an welchen Stellen die aktuelle Arbeit ansetzt, um entsprechende Fortschritte zu erreichen.

2.2.1 Freihandgesten

Im Bereich der Freihandgesten existieren zahlreiche Arbeiten mit medizinischem Kontext. Die meisten beschäftigen sich dabei mit der Navigation und Manipulation innerhalb von medizinischen Bilddaten.

So entwickelten Hettig et al. [19] ein Gesteninterface für das Myo-Armband, welches die Bewegung innerhalb von 2D- und 3D-Bilddaten mithilfe von Panning, Zoom und Rotation ermöglicht. Die hier entwickelte Gestensteuerung weist jedoch Schwächen in Form mangelnder Robustheit auf und auch das verwendete Armband entspricht in seinem Tragekomfort nicht vollends den Anforderungen der Ärzte. Ähnliche Arbeiten stammen außerdem von Riduwan et al. [20] und Luigi Gallo [21], beinhalten allerdings keine Studien zur Validierung der entwickelten Konzepte.
Andere Gruppen [4, 22] bezogen zusätzlich die Navigation in 2D-Schichtdaten in ihre Untersuchungen mit ein. So gelang es Mewes et al. [4] ein intuitives Gestenset zu entwickeln, welches jedoch noch Schwachstellen in den Bereichen Ergonomie und Robustheit besitzt. Neben den Gesten zur Navigation in den Bilddaten erarbeiteten sie auch ein Konzept für das Zeigen und Selektieren von virtuellen Objekten.
Weitere gestenbasierte Lösungsansätze in diesem Bereich wurden außerdem von Soutschek et al. [23] und Ferracani et al. [24] vorgestellt. Erstere entwickelten ein System zur Exploration und Navigation in medizinischen 3D-Bilddaten und konnten anhand einer Nutzerstudie dessen Machbarkeit hinsichtlich einer sehr guten Erkennungsrate, einer akzeptablen Usability sowie einer intuitiven und effizienten Handhabung zeigen. Demgegenüber dient das von Ferracani et al. [24] umgesetzte, gesten- und sprachgesteuerte Interface dem Training von medizinischem Personal in Bezug auf die Einhaltung bestimmter Sicherheitsabläufe. Es wurde jedoch bisher noch keiner Evaluierung unterzogen.
Noch breiter gefächerte Arbeiten stammen schließlich von Shen et al. [25] and Druckmiller et al. [26]. Erstere nutzen eine teilweise bimanuelle Gestensteuerung für den Leap Motion Controller, welche sich im Rahmen einer Evaluierung als natürlicher und schneller als eine einhändige Bedienung erwies. Sie dient unter anderem der Platzierung von Schnittebenen sowie der Definition eines Volumenclippings über Regions of Interest (ROIs). Währenddessen ermöglicht die von Druckmiller et al. [26] entwickelte Benutzungsschnittstelle die Konfiguration der Fensterung medizinischer Bilddaten und das Markieren von Läsionen in selbigen über die Kinect. Des Weiteren ist das Erstellen sowie Bearbeiten von Resektionslinien bzw. -ebenen möglich. Studien zu dieser Arbeit existieren jedoch keine.

Darüber hinaus gibt es kaum Arbeiten, die sich mit der Vermessung von Objekten mithilfe von Freihandgesten auseinandersetzen. So haben nur Abate et al. [27] eine Gestensteuerung entwickelt, die sowohl die Manipulation von 3D-Objekten durch Selektion, Rotation und Translation als auch die Erstellung von Distanzmessungen ermöglicht. Eine Geste für das Abbrechen von Aktionen wurde ebenfalls definiert. Als Eingabegerät dient ein Datenhandschuh in Kombination mit einem Head-mounted Display (HMD). Eine in diesem Rahmen durchgeführte Studie stellte die erarbeiteten Konzepte dabei als vielversprechend heraus, da gezeigt werden konnte, dass die Probanden zunehmend bessere Leistungen in der Anwendung der Benutzungsschnittstelle erzielten.

2.2.2 Vermessung

Einige Arbeiten beschäftigen sich unabhängig von Freihandgesten mit dem Entwurf angemessener Interaktionstechniken und Visualisierungen zur Vermessung. Zu jenen gehört auch eine Reihe von Untersuchungen von Preim et al. [28, 29, 30], welche sich mit Distanz-, Winkel- und Volumenmessungen beschäftigen. Dabei wird nicht nur die manuelle Vermessung in medizinischen Bilddaten aufgegriffen, sondern auch die automatischen Bestimmung der Messwerte. Allerdings steht eine Evaluierung der hier vorgestellten Techniken noch aus.
Ebenfalls im medizinischen Bereich angesiedelt ist die Dissertation von Ivo Rössling [2]. Sie beschäftigt sich mit der automatischen und generischen Berechnung abstandsbasierter Maße sowie deren optimaler Visualisierung im Rahmen medizinischer Segmentierungen. Den hier angestellten Untersuchungen ging außerdem eine Arbeit von Rössling et al. [1] voraus. Diese leitete aus einer Studie mit Medizinern Empfehlungen für die Visualisierung von Vermessungen der Distanz und Ausdehnung im dreidimensionalen Raum ab.
Des Weiteren wurden komplexe Vermessungsinteraktionen von Hagedorn et al. [31] vorgestellt. Diese umfassen Distanzmessungen von Punkt zu Punkt sowie die Erhebung verschiedener Messgrößen mithilfe von ellipsoiden bzw. zylindrischen Volumenkörpern. Außerdem werden verschiedene Clippingmöglichkeiten zur Verfügung gestellt. Die Steuerung erfolgt hier mithilfe eines in der Hand gehaltenen, getrackten Controllers und einer auf diesem befindlichen Taste. Es ließen sich jedoch noch einige, ungelöste Probleme am entwickelten Prototypen feststellen. So mangelte es an Genauigkeit und Zuverlässigkeit bei der Vermesssung, welche sich insbesondere im Fall der Messung mit dem Zylinder sehr zeitaufwendig gestaltete. Dies führte zu der Schlussfolgerung, dass Mechanismen zur Steigerung der Genauigkeit notwendig sind.
Eine mögliche Vorgehensweise zur Verbesserung der Genauigkeit kommt dabei im Rahmen der Arbeit von Reitinger et al. [32] zum Einsatz. So wurde ein Modus integriert, welcher es dem Nutzer ermöglicht, den nächsten Punkt auf der Oberfläche des nächstgelegenen Objektes anzuvisieren. Des Weiteren werden verschiedene Konzepte zur Distanz- und Volumenmessungen sowie zur Vermessung von Winkeln, minimalen Abständen und der Ausdehnung von Objekten vorgestellt. Die Eingabe erfolgt je nach Art der Vermessung mit getrackten Objekten (z. B. einem Lineal für die Distanzmessung) oder einem Stift mit einem Knopf zur Bestätigung von Interaktionen (z. B zur Platzierung von Messpunkten). Als problematisch könnte sich allerdings der hohe Preis für das verwendete Virtual-Reality-System erweisen. Andererseits ist zu bedenken, dass die entwickelte Anwendung positive Ergebnisse in Bezug auf die Effizienz, Erlernbarkeit und Genauigkeit der Vermessungsinteraktionen erzielen konnte.
Demgegenüber setzen sich Ozmen et al. [33] mit der Vermessung von Kulturgütern auseinander, indem sie die realen Messvorgänge und die dazu notwendigen Werkzeuge in den virtuellen Raum übertragen. So wird beispielsweise der Radius eines Tontopfes mit einer Kreisscheibe gemessen oder der Abstand mithilfe eines Messschiebers bestimmt. Das entwickelte System weist jedoch noch Probleme in der Genauigkeit der Messungen auf.

Eine Evaluierung mit potenziellen Nutzern wurde ebenfalls noch nicht durchgeführt.

2.2.3 Einordnung der Arbeit

Die vorhergehenden Abschnitte zeigen, dass bereits Arbeiten existieren, die die Vermessung mithilfe von Gesten bzw. Freihandgesten thematisieren. Ihnen allen ist gemein, dass sie nicht gänzlich ohne den Kontakt des Nutzers mit entsprechenden Eingabegeräten auskommen und somit das Kriterium der Sterilität nicht ohne die in Kapitel 1 beschriebenen Umwege erfüllen können. An diesem Punkt soll die aktuelle Arbeit ansetzen, indem sie ein berührungsloses Interaktionskonzept zur Vermessung mit Freihandgesten entwickelt. Dies geschieht mithilfe sichtbasierter Gestenerkennung, für die nur der Sichtkontakt zwischen Eingabegerät und Händen notwendig ist.
Ein weiterer Fortschritt besteht darin, dass die entsprechenden Interaktionstechniken nicht nur in einem theoretischen Rahmen entworfen und bewertet, sondern auch mit Nutzern evaluiert werden sollen.
Darüber hinaus wird das Problem der teilweise mangelnden Genauigkeit bei der Platzierung der Messungen (z. B. durch unruhige Hände oder Einschränkungen des Eingabegerätes) aufgegriffen. Zu diesem Zweck sollen Mechanismen entwickelt werden, die den Nutzer bei der Eingabe unterstützen.

2.3 Zusammenfassung

In diesem Kapitel erfolgte zunächst eine Einführung in die Grundlagen der Freihandgesten, indem sowohl eine Definition der entsprechenden Begrifflichkeiten sowie einer Taxonomie der verschiedenen Gestenstile erfolgte. So umfassen Freihandgesten alle Bewegungen der Hände, die frei im Raum und mit einer kommunikativen Absicht ausgeführt werden und lassen sich in folgende Arten von Gesten unterteilen: Deiktische Gesten zum Zeigen und Verweisen, manipulative und symbolische Gesten, Gesten der Zeichen- und Gebärdensprache und sprachbegleitende Gesten.
Außerdem wurden die Phasen der Gestenausführung geschildert. Von diesen ist dabei nur die Phase der Gestenbewegung obligatorisch, während die Ruhe- und Vorbereitungsphase sowie das bewegungslose Halten und die Rücknahme- bzw. Erholungsphase optional sind. Des Weiteren wurden die zur Detektion der Freihandgesten zur Verfügung stehenden Typen von Eingabegeräten vorgestellt. Diese können entweder auf sichtbasierten oder nicht-sichtbasierten Detektionsverfahren beruhen. Erstere werden sowohl beim optischen Tracking mit Markern als auch von Kameras zur Extraktion von Bildmerkmalen und von Tiefenkameras wie der Kinect angewandt. Währenddessen sind Datenhandschuhe, myoelektrische Eingabegeräte und das magnetische Tracking nicht auf eine direkte Sichtverbindung zwischen Hand und Gerät angewiesen.

Darüber hinaus wurden anhand einer systematischen Literaturrecherche verwandte Arbeiten ermittelt und sich anschließend mit diesen auseinandergesetzt. Dabei stellte sich heraus, dass bereits Arbeiten existieren, die Vermessungsinteraktionen mithilfe von Gesten bzw. Freihandgesten umsetzen. Außerdem beschäftigen sich zahlreiche Untersuchungen mit dem Zeigen und Selektieren sowie der Navigation in medizinischen Bilddaten mit Freihandgesten. Auch sind einige Arbeiten zum Thema der Optimierung der Vermessungsvisualisierungen und manuellen sowie automatischen Vermessungsinteraktionen mithilfe von herkömmlichen Eingabegeräten vorhanden.
Anhand der dabei gewonnen Erkenntnisse konnte die Arbeit schließlich in den aktuellen Stand der Wissenschaft eingeordnet werden. So soll die Lücke bezüglich angemessener, steriler Interaktionskonzepte zur Vermessung in medizinischen Bilddaten mithilfe der Freihandgesten geschlossen und anhand von Mechanismen zur Nutzerunterstützung die Genauigkeit der Gesteneingaben erhöht werden. Des Weiteren wird eine Evaluierung der entwickelten Interaktionstechniken, welche in vielen der verwandten Arbeiten noch aussteht, stattfinden.

3 Anforderungsanalyse

Die Grundlage für die Entwicklung der verschiedenen Interaktionskonzepte bildet die Anforderungsanalyse. Diese basiert auf den Erkenntnissen früherer Untersuchungen zu verwandten Themen, die im Rahmen der strukturierten Literaturrecherche gewonnen werden konnten, sowie Beobachtungen und Interviews am Institut für Neuroradiologie des Universitätsklinikums Magdeburg. Darüber hinaus diente eine Recherche im Bereich der interaktiven Systeme und der Medizin zur Bildung des notwendigen Hintergrundwissens. Die medizinischen Grundlagen unterstützten außerdem die Vorbereitung auf die anschließenden Beobachtungen im klinischen Arbeitsumfeld. Hinzu kommt, dass während der Hospitation nur einer von vielen Fachbereichen abgedeckt werden konnte und daher die Erkenntnisse aus der Literaturrecherche zur Definition allgemeinerer Anforderungen benötigt wurden. Auf diese Weise lassen sich sowohl die Anforderungen an die Interaktion mit Freihandgesten als auch die Bedürfnisse der Ärzte im Rahmen der Vermessung medizinischer Strukturen auf 2D- und 3D-Bilddaten während der Planung und Durchführung einer Intervention identifizieren.
Dieser Prozess wird im Folgenden nachvollzogen. So sollen zunächst die Vorgehensweisen und Ergebnisse der Hospitation am Universitätsklinikum Magdeburg genauer erläutert werden. Diese fließen anschließend zusammen mit den Hintergrundinformationen aus den Recherchen in die Ableitung der nicht-funktionalen und funktionalen Anforderungen ein.

3.1 Beobachtungen

Um neben den theoretischen Erkenntnissen der Literaturrecherche auch einen Einblick in das praktische Umfeld der medizinischen Vermessung auf 2D- und 3D-Bilddaten zu erhalten, wurden mit den Mitarbeitern des Instituts für Neuroradiologie des Universitätsklinikums Magdeburg Termine vereinbart, die dazu dienten bei Interventionen zu hospitieren und ausgewählte Fragen an die beteiligten Mediziner zu stellen. Die wichtigsten Ergebnisse dieser Befragungen und Beobachtungen werden nachfolgend beschrieben.

3.1.1 Planung und Durchführung

In Vorbereitung auf die Befragungen und zur Orientierung während der Hospitation am Institut für Neuroradiologie wurde ein Fragebogen (siehe Anhang A) entworfen. Dieser sollte klären, welche Messungen zum Einsatz kommen und von wem sie unter welchen Umständen durchgeführt werden. Außerdem galt es aktuelle Probleme bei der Vermessung sowie die Voraussetzungen für eine Interaktion mit Freihandgesten zu erfassen. Die Interviews mit den zwei Mitarbeitern des Instituts erfolgten dabei zwischen der Teilnahme an den Interventionen, bei denen es sich um zwei Behandlungen eines Aneurysmas durch das sogenannte „Auscoilen“mit Platindrähten handelt. Während der Befragungen konnten nach der Einverständniserklärung der Mediziner außerdem Tonaufnahmen angefertigt und zu einem späteren Zeitpunkt ausgewertet werden.
Über die Interviews und die Hospitationen hinaus konnte eine existierende Software zur Vermessung außerhalb des Operationssaals betrachtet und die Probleme der Mediziner mit dieser in Erfahrung gebracht werden.

3.1.2 Beobachtete Messungen

Im Zuge der Befragungen sowie der Hospitation bei der Behandlung der Aneurysmen konnten Erkenntnisse über verschiedene Arten der Vermessungen gewonnen werden. Diese sollen im Folgenden näher erläutert werden.

Distanzmessung

Während den Interventionen selbst kam es ausschließlich zur Messung von Distanzen zwischen zwei Punkten. Auf diese Weise wurde der vertikale und horizontale Durchmesser eines Aneurysmas ermittelt, um abzuschätzen, welche Coils zur Behandlung zum Einsatz kommen sollen. Dabei erfolgte die Vermessung stets in den 3D-Bilddaten. Ob in der 2D- oder 3D-Ansicht gemessen wird, ist jedoch nach Aussage des operierenden Arztes auch von den Präferenzen des jeweiligen Mediziners abhängig. Im Gegensatz dazu deuten die Antworten in den Interviews darauf hin, dass die Messung in einer 2D-Ansicht bevorzugt wird, da es während der 3D-Verarbeitung zu Problemen kommen kann (z. B. durch die Fensterung).

Winkelmessung

Im Zuge der Interviews wurde auch der Bedarf der Messung von Winkeln bei Hüft- und Knieoperationen in der Orthopädie angesprochen. So wird beispielsweise im Falle eines

Oberschenkelbruchs das räumliche Verhältnis des teilweise zu ersetzenden Oberschenkelknochens zum Hüfknochen gemessen, um beim Einsatz der Teilprothese genau diesen Winkel wiederherzustellen. Dies ist zur Vermeidung späterer Probleme, wie z. B. Rotationsfehlstellungen, wichtig. Doch laut der Aussage eines Arztes aus dem entsprechenden Fachbereich wird die Messung momentan nicht mithilfe einer Software durchgeführt.

Flächenmessung und Grauwertanalyse

Weitere Möglichkeiten der Vermessung wurden von der Software außerhalb des Operationssaals angeboten. So können verschiedene geometrische Formen wie Kreise, Rechtecke und Polygonzüge auf den 2D-Bilddaten erstellt werden, um sowohl die Fläche als auch die Grauwertinformationen der ausgewählten Region zu ermitteln.

Volumenmessung

Eher am Rande erwähnt blieb die Vermessung des Volumens. Diese ist zwar auch in der medizinischen Software vor Ort in Form anpassbarer Ellipsenformen umgesetzt, wird jedoch selten bis nie während eines Eingriffs genutzt. Ein möglicher Grund hierfür wurde im Interview benannt. So werden für die Behandlung von Aneurysmen nur sehr erfahrene Ärzte eingesetzt, welche sich allein anhand der verschiedenen Distanzmessungen in den 2D-Ansichten ein Bild vom Volumen einer Struktur machen können und daher eine explizite Volumenmessung nicht benötigen.

Messung der Länge von Gefäßen

Darüber hinaus wird durch die Software außerhalb des Operationssaals auch die Messung der Länge von Gefäßen ermöglicht. Das hier verwendete Verfahren verlangt dabei das möglichst mittige Platzieren von Punkten auf der 3D-Oberfläche eines solchen Blutgefäßes. Anschließend wird dem Nutzer sowohl die Länge als auch der Querschnitt des entsprechenden Gefäßes angezeigt. Die gemachten Eingaben lassen sich darüber hinaus auch nach der Berechnung der Ergebnisse noch interaktiv verändern.

Semi-automatische Berechnung von Aneurysmen-Daten

Eine weitere Vermessungsmöglichkeit im Kontrollraum besteht darin das Volumen und die Höhe eines Aneurysmas ausgehend von seiner Basis durch die Eingabe bestimmter

Abbildung 3.1: Exemplarischer Aufbau eines Operationssaals. Quelle: Institut für Neuroradiologie am Universitätsklinikum Magdeburg

Punkte auf den Bilddaten automatisch zu berechnen. Die Anweisungen für das richtige Setzen erhält der Arzt dabei vom Programm.

3.1.3 Interaktionsumfeld

Der Arbeitsplatz im Operationssaal erwies sich als Umgebung mit beschränkten Platzverhältnissen. So befindet sich direkt vor dem Mediziner der Operationstisch, an dessen unterem Ende sich eine Steuereinheit befindet (siehe Abbildung 3.1). Mit dieser lässt sich sich sowohl die Position des Tisches und der zur Bildaufnahme verwendeten C-Arme steuern als auch die Interaktion mit den Bilddaten realisieren. Die Steuerung erfolgt hier über verschiedene mechanische Knöpfe, einen Touchscreen und mehrere Joysticks. Letztere sind teilweise mit drei Tasten am Kopf ausgestattet. Während einer Intervention sind außerdem alle Eingabegeräte mit einer sterilen Folie bedeckt, die an dem medizinischen Tuch zur Abdeckung des Patienten befestigt ist. Darüber hinaus werden bestimmte Aktionen auch über Fußpedale ausgelöst.
Des Weiteren befindet sich dem operierenden Arzt gegenüber und parallel zum Operationstisch ein großer Flachbildmonitor als Ausgabegerät, welches die Anzeige verschiedener 2D- und 3D-Bilddaten ermöglicht. Für die Vermessung erfolgte dabei stets eine Anzeige von drei 2D-Ansichten, die die gewünschte Struktur jeweils aus verschiedenen Perspektiven zeigen, und einer 3D-Ansicht des mithilfe von Kontrastmitteln sichtbar gemachten

Gefäßes. Zusätzlich wird der für Interaktionen zur Verfügung stehende Platz durch umstehende Assistenten weiter eingeschränkt, so dass sich der für Gesten zur Verfügung stehende Raum auf den Körperumfang reduziert.

3.1.4 Ablauf der Vermessung während einer Intervention

Bei einer Intervention zur Behandlung von Aneurysmen geht der Vermessung die **Exploration der Bilddaten** voraus. Sie dient vor allem dazu die richtigen Perspektiven für den nachfolgenden Vermessungsschritt zu finden. Dabei wurden die 3D-Bilddaten häufig rotiert, während eher selten bis gar nicht innerhalb der Ansicht gezoomt oder der gezeigte Bildausschnitt verschoben wurde. Als Eingabegerät diente hierbei der Joystick. Anschließend wird die **Vermessung** ebenfalls mithilfe des Joysticks durchgeführt. So kann der Mauszeiger innerhalb der Bilddaten zu einem gewünschten Punkt bewegt und dieser durch einen Klick auf die linke Taste am Kopf des Joysticks als Anfangspunkt der Distanzmessung festgelegt werden. Die Eingabe des Endpunktes erfolgt schließlich auf dieselbe Weise. Danach ist die Interaktion beendet. Der Vorgang kann so oft wiederholt werden, wie es zur vollständigen Bemaßung der entsprechenden Struktur notwendig ist. Der **Wechsel zwischen den einzelnen Modi** zur Interaktion innerhalb der Bilddaten erfolgt dabei über im aktuellen Modus nicht belegte Tasten am Kopf des Joysticks oder ein Drop-Down-Menü. Darüber hinaus wurde ein weiterer Joystick ohne Tasten zur **Navigation durch zeitlich aufeinander folgende Bildaufnahmen** genutzt.

Nach Aussage der Mediziner werden während der Intervention jegliche, mit der Vermessung in Zusammenhang stehenden Interaktionen vom operierenden Arzt selbst ausgeführt. Nur in Notfällen bzw. unter Zeitdruck wird die Vermessung von der Assistenz im Kontrollraum übernommen, um den Prozess zu beschleunigen. Des Weiteren hat der Operateur für die Interaktionen im Operationssaal stets beide Hände frei, da die Arbeitsgeräte während dieses Zeitraums auf einer sterilen Ablage verbleiben müssen.

3.1.5 Probleme

Sowohl während den Befragungen als auch im Rahmen der Hospitation konnten einige Probleme herausgestellt werden, die den Arbeitsfluss teilweise wesentlich beeinflussen. Diese sollen im Folgenden aufgelistet und beschrieben werden.

Behinderung durch sterile Folie

Wie bereits in früheren Abschnitten beschrieben, stellt auch im Operationssaal des Instituts für Neuroradiologie sterile Folie zur Abdeckung der nicht sterilen Eingabegeräte ein Problem für die komfortable Handhabung dar. So musste diese häufig hin und her geschoben werden, da Knicke die Anzeige auf dem Touchscreen verdeckten oder die Eingabe erschwerten.

Lage der Steuereinheit nicht zentral

Obwohl sich die Eingabegeräte direkt am Operationstisch befinden, war es häufig notwendig, dass der operierende Mediziner zu ihnen herüber läuft oder sich über den Tisch streckt, um diese zu erreichen. Dies führte in einem Fall sogar dazu, dass der gesamte Tisch versehentlich mit einem der Joysticks ruckartig verschoben wurde.

Behinderung der Handhabung durch mechanische Eingabeprobleme

Im Rahmen der Befragungen wurde schnell ersichtlich, dass die aktuelle Handhabung des Joysticks keine schnelle und präzise Vermessung ermöglicht. So dauerte das Anfahren eines bestimmten Punktes entweder durch zu zaghafte Handhabung sehr lange oder die Bewegung geschah viel zu schnell, wenn der Nutzer den Druck zu stark erhöhte. Kritisch wird dieser Umstand schließlich, wenn während einer nächtlichen Notfallsituation kein Assistent zur Verfügung steht, so dass der operierende Arzt ggf. den Operationssaal verlassen und Zeiteinbußen in Kauf nehmen muss, um eine präzise Messung im Kontrollraum vorzunehmen.
Darüber hinaus wies einer der Mediziner jedoch darauf hin, dass das Prinzip des Joysticks an sich nicht schlecht sei und ältere Geräte durch eine leichtere mechanische Handhabung deutlich besser funktioniert hätten.

Mehrere Messungen direkt hintereinander nicht möglich

Die Tatsache, dass der Vermessungsmodus nach jeder einzelnen Messung beendet wird und neu gestartet werden muss, trug zusätzlich dazu bei, dass sich die Interaktion für den operierenden Arzt umständlich gestaltete.

Teilweise zu hohe Interaktionskomplexität

Bei der Vorstellung der komplexeren Messverfahren, welche die örtliche Software zur Verfügung stellt, zeigte sich, dass diese zwar viele einzelne und einfache Messschritte ersparen, sich dabei jedoch als zu kompliziert und umfangreich für eine Anwendung im Operationssaal erweisen. Das führt schließlich dazu, dass die Mediziner lieber mehrere Einzelschritte mit einfachen Distanzmessungen durchführen, da dies als schneller und einfacher empfunden wird. Beispiele hierfür sind die Messung der Länge von Gefäßen sowie die automatische Bestimmung der Aneurysmen-Daten (siehe Abschnitt 3.1.2).

Mangel an intuitiver Handhabung

Für Vermessungen, die innerhalb von vom Nutzer festzulegenden Regionen erfolgen, stellt vor allem die wenig intuitive Vorgehensweise zur Definition des zu betrachtenden Bereiches ein Problem dar. Diese macht das betreffende Interaktionskonzept zu komplex für die Verwendung während einer Intervention. So schien den Medizinern beispielsweise das Aufziehen eines Kreises, beginnend in seiner oberen rechten Ecke, wenig sinnvoll und erforderte zu viel nachträgliche Bearbeitung des selektierten Bereiches.

Wenig Vertrauen in komplexe, semi-automatische Vermessung

Als problematisch erwiesen sich auch komplexe, semi-automatische Vermessungsmethoden, bei denen nicht klar verständlich ist, inwiefern sich die Eingaben auf die Berechnung der Messwerte auswirken. Ein Beispiel hierfür ist die semi-automatische Berechnung der Aneurysmen-Daten anhand der im Bereich der Aneurysmabasis gesetzten Punkte. So wird hier die Zuverlässigkeit der Messmethoden infrage gestellt, wodurch eine Nachkontrolle der Ergebnisse stattfindet. Aufgrund des hier entstehenden Mehraufwands wird lieber die manuelle Messung verwendet. Darüber hinaus wurden jedoch Präferenzen für die semi-automatische Vermessung durch einfaches Antippen der jeweiligen Struktur geäußert. Eine Konkretisierung in diesem Punkt erfolgte durch die Aussage, dass dabei die halbautomatische Berechnung des minimalen bzw. maximalen Gefäßdurchmessers nützlich sein könnte.

3.1.6 Wichtige Aspekte während der Vermessung

Während der Beobachtungen am Institut für Neuroradiologie stellten sich folgende Aspekte als besonders wichtig für die Vermessung medizinischer Strukturen heraus:

- Hohe Genauigkeit, d. h. Messtoleranzen zwischen einem Millimeter und einem Zehntel Millimeter
- Unterstützung der räumlichen Vorstellungskraft durch Angabe der aktuellen Bildschicht und Rotationsangaben in Grad
- Übersichtlichkeit, d. h. nicht zu viele Informationen auf einmal

3.1.7 Voraussetzungen für Freihandgesten

Bei keinem der befragten Mediziner existierten vorherige Erfahrungen mit Freihandgesten im Arbeitsalltag. Dabei ließ sich jedoch keine ablehnende Haltung der Nutzung selbiger für die Vermessung oder Steuerung bestimmter Funktionen feststellen. Bei den Befragungen erfolgte außerdem die Aussage, dass die Steuerung der Geräte und die Vermessung mithilfe von Freihandgesten durchaus vorstellbar wäre.

3.2 Nicht-funktionale Anforderungen

Im Folgenden sollen alle Anforderungen aufgelistet und erläutert werden, die die allgemeinen Qualitäten des geplanten Systems zur Vermessung mit Freihandgesten betreffen.

3.2.1 Sterilität

Wie bereits in Kapitel 1 beschrieben, handelt es sich bei der Sterilität um eine notwendige Voraussetzung innerhalb von sterilen Umgebungen wie dem Operationssaal. Aus diesem Grund darf während der Interaktion mit dem System kein Kontakt zu nicht sterilen Materialien stattfinden.

3.2.2 Einhaltung der Usability-Kriterien

Die Berücksichtigung der Usability-Kriterien stellt eine grundlegende Voraussetzung zur Sicherstellung der Benutzungsfreundlichkeit dar. Die nachfolgende Aufteilung der Kriterien ist dabei stark an der Arbeit von Manresa-Yee et al. [34] orientiert, da diese einen aktuellen Ansatz zur Herausarbeitung spezieller Usability-Kriterien für die Interaktion mit

Freihandgesten durch sichtbasierte Gestenerkennung darstellt. Den fundierten Ausgangspunkt bilden dabei die durch die ISO-Norm 9241-11 festgelegten „Anforderungen an die Gebrauchstauglichkeit“, welche die Effektivität, Effizienz und Zufriedenheit umfassen. Durch diese Strukturierung werden außerdem die in den von Jacob Nielsen definierten, allgemeinen Usability-Kriterien [35] vorkommenden Redundanzen weitestgehend beseitigt. So sorgen beispielsweise sowohl schnelle Erlernbarkeit als auch die Einprägsamkeit von Freihandgesten für eine höhere Effizienz. Von Nielsen werden diese Eigenschaften jedoch getrennt behandelt.

Effektivität

Für die Handhabung einer Software ist das Erreichen der Interaktionsziele von essenzieller Bedeutung. Daher ist es wichtig, dass die Eingaben des Nutzers richtig erkannt werden. Wird eine Geste falsch oder gar nicht erfasst, führt dies zu unerwünschten Fehlern [34]. Daher sollte bei der Umsetzung der Gestensteuerung eine gewisse Fehlertoleranz eingeplant werden. So dürfen beispielsweise kleinere Fehler in der Ausführung nicht sofort zu einer fehlerhaften Eingabe führen [11]. Andererseits sollten die Toleranzen in der Gestenerkennung nicht zu groß werden, da sonst eine erhöhte Anfälligkeit gegenüber falschen Detektionen besteht.
Des Weiteren sind auch die übrigen Aspekte der Bedienung so zu gestalten, dass möglichst wenig Fehler auftreten. Sollte es dennoch zu Fehlern kommen, müssen diese leicht zu beheben sein. [11]

Effizienz

Die in Abschnitt 3.1.5 beschriebenen, durch ineffiziente Interaktionsmöglichkeiten hervorgerufenen Probleme in medizinischen Notfallsituationen zeigen, wie wichtig eine effiziente Arbeitsweise ist. Doch auch unter normalen Umständen ist es stets von Vorteil, wenn die nötigen Messungen schnell und präzise durchgeführt werden können. Dabei hängt die Effizienz eines Interaktionskonzepts von verschiedenen Einzelfaktoren ab, welche nachfolgend beschrieben werden sollen.

Vermeiden von Ermüdung Je mehr Erschöpfung durch die Ausführung einer Handgeste hervorgerufen wird, desto weniger Eingaben können vom Nutzer gemacht werden. Dies führt dazu, dass seine Effizienz sinkt [34]. Darüber hinaus wäre eine zu große Ermüdung nach der Vermessung auch für den weiteren Verlauf einer medizinischen Intervention, in der den Händen ein gewisses Feingefühl abverlangt wird, sehr problematisch. Aus diesen Gründen sollten Gestenkommandos präzise sowie schnell auszuführen sein und nicht über einen langen Zeitraum eine hohe Genauigkeit verlangen [34].

Geringer kognitiver Aufwand Zur Schonung des Arbeitsgedächtnisses des Nutzers ist es wünschenswert, dass dieser während der Arbeit mit dem System bezüglich der zu merkenden Informationen unterstützt wird [34]. Andernfalls kann es zur Überforderung kommen.

Schnelle Erlernbarkeit und Einprägsamkeit Während eine schnelle Erlernbarkeit der Freihandgestensteuerung dafür sorgt, dass das System innerhalb weniger Zeit effizient genutzt werden kann, stellt eine gute Einprägsamkeit sicher, dass auch nach längerer Abwesenheit rasch wieder eine gewisse Effizienz erreicht werden kann [35]. So sollten die entwickelten Gesten im Optimalfall sowohl leicht zu erlernen als auch leicht zu merken sein.

Nutzerzufriedenheit

Ein weiteres wichtiges Kriterium ist die Zufriedenheit des Nutzers mit dem System, da ein angenehmes Nutzungserlebnis dazu führt, dass das System gerne wieder verwendet wird [35]. Auch hier sollen im Folgenden verschiedene Einzelkriterien erläutert werden.

Natürliche und intuitive Gestaltung Die Bedienung mit den entwickelten Freihandgesten sollte so intuitiv und natürlich wie möglich sein. Diese Eigenschaften machen das System für den Nutzer leichter zugänglich und beeinflussen Erlernbarkeit, Einprägsamkeit sowie den kognitiven Aufwand positiv [34], so dass auch die Effizienz gesteigert wird.
Eine kritische Eigenschaft von Freihandgesten bezüglich der intuitiven Handhabung besteht dabei darin, dass ihnen der sichtbare Aufforderungscharakter (engl. *affordance*) fehlt, den beispielsweise Buttons mit einem aussagekräftigen Icon besitzen [11]. Daher weiß der Nutzer nicht immer intuitiv, was er mit dem System tun kann und wie dies zu bewerkstelligen ist. Aus diesem Grund sollte ein Weg gefunden werden diesen Zustand auszugleichen bzw. trotzdem einen Aufforderungscharakter zu erzeugen.

Komfortable und einfache Nutzung Die Nutzung des zu entwickelnden Systems sollte weder aufwendig sein noch sollte die Ausführung der Gesten dem Nutzer Schmerzen oder Unbehagen bereiten [34]. Letzteres wäre beispielsweise der Fall, wenn ausgefallene motorische Fähigkeiten verlangt werden würden.

Soziale Akzeptanz Die genutzten Freihandgesten sollten sowohl vom Anwender als auch von den anderen anwesenden Personen als angemessen empfunden werden [34]. Dies ist relevant, da die soziale Akzeptanz einen wichtigen Faktor für die Bereitschaft zur Ausführung einer Geste darstellt. Darüber hinaus entscheidet auch die Arbeitsumgebung darüber, was als angebracht empfunden wird und was nicht

[11]. So benötigt der Operationssaal vermutlich weniger diskrete Eingabemethoden als die Verwendung eines Handys im öffentlichen Raum.

3.2.3 Sichtbarkeit des Systemzustandes und angemessenes Feedback

Zahlreiche Quellen weisen darauf hin, dass Informationen zur aktuell im System aktiven Steuerungsfunktion für die Bedienung mithilfe von Freihandgesten eine große Bedeutung besitzen [6, 8, 11, 36, 37]. Daher sollte das Feedback zur Gestensteuerung nicht nur nach der Eingabe, sondern auch währenddessen erfolgen, um Vertrauen und Bediensicherheit bezüglich der Ausführung und Erkennung der Freihandgesten zu steigern [11]. In diesem Kontext betonen O'Hara et al. [37] beispielsweise die Wichtigkeit des sichtbar Machens des vom System Gesehenen für den Mediziner, um diesem zu ermöglichen seine Haltung entsprechend anzupassen.

3.2.4 Genauigkeit

Anhand der vorgelagerten Beobachtungen am Universitätsklinikum Magdeburg ließ sich der Bedarf an hoher Genauigkeit bei der Vermessung ermitteln. Dieser liegt darin begründet, dass Messfehler bereits ab einer Abweichung von einem Millimeter im medizinischen Kontext aufgrund möglicher schwerwiegender Folgen kritisch sein können. Dies stellt insbesondere für die Interaktion mit Freihandgesten eine Herausforderung dar, da die Präzision von Zeigegesten bisher eher gering [10] und daher eine Erhöhung der Genauigkeit erforderlich ist [38].

3.2.5 Einfachheit

Während der Befragungen der Mediziner stellte sich heraus, dass eine zu hohe Komplexität der Interaktionen sowie eine Überladung mit Informationen dazu führen, dass die betreffende Funktionalität im Operationssaal nicht genutzt wird. Daher sind diese Eigenschaften zu vermeiden und Kompromisse zwischen Einfachheit bzw. schneller Erlernbarkeit und Komplexität bzw. Kontrolle zu finden [11].

3.2.6 Begrenzung des benötigten Raums

Um die Anwendbarkeit der entwickelten Interaktionskonzepte zu gewährleisten, gilt es die in Abschnitt 3.1.3 beschriebenen und auch von O'Hara et al. [37] beobachteten, eingeschränkten räumlichen Verhältnisse im Operationssaal während der Konzeption zu beachten.

3.2.7 Flexibilität

Da verschiedene Nutzer unterschiedliche Vorgehensweisen zur Lösung eines Problems bzw. zur Ausführung einer Aufgabe verwenden, muss die Bedienung einer Software flexibel gestaltet werden. Auf diese Art und Weise ist es möglich, dass eine größere Bandbreite von Anwendern ein System sinnvoll nutzen kann [39].

3.3 Funktionale Anforderungen

Nachdem im vergangenen Abschnitt die zu erfüllenden, nicht-funktionalen Qualitäten beschrieben wurden, sollen nachfolgend die von den Interaktionskonzepten abzudeckenden funktionalen Aufgaben näher erläutert werden.

3.3.1 Grundfunktionalitäten

Damit die Interaktion mit den Bilddaten überhaupt möglich wird, gilt es zunächst einige Grundfunktionen umzusetzen. So steht am Anfang jeder Vermessung das **Laden der Bilddaten**, wobei diese in einem Dialog ausgewählt und schließlich bestätigt werden müssen. Da die Aufnahmen meist im DICOM-Format vorliegen, steht eine Vielzahl von Zusatzinformationen zur Verfügung. Diese spielen für die Diagnose und computergestützte Analyse eine Rolle [42]. Während der Befragungen zur Vorbereitung der Anforderungsanalyse wurde in diesem Rahmen außerdem die Relevanz der Unterstützung der räumlichen Vorstellungskraft durch entsprechende Informationen zur Aufnahme der Bilddaten hervorgehoben, welche ebenfalls vom DICOM-Format zur Verfügung gestellt werden. Daher sollte die **Bereitstellung der DICOM-Zusatzinformationen** erfolgen.

3.3.2 Navigation in den Bilddaten

Um anschließend Messungen durchführen zu können, ist es notwendig innerhalb der umfangreichen Bilddaten die Möglichkeit zur Navigation zu besitzen. Zu diesem Zweck sind folgende Funktionen umzusetzen:

- Verschieben des sichtbaren Bildausschnitts (auch *Panning* genannt)
- Heran- bzw. Herauszoomen
- Rotation innerhalb der 3D-Ansicht
- „Blättern“ durch die einzelnen 2D-Schichtbilder oder Zeitstufen einer Ansicht

3.3.3 Selektion von Objekten

Bei der Selektion von Objekten auf der Benutzungsoberfläche, welche sowohl Steuerungselemente als auch die Anzeige der Bilddaten umfasst, handelt es sich um einen essenziellen Vorgang. Durch ihn werden Funktionalitäten wie das Erstellen, Bearbeiten und Löschen von Messungen bzw. Messpunkten oder das Aktivieren von Funktionen auf der Oberfläche überhaupt erst ermöglicht. Aus diesem Grund ist auch die Selektion in der Entwicklung der Interaktionskonzepte zu berücksichtigen.

3.3.4 Vermessungsaufgaben

Ausgehend von den Beobachtungen am Universitätsklinikum Magdeburg und den medizinischen Hintergrundrecherchen lässt sich ermitteln, welche räumlichen Maße für die Vermessung medizinischer Strukturen in 2D- und 3D-Bilddaten benötigt werden. Diese sind im Folgenden beschrieben.

Distanzmessung

Der Anwendungsbereich von Distanzmessungen ist vielfältig. So werden sie nicht nur gebraucht, um den Durchmesser von Aneurysmen zu bestimmen, sondern auch um das Stadium und mögliche Therapien eines Tumors festlegen zu können. Letzteres geschieht dabei unter anderem über die Bestimmung des Abstandes zwischen krankhaftem und gesunden Gewebe [30]. Daher soll es dem Nutzer möglich sein einen Start- und einen

Endpunkt innerhalb der Bilddaten zu definieren, zwischen denen schließlich der entsprechende Abstand vom System berechnet wird.

Winkelmessung

Mithilfe der entwickelten Interaktionskonzepte soll der Anwender Eingaben tätigen können, die es der Software ermöglichen den Winkel zwischen zwei Strukturen zu berechnen. Dies ist wichtig, da Winkelmessungen zur Feststellung des Behandlungsbedarfes von Fehlstellungen der Knochen und Zähne beitragen [3]. Außerdem wird diese Art der Messung beim Einsetzen von Prothesen gebraucht, da hier bestimmte räumliche Verhältnisse beachtet und eingehalten werden müssen. Ein Beispiel hierfür ist die in Abschnitt 3.1.2 beschriebene, Teilprothese beim Bruch des Oberschenkelknochens.

Flächenmessung

Die medizinischen Hintergrundrecherchen zeigen, dass die Vermessung des Gefäßdurchmessers nicht immer die beste Alternative für die Diagnose einer Stenose darstellt. So ist eine Unterschätzung des Stenosegrades möglich. Dies kann jedoch mithilfe einer Flächenvermessung des Gefäßquerschnitts verhindert werden [40]. Aus diesem Grund soll die Festlegung einer Region zur Flächenberechnung Teil der Funktionen des Systems sein.

Volumenmessung

Auch die Messung des Volumens bestimmter Strukturen spielt in der Medizin eine Rolle. So bestimmt die Größe eines Tumors dessen Stadium und Behandlungsmöglichkeiten [3]. Darüber hinaus ist die Volumenberechnung auch bei der Teilspende von Organen wichtig, da sich auf diese Weise feststellen lässt, welches Volumen gespendet werden kann [2]. Ähnliche Messungen erfolgen außerdem zur Bestimmung der verbleibenden Leber nach einer entsprechenden Operation [32]. Diese Umstände führten zu dem Entschluss auch die Bestimmung des Volumens in das System mitaufzunehmen.

Graustufenanalyse

Eine häufig in Zusammenhang mit ROIs durchgeführte Untersuchung stellt die Graustufenanalyse dar. Sie wird beispielsweise genutzt, um eine frühe Erkennung von Abstoßungsreaktionen nach einer Transplantation zu ermöglichen. Dies geschieht über die

Diagnose von Entzündungen, Proteinablagerungen und Umbauprozessen des Bindegewebes anhand der Grauwerte der Bilddaten. Außerdem lässt sich mithilfe des mittleren, maximalen und minimalen Grauwertes sowie der Grauwertverteilung auch gesundes von verfettetem Lebergewebe gut unterscheiden [41]. Darüber hinaus ist die Varianz der Grauwerte ein typischer Kennwert für ROIs [42]. Aus diesen Gründen sollte der Nutzer im Rahmen der Untersuchung medizinisch relevanter Bereiche folgende Informationen in Bezug auf die Grauwerte erhalten:

- Minimaler und maximaler Wert
- Mittelwert
- Varianz
- Grauwertverteilung in Form eines Histogramms

Die Regionen sollen dabei mithilfe einer rechteckigen, kreisförmigen oder polygonzugbasierten Auswahl markiert werden können. Diese Vorgehensweise stellt eine bekannte Funktionalität in radiologischer Software dar [42] und scheint daher geeignet. Außerdem ist diese Funktion auch im System des Instituts für Neuroradiologie zu finden.

Semi-automatische Berechnung von Abständen und Ausdehnungen

In der Medizin gibt es Situationen, in denen ein **minimaler bzw. maximaler Abstand zwischen zwei Strukturen** bestimmt werden muss. Beispiele hierfür sind die Messung des minimalen Abstands zwischen einem Tumor und den Blutgefäßen vor einer Intervention [43] sowie die Ermittlung des maximalen Durchmessers der aufsteigenden Aorta als Prädiktor für Komplikationen bei der Behandlung von Aneurysmen [44]. Darüber hinaus besitzt auch die **maximale Ausdehnung** einer medizinischen Struktur eine nicht zu unterschätzende Bedeutung. So stellt die maximale Ausdehnung eines Tumors einen weiteren Faktor für dessen Einschätzung und Behandlung dar [3].
Diese Größen manuell zu ermitteln, wäre ausgesprochen zeitaufwendig, da sehr viele Messungen notwendig sind, um sicher das geringste bzw. größte Maß bestimmen zu können. Daher soll eine Möglichkeit gefunden werden diese semi-automatisch mithilfe von Nutzereingaben zu berechnen. Dabei ist sicherzustellen, dass diese Messungen stets korrekt und zuverlässig erfolgen, da sich bereits in Abschnitt 3.1.5 herausstellte, dass nicht vertrauenswürdige automatische Messungen in der Praxis nicht verwendet werden. Die Begründung hierfür besteht darin, dass bereits kleinste Fehler schwerwiegende Folgen haben können.

3.3.5 Messungen auf 2D-Bild- und 3D-Planungsdaten

Es ist darauf zu achten, dass sowohl 2D-Bilddaten als auch 3D-Planungsdaten für die Vermessungen zur Verfügung stehen. Dies liegt zum einen darin begründet, dass jede der beiden Ansichten für sich bestimmte Ungenauigkeiten aufweist. Zum anderen lassen sich unterschiedliche Maße unterschiedlich gut innerhalb der verschiedenen Ansichten messen. So kann beispielsweise die Berechnung der dreidimensionalen Ausdehnung eines Objektes in einer zweidimensionalen Ansicht nur eingeschränkt und eher grob erfolgen [28], während sich eine Graustufenanalyse nur auf den 2D-Schichtdaten ergibt. Zu diesem Zweck soll die Gesamtansicht in folgende verschiedene Einzelansichten unterteilt werden:

- 3D-Ansicht des Planungsmodells
- Zweidimensionale Schichtdaten aus der ...
  - Seitenansicht (Sagittalebene)
  - Draufsicht (Transversalebene)
  - Frontalansicht (Frontalebene)

Diese Art der Unterteilung scheint sinnvoll, da sie auch am Institut für Neuroradiologie des Universitätsklinikums Magdeburg beobachtet worden ist. Außerdem kann auf diese Weise den beschriebenen, unterschiedlichen Präferenzen der einzelnen Ärzte bezüglich der Vermessung in 2D oder 3D nachgekommen werden. Darüber hinaus bestätigen Preim et al. [29], dass eine Kombination aus 2D- und 3D-Ansichten wünschenswert ist, da sich die dreidimensionale Darstellung aus einer Kombination der 2D-Schichten ergibt und zusätzlich zur Orientierung genutzt werden kann.

3.3.6 Bearbeiten und Löschen von Messungen und Messpunkten

Neben der Erstellung von Messungen ist es notwendig dem Nutzer die Überarbeitung seiner Eingaben bzw. das Löschen der gesamten Messung zu ermöglichen. Auf diese Weise wird ein ausreichendes Maß an Flexibilität und Fehlertoleranz gewährleistet sowie eine Möglichkeit zur Feinjustierung geboten.

3.3.7 Visualisierung

Im Anschluss an die Eingaben des Nutzers und die darauffolgende Berechnung des Messwertes sollen die entsprechenden Ergebnisse innerhalb der Bilddaten visualisiert werden. Dies bietet sich an, da sich so die Wirkung des Gesetzes der Nähe [45] gut ausnutzen lässt. Dabei sind folgende Vorgaben einzuhalten:

- Die zum aktuellen Kontext gehörigen Bemaßungen müssen jederzeit lesbar sein [30].
- Die Maßzahlen sollen eindeutig zuordenbar sein [1, 28].
- Es dürfen keine auffälligen, schnellen Änderung in der Visualisierung der Bemaßungen erfolgen [1].

3.3.8 Hilfswerkzeuge zur korrekten Platzierung von Messpunkten

Die Fähigkeit Messpunkte auf den Bilddaten präzise platzieren zu können, ist für die Genauigkeit der Vermessung von großer Bedeutung [30]. Gleichzeitig gestaltet sich eine exakte Positionierung mit Freihandgesten schwierig, da die Hände nie gänzlich ruhig gehalten werden können [31]. Aus diesem Grund ist es sinnvoll dem Nutzer Eingabehilfen bereitzustellen, so dass exakt ruhige Hände gar nicht notwendig sind.

3.4 Zusammenfassung

In diesem Kapitel erfolgte die für den Entwurf der Interaktionskonzepte grundlegende Anforderungsanalyse. Dabei wurden zunächst die im Rahmen zweier medizinischer Interventionen und Interviews am Institut für Neuroradiologie des Universitätsklinikums Magdeburg gemachten Beobachtungen geschildert. Diese gaben nicht nur Aufschluss über die verwendeten Messungen und deren Ablauf während einer Intervention, sondern auch über das Interaktionsumfeld und die Probleme mit den bisher verwendeten Interaktionskonzepten. Des Weiteren konnten wichtige Aspekte für die Vermessung sowie die Voraussetzungen für deren Durchführung mithilfe von Freihandgesten herausgestellt werden.
Aufbauend auf den so gewonnenen Erkenntnissen wurden schließlich Anforderungen erarbeitet. Die nicht-funktionalen Anforderungen umfassen dabei Qualitätskriterien, die

vom System erfüllt werden müssen und sind in der folgenden Tabelle noch einmal kurz zusammengefasst.

Anforderung	Beschreibung
Sterilität	Während der Interaktion darf kein Kontakt zu nicht sterilen Materialien stattfinden.
Effektivität	Fehler sollten selten auftreten und leicht zu beheben sein.
Effizienz	Die Vermessungen müssen mithilfe des Systems schnell zu erledigen sein.
Nutzerzufriedenheit	Die Nutzung des Systems muss angenehm und einfach sein.
Feedback	Der Systemzustand muss stets durch angemessenes Feedback sichtbar gemacht werden.
Genauigkeit	Die Vermessung medizinischer Strukturen erfordert eine sehr hohe Präzision.
Einfachheit	Die Interaktionen dürfen keine zu hohe Komplexität aufweisen.
Platzsparend	Der zur Ausführung der Gesten benötigte Platz ist an den eingeschränkten Interaktionsraum anzupassen.
Flexibilität	Die Interaktionsmöglichkeiten sind so zu gestalten, dass mehrere Wege zur Lösung einer Aufgabe führen.

Die funktionalen Anforderungen hingegen betreffen die Funktionalitäten, die von den Interaktionskonzepten abzudecken sind. Nachfolgend sollen auch diese kurz zusammengefasst werden.

Anforderung	Beschreibung
Darstellung der Bilddaten	Die Bilddaten sind zusammen mit Patienten- und Aufnahmeinformationen in 2D und 3D darzustellen.
Navigation	Der Nutzer muss sich innerhalb der Bilddaten bewegen können.

Selektion	Interaktionsmöglichkeiten zur Selektion von Strukturen und Objekten sind bereitzustellen.
Vermessung	Es müssen Distanzen, Winkel, Flächen und Volumen messbar sowie eine Grauwertanalyse durchführbar sein.
Visualisierung	Die Vermessungsergebnisse sind innerhalb der Bilddaten entsprechend zu visualisieren.
Bearbeitungsmöglichkeiten	Messungen müssen auch bearbeitet und gelöscht werden können.
Hilfswerkzeuge	Zur Erhöhung der Genauigkeit der Eingabe durch Freihandgesten sind entsprechende Mechanismen bereitzustellen.

4 Konzept

Nachdem im vorangegangenen Kapitel die Anforderungen an die Interaktionstechniken zur Vermessung medizinischer Strukturen definiert wurden, widmet sich dieses Kapitel nun der Entwicklung eines entsprechenden Konzeptes. Wie schon bei der Herausarbeitung der Anforderungen sollen dabei Fortschritte und Erfahrungen aus früheren Arbeiten verwandter Themenbereiche berücksichtigt werden, um auf diesen aufbauen zu können. So werden zunächst allgemeine Entwurfsgrundsätze festgelegt, die während des gesamten Entwurfs Anwendung finden sollen. Anschließend erfolgt die Beschreibung der Konzepte für die Benutzungsoberfläche sowie das visuelle Feedback. Nachdem auf diese Weise der Grundstein für die Anwendung der Gesteninteraktion gelegt wurde, soll der Entwurfsprozess selbiger erläutert werden. Den Abschluss bilden schließlich die Konzeptionierung der Mechanismen zur Nutzerunterstützung sowie die Auswahl eines entsprechenden Eingabegerätes zur optischen Gestenerkennung.

4.1 Allgemeine Entwurfsgrundsätze

Im Folgenden werden allgemeingültige Grundsätze für die Gestaltung von Freihandgesten definiert. Sie sollen eine Grundlage für den folgenden Entwurfsprozess schaffen und die allgemeine Qualität der gestenbasierten Interaktionstechniken sicherstellen.

4.1.1 Vermeiden einer Kommandosprache

Soweit wie möglich sollte auf die Verwendung symbolischer Gesten verzichtet werden, da dies zur Entwicklung einer Kommandosprache führen kann. Die Folge wäre, dass sehr viele Befehle gelernt und behalten werden müssen, was in einem hohen Lernaufwand und einem schweren Wiedereinstieg resultiert [46, 47]. Außerdem variiert die Bedeutung symbolischer Gesten je nach Nutzer, Kontext und Kultur, was die Wahl geeigneter Gesten in diesem Bereich zusätzlich erschwert [48].
Stattdessen ist es empfehlenswert die durch die Geste herbeigeführte Manipulation entsprechend abzubilden, da die gestische Manipulation eine Art physische, natürliche Sprache darstellt, die alle Menschen teilen [48]. Lässt sich der Einsatz einer symbolischen

Geste nicht vermeiden, ist der Nutzer durch visuelles Feedback und gute Affordances zu unterstützen.

4.1.2 Vermeiden statischer Gesten

Die Verwendung statischer Gesten ist ebenfalls zu vermeiden. Der Grund für die geringe Eignung dieser Art von Geste ist, dass sie mehr Training sowie Konzentration erfordert [49]. Stattdessen sollte auf dynamische Gesten zurückgegriffen werden, da frühere Untersuchungen zeigen, dass diese besser geeignet sind [49, 50].

4.1.3 Bevorzugung bekannter Interaktionskonzepte

Wenn möglich, sollte auf bekannte Interaktionstechniken zurückgegriffen bzw. versucht werden diese auf die gestenbasierten Interaktionen zu übertragen, da Gesten, die unter gewohnten Bedingungen ablaufen, als leichter und intuitiver wahrgenommen werden [50]. Der Grund hierfür ist, dass Menschen von existierenden Interaktionskonzepten in ihrem Denken beeinflusst werden [51, 52].

4.1.4 Ausführung der Gesten aus egozentrischer Perspektive

Beim Entwurf der Gesten ist darauf zu achten, dass deren Ausführung aus der Perspektive des Nutzers erfolgt. Dies verbessert die Geschicklichkeit während der gestischen Interaktion, da der Nutzer seinen Handlungsrahmen nicht auf eine andere Perspektive umorientieren muss. Belegt wird dies durch Studien, in denen festgestellt werden konnte, dass dem Menschen das Nachahmen von Tätigkeiten aus der eigenen Perspektive leichter fällt, als wenn sie aus der Perspektive eines anderen zeigen müssen, was zu tun ist [50].

4.1.5 Bevorzugung beidhändiger Interaktion

Soweit wie möglich sollte während der Konzeption der gestenbasierten Interaktionstechniken auf beidhändige Freihandgesten zurückgegriffen werden. Dies hat vielfältige Gründe. So ermöglichen beidhändige Gesten sowohl ein effizienteres Arbeiten [53] als auch eine bessere Wahrnehmung des realen Raumes, so dass der Nutzer die Bewegung seiner Hände nicht kontinuierlich überwachen muss [50, 54, 55]. Auf diese Weise lässt sich Zeit sparen und der kognitive Aufwand reduzieren [54, 56]. Außerdem kann die Arbeitslast

auf beide Hände verteilt werden. Auch widerspricht eine beidhändige Vorgehensweise nicht den Bedingungen im klinischen Kontext, da sowohl eigene als auch frühere Beobachtungen zeigen, dass es eher unüblich ist während der Interaktion mit den Bilddaten medizinische Instrumente in der Hand zu halten [6, 37]. Darüber hinaus empfiehlt es sich bei der Verwendung beidhändiger Freihandgesten die nicht-dominante Hand als Referenz für die Bewegungen der dominanten Hand zu nutzen [50, 57]. Auf diese Weise kann der natürlichen Vorgehensweise des Menschen für die Arbeit mit zwei Händen entsprochen werden [57].
Es existieren jedoch auch Einschränkungen für die beidhändige Interaktion. So eignet sie sich nicht in Situationen, wo durch sie eine geteilte Aufmerksamkeit provoziert wird oder der Nutzer sich die Durchführung einer Tätigkeit mit zwei Händen nicht vorstellen kann [58]. Dies wäre beispielsweise bei der Erledigung von zwei voneinander unabhängigen Aufgaben der Fall [54].

4.1.6 Einhaltung ergonomischer Kriterien

Es ist stets darauf zu achten, dass die entwickelten Freihandgesten komfortabel und leicht auszuführen sind. Außerdem dürfen sie keine Schmerzen oder ähnliche Unannehmlichkeiten verursachen. Aus diesem Grund sind für die Gesteninteraktion folgende Verhaltensweisen bzw. Körperhaltungen anzustreben [59, 60, 61, 62]:

- Entspannung der Muskeln zwischen den Bewegungen
- Agieren jeder Hand auf der jeweils eigenen Körperhälfte
- Halten der Hände nahe der Körpermitte
- Positionierung der Ellenbogen in Höhe des Bauches und des unteren Brustkorbs

Darüber hinaus sollen nachfolgende Positionen vermieden werden [59, 60, 62]:

- Störung des Blutflusses
- Straff ausgestreckte Finger oder Arme
- Wiederholte vollständige Rotationen oder Beugungen der Handgelenke
- Widersprechende Bewegungen benachbarter Finger

4.2 Interface-Konzept

Im Folgenden soll kurz das Konzept der Benutzungsoberfläche erläutert werden. Abbildung 4.1 zeigt daher den Prototypen des Interfaces. [1] [2] [3]
Dabei ist eine Teilung der Benutzungsoberfläche in drei Teile zu erkennen. So befinden sich auf der linken Seite die notwendigen Schaltflächen, deren Funktionen folgende sind (beginnend von links oben):

- Menüschaltfläche des Hauptmenüs zum Laden der Bilddaten, Tätigen verschiedener Einstellungen und Schließen der Anwendung
- Menüschaltfläche für die Auswahl verschiedener Werkzeuge und Messungen
- Zustandsschaltfläche zur Aktivierung/Deaktivierung des Snappings (siehe Abschnitt 4.5.2)
- Zustandsschaltfläche zur Aktivierung/Deaktivierung der Mehrfachselektion

Im Falle der Mehrfachselektion und des Snappings wurde dabei auf eine Aktivierung bzw. Deaktivierung über Gesten verzichtet. Dies hätte mithilfe von Kommandogesten erfolgen müssen, was zu den in Abschnitt 4.1.1 erwähnten Nachteilen führen würde. Der mittlere Teil des Interfaces wurde schließlich so groß wie möglich gewählt, um eine gute Sichtbarkeit der Bilddaten zu gewährleisten. Dabei ist immer jeweils eine Bildansicht als aktiv gewählt, wobei die aktive Ansicht durch die Selektion einer anderen Bildansicht geändert werden kann. Währenddessen ist auf der rechten Seite der Gesten-Guide zusammen mit der Anzeige der vom Gerät erkannten 3D-Handmodelle zu finden.
Insgesamt wurden beim Entwurf der Benutzungsoberfläche möglichst große Schaltflächen angestrebt, um diese auch für Nutzer mit unruhigen Händen leicht selektierbar zu gestalten. Um dies zu ermöglichen, musste die Menge der dauerhaft sichtbaren Schaltflächen minimiert werden, so dass auch ohne ein übermäßig großes Interface genug Platz für alle Elemente vorhanden ist. Gestützt wird diese Vorgehensweise durch die Untersuchungen von Argelaguet et al. [63]. Sie fanden heraus, dass das Maximieren der Selektionsflächen und das Minimieren des Abstands des Cursors zu selbigen zur Erhöhung der Performance bei der Selektion beiträgt.

[1] Snapping-Icon von [100]
[2] Hand im Icon für die Mehrfachselektion von [101]
[3] Icons für die Gestenvisualisierungen von [102]

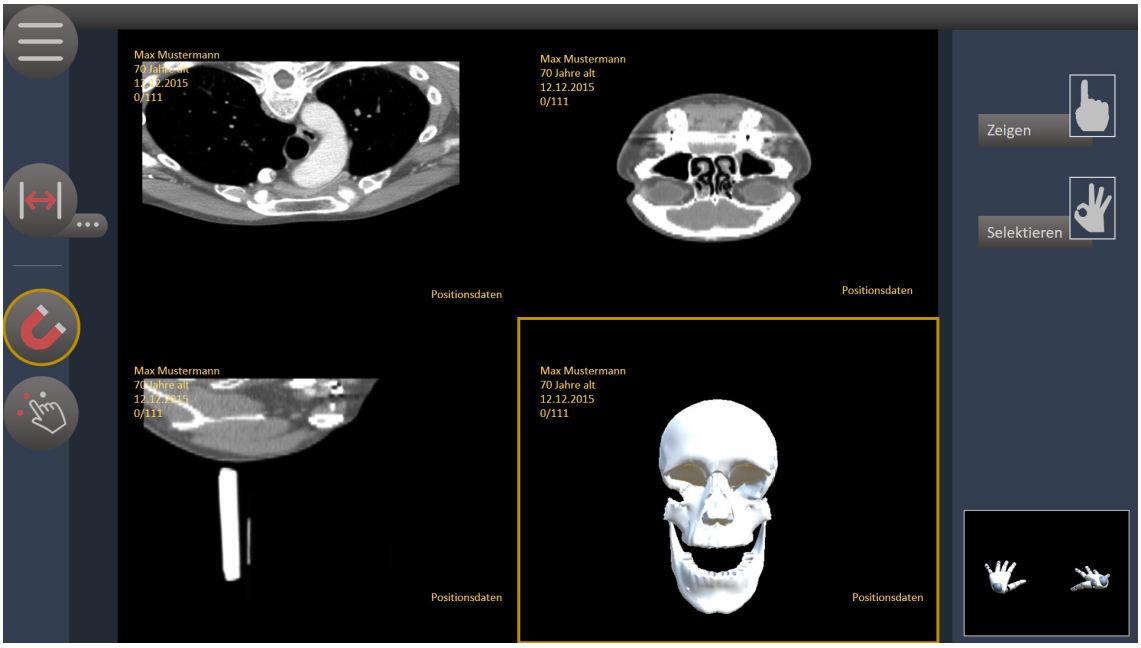


Abbildung 4.1: Interface Prototyp mit der Seitenleiste (links), den verschiedenen Ansichten zur Anzeige der Bilddaten (mittig) und dem Gesten-Guide (rechts)

4.2.1 Farbschema für Markierungen in den Bilddaten und im Interface

Für die gesamte Benutzungsoberfläche gilt ein allgemeines Farbschema für Markierungen.
So steht Grün für fertige und gerade nicht selektierte bzw. gerade nicht bearbeitete Messungen. Hierdurch soll angezeigt werden, dass mit dem jeweiligen Objekt alles in Ordnung ist.
Dahingegen wird Gelb für die Vorschau möglicher Änderungen sowie zur Markierung selektierbarer Objekte unter dem Pointer verwendet. Es fungiert somit, ähnlich dem gelben Ampellicht, als eine Art Warnung davor, dass der Nutzer kurz davor steht eine Änderung am aktuellen Zustand vorzunehmen. Währenddessen markiert Rot stets selektierte oder in Bearbeitung befindliche Objekte. Inspiriert ist diese Farbwahl von Anwendungen wie 3ds Max, die selektierte Objekte ebenfalls auf diese Weise hervorheben.
Die einzige Ausnahme von diesem Farbschema bildet die Markierung der aktiven Zustandsschaltflächen sowie der als aktiv selektierten Bildansicht. Hier könnte die ständige Hervorhebung des aktuellen Zustands durch eine rote Umrandung ablenkend wirken. So wurde stattdessen eine weniger auffällige, gelbe Umrandung gewählt. Diese ermöglicht außerdem ein ständiges Feedback darüber, welche Modi zur Zeit aktiv bzw. inaktiv sind.

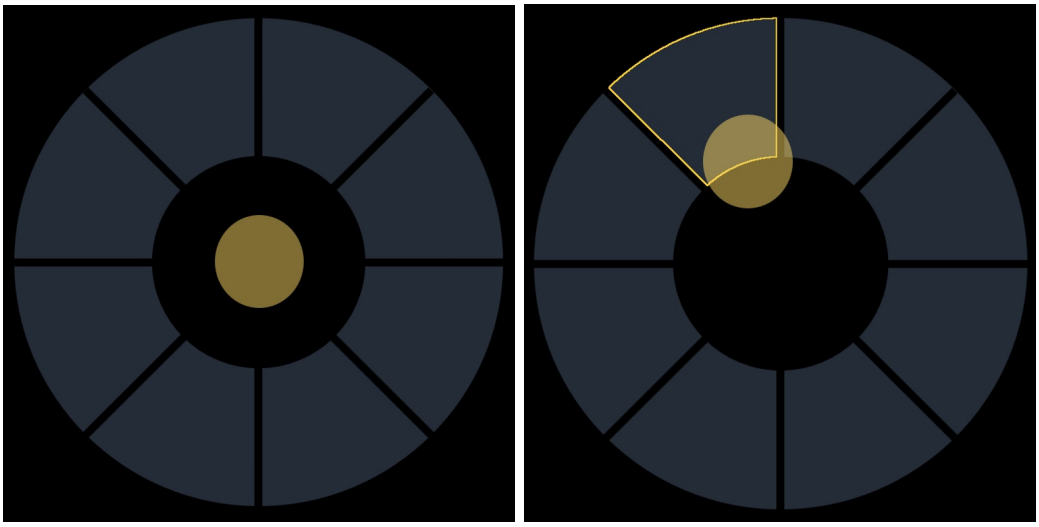


Abbildung 4.2: Prototypisches Radialmenü mit mittig initiiertem, semitransparentem Pointer (links) und markiertem Element (rechts)

4.2.2 Radialmenüs

Die Radialmenüs (siehe Abbildung 4.2) werden zur Auswahl der unter den Menüschaltflächen verborgenen Funktionen und als Kontextmenü verwendet. Dabei ermöglichen sie einen schnellen Zugang zu den im Menü vorhandenen Funktionen [64, 65, 66] ohne dauerhaft Platz einzunehmen. Ihre Effizienz ist zum einen dadurch bedingt, dass der mittig initiierte Pointer zu jedem der Menüelemente denselben geringen Abstand besitzt und zum anderen dadurch, dass der Nutzer von seinem Muskelgedächtnis profitieren kann [67, 68]. Letzteres beschreibt den Effekt, dass der Anwender nach mehrfacher Ausführung der Bewegung zu einem bestimmten Element des Radialmenüs nicht mehr auf das visuelle Feedback angewiesen ist, da er sich den immer gleichen Bewegungsablauf eingeprägt hat. Auf diese Weise wird auch der kognitive Aufwand reduziert. Darüber hinaus sorgt eine semitransparente Darstellung des Menüs dafür, dass der Arbeitskontext im Auge behalten werden kann [56].

4.2.3 Laden der Bilddaten

Für das Laden der Bilddaten gilt es lediglich die Steuerung der bisher typischen Menüs anzupassen. So erfolgt das Scrollen innerhalb der in Abbildung 4.3 gezeigten Bereiche durch das Greifen und Bewegen mit der zur lockeren Faust geballten Hand. Dies scheint geeignet, da es sich hierbei um eine bekannte, wenn auch bisher nur mit der Maus gesteuerte, Interaktionstechnik in Anwendungen wie dem Acrobat Reader oder der Windows-Fotoanzeige handelt.
Außerdem wird die in Abschnitt 4.4.6 vorgestellte Wischgeste von links nach rechts ver-

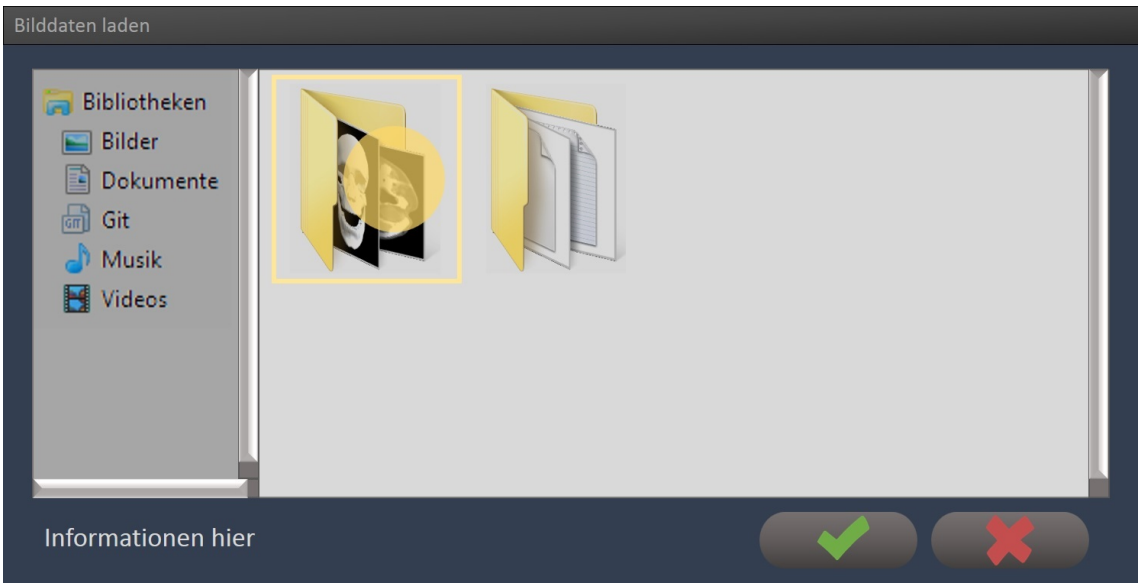


Abbildung 4.3: Prototyp des Menüs zum Laden der Bilddaten mit semitransparentem Pointer

wendet, um eine Stufe in der Ordnerstruktur nach oben zu navigieren. Diese Art der Interaktion ist bereits im Rahmen von Touchscreen-Anwendungen für die Vorwärts- und Rückwärtsnavigation (z. B. bei mobilen Apps zur Bildanzeige oder der Dropbox-App) geläufig und daher angemessen. Die restliche Steuerung des Menüs verläuft wie bereits aus mausgesteuerten Anwendungen gewohnt. [4] [5]

4.3 Visuelles Feedback

Wie bereits in Abschnitt 3.2.3 erwähnt, ist ein kontinuierliches Feedback essentiell. Auf diese Weise erhält der Nutzer sowohl Rückmeldung über den aktuellen Zustand des Systems als auch wichtige Informationen. Daher wurden verschiedene Mechanismen zur Bereitstellung von schnell zugänglichem, visuellem Feedback entwickelt. Diese sollen nachfolgend vorgestellt werden.

[4] Häkchen- und Kreuz-Icon von [103]
[5] Icons für Ordnerhierarchie und Ordnerdarstellung beispielhaft aus Windows 7 von Microsoft entnommen

4.3.1 Gesten-Guide

Der Gesten-Guide dient dem Zweck das teilweise Fehlen sichtbarer Affordances bei der Freihandgesteninteraktion auszugleichen. So zeigt er stets die Gesten für die im aktuellen Zustand möglichen Interaktionen an. Dieser Ansatz stammt von Brandl et al. [69] und mindert die Nachteile, die die Gestensteuerung in Bezug auf das notwendige Erlernen und Behalten der Gesten mit sich bringt. Außerdem werden hier Erkenntnisse aus den Untersuchungen von Walter et al. [70] genutzt. Sie konnten zeigen, dass eine räumliche Teilung zwischen der Anwendung und dem Gesten-Guide sinnvoll ist, da hierdurch unter anderem die Handlungen des Nutzers nicht unterbrochen werden.
Darüber hinaus werden die vom Eingabegerät erkannten Handdaten in Form von 3D-Handmodellen im unteren Bereich dargestellt, um dem Nutzer die Möglichkeit zu geben im Falle eines Fehlers in der Gestenerkennung die Ursache der falschen bzw. nicht erfolgten Erkennung herauszufinden und ggf. zu korrigieren. Gestützt wird die Verwendung solch eines dynamischen visuellen Feedbacks von Rovelo at el. [71], die dieses in Bezug auf die Interaktion mit Gesten als performanter als statisches visuelles Feedback herausstellten. Der Nachteil dieser Vorgehensweise besteht allerdings darin, dass der Gesten-Guide permanent Platz einnimmt [70].

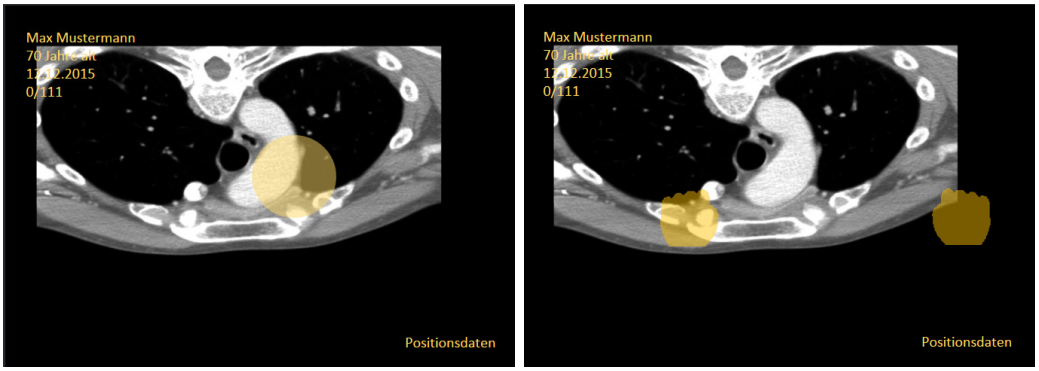


Abbildung 4.4: Darstellung des von der Zeigegeste hervorgerufenen Pointers (links) und der Gestenvisualisierung der Kamerakontrollgeste (rechts)

4.3.2 Repräsentation der Gesten

Um dem Nutzer Feedback darüber zu geben, welche Geste gerade vom System erkannt wird, soll die jeweilige Visualisierung der Geste an der aktuellen Handposition auf dem Bildschirm dargestellt werden (siehe Abbildung 4.4). So werden zusätzlich Informationen darüber geliefert, wo im Interaktionsraum der Anwender sich mit seinen Händen befindet

[72]. Die semitransparente, gelbe Farbe sorgt außerdem dafür, dass sich die Gestenvisualisierungen sowohl von den Bilddaten abheben als auch die darunterliegenden Strukturen nicht verdecken, so dass diese im Auge behalten werden können.
Kleinere Abweichungen in der Visualisierung existieren dabei für den durch die Zeigegeste hervorgerufenen Pointer. Dieser passt seine Größe unter anderem seiner Bewegungsgeschwindigkeit an. So ist er größer, wenn eine schnelle Bewegung erfolgt und daher keine besondere Genauigkeit in der Selektion wohl aber eine bessere Sichtbarkeit des Pointers benötigt wird. Entsprechend kleiner wird er dargestellt, wenn eine langsame Bewegung erfolgt, bei der davon auszugehen ist, dass eine präzise Selektion durchgeführt werden soll. Hier wäre eine große Pointerfläche hinderlich, da so leicht andere benachbarte Objekte statt des gewünschten getroffen werden könnten. Des Weiteren hängt die Größe des Pointers von der Größe des unter ihm befindlichen Objektes ab, da frühere Arbeiten gezeigt haben, dass eine Cursorgröße, die in etwa dem zu selektierenden Objekt entspricht, am effizientesten ist [72]. Darüber hinaus spart die Verwendung eines Flächencursors Zeit bei der Selektion kleiner Elemente [73].

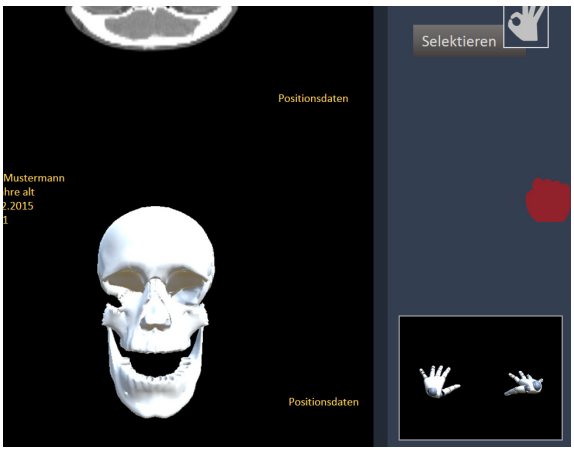


Abbildung 4.5: Darstellung einer Stopper-Visualisierung am rechten Bildschirmrand

4.3.3 Stopper

Eine weitere visuelle Hilfestellung erfolgt in Form der sogenannten „Stopper“. Dieser Begriff wurde von Chattopadhyay et al. [72] geprägt und steht für das Feedback über das Verlassen des Interaktionsraums durch eine entsprechende Veränderung der Gestenvisualisierung. Dabei haben Studien festgestellt, dass es den Probanden so auf intuitive Weise erleichtert wird, nach dem Verlassen des Interaktionsraums schnell in diesen zurückzukehren und die Interaktion fortzusetzen. Letzteres wird dadurch gewährleistet, dass der aktuelle Vorgang nur gestoppt und beim Wiedereintritt fortgesetzt wird. Dabei soll durch

die Änderung der Farbe von Gelb zu Rot außerdem die Aufmerksamkeit des Nutzers auf die aktuelle Situation gelenkt werden.

4.4 Interaktionstechniken

Der Schwerpunkt der Konzeption liegt in der Entwicklung angemessener, gestenbasierter Interaktionstechniken. Diese sollen im Folgenden erarbeitet und näher erläutert werden. Dabei sind zunächst die grundlegenden Techniken vorzustellen, um darauf aufbauend die Vermessungsinteraktionen beschreiben zu können.

4.4.1 Zeigen

Abbildung 4.6: Zeigegeste

Das Zeigen erfolgt mithilfe der in Abbildung 4.6 dargestellten Geste. So wird der Zeigefinger wie eine Art Laserpointer verwendet, d.h. es wird ein imaginärer Strahl vom Finger auf den Bildschirm geschossen. Wo dieser aufkommt, wird die Pointervisualisierung gezeichnet. Dieses Vorgehen wird auch „Raycasting“ genannt [52] und ist eine natürliche und intuitive Vorgehensweise für das Zeigen auf Objekte. Das Raycasting soll verwendet werden, da es im Vergleich zur Darstellung der bloßen Handposition (auch *Hand Extension Metapher* genannt) eine bessere Performance besitzt und von den Nutzern bevorzugt wird [60, 52]. Des Weiteren muss zum Zeigen nicht der gesamte Arm bewegt werden, sondern die Bedienung kann bequem und platzsparend aus dem Handgelenk

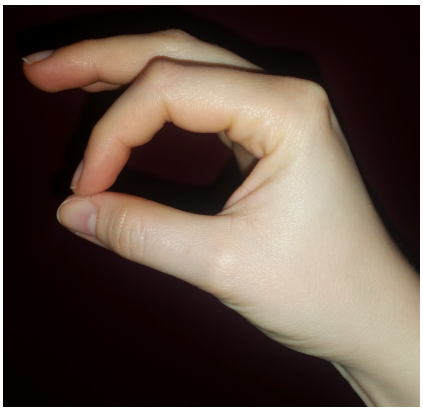


Abbildung 4.7: Pinch-Geste

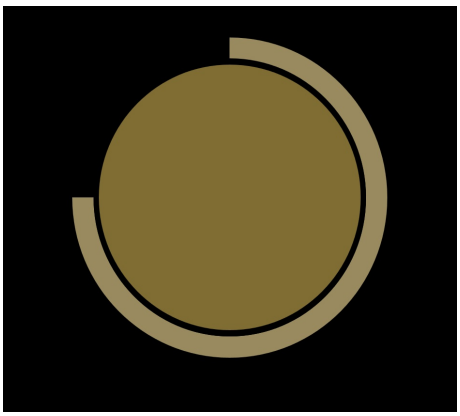


Abbildung 4.8: Wait-to-Click-Pointer

erfolgen [56]. So wirkt die Interaktion weniger ermüdend [52]. Als nachteilig ist hier jedoch der möglicherweise geringere Komfort zu nennen, der durch das häufige Beugen des Handgelenks entstehen kann.

4.4.2 Pinch-to-Click

Pinch-to-Click wird als Pendant für den Klick mit der Maus verwendet und durch die Pinch-Geste (siehe Abbildung 4.7) ausgelöst. Diese schien als Geste im Allgemeinen geeignet, da sie sowohl schnell und unmissverständlich in der Ausführung als auch natürlich ist [74]. Außerdem bietet sie kinästhetisches Feedback und mindert so einen Nachteil der Freihandgesten, der durch das fehlende taktile Feedback aufgrund des nicht vorhandenen Kontakts zum Eingabegerät entsteht. Des Weiteren wurden Gesten mit kinästhetischem Feedback schon in früheren Untersuchungen als positiv bewertet [75]. Darüber hinaus wirkt sich die Aufteilung des Zeigens und Selektierens auf zwei Hände insofern positiv aus, als dass die gezeigte Position nicht durch die „Klickbewegung“ verzerrt wird [75, 76].

4.4.3 Wait-to-Click

Diese einhändige Interaktionstechnik besteht darin, dass der Nutzer den Zeiger über das zu selektierende Objekt bewegt und dort verweilen lässt bis eine gewisse Wartezeit abläuft. Ist dies geschehen, erfolgt der Klick.
Im aktuellen Konzept wird Wait-to-Click zur Aktivierung bzw. Deaktivierung von Zustandsschaltflächen genutzt, um andere Interaktionen für das Aktivieren des Snappings oder der Mehrfachselektion nicht vollständig unterbrechen zu müssen. So kann beispiels-

weise die gerade mit dem Pointer anvisierte Position in den Bilddaten beibehalten werden, während die andere Hand das Snapping einschaltet.
Das Wait-to-Click-Paradigma scheint angemessen, da es aufgrund seiner Einfachheit und Genauigkeit bereits in mehreren Arbeiten positiv abschnitt [49, 75]. Um dabei eine versehentliche Aktivierung der Schaltflächen durch Unaufmerksamkeit zu vermeiden, wird die in Abbildung 4.8 gezeigte Pointervisualisierung verwendet, die das Ablaufen der Zeit um den Pointer herum darstellt. Sie soll die Aufmerksamkeit des Nutzers durch Bewegung auf den Wait-to-Click-Prozess lenken [39].

4.4.4 Steuerung der Radialmenüs

Für die Radialmenüs gibt es zwei verschiedene Einsatzszenarios, die anhand des beim Aufruf vom Pointer angezeigten Objektes unterschieden werden. Im Folgenden sollen daher die Mechanismen zum Aufrufen der verschiedenen Menüs sowie die anschließende Handhabung selbiger erläutert werden.

Aufruf der Radialmenüs

Sowohl das Haupt- und Werkzeugmenü als auch die Kontextmenüs werden durch das Halten der Pinch-Geste aufgerufen. Für Erstere muss dabei der Pointer auf die entsprechende Menüschaltfläche zeigen, während Kontextmenüs nicht durch das Anzeigen von Schaltflächen geöffnet werden können. Stattdessen ist das Zeigen auf andere Objekte oder die Abwesenheit des Pointers notwendig. Je nachdem welche der beiden Situationen eintritt bzw. auf welche Objekte gezeigt wird, ändert sich der Inhalt des Kontextmenüs, wobei die Abwesenheit des Pointers zum Aufruf eines allgemeinen Menüs führt. Über dieses lassen sich jederzeit das Snapping und die Mehrfachselektion aktivieren bzw. deaktivieren sowie die aktive Funktion ändern. Sind Messungen selektiert, wird es außerdem möglich diese zu bearbeiten oder zu löschen. Dabei gilt für alle Radialmenüs, dass die beim Aufruf ausgeführte Zeigegeste nach dem Öffnen des Radialmenüs beendet werden kann, da nicht weiter auf das jeweilige Objekt gezeigt werden muss.
Die Vorgehensweise zum Öffnen der Kontextmenüs ist bereits aus Anwendungen auf dem Touchscreen bekannt und sollte dem Nutzer daher leicht fallen. Währenddessen weicht das Aufrufen der unter den Schaltflächen verborgenen Menüs über das Halten der Pinch-Geste leicht vom gewohnten Prinzip des einfachen Klicks ab. Dies ist notwendig, um das im nächsten Abschnitt beschriebene Verfahren zur Selektion der Menüelemente anwenden und so die in Abschnitt 4.2.2 beschriebenen Vorteile nutzen zu können.

Selektion im Radialmenü

Ist das Radialmenü aufgerufen, wird die Visualisierung der Pinch-Geste in einen, durch die Handposition gesteuerten „Pinch-Pointer“ umgewandelt, der durch das Aufbauen des Menüs um die Geste herum in der Mitte des Menüs platziert wird. Befindet sich die Hand dabei zu weit am Rand des Bildschirms, wird das Menü so verschoben, dass es ganz sichtbar ist. Anschließend wird der Position des Pinch-Pointers ein gewisser Offset hinzugefügt, so dass dieser sich weiterhin mittig befindet.
Mithilfe des Pinch-Pointers kann anschließend das gewünschte Element angezeigt und mit einer Pinch-Geste der anderen Hand bestätigt werden. Die Verwendung der Hand Extension Metapher für die Steuerung des Pinch-Pointers ist dabei angemessen, da durch dessen Positionierung stets nur sehr kurze Bewegungen auszuführen sind, so dass die Nachteile dieser Interaktionstechnik (siehe Abschnitt 4.4.1) kaum zum Tragen kommen. Außerdem lassen sich so die zuvor beschriebenen Nachteile des Raycastings für die Steuerung des Radialmenüs vermeiden.

Schließen des Radialmenüs

Die erste Möglichkeit ein Radialmenü zu schließen, besteht darin die aufrufende Pinch-Geste zu beenden. Dies schien bei der gewählten Aufrufmethode der intuitivste Weg dem Nutzer die Möglichkeit zu geben das Menü nach dem Öffnen wieder beenden zu können, ohne dass eine Funktion aktiviert wird.
Das Aktivieren einer Funktion durch die Selektion des entsprechenden Menüelements stellt schließlich auch die zweite Möglichkeit zur Schließung des Radialmenüs dar. Die einzige Ausnahme bilden hier die als Zustandsschaltflächen verwendeten Menüelemente für das Snapping und die Mehrfachselektion. Sie haben nicht das Beenden des Menüs zur Folge, da Situationen eintreten können, in denen der Nutzer mehrere Modi innerhalb eines Menüaufrufes aktivieren bzw. deaktivieren möchte.

4.4.5 Manipulation von Objekten

Bei der Manipulation wird grundlegend zwischen der Kamera und anderen Objekten (z. B. Messungen) unterschieden, so dass hier kleinere Unterschiede existieren, das grundlegende Prinzip jedoch gleich bleibt. So beginnt jede Manipulation der Kamera damit, dass beide Hände nebeneinander zu einer lockeren Faust geballt werden, während zur Einleitung der Manipulation anderer Objekte beide Hände nebeneinander die Pinch-Geste halten müssen. Mithilfe dieser Gesten soll dabei das Greifen beider Enden eines Stabes, auf dem das jeweilige Objekt aufgespießt wurde, symbolisiert werden. Sie sind bis zum Abschluss der Interaktion aufrechtzuerhalten. Welche Achse dabei manipuliert wird,

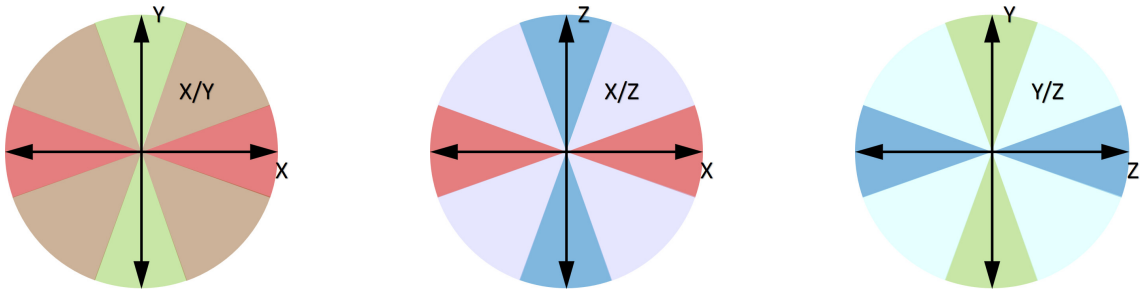


Abbildung 4.9: Manipulationsbereiche der Achsen

hängt von der Bewegungsrichtung der Hände ab (siehe Abbildung 4.9). Die Kamerasteuerung bezieht sich außerdem stets auf die aktive Bildansicht.
Bei diesem Konzept handelt es sich um eine leicht abgewandelte *Handle-Bar-Metapher*. Sie stammt ursprünglich aus der Arbeit von Song et al. [77] und wird auch von einer Studie von Mendes et al. [78] sehr unterstützt. Die Handle-Bar-Metapher ähnelt dabei der realen Manipulation von Objekten und bietet so eine natürliche und intuitive Handhabung, wie sie auch in [77] bestätigt wird. Dies kann außerdem durch die enge Kopplung zwischen der manipulierten Achse und den Bewegungen der Hände noch verstärkt werden. Nachfolgend wird die Funktionsweise der Handle-Bar-Metapher genauer und getrennt nach Art der Manipulation erläutert.

Translation

Die Translation beschreibt das Verschieben eines Objektes. Dies erfolgt bei der Handle-Bar-Metapher durch das Bewegen der nebeneinander gehaltenen Hände entlang der entsprechenden Achse. Für die Kamera ist allerdings keine Bewegung in Richtung der z-Achse möglich, da dies dem Zoomen entspricht und daher von der Interaktion zur Skalierung abgedeckt wird.

Skalierung

Für die Skalierung von Objekten bzw. den Zoom bei der Kamera müssen die Hände auseinander- oder zusammengeführt werden. Ersteres führt dabei zum Vergrößern bzw. Heranzoomen, während Letzteres ein Verkleinern bzw. Herauszoomen zur Folge hat. Die Skalierung von Objekten kann auch einseitig erfolgen, indem nur die rechte oder linke Hand bewegt wird. Auf diese Weise soll die Manipulation von Objekten zusätzlich erleichtert und flexibel gestaltet werden. Des Weiteren ist das Zoomen entsprechend seiner grundlegenden Eigenschaften nicht entlang einzelner Achsen, sondern nur uniform durch eine Bewegung entlang der x-Achse möglich.

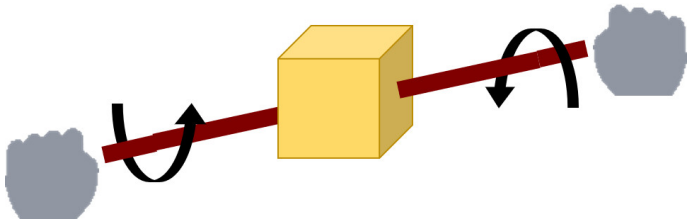


Abbildung 4.10: Metapher zur Rotation um die x-Achse

Rotation

Die Rotation findet, leicht abweichend von dem ursprünglichen Konzept der Handle-Bar-Metapher, zweigeteilt statt. So erfolgt die Rotation um die z- und y-Achse durch das Drehen des imaginären Stabes um die jeweilige Achse, während der Stab für die Rotation um die x-Achse leicht schräg gehalten und anschließend eine entsprechende Kreisbewegung um die x-Achste ausgeführt werden muss (siehe Abbildung 4.10 [6]). Alternativ hätte der imaginäre Stab schlicht durch eine Beugung der Handgelenke um die x-Achse gedreht werden können. Doch dies würde zu einer wenig komfortablen und eher umständlichen Handhabung führen.

Trennung der Manipulationsmodi

Um den Umgang mit den Gesteninteraktionen zur Manipulation der Kamera bzw. von Objekten einfacher zu gestalten, ist ein nahtloser Übergang zwischen Translation, Skalierung und Rotation (z. B. gleichzeitiges Bewegen und Heranzoomen) nicht möglich. So wird der Manipulationsmodus für die Kamerasteuerung mit der ersten Bewegung festgelegt und kann nur gewechselt werden, indem die aktuelle Geste durch das Öffnen einer der beiden Hände beendet wird. Ähnliches gilt für die Handhabung der Objekte, bei denen mit der ersten Bewegung nicht nur die Art der Manipulation, sondern auch die betroffene Achse festgelegt wird. Auf diese Weise ist es nicht nötig permanent auf die genaue Positionierung der Hände zueinander zu achten, da versehentliche Manipulationen erschwert werden. Außerdem wird eine höhere Präzision ermöglicht, da nicht überwacht werden muss, ob während der Änderung entlang einer Achse noch eine weitere Achse manipuliert wird.

[6] Hand-Icons von [102]

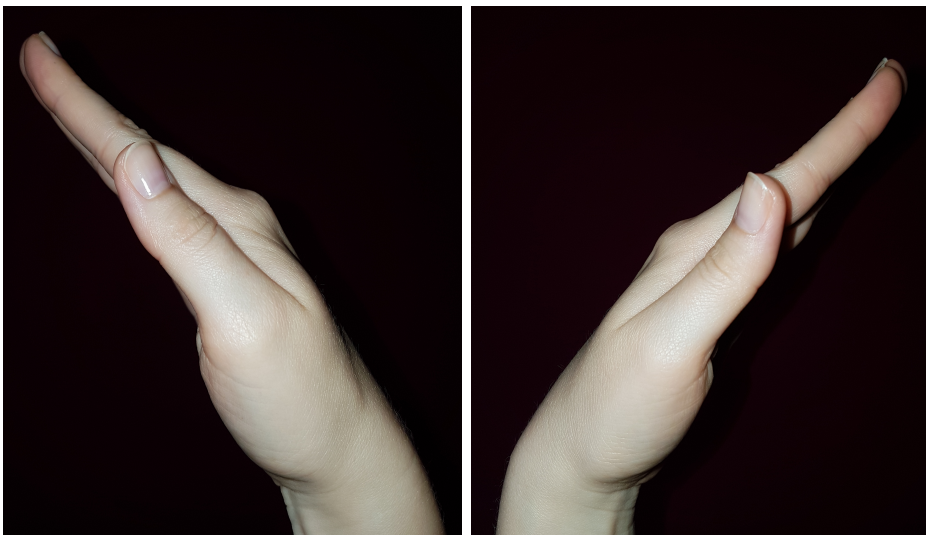


Abbildung 4.11: Wischgeste nach links und nach rechts

4.4.6 Navigation durch die 2D-Schichtdaten

Zur Navigation innerhalb der 2D-Schichtdaten wird die in Abbildung 4.11 gezeigte Wischgeste verwendet. Dabei entspricht eine Bewegung von rechts nach links der Vorwärtsnavigation und eine Bewegung von links nach rechts der Rückwärtsnavigation. Diese erfolgen jeweils in der aktuell aktiven Ansicht der Bilddaten.
Um häufige Beugungen des Handgelenks bei der Navigation durch mehrere Schichten zu vermeiden, ist es außerdem möglich die Geste zu halten, um nicht nur eine, sondern mehrere Schichten mit einer Interaktion zu durchlaufen. Dafür ist es zusätzlich notwendig das Handgelenk ein kleines Stück weiter in die entsprechende Richtung zu beugen. Auf diese Weise soll die Einzelnavigation besser von der Mehrfachnavigation zu unterscheiden sein.
Der Vorteil dieser Freihandgeste liegt dabei darin, dass sie die bereits vom Touchscreen bekannte Wischgeste nachahmt und daher für den Nutzer intuitiver sein könnte. Des Weiteren wurde sie im Rahmen einer Nutzerstudie als intuitive Geste für das Vor- und Zurücknavigieren gewählt [51]. Allerdings wird im aktuellen Konzept eine Bewegung aus dem Handgelenk statt aus dem Ellenbogen genutzt, um die Geste weniger ermüdend und platzsparend zu gestalten.

4.4.7 Messungen erstellen

Nachdem in den vergangenen Abschnitten die grundlegenden Interaktionen zur Bedienung der Benutzungsoberfläche sowie zur Navigation in den Bilddaten vorgestellt worden sind, kann die Aufmerksamkeit nun auf die Vermessungsinteraktionen gerichtet werden.

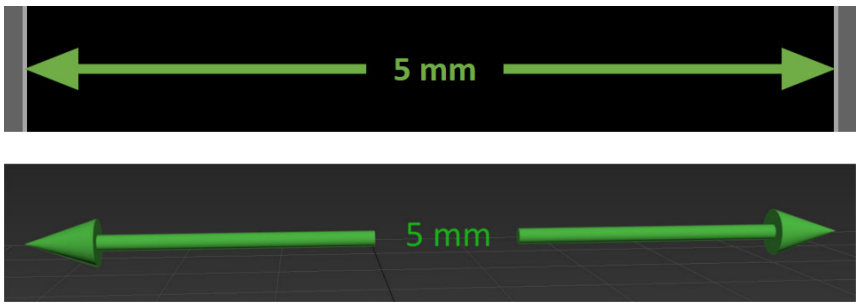


Abbildung 4.12: Beispielhafte Visualisierung der Distanzmessung in 2D und 3D

Hierzu sollen zunächst das allgemeine Visualisierungskonzept sowie die zwei grundlegenden Interaktionstechniken beschrieben werden, um anschließend die Vorgehensweisen zur Vermessung einfacher ableiten und erläutern zu können.

Allgemeines Visualisierungskonzept

Wie bereits in Abschnitt 4.2.1 erwähnt, sind fertige Messungen stets grün, während Messungen, die sich noch in der Erstellung befinden, immer gelb dargestellt werden. Zudem erfolgt eine kontinuierliche Anpassung der Schriftgröße der Bemaßungsbeschriftung an den Zoomfaktor, um zu verhindern, dass diese zu klein oder zu groß ausfällt. Außerdem wird sie bei einer Rotation erneut der Kamera zugewandt. Auf diese Weise soll die Beschriftung stets lesbar sein [28]. Des Weiteren wird die Maßzahl möglichst nah an der Messung platziert, um eine eindeutige Zuordnung zu gewährleisten.
Darüber hinaus gilt für jede Messungserstellung, dass während der Platzierung des letzten, bestimmenden Messpunktes bzw. während der letzten bestimmenden Manipulation eine entsprechende Vorschau der fertigen Messung bereitgestellt werden soll. Dies dient dem Zweck, dass der Nutzer bereits vor der endgültigen Fertigstellung einen ersten Eindruck von den räumlichen Verhältnissen bei der Vermessung bekommt. Ein Beispiel für solch eine Vorschau ist in Abbildung 4.13 zu sehen.

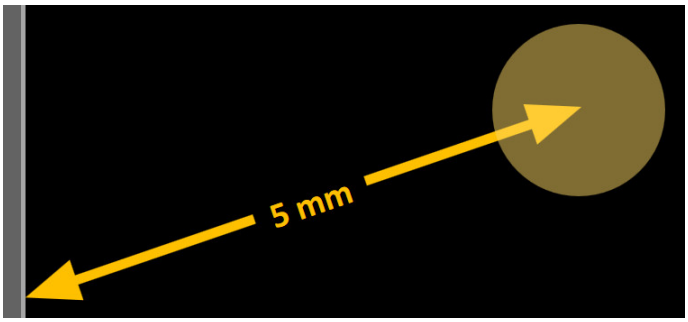


Abbildung 4.13: Beispielhafte Visualisierung der Vermessungsvorschau

Platzierung von Messpunkten

Die Platzierung von Messpunkten bildet die Grundlage für fast alle Arten von Messungen. So besteht die Distanzmessung aus zwei Messpunkten, die zu einer Linie verbunden werden, während Flächen- und Volumenmessungen bestimmte Eckpunkte als Eingabe benötigen.
Das Setzen eines Messpunktes erfolgt dabei durch das Zeigen auf die gewünschte Position in den Bilddaten und die anschließende Bestätigung durch Pinch-to-Click. Welche Hand die Zeige- und welche die Pinch-Geste ausführt, bleibt hier dem Nutzer überlassen. Auf diese Weise wird es ihm ermöglicht, die für ihn komfortabelste Haltung einzunehmen.

Aufspannen von Objekten

Ein weiteres wichtiges Konzept für die Erstellung von Messungen ist das Aufspannen von Objekten. Es kommt hauptsächlich bei der Erzeugung von Flächen- und Volumenmessungen zum Einsatz.
Der erste Schritt ist dabei stets die Festlegung des Mittelpunktes, um den herum das Objekt skaliert werden soll. Dieser wird mithilfe der Zeigegeste angezeigt und durch eine Pinch-Geste der anderen Hand bestätigt. Mithilfe des Haltens dieser Pinch-Geste wird schließlich auch die Skalierung des Objektes eingeleitet. Diese hängt dabei vom Abstand zwischen dem gesetzten Mittelpunkt und der Position der Pinch-Geste ab und erfolgt, ähnlich dem in Abschnitt 4.4.5 erklärten Prinzip, abhängig von der Bewegung der Hand entlang der drei Achsen. Daher besitzt diese Interaktionstechnik auch die dort bereits beschriebenen Vorteile. Welche der Achsen dabei manipuliert werden können, wird durch das aufzuspannende Objekt bestimmt. Außerdem ist hier ein fließender Übergang zwischen der Manipulation einzelner Achsen erlaubt. Auf diese Weise wird eine schnelle Erstellung ermöglicht, während Feinheiten anschließend ggf. angepasst werden können. Abschließend erfolgt die Bestätigung der Skalierung durch eine Pinch-Geste der Hand, die im ersten Schritt den Mittelpunkt angezeigt hatte. So führt eine zwischenzeitliche Unterbrechung der aufziehenden Pinch-Geste nicht zu einer fehlerhaften Messungserstellung.

Distanzmessung

Für die Erstellung einer Distanzmessung ist die Definition eines Anfangs- und eines Endpunktes notwendig. Aus diesem Grund liegt es nahe diese auf einfache Art und Weise durch die Platzierung von zwei Messpunkten zu erzeugen und so die fertige Messung zu erhalten.
Die in Abbildung 4.12 zu sehende, innenliegende Bemaßungsbeschriftung entstammt

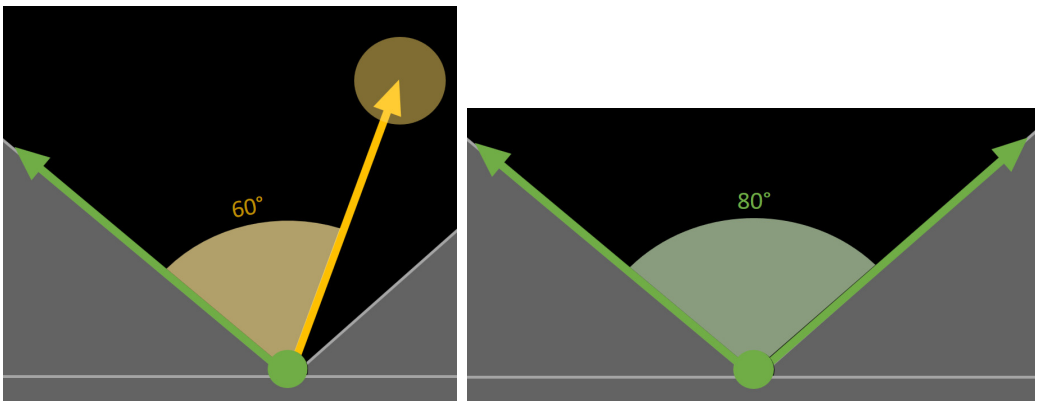


Abbildung 4.14: Aufspannen der zweiten Winkelkante nach der Bestimmung der ersten durch das Setzen von Anfangs- und Mittelpunkt (links) und fertige Winkelmessung nach Variante 1 (rechts)

dabei ebenfalls der Arbeit von Rössling et al. [1], da dort gezeigt werden konnte, dass diese von Ärzten bevorzugt wird.

Winkelmessung

Während die Distanzmessung kaum Variationen in ihrer Erstellung zulässt, sind für die Winkelmessung zwei Varianten denkbar. Eine erste Möglichkeit besteht darin die Messung durch die serielle Platzierung dreier Punkte (Anfangs-, Mittel- und Endpunkt) zu erzeugen. Die entsprechende Visualisierung orientiert sich dabei an den Konzepten von Preim et al. [29] und ist in Abbildung 4.14 zu sehen. Vorteilhaft ist hier, dass diese Art des Aufspannens eines Winkels bereits bekannt ist und somit eher eine intuitive Vorgehensweise darstellt. Andererseits existieren Fälle, für die sie nicht optimal ist. So bauen manchen Winkelmessungen auf einer Konstruktion aus mehreren Linien auf, die für die Ableitung der endgültigen Winkelmessung notwendig ist. Beispielhaft sei hier die Messung des Skoliosewinkels nach Cobb [79] erwähnt.
Abhilfe schafft die zweite Variante, da hier zunächst beliebig viele Linien durch das Platzieren der jeweiligen Anfangs- und Endpunkte erzeugt werden können. Anschließend ist zu bestimmen, zwischen welchen Schnittpunkten Winkel zu berechnen sind. Letzteres erfolgt dabei durch das Zeigen in die Nähe des jeweiligen Schnittpunktes und die Bestätigung des Winkels durch Pinch-to-Click (siehe Abbildung 4.15). Auf diese Weise müssen außerdem statt vier nur drei Winkelkanten platziert werden, sobald sich zwei Winkel eine Kante teilen. Dies hat eine höhere Effizienz zur Folge. Im Allgemeinen bietet diese Interaktionstechnik außerdem ein hohes Maß an Flexibilität. Allerdings könnte sie auch zu Irritationen bei den Nutzern führen, da sie weit weniger bekannt sein dürfte als die erste Variante.

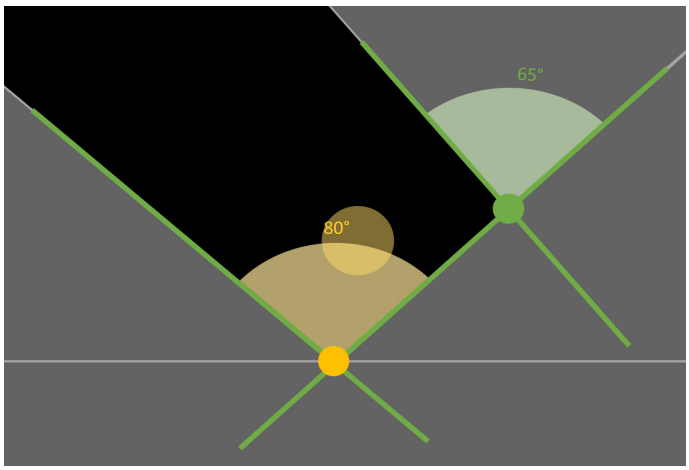


Abbildung 4.15: Bestimmung eines Winkels zwischen zwei Kanten nach der zweiten Variante

Flächenmessung und Grauwertanalyse in 2D über Regions of Interest

Da sowohl für die Flächenmessung als auch die Grauwertanalyse eine Fläche in den 2D-Bilddaten definiert werden muss, lassen sich diese beiden Messungen kombinieren und so dasselbe Interaktionsprinzip verwenden. Da dabei üblicherweise Kreis-, Ellipsen- oder Rechteckflächen bzw. Polygonzüge festzulegen sind, sollen diese im Folgenden betrachtet werden.
Die Visualisierung erfolgt hier, indem der Umriss der Fläche zusammen mit der Flächenangabe oben rechts und einer Schaltfläche oben links dargestellt wird (siehe Abbildung 4.16 rechts). Letztere ermöglicht es die Ergebnisse der Grauwertanalyse aus- bzw. einzublenden, um Platz zu sparen.

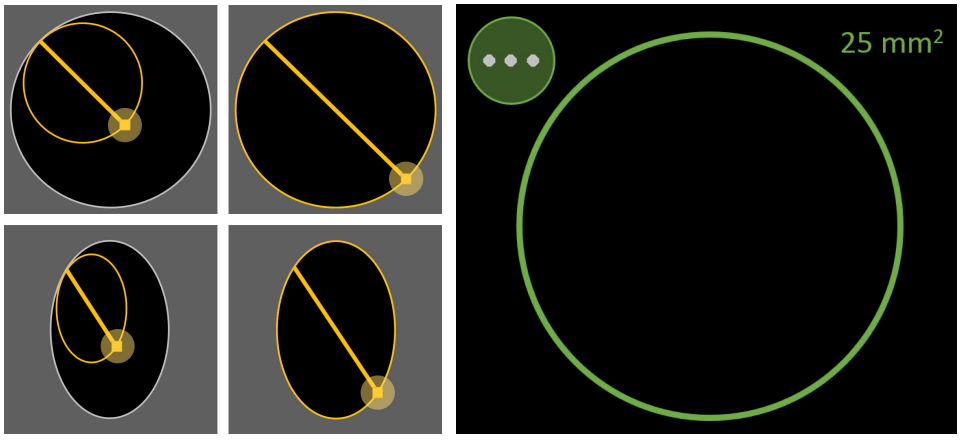
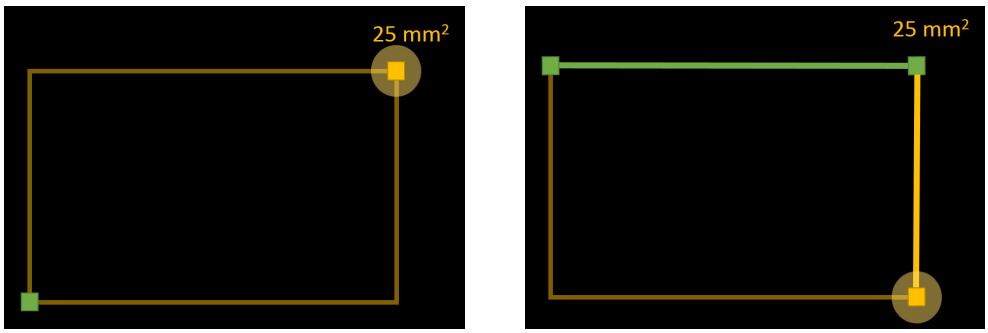


Abbildung 4.16: Aufspannen einer Kreis- bzw. Ellipsenfläche über eine Linie (links) und fertige, kreisförmige Flächenmessung (rechts)



(a) (b)

Abbildung 4.17: Varianten zum Aufspannen einer Rechteckfläche

Kreis-/Ellipsenflächen Die erste Möglichkeit zur Definition eines Kreises bzw. einer Ellipse besteht darin zunächst den Mittelpunkt zu bestimmen und ausgehend von diesem den Kreis entlang der x- und y-Achse aufzuziehen. Das hierfür verwendete Interaktionskonzept entspricht dem im Abschnitt zum Aufspannen von Objekten beschrieben Verfahren. Nachteilig ist dabei, dass der Nutzer sehr genau den Mittelpunkt der kreis- bzw. ellipsenförmigen Fläche abschätzen können muss, um eine gute Annäherung der Fläche zu erhalten. Kann er dies nicht, sind Nachbearbeitungen erforderlich, die zusätzliche Zeit kosten. Um dieses Problem zu vermeiden, wurde eine weitere Variante entwickelt.
Bei dieser zweiten Variante wird der Durchmesser der Fläche entlang der x- und y-Achse durch eine diagonal verlaufende Linie mit zwei Messpunkten definiert (siehe Abbildung 4.16 links). Dabei ist es sehr wichtig nicht nur den aufgezogenen Kreis bzw. die aufgezogene Ellipse, sondern auch die definierende Linie zu visualisieren, um dem Nutzer die Funktionsweise zu verdeutlichen. Dies ist schließlich auch der Unterschied zu bestehenden Verfahren, die im Universitätsklinikum als wenig intuitiv beschrieben wurden (siehe Abschnitt 3.1.5). Der Grund hierfür könnte jene fehlende Visualisierung der definierenden Linie sein.

Rechteckflächen Auch für die Definition einer Rechteckfläche durch das Setzen von Messpunkten existieren zwei mögliche Varianten. Die in der Erstellung schnellere von beiden besteht darin, das Rechteck über zwei Eckpunkte zu definieren (Abbildung 4.17a). Hier ist jedoch problematisch, dass auf diese Weise keine schräg liegenden, rechteckigen Flächen während der Erzeugung erfasst werden können.
Abhilfe schafft die zweite Variante. Diese erfordert die, in Abbildung 4.17b) dargestellte, Definition des Rechtecks über drei Punkte, benötigt dabei jedoch mehr Arbeitsschritte. Sie könnte dennoch effizienter sein, da im Falle schräg liegender Rechteckflächen keine aufwendige Nachbearbeitung notwendig ist und vollständig achsenparallele Flächen in anatomischen Strukturen eher selten sind.

Polygonflächen Für die Erzeugung von Polygonzügen wird das schrittweise Festlegen

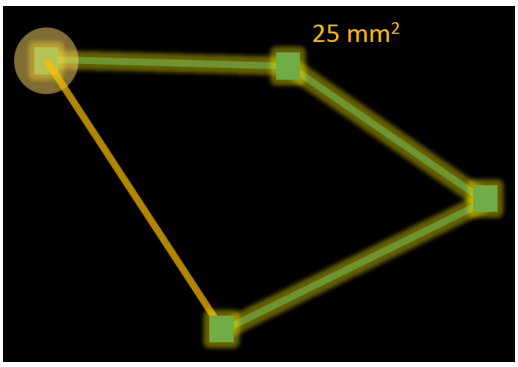


Abbildung 4.18: Visuelles Feedback beim Schließen eines Polygonzuges in der Ebene zur Erzeugung einer Polygonfläche

der notwendigen Messpunkte in einer Ebene der 2D-Ansicht als die intuitivste Vorgehensweise beurteilt, da sie ein flexibles Zeichnen der Polygonfläche ermöglicht. Um dabei sicherzugehen, dass das Polygon mit dem letzten, gesetzten Punkt geschlossen wird, erhält der Nutzer außerdem das in Abbildung 4.18 gezeigte visuelle Feedback beim Schließen der Fläche.

Volumenmessung

Die Wahl der Primitive zur Erstellung einer Volumenmessung orientiert sich an den möglichen Formen anatomischer Strukturen, die anhand solch einer Messung annähernd erfasst werden können. Daher wurden Interaktionskonzepte für die Erzeugung von Kugeln bzw. Ellipsoiden, Quadern und Zylindern zur manuellen Messung des Volumens entwickelt.

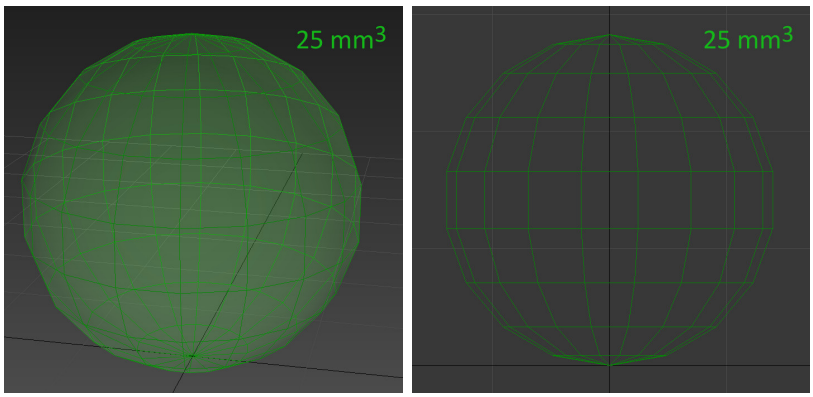


Abbildung 4.19: Beispielhafte Visualisierung einer fertigen Volumenmessung in 3D und 2D

Kugeln/Ellipsoide Für die Erstellung eines kugelförmigen bzw. ellipsoiden Volumenkörpers stellen sich ähnliche Fragen wie bei der Vermessung einer Fläche mithilfe eines Kreises. Daher sollen die dort erzielten Ergebnisse auch auf den dreidimensionalen Raum übertragen werden. Dabei bietet sich das Aufspannen einer Kugel über eine diagonal durch das Volumen verlaufende Linie umso mehr an, da sich der Mittelpunkt einer dreidimensionalen Struktur nicht so einfach abschätzen lässt. Zwar gestaltet sich hier das Platzieren einer Diagonalen ebenfalls aufwendiger, wird jedoch durch das in Abschnitt 4.5.2 beschriebene Snapping vereinfacht.

Quader Auch für den Entwurf der Interaktionen zur Erzeugung eines quaderförmigen Volumens lassen sich einige Konzepte aus der Flächenmessung nutzen. So stellte sich das Aufspannen über zwei Messpunkte bereits als wenig flexibel heraus. Aus diesem Grund ist für Volumenkörper, die nicht achsenparallel verlaufen, eine andere Vorgehensweise zu bevorzugen.
Hier könnte es sich als nützlich erweisen das Volumen über die Grundfläche mit drei Messpunkten aufzuspannen und anschließend die Höhe zu bestimmen (siehe Abbildung 4.20). Letzteres erfolgt dabei über den Abstand des Pointers zur Grundfläche und kann mithilfe von Pinch-to-Click bestätigt werden. Auf diese Weise wird schon während der Erzeugung eine höhere Genauigkeit und Flexibilität erzielt und aufwendige Nachbearbeitungen vermieden. Ein sehr ähnliches Vorgehen findet außerdem bereits in Modellierungsprogrammen wie 3ds Max Anwendung. Nachteilig ist jedoch auch hier die höhere Anzahl an Arbeitsschritten im Vergleich zum Aufspannen über zwei gegenüberliegende Eckpunkte.

Zylinder Für die Erstellung eines zylindrischen Volumenkörpers bietet es sich ebenfalls an zunächst die Grundfläche zu erzeugen, um anschließend die Höhe festzulegen. Für Ersteres lassen sich dabei die Erkenntnisse aus der Erzeugung kreis- bzw. ellipsenförmiger Flächen nutzen, so dass sich das Aufspannen der Grundfläche über eine diagonale Linie empfiehlt. Die Festlegung der Höhe erfolgt dann, wie im vorherigen Abschnitt

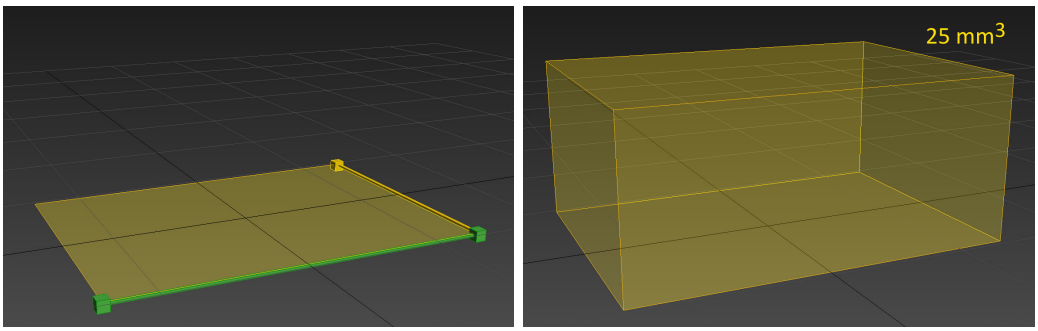


Abbildung 4.20: Aufspannen der Grundfläche des Quaders über drei Eckpunkte (links) und anschließendes Bestimmen der Höhe des Volumens (rechts)

Abbildung 4.21: Darstellung eines maximalen Abstandes zwischen zwei Strukturen

beschrieben, über den Abstand des Pointers zur Grundfläche. So wird in nur drei Arbeitsschritten ebenfalls eine hohe Flexibilität und Genauigkeit bei der Erstellung erreicht, wodurch Nachbearbeitungen kaum noch nötig sind.

Semi-automatische Messungen

Während die manuelle Vermessung dem Arzt zwar die volle Kontrolle über die Messungen ermöglicht, besitzt sie einen Nachteil: sie kann je nach Art der Vermessung recht aufwendig werden. Dies gilt insbesondere für das Messen von komplexen Flächen und Volumen über das Anpassen der entsprechenden Flächenformen und Volumenkörper. Für diese Fälle wären semi-automatische Vermessungen eine Alternative, da sie bereits während der Beobachtungen am Universitätsklinikum Magdeburg (siehe Abschnitt 3.1) als mögliche, effizientere Lösung Erwähnung fanden.
So sollen durch simples Selektieren der jeweiligen Struktur automatisch die Fläche, das Volumen oder die maximale Ausdehnung berechnet werden. Dies kann entweder durch das Zeigen mit dem Pointer und der Bestätigung durch Pinch-to-Click im entsprechenden Vermessungsmodus oder mithilfe des Kontextmenüs zu jeder Zeit geschehen. Letzteres ist dabei am effizientesten, da nicht zusätzlich in den entsprechenden Modus gewechselt werden muss. Die Berechnung des minimalen bzw. maximalen Abstandes zwischen zwei Strukturen soll ebenfalls möglich sein und durch deren Selektion erfolgen. Auf diese Weise lassen sich einige Messungen durch das Wegfallen diverser Arbeitsschritte einfacher und effizienter gestalten. Des Weiteren wurde diese Vorgehensweise während der Beobachtungen als bevorzugt beschrieben.
Dabei bleibt das Visualisierungskonzept für die Flächen- und Volumenmessung gleich, während minimale bzw. maximale Abstände entsprechend der Darstellung in Abbildung 4.21 gekennzeichnet sind, um nicht mit normalen Distanzen verwechselt zu werden. Die Visualisierung der maximalen Ausdehnung (siehe Abbildung 4.22) hingegen orientiert sich an der Arbeit von Rössling et al.[1]. Sie stellten zwar die Darstellung mithilfe einer Bounding Box als irritierend heraus, zeigten jedoch nur die Kanten selbiger, während die Flächen transparent blieben. Dies könnte möglicherweise zu einer schlechteren Wahrnehmung geführt haben und soll mithilfe von semitransparenten Flächen in der aktuellen Visualisierung behoben werden. Darüber hinaus ist die entsprechende Markierung der aktuell vom Pointer angezeigten Strukturen während der Selektion im jeweiligen semi-automatischen Vermessungsmodus wichtig. Auf diese Weise soll dem Nutzer mitgeteilt

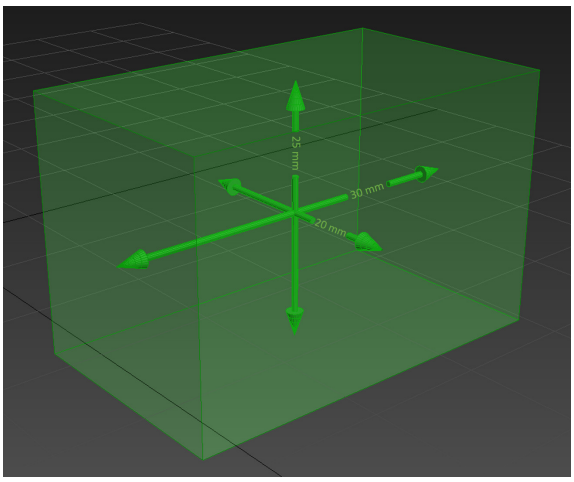


Abbildung 4.22: Visualisierung zur Darstellung der maximalen Ausdehnung einer Struktur

werden, welches Objekt bei einer Bestätigung zur Vermessung erfasst werden würde. Außerhalb des Vermessungsmodus, bei der Selektion über das Kontextmenü, ist dies allerdings nicht möglich, da vorherige Markierungen bei anderen Interaktionsaufgaben störend sein könnten.

4.4.8 Bearbeiten und Löschen von Messungen

Zusätzlich zur Erstellung lässt sich jede Art von Messung bearbeiten, um etwaige Fehler auszugleichen oder ggf. Feinheiten anzupassen. So können auch semi-automatisch erstellte Messungen nachbearbeitet werden, wenn der Nutzer mit der automatischen Erfassung der Struktur nicht zufrieden ist.
Um eine Messung zu ändern, kann der Anwender diese entweder selektieren und den Bearbeitungsmodus per Kontextmenü auslösen oder zuerst in den entsprechenden Modus wechseln, um anschließend die gewünschte Messung auszuwählen. Ist dies getan, können andere Messungen nur selektiert und bearbeitet werden, indem der aktuelle Bearbeitungsvorgang über das Kontextmenü abgebrochen oder bestätigt wird. Dieses Vorgehen verhindert versehentliche Interaktionen und bietet dennoch die Möglichkeit eines schnellen Wechsels zwischen den zu bearbeitenden Messungen sowie zwischen der Bearbeitung und anderen Funktionen.
Nach der Selektion zur Anpassung wird die Messung außerdem wie in Abbildung 4.23 rot gefärbt und die während der Erstellung platzierten Messpunkte markiert. Diese können nun per Pointer angezeigt und mit Pinch-to-Click selektiert werden, woraufhin es möglich ist sie neu zu positionieren. Letzteres erfolgt dabei auf dieselbe Weise wie das anfängliche Platzieren der Messpunkte. Die Selektion selbiger wird des Weiteren dadurch vereinfacht, dass stets derjenige Punkt ausgewählt wird, der dem Pointer am nächsten

Abbildung 4.23: Beispielhafte Visualisierung während der Messungsbearbeitung mit selektiertem Messpunkt

ist. Auf diese Weise ist es nicht notwendig exakt den Messpunkt zu treffen, wodurch mit weniger Aufwand eine höhere Effizienz erzielt wird.
Da sich die Bearbeitung von Messpunkten im Falle der Flächen- und Volumenmessungen komplexer gestaltet kann als bei den Distanz- und Winkelmessungen existiert für sie außerdem eine weitere Bearbeitungsmöglichkeit. So lassen sich Flächen- und Volumenmessungen auch mithilfe der in Abschnitt 4.4.5 beschriebenen Interaktionsmöglichkeiten zur Manipulation von Objekten anpassen. Des Weiteren ist dies die einzige Möglichkeit durch Aufziehen erzeugte Objekte sowie die Höhe von Flächen und Volumenkörpern zu verändern.

Darüber hinaus können Messungen auch gelöscht werden. Dies erfolgt stets über das Kontextmenü, da es eine schnellere Interaktion ermöglicht als das Auslösen des Löschvorgangs über eine gestische Manipulation. Auch lassen sich mithilfe der Mehrfachselektion mehrere Messungen gleichzeitig selektieren und löschen, so dass nicht jede Messung einzeln entfernt werden muss.

4.4.9 Rückgängig machen und Wiederholen von Aktionen

Um eine flexible und gegen Fehler robuste Arbeitsweise zu ermöglichen, ist das Rückgängig machen und Wiederholen von Aktionen unabdingbar. Zu diesem Zweck kann sowohl das Setzen von Messpunkten, das Aufziehen und Manipulieren von Messungen sowie das Löschen von Objekten rückgängig gemacht bzw. wiederholt werden.
Der Vorgang wird dabei über eine Geste ausgelöst. Dies ist effizienter als das Betätigen eines Buttons, da der Nutzer den Pointer zu diesem erst hinbewegen müsste. Die Grundposition für die verwendete Freihandgeste ist dabei in Abbildung 4.24 zu sehen. Von ihr ausgehend ist ein Kreis entgegen dem Uhrzeigersinn zu beschreiben, um eine Aktion rückgängig zu machen, während ein Kreis mit dem Uhrzeigersinn diese wiederholt. Die sogenannte Circle-Geste wurde gewählt, da sie im Rahmen einer Nutzerstudie von einer Mehrzahl der Probanden als am natürlichsten bestimmt wurde [51]. Allerdings beinhaltet die dort gezeigte Geste nur das Ausstrecken eines Fingers. Eine derartige Haltung konnte im aktuellen Konzept jedoch nicht umgesetzt werden, da dies zu Konflikten mit der Zeigegeste geführt hätte.

Abbildung 4.24: Grundhaltung der Geste zum Rückgängig machen und Wiederholen von Aktionen

4.5 Nutzerunterstützung

Bisher ist die Präzision von Zeigegesten eher gering. Dies führt zu Ermüdungen, wenn akkurate Eingaben erforderlich sind. Aus diesem Grund ist eine Erhöhung der Genauigkeit erforderlich. Zu diesem Zweck wurden Mechanismen entwickelt, die den Nutzer bei der Eingabe unterstützen und so deren Präzision erhöhen. Diese sollen nachfolgenden beschrieben werden.

4.5.1 Exponentielle Glättung

Das Prinzip der einfachen exponentiellen Glättung besteht darin, dass eine gewisse Anzahl n an zurückliegenden Werten y_{t-i} sowie der aktuelle Wert y_t gewichtet und aufaddiert werden. Dabei ist die Gewichtung umso kleiner desto älter der jeweilige Wert ist. Das Ergebnis der Berechnung ist schließlich eine Vorhersage über den nächsten Wert y_{t+1}, wobei auch Rauschen aus den Daten herausgefiltert wird [80]. Die entsprechende Formel lautet dabei wie folgt: [81]

$$y_{t+1} = \sum_{i=0}^{n} \alpha(1-\alpha)^i y_{t-i} \tag{4.1}$$

Die Verteilung der Gewichte hängt hier von der Glättungskonstanten α ab. Diese kann

(a) (b)

Abbildung 4.25: Visualisierung des Snappings beim Erreichen der Sichtweite eines Snap-Punktes (a) und beim Erreichen der Snapping-Reichweite (b)

Werte zwischen 0 und 1 annehmen. Desto größer sie ist umso mehr Gewicht erhalten die neueren Werte. Dies führt zu einer schnelleren Reaktion auf die aktuelle Wertentwicklung, während die Rauschfilterung verringert wird. Eine kleineres α hingegen führt zu einer besseren Rauschfilterung und einer zögerlicheren Reaktion auf die aktuellen Werte. Die einfache exponentielle Glättung eignet sich dabei zur Vorhersage von Daten, die keinem speziellen Trend oder Muster folgen [81] und kommt daher auch für die Position der Hand sowie des Pointers infrage. Diese werden durch die rauschunterdrückende Wirkung stabilisiert, indem kleine, schnelle Zitterbewegungen entfernt werden. Auf diese Weise ist ein exaktes ruhig Halten der Hände nicht mehr notwendig.

4.5.2 Snapping

Der Einsatz von Snapping bei der Verwendung von Freihandgesten wird von mehreren vorhergehenden Arbeiten empfohlen [31, 56, 82] und findet daher auch im aktuellen Konzept Anwendung. So sollen Genauigkeit und Komfort bei der Platzierung der Messpunkte weiter gesteigert werden.
Die Funktionsweise des Snappings besteht dabei darin, dass der Messpunkt während der Positionierung ab einem bestimmten Mindestabstand zu Kanten, Eckpunkten und Oberflächen zu diesen herangezogen wird. Dort wird er bis zum Erreichen einer gewissen Differenz zwischen Snap-Punkt und eigentlicher Pointerposition gehalten, so dass weniger Genauigkeit beim Zeigen notwendig ist. Darüber hinaus existiert ein weiterer Schwellenwert für die Distanz zwischen Pointer und Snap-Punkt, ab dem der mögliche, nächstgelegene Snap-Punkt bereits sichtbar wird, um dem Nutzer die Snappingmöglichkeit aufzuzeigen (siehe Abbildung 4.25). Die durch diesen Schwellenwert angegebene Sichtweite ist dabei stets größer als der zum Snappen benötigte Abstand. Das Snapping lässt sich außerdem über die bereits beschriebene Zustandsschaltfläche aktivieren und deaktivieren, um es in Situationen ausschalten zu können, in denen es eher hinderlich erscheint.

4.6 Auswahl des Eingabegerätes

Nachdem in den vergangenen Abschnitten die Konzepte für das Interface, die Interaktionstechniken und die Nutzerunterstützung beschrieben worden sind, soll nun die Auswahl eines angemessenen Eingabegerätes zur optischen Gestenerkennung stattfinden. Dabei wurde zunächst eine Vorauswahl an Geräten getroffen, um nur diejenigen in den Vergleich miteinzubeziehen, die grundlegende Anforderungen erfüllen. Auf diese Weise konnten der Mechaless Sensor [83] und das Myo Armband [84] ausgeschlossen werden, da sie weder eine kontinuierliche Positionserfassung im dreidimensionalen Raum noch die Umsetzung aller erforderlichen Gesten ermöglichen. Ersteres ist für die Umsetzung des Raycasting-Ansatzes der Zeigegeste unentbehrlich und somit zusammen mit der fehlenden Erkennungsmöglichkeit der benötigten Gesten ein Ausschlusskriterium. Währenddessen erschienen der NUIMO von Senic [85] sowie die Intel RealSense Kamera [86] vielversprechend, waren jedoch zu Beginn der Umsetzung des Prototyps noch nicht erhältlich.
So verbleiben der Leap Motion Controller [87], die Kinect [88] und die Depthsensor 525 Kamera [89, 90], welche in Tabelle 4.1 vergleichend dargestellt werden. Basierend auf den vorhandenen Daten fiel die Entscheidung auf den Leap Motion Controller in der Version V2. Dabei waren seine überlegene Genauigkeit und umfangreiche, vordefinierte Unterstützung der entwickelten Gesten entscheidend. Zwar könnten die Funktionalitäten der Kinect bezüglich der Gestenunterstützung durch Softwarebibliotheken von Drittanbietern erweitert werden, doch mit ihrem weitaus kleineren Sichtwinkel und der sehr viel geringeren Genauigkeit ist sie dem Leap Motion Controller dennoch unterlegen.

	Leap Motion Controller [91, 92, 95, 96]	**Kinect [93, 97, 98, 99]**	**Depthsense 525 [89, 94]**
Maximale Positionsabweichung	2,5 mm	15 mm	13,99 mm
Maximale Samplingrate	39 Hz	30 Hz	60 Hz
Interaktionsdistanz	25 mm	0,5 m	0,15 m
Interaktionsraumtiefe	575 mm	4,5 m	0,85 m
Sichtwinkel (HxV)	150°x120°	70°x60°	63.2°x49.3°
Granularität erkannter Gelenke	Handmitte und Gelenke der Finger	Handgelenk, Handbasis, Handspitzen und Daumen	Handmitte und zugehörige Finger

Vordefinierte Gestenunterstützung	Alle	Faustgeste	Faust- und Pinch-Geste

4.7 Zusammenfassung

Das aktuelle Kapitel diente der Herleitung und Erläuterung der für die Vermessung von medizinischen Strukturen notwendigen Interaktionskonzepte. Dabei wurden zunächst allgemeine Entwurfsgrundsätze für die Entwicklung angemessener Freihandgesten aufgestellt. So sollen Kommandosprachen und statische Gesten vermieden werden, während bekannte Interaktionskonzepte und eine beidhändige Interaktion zu bevorzugen sind. Außerdem gilt es Gesten stets aus der egozentrischen Perspektive zu entwickeln und ergonomische Kriterien einzuhalten.
Anschließend wurde das für die Interaktionstechniken benötigte Interface-Konzept vorgestellt und die Entwürfe für das visuelle Feedback präsentiert. Letztere beinhalten die visuelle Repräsentation der Gesten sowie den Gesten-Guide, welcher den Nutzer bezüglich der im aktuellen Zustand des Systems ausführbaren Freihandgesten leitet. Die visuelle Gestenrepräsentation soll dem Anwender dabei sowohl ein Feedback über die jeweils erkannten Freihandgesten als auch die entsprechende Position der Hände im Interaktionsraum geben.
Nachdem auf diese Weise die Basis für die Interaktionstechniken geschaffen wurde, sind diese ausführlich erarbeitet und beschrieben worden. So bilden die Zeige- und die Pinch-Geste ein Pendant zu der Kombination aus Mauszeiger und Maustasten, während mithilfe zweier Faustgesten die Kamerasteuerung erfolgt. Außerdem wird die Navigation durch die 2D-Schichtdaten mithilfe einer Wischgeste ermöglicht. Darüber hinaus kann die Pinch-Geste zur Manipulation von bestehenden Messungen durch Verschieben, Skalieren und Rotieren genutzt werden. Diese Interaktionen stellen die Grundlage für das Erstellen, Bearbeiten und Löschen der Messungen dar, so dass diese daraufhin ebenfalls erläutert werden konnten. Den Abschluss für die Interaktionstechniken bildete schließlich die Circle-Geste zum Rückgängig machen und Wiederholen von Aktionen.
Des Weiteren wurden Mechanismen zur Unterstützung des Nutzers bei der Steuerung mithilfe der Freihandgesten entwickelt. So wird die Pointerposition durch eine exponentiellen Glättung stabilisiert und für die Platzierung der Messpunkte ein Snapping-Mechanismus zur Verfügung gestellt, der die Genauigkeit und Effizienz weiter erhöhen soll.
Abschließend erfolgte die Auswahl eines für die Umsetzung der entwickelten Konzepte geeigneten Eingabegerätes zur optischen Gestenerkennung. Dabei fiel die Wahl auf den Leap Motion Controller, da dieser den anderen Geräten sowohl in seiner Genauigkeit als auch in der gebotenen Gestenunterstützung überlegen ist.

5 Implementierung

Während im vorangegangenen Kapitel die zur Vermessung medizinischer Strukturen notwendigen Interaktionskonzepte herausgearbeitet wurden, soll das aktuelle Kapitel die wichtigsten Details zu deren Umsetzung im Rahmen eines Prototyps klären. Dieser soll dazu dienen die entwickelten Interaktionstechniken mithilfe einer Nutzerstudie zu evaluieren. Dabei würde jedoch die Umsetzung aller beschriebenen Messungsarten den zeitlichen Rahmen der aktuellen Arbeit übersteigen. Aus diesem Grund wird neben den grundlegenden Interaktionen zur Steuerung der Benutzungsoberfläche sowie zur Navigation in den Bilddaten die Distanzmessung exemplarisch umgesetzt. Zusätzlich erfolgt die Implementierung einer zweiten Version des Prototyps, die mithilfe von Maus und Tastatur bedient wird. Sie wird in der späteren Evaluierung für den Vergleich zwischen der Steuerung über Freihandgesten und der Steuerung mit klassischen Eingabegeräten benötigt.
Im Folgenden soll zunächst die dabei verwendete Entwicklungsumgebung beschrieben und anschließend das Vorgehen zur Gestenerkennung erläutert werden. Nachdem auf diese Weise die Grundlagen zur Handhabung des Prototyps geschaffen sind, erfolgt die Darstellung der Mechanismen zur Nutzerunterstützung sowie der alternativen Steuerung über Maus und Tastatur.

5.1 Entwicklungsumgebung

Die Entwicklung des Prototyps erfolgt mithilfe der Unity Game Engine in Kombination mit Microsoft Visual Studio Community 2015. Bei Unity handelt es sich dabei um eine umfassende Software zur Entwicklung von Spielen und anderen Anwendungen mithilfe eines grafischen Editors und Scripten. Ersteres wird nicht nur genutzt, um die visuelle Umsetzung (z. B. das Erstellen der Benutzungsoberfläche) zu ermöglichen, sondern bietet auch die Möglichkeit benötigte Ressourcen wie Grafiken, 3D-Modelle, Audio-Dateien, Vorlagen, Scripte und ähnliches zu verwalten. Hinzu kommt, dass viele Funktionalitäten, die zur Erstellung von Spielen benötigt werden, bereits vorhanden sind. Beispielhaft seien hier die Physik-Engine zur Simulation physikalischer Vorgänge und die Organisation sowie Manipulation von 2D- und 3D-Elementen in entsprechenden Koordinatensystemen genannt. Die Scripte dienen dabei der Implementierung der Programmlogik und sollen im Falle des Prototyps in C# programmiert werden. Der große Funktionsumfang und die daraus resultierende Möglichkeit einer schnelleren Entwicklung des Prototyps waren

maßgebend bei der Entscheidung für die Unity Game Engine.
Des Weiteren werden für die Steuerung mit dem Leap Motion Controller das Leap Motion Windows Software Development Kit (SDK) 2.3.1 sowie die Leap Motion Core Assets 3.0 genutzt. Letztere sind für die Verwendung in Unity erforderlich. Außerdem wurde das ältere SDK dem neuesten vorgezogen, da es eine umfangreichere vordefinierte Gestenunterstützung bietet als sein Nachfolger. Zum Laden der DICOM-Bilddaten für die Darstellung der medizinischen Strukturen wird darüber hinaus ein auf der EvilDICOM-Bibliothek basierendes Framework aus einer studentischen Vorarbeit verwendet.

5.2 Gestenerkennung

Die Gestenerkennung bildet die Grundlage zur Steuerung des Prototyps. Die dabei verwendeten Algorithmen basieren auf den vom Leap Motion Controller gelieferten Handdaten und sollen im Folgenden näher erläutert werden.

5.2.1 Zeigegeste

Für die Detektion der Zeigegeste wird eine Funktion des Leap Motion Controllers genutzt, die es ermöglicht zu erkennen, welche Finger aktuell ausgestreckt sind. So muss lediglich abgefragt werden, ob der Zeigefinger der einzige gestreckte Finger ist.
Währenddessen lässt sich mithilfe der Position der Zeigefingerspitze und des oberen Endes des Mittelhandknochens sowie der Entfernung zwischen Leap Motion Controller und Bildschirm das durch die Zeigegeste erfolgende Raycasting realisieren. Die benötigten Positionen der Gelenke werden dabei vom Leap Motion Controller bereitgestellt. Außerdem geht der Berechnung die Annahme voraus, dass sich das Eingabegerät mittig vor dem Monitor befindet. Basierend auf diesen Informationen wird die Position der Benutzungsoberfläche in Unity mit der des Bildschirms gleichgesetzt sowie die erfassten Gelenkpositionen so in das Unitykoordinatensystem transformiert, dass diese dasselbe räumliche Verhältnis zum Interface haben wie die realen Hände des Nutzers zum Bildschirm. Anschließend lässt sich die Richtung, in die der Zeigefinger zeigt, über den Vektor zwischen dem oberen Ende des Mittelhandknochens und der Spitze des Zeigefingers berechnen. Des Weiteren kann die Position der Zeigefingerspitze durch die zuvor erfolgte Transformation als direkter Ursprung für das Raycasting genutzt werden. Mithilfe der in Unity vordefinierten Raycastingfunktion wird schließlich anhand des Ursprungs und der Zeigerichtung ein Strahl berechnet, dessen Kollisionspunkt mit der Benutzungsoberfläche die Position des Pointers auf dieser bestimmt.

5.2.2 Pinch-Geste

Bei der Erkennung der Pinch-Geste kann auf die vom Leap Motion Controller erfasste, sogenannte „Pinch-Stärke“ zurückgegriffen werden. Diese gibt an, wie stark der Daumen und ein beliebiger anderer Finger der Hand zusammengedrückt werden und kann Werte zwischen 0 (kein Pinch) und 1 (vollständiger Pinch) annehmen. Um anhand dieser Informationen feststellen zu können, ob eine Pinch-Geste zwischen Daumen und Zeigefinger ausgeführt wird, ist ein zweistufiger Test notwendig. Dabei wird zunächst geprüft, ob die Pinch-Stärke hoch genug ist. Um hier sowohl eine zu empfindliche als auch eine zu zögerliche Erkennung zu vermeiden, wurden mehrere Schwellenwerte erprobt und anhand der gemachten Beobachtungen ein Wert von 0,8 als geeignet ermittelt. Ist dieser überschritten, muss ausgeschlossen werden, dass es sich um keine andere Pinch-Geste als die zwischen Zeigefinger und Daumen handelt. Dies geschieht, indem sichergestellt wird, dass die Entfernung zwischen diesen beiden Fingern am geringsten ist.

5.2.3 Kamerasteuerung

Die Kamerasteuerung setzt sich aus mehreren Gesten zusammen. So bildet die Faustgeste die Grundlage und muss von beiden Händen ausgeführt werden, um die Grundhaltung für die Kamerasteuerung herzustellen. Anschließend ist eine von vier verschiedenen Gesten durchzuführen, damit eine Translation, Skalierung oder Rotation der Ansicht stattfindet (siehe Abschnitt 4.4.5). Die hierfür notwendigen Mechanismen zur Detektion sollen nachfolgend beschrieben werden.

Grundhaltung

Für die Detektion der Faustgeste werden ebenfalls die vom Leap Motion Controller gelieferten Daten genutzt. So bildet die sogenannte „Griffstärke“ den Grad des Greifens mit der Hand auf eine Skala von 0 bis 1 ab. Dabei steht 0 für eine geöffnete Hand und 1 für eine geschlossene Faust. Hier fand daher ebenfalls eine Erprobung verschiedener Schwellenwerte statt, um einen geeigneten Wert zu bestimmen, dessen Überschreitung eine Faustgeste anzeigt. Dabei ergab sich ein Schwellenwert von 0,8. Dieser sorgt dafür, dass die Geste weder zu früh noch zu spät erkannt wird. Die Detektion der Grundhaltung der Kamerasteuerung erfolgt schließlich, sobald für beide Hände eine Faustgeste erfasst ist.

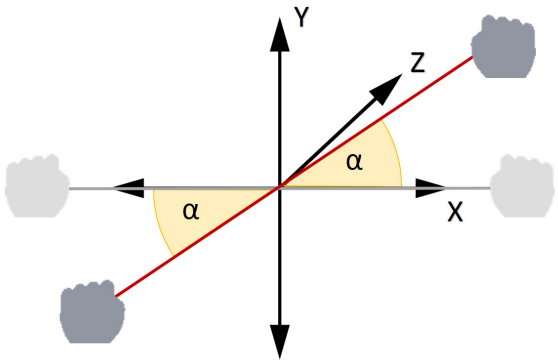


Abbildung 5.1: Beispielhafte Visualisierung der Bestimmung des Rotationswinkels α bei der Rotation der Ansicht um die z-Achse anhand der entsprechende Geste. Die hellgrauen Hände markieren die Position vor der Manipulation, während die dunkelgrauen Hände die Positionierung danach darstellen. Der zwischen diesen beiden Stellungen messbare Winkel legt fest, um wie viel Grad um die z-Achse rotiert wird.

Kameramanipulation

Nachdem die Grundhaltung eingenommen ist, kann die Kamera gesteuert werden. Um dabei festzustellen, welche der Manipulationsgesten ausgeführt wird, findet eine kontinuierliche Berechnung des Mittelpunktes sowie des Abstands zwischen den Händen und deren Veränderung über die Zeit statt.
So erfolgt das Verschieben der aktuellen Ansicht auf die Bilddaten, indem sich nur der Mittelpunkt zwischen den Händen verändert, während der Abstand annähernd gleich bleibt. Demgegenüber wird heran- bzw. herausgezoomt, wenn sich der Mittelpunkt zwischen den Händen kaum bewegt und stattdessen eine Änderung des Abstands stattfindet. Darüber hinaus wird zur Erkennung der Kamerarotation die Distanz zwischen den Händen für jede Achse des Koordinatensystems einzeln erfasst. Dabei führt eine signifikante Veränderung des Abstandes auf der y- oder z-Achse zur Detektion einer Rotation. Findet diese auf der z-Achse statt, wird um die y-Achse rotiert, während bei einer Abstandsänderung auf der y-Achse weiter unterschieden werden muss. Hier ist zu prüfen, ob die Hände so zu einander versetzt sind, dass die Bewegung der im Konzept beschriebenen Rotation um die x-Achse entspricht (siehe Abbildung 4.10 in Abschnitt 4.4.5). Dies wird anhand des Abstandes der Hände auf der z-Achse gemessen, so dass hier ein gewisser Schwellenwert überschritten werden muss, ab der die Lage des Stabs als Schräglage gilt. Ist dies nicht der Fall, wird eine Rotation um die z-Achse ausgelöst. Der Umfang der Rotation wird dabei über den Winkel zwischen der Position der Hände vor der Veränderung und ihrer Position nach der Veränderung berechnet. Ein Beispiel für diese Vorgehensweise bei einer Rotation um die z-Achse ist in Abbildung 5.1 zu sehen. Sie lässt sich analog auf Rotationen um die x- und y-Achse übertragen.

Für alle Gesten zur Kameramanipulation gilt, dass sowohl die Änderung des Mittelpunktes als auch die des Abstandes einen gewissen Schwellenwert überschreiten muss, um als gewollte Bewegung wahrgenommen und einbezogen zu werden. So soll die Gestenerkennung weniger empfindlich gestaltet werden. Nach der Detektion der jeweiligen Manipulation muss die Veränderung der räumlichen Verhältnisse jedoch keinen Schwellenwert mehr überschreiten, da das Auslösen eine andere Art der Kamerasteuerung erst nach Beendigung der aktuellen Geste möglich ist. Auf diese Weise soll eine kontinuierliche Manipulation gewährleistet werden.

5.2.4 Wischgeste

Die Detektion der Wischgeste sollte ursprünglich mithilfe der vordefinierten Swipe-Geste des Leap Motion Controllers erfolgen. Diese wird jedoch nur im Falle einer Auf- bzw. Abwärtsbewegung der Hand zuverlässig erkannt. Aus diesem Grund wird ein eigener Algorithmus implementiert, der im Folgenden beschrieben werden soll.

Die Basis der Wischgeste bildet die senkrecht über dem Leap Motion Controller gehaltene, flache Hand. Sie wird anhand eines dreistufigen Detektionsverfahrens erkannt. Dabei ist zunächst zu prüfen, ob alle Finger oder zumindest alle Finger außer dem Daumen gestreckt sind. Letzteres liegt darin begründet, dass der flach an die Hand gelegte Daumen nicht immer einwandfrei als ausgestreckt klassifiziert wird. Des Weiteren ist sicherzustellen, dass sich die Hand tatsächlich senkrecht über dem Eingabegerät befindet. Dies geschieht, indem der vom Leap Motion SDK bereitgestellte Richtungsvektor der Hand abgefragt wird. Dabei fällt der Test nur positiv aus, wenn dieser nach links oder rechts zur Seite zeigt. Abschließend gilt es zu prüfen, ob der Daumen mit der Hand annähernd eine Fläche bildet. Hierfür werden die vom Leap Motion Controller erfassten Richtungsvektoren der Finger genutzt. Auf diese Weise lässt sich sicherstellen, dass der Daumen nicht zur linken bzw. rechten Seite zeigt. Diese Vorgehensweise ist außerdem zur Unterscheidung zwischen Wisch- und Pinch-Geste essenziell.
Ist die Grundhaltung detektiert, muss festgestellt werden, ob es sich im weiteren Verlauf der Bewegung um eine Wischgeste von links nach rechts oder rechts nach links handelt. Dies wird anhand des Winkels zwischen dem gebeugten Zustand der Hand und der gerade gehaltenen Grundstellung geprüft. Ist dieser negativ, handelt es sich um ein Wischen von rechts nach links. Ein positiver Winkel hingegen beschreibt eine Wischgeste von links nach rechts.
Dabei sind außerdem zwei Schwellenwerte in jede Richtung zu beachten. So zeigt das Überschreiten des ersten an, dass die Hand genug gebeugt wurde, um eine Wischgeste auszulösen, während die Überschreitung des zweiten das Eintreten in eine gehaltene Wischgeste zum Navigieren über mehr als eine Schicht anzeigt. Diese beiden Schwel-

lenwerte sind im aktuellen Prototyp mit den Werten von 15 und 30 Grad Unterschied zur Grundhaltung belegt. Sie konnten anhand einer praktischen Prüfung verschiedener Winkel als geeignet bestimmt werden. So stellt der erste Schwellenwert sicher, dass nicht schon bei einer kleinen Bewegung eine Navigation durch die Schichtdaten erfolgt, während der zweite so gewählt wurde, dass er zum ersten Wert den größtmöglichen Unterschied aufweist und gleichzeitig komfortabel durch eine Beugung des Handgelenks zu erreichen ist. Die Wahl der Beugungswinkel wird außerdem durch die Handerkennung des Leap Motion Controllers begrenzt, die sich bei zu großen Winkeln häufig als unzuverlässig erweist.

5.2.5 Circle-Geste

Auch für die Erkennung der Circle-Geste sollte zunächst die vom Leap Motion Controller vordefinierte Implementierung genutzt werden. Doch das Ergebnis ist aufgrund einer sehr empfindlichen Detektion der Geste nicht zufriedenstellend, weshalb ebenfalls die Entscheidung für eine eigene Umsetzung erfolgte. Diese wird nachfolgend geschildert.

Die Grundhaltung der Circle-Geste bildet eine parallel zum Eingabegerät gehaltene Hand, bei der nur der Zeige- und Mittelfinger ausgestreckt sind. Ob diese Haltung eingenommen wird, kann ähnlich wie bei der Wischgeste getestet werden. So wird anhand des Richtungsvektors der Hand geprüft, ob deren Handfläche tatsächlich nach unten zeigt, während die gestreckten Finger wie bereits beschrieben erfasst werden.
Ausgehend von der Grundgeste erfolgt die Berechnung des mit den beiden Fingern zu beschreibenden Kreises. Hierfür ist die Angabe des minimalen Radius notwendig. Dieser wurde im Rahmen des Prototyps auf vier Grad festgelegt, um zu vermeiden, dass der Nutzer eine zu weit ausholende Bewegung ausführen muss. Während der folgenden Berechnungen wird außerdem davon ausgegangen, dass die Startposition der Circle-Geste den in Abbildung 5.2 links dargestellten Punkt auf dem Kreis beschreibt. Basierend auf dieser Information lässt sich der Mittelpunkt des Kreises definieren und anschließend in 90-Grad-Schritten Kontrollpunkte festlegen, die von der Hand nacheinander überschritten werden müssen.
Ein Kontrollpunkt gilt dabei als erreicht, wenn der Abstand zwischen Hand und Kreismittelpunkt mindestens so groß ist wie der als minimal festgelegte Radius und die Position der Hand annähernd denselben Winkel zum Mittelpunkt besitzt wie der Kontrollpunkt. Um Letzteres zuverlässig feststellen zu können, wird ein Toleranzbereich festgelegt, in dem der Winkel des Kontrollpunktes als erreicht gilt (siehe Abbildung 5.2 rechts). Aktuell liegt hier die empirisch ermittelte Toleranz bei ± 6 Grad. Dabei wird anhand des ersten, passierten Kontrollpunkts bestimmt, ob die Circle-Geste entgegen oder im Uhrzeigersinn verläuft. Sind alle Kontrollpunkte überschritten, wird die entsprechende Circle-Geste als Eingabe akzeptiert. Wird jedoch vor dem Erreichen des letzten Kontrollpunktes die Startposition wieder passiert, erfolgt ein Zurücksetzen des Fortschritts auf null. Anschließend

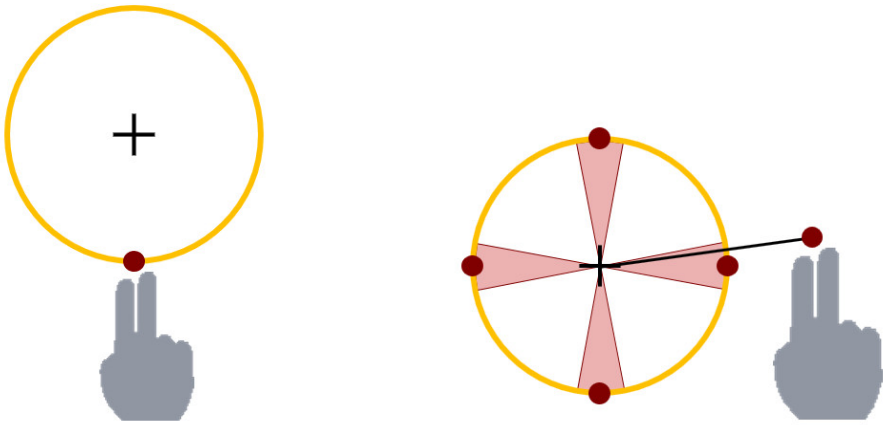


Abbildung 5.2: Startposition der Circle-Geste (links) und Toleranzbereiche der Kontrollpunkte mit erreichtem, ersten Punkt (rechts)

gilt es erneut den Verlauf der Circle-Geste anhand des nächsten, überschrittenen Kontrollpunktes zu bestimmen. Auf diese Weise wird eine flexible Korrektur ermöglicht, falls die Richtung während der Ausführung der Geste geändert werden soll.

5.3 Mechanismen zur Nutzerunterstützung

Nachdem im vorhergehenden Abschnitt die Algorithmen zur Gestenerkennung genauer erläutert wurden, kann nun zu den Mechanismen zur Unterstützung der Gesteninteraktion übergegangen werden. So soll zunächst die Filterung zur Stabilisierung der detektierten Positionen der Hand und ihrer Gelenke beschrieben werden, um anschließend das für das Snapping verwendete Vorgehen näher zu beleuchten.

5.3.1 Filterung der Handpositionen

Wie bereits in Abschnitt 4.5.1 dargelegt, wird für die Filterung der Positionen der Hand und ihrer Gelenke die exponentielle Glättung verwendet. Dabei gilt es zunächst zwei Werte zur Parametrisierung festzulegen: die Glättungskonstante α und die Anzahl der zu speichernden, zurückliegenden Werte der jeweiligen Position. Während Letztere für alle Gesten bei 15 liegt, unterscheidet sich die Glättungskonstante zwischen den Gesten. So liegt sie für den Pointer bei 0,35, da dieser zwar einer Stabilisierung bedarf, aber dennoch schnell genug auf die Eingaben des Nutzers reagieren muss. Für die Darstellung der Handposition während der Ausführung der anderen Gesten ist dies jedoch weniger wichtig, weil hier kein konkretes Ziel getroffen werden soll. Im Falle der Kamerasteue-

rung wäre eine zu empfindliche Reaktion sogar kontraproduktiv, weil sie beim Öffnen der Faustgeste ungewollte Manipulationen auslösen würde. Aus diesem Grund liegt die Glättungskonstante α hier bei 0,25. Die Parameter wurden dabei anhand von Tests und Beobachtungen bestimmt.
Basierend auf dieser Parametrisierung kann die Berechnung der gefilterten Handpositionen erfolgen. Die zurückliegenden Positionen werden dabei in einer Historie gespeichert. Diese wird sofort nach der Detektion einer Geste mit deren Startposition gefüllt, um Werte für die nachfolgende Berechnung bereitzustellen. Ob es sich dabei um die Position der Hand und ihrer Gelenke oder die durch das Raycasting erhaltene Pointerposition handelt, hängt dabei von der erkannten Geste ab. Die gefilterte Position wird in jedem Fall über die in Abschnitt 4.5.1 erläuterte Funktion ermittelt und danach der Historie hinzugefügt. Auf diese Weise kann sie für das weitere Vorgehen berücksichtigt werden. Anschließend gilt es die der Geste entsprechende Position kontinuierlich zu erfassen, zu glätten und den zurückliegenden Werten zuzuordnen. Kommt es während des Hinzufügens einer Position zur Historie zu einer Überschreitung der Anzahl zu speichernder Werte, wird der jeweils älteste entfernt, um Platz für den neuen Wert zu schaffen.
Der Filterungsprozess wird gestoppt, sobald die entsprechende Geste nicht mehr erkannt wird. Dies führt außerdem zur Löschung der Historie, da die Werte nun nicht mehr benötigt werden.

5.3.2 Snapping

Für die Umsetzung des Snappings ist zunächst die Berechnung der Pointerposition auf den Bilddaten sowie die Bereitstellung einer Datengrundlage für die Positionierung der Snap-Punkte notwendig. Letztere soll die Strukturen in den Bilddaten repräsentieren, an denen gesnappt werden kann (siehe Abschnitt 4.5.2). Sobald die entsprechenden Daten zur Verfügung stehen, kann der zum Pointer nächstgelegene Snap-Punkt ermittelt und basierend auf diesem ggf. ein Snapping-Vorgang durchgeführt werden. Die hierfür erforderlichen Prozesse sollen im Folgenden näher erläutert werden.

Berechnung der Pointerposition in den Bilddaten

In Abschnitt 5.2.1 wurde bereits beschrieben wie die Position des Pointers auf der Benutzungsoberfläche bestimmt wird. Auf dieser werden sowohl die 2D- als auch die 3D-Bilddaten mithilfe sogenannter „Rendering Textures“ dargestellt. Rendering-Texturen projizieren ein von einer Kamera aufgenommenes Bild in einer beliebigen Größe und an einer beliebigen Position auf das Interface. So existiert für jede der vier Bildansichten eine Kamera, die die originalen Bilddaten aus einer bestimmten Position heraus aufnimmt. Die Positionierung wird dabei vom Nutzer durch die Manipulation der jeweiligen Kamera bestimmt.

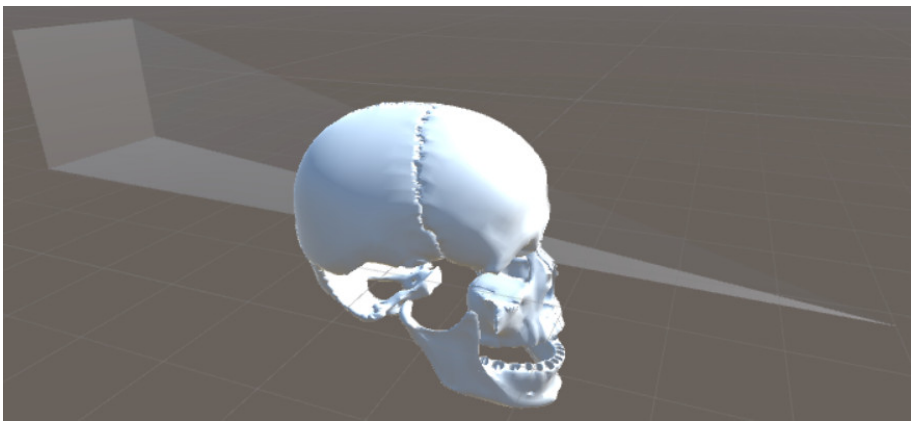


Abbildung 5.3: Darstellung des transparenten Ghostpointers auf den 3D-Bilddaten

Aufgrund dieser Konstellation entspricht die Pointerposition auf der Benutzungsoberfläche nicht der auf den Bilddaten. Um Letztere berechnen zu können, wird die Position auf der Benutzungsoberfläche normalisiert und mithilfe der jeweiligen Kameraposition sowie den Daten des dazugehörigen View Frustums auf die Bilddaten übertragen. Dabei wird auch die Ausdehnung des Pointers entsprechend berücksichtigt. Auf diese Weise entsteht neben dem Pointer auf der Benutzungsoberfläche zusätzlich ein Pointer auf den Bilddaten. Während dieser in den zweidimensionalen Bilddaten weiterhin als Kreisfläche dargestellt wird, ergeben die Berechnungen im dreidimensionalen Raum einen Pyramidenstumpf (siehe Abbildung 5.3). Letzteres müsste aufgrund der Kreisform des Pointers zwar ein Kegelstumpf sein, wurde jedoch aus Performancegründen durch den Pyramidenstumpf angenähert. Repräsentiert wird der Pointer daher in den Bilddaten als Fläche bzw. Volumenkörper. Er ist jedoch für den Nutzer unsichtbar und wird daher als „Ghostpointer“ bezeichnet. Der Ghostpointer erhält außerdem einen ebenfalls nicht sichtbaren, sogenannten „Collider“, der Kollisionsabfragen zwischen ihm und den Bilddaten sowie Objekten auf selbigen ermöglicht. Wichtig ist dies auch für die Selektion der fertigen Messungen und Messpunkte.

Berechnung der Datengrundlage

Bevor die Suche nach dem nächstgelegenen Snap-Punkt stattfinden kann, wird eine Datenstruktur benötigt, die die Positionen der möglichen Snap-Punkte beschreibt. Deren Berechnung unterscheidet sich abhängig davon, ob es sich um 2D- oder 3D-Bilddaten handelt.

So lassen sich Ecken und Kanten aus 2D-Bilddaten mithilfe des sogenannten „Sobel-Operators“ extrahieren [104]. Dabei sind die folgenden zwei Faltungskerne auf das Bild

anzuwenden:

$$\begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad und \quad \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \tag{5.1}$$

Die Anwendung von Faltungskernen geschieht durch eine diskreten Faltung, bei der die Pixel des Ergebnisbildes anhand der Formel 5.2 berechnet werden [105].

$$I^*(x,y) = \sum_{i=1}^{n} \sum_{j=1}^{n} I(x+i-a, y+j-a)k(i,j) \tag{5.2}$$

Hier stellt I das Bild dar, auf welches die Faltung angewendet wird, während $k(i,j)$ ein Element der Faltungsmatrix ist und a die Koordinate der Matrixmitte.
Um auf diese Weise Kantenbilder der 2D-Schichtdaten zu erhalten, wurde in der Umsetzung des Prototyps eine 3x3 Matrix aus Vektoren erstellt. Deren x-Werte enthalten den ersten in Formel 5.1 genannten Faltungskern, während der zweite in den y-Werten gespeichert ist. Anschließend wird die Matrix mithilfe der Formel 5.2 auf das jeweilige Schichtbild angewendet. So wird für jedes Pixel der Gradient in Form eines Richtungsvektors mit einem x- und einem y-Wert berechnet. Dessen Betrag stellt die Kantenstärke des jeweiligen Pixels dar [106] und wird daher ermittelt. Da jedoch nur stark ausgeprägte Kanten als Snap-Punkte infrage kommen, wird auf die Kantenstärke zusätzlich ein Schwellenwert von 0,5 angewendet. Dieser stellte sich im Test des Prototyps als guter Kompromiss zwischen zu vielen und zu wenigen erkannten Kanten heraus. Wird er überschritten, wird das Pixel im resultierenden Bild weiß dargestellt. Ist dies nicht der Fall, bekommt es die Farbe Schwarz. Außerdem wird die Position eines jeden Pixels, dessen Kantenstärke hoch genug ist, in einer Liste für Kantenpixel gespeichert, damit diese später nicht während jedes Snapping-Vorgangs erneut anhand des Ergebnisbildes ermittelt werden müssen. Diese Liste wird, genau wie die Kantenbilder, für jedes Schichtbild einzeln erstellt und bildet die Datengrundlage für das Snapping in diesem.

Dahingegen basiert das Snapping in den 3D-Bilddaten auf den Positionen der Eckpunkte des Dreiecksnetzes, welches das 3D-Modell der jeweiligen medizinischen Struktur ausmacht. Ein Mechanismus, der dafür sorgt, dass stets die Oberfläche des Modells getroffen wird, ist dabei nicht notwendig, da hierfür schon das Raycasting-Prinzip des Pointers sorgt. So wird mithilfe des Raycastings stets nur das nächste Objekt getroffen, so dass der Pointer sich nicht hinter der Oberfläche befinden kann.
In Unity besteht bereits eine einfache Importmöglichkeit für 3D-Modelle, die die dazugehörigen Eckpunkte in einer jederzeit abrufbaren Datenstruktur organisiert. Aus diesem Grund sind keine weiteren Verarbeitungsschritte erforderlich.

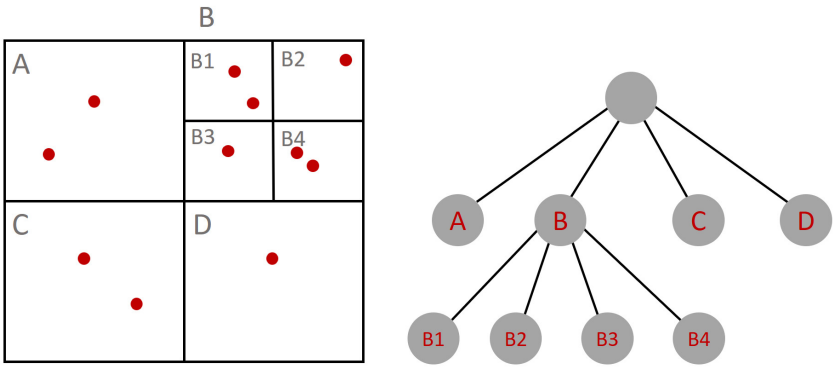


Abbildung 5.4: Aufbau eines Quadtrees basierend auf den Punkten in einer Region

Suche des nächstgelegenen Snap-Punktes

Nachdem die Positionen der möglichen Snap-Punkte sowohl für die 2D- als auch die 3D-Bilddaten zur Verfügung stehen, gilt es nun festzustellen, welcher Punkt der Pointerposition am nächsten ist. Dies soll anhand einer sogenannten „Nearest-Neighbor-Suche“ (dt. Nächster-Nachbar-Suche) geschehen. Um dabei eine gute Performance zu erzielen, wird im zweidimensionalen Raum mit Quadtrees und im dreidimensionalen Raum mit Octrees gearbeitet.
Bei Quad- bzw. Octrees handelt es sich um Baumstrukturen, die die Snap-Punkte in den Bilddaten räumlich sortieren. Abbildung 5.4 veranschaulicht das Prinzip des Aufbaus eines Quadtrees, welches sich analog auf den Aufbau eines Octrees übertragen lässt. Der einzige Unterschied besteht hier darin, dass der Raum pro Stufe aufgrund der zusätzlichen Dimension in acht Quader anstatt vier Quadrate aufgeteilt wird. Ob eine bestehende (Teil-)Region unterteilt wird, hängt davon ab, ob eine Maximalanzahl an Punkten in dieser überschritten ist. Die Anzahl sollte dabei möglichst klein sein und beträgt daher für die 2D-Schichtdaten eins. Für das 3D-Modell liegt sie pro Teilregion bei zehn, da das Modell aus vielen tausend Eckpunkten besteht und eine kleinere Anzahl somit zu sehr vielen Teilungen mit einem sehr hohen Rechenaufwand führen würde.
Mithilfe der Quad- und Octrees gilt es anschließend beginnend bei der Wurzel zu bestimmen, in welcher, durch einen Blattknoten repräsentierten, Teilregion sich die Position des Pointers befindet. Auf diese Weise muss für das Finden des nächsten Nachbarn nur zwischen den wenigen, dort befindlichen Snap-Punkten verglichen werden.
Für die Entscheidung zwischen den verschiedenen Teilregionen sollen außerdem nicht alle in diesen befindlichen Punkte auf ihre Nähe zum Pointer getestet werden. Aus diesem Grund werden für jede Teilregion repräsentative Positionen gewählt. Deren Auswahl erfolgt dabei so, dass pro Achse der Punkt bestimmt wird, der auf dieser den geringsten Abstand zur Grenze zwischen den Teilregionen besitzt. Auf diese Weise werden im zweidimensionalen Raum zwei und im dreidimensionalen Raum drei Positionen festgelegt, die dann reduzierte Teilregionen bilden (siehe Abbildung 5.5). Diese enthalten alle Snap-Punkte, die sich in der jeweiligen Teilregion befinden, so dass sich anhand des Abstandes

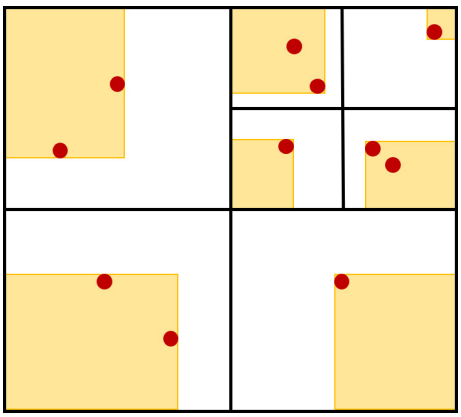


Abbildung 5.5: Visualisierung der reduzierten Teilregionen eines Quadtrees

zu der reduzierten Teilregion die Nähe zu allen in ihr befindlichen Punkten bestimmten lässt. Daher wird der Pointer derjenigen Teilregion zugeordnet, zu deren reduzierten Teilregion er den geringsten Abstand besitzt. Berechnet wird dieser anhand der folgenden Formeln, in denen $Mitte$ für die Mitte der reduzierten Teilregion steht.

$$d_x = |Mitte_x - Pointer_x| - 0.5 \cdot Breite \tag{5.3}$$
$$d_y = |Mitte_y - Pointer_y| - 0.5 \cdot Höhe \tag{5.4}$$
$$d_z = |Mitte_z - Pointer_z| - 0.5 \cdot Tiefe \tag{5.5}$$
$$\tag{5.6}$$

$$Abstand = \sqrt{d_x^2 + d_y^2 + d_z^2} \tag{5.7}$$

Ist der $Abstand$ dabei kleiner als null, befindet sich der Pointer innerhalb der reduzierten Teilregion und wird sofort der dazugehörigen Teilregion zugeordnet, so dass sich zusätzlich Rechenaufwand sparen lässt. Da die reduzierten Teilregionen außerdem bereits während der Erstellung des Quad- bzw. Octrees berechnet werden, ist somit nur eine Abstandberechnung pro Teilregion notwendig.
Die Definition der repräsentativen Punkte geschieht beim Aufbau der Baumstruktur während der Zuweisung der Snap-Punkt-Positionen zu den Teilregionen. So wird jeder Punkt darauf getestet, ob er sich auf einer oder mehr der zwei bzw. drei Achsen näher an der jeweiligen Regionsgrenze befindet als die vor ihm zugeordneten Punkte. Ist dies der Fall, wird der Snap-Punkt der neue repräsentative Punkt der jeweiligen Achse. Auf diese Weise müssen die Teilregionen nach dem Aufbau nicht noch einmal für die Feststellung der repräsentativen Positionen durchlaufen werden.

Snapping-Vorgang

Ist der nächstgelegene Snap-Punkt gefunden, kann basierend auf dem in Abschnitt 4.5.2 beschriebenen Verfahren geprüft werden, ob dieser sich in Sichtweite oder Snapping-Reichweite befindet.
Kommt es dabei zum Snapping auf einen Punkt in den Bilddaten, wird der Pointer auf der Benutzungsoberfläche an der entsprechenden Position fixiert, während sich der Ghostpointer auf den Bilddaten weiterbewegt. So lässt sich feststellen, ob die eigentliche Position des Pointers noch eine ausreichende Nähe zum Snap-Punkt aufweist. Solange dies der Fall ist, werden die anderen Snap-Punkte nicht auf ihre Nähe zum Ghostpointer geprüft.

5.4 Umsetzung der Steuerung über Maus und Tastatur

Da die gestengesteuerte Version des Prototyps eine modulare Eingabeverwaltung besitzt, gilt es für die Implementierung einer durch Maus und Tastatur gesteuerten Version lediglich die Abfrage der Freihandgesten durch eine Abfrage der Maus- und Tastatureingaben zu ersetzen. Dabei erfolgt eine Orientierung an bekannten Interaktionstechniken. Auf diese Weise soll die Steuerung intuitiver gestaltet werden (siehe Abschnitt 4.1.3). Außerdem ist darauf zu achten, dass bei der Übersetzung zwischen der Steuerung mit den Freihandgesten und der Steuerung mit Maus und Tastatur keine allzu großen Unterschiede im grundlegenden Interaktionskonzept entstehen. Nachfolgend soll die dabei entstandene Maussteuerung erläutert werden.

Zeigen Die Pointerposition wird anhand der Position des Mauszeigers bestimmt.

Anklicken und Selektieren von Objekten Das Anklicken von Objekten erfolgt in gewohnter Weise über einen Linksklick. Dieser muss zum offen Halten der über entsprechende Schaltflächen aufgerufenen Menüs nicht gehalten werden, da für die Selektion eines Elements im Menü ebenfalls ein Linksklick benötigt wird. Dieser wäre bei gedrückter, linker Maustaste nicht verfügbar.

Öffnen des Kontextmenüs Das Kontextmenü wird über das Halten der rechten Maustaste geöffnet, um sowohl die Vorgehensweise bei der Gestensteuerung nachzustellen als auch das bekannte Interaktionskonzept für den Aufruf solcher Menüs zu berücksichtigen. Die Selektion eines Menüelements kann, wie bereits erwähnt, über einen Linksklick erfolgen, während das Loslassen der rechten Maustaste das Kontextmenü schließt.

Einleiten der Kamerasteuerung Die Steuerung der Kamera wird durch das Halten der Alt-Taste ausgelöst und ist nur möglich während diese gedrückt wird. Diese Vorgehensweise sowie die drei folgenden Interaktionen zur Manipulation der Bildansicht entsprechen dem im Unity-Editor verwendeten Konzept, mit dessen Hilfe die Mausbewegungen auf einfache und intuitive Weise auf die Kamerasteuerung übertragen werden. Durch dessen Nutzung ist es außerdem möglich die horizontalen und vertikalen Bewegungen der Maus mehrfach zu nutzen, ohne dass die Gefahr versehentlicher Interaktionen besteht.

Verschieben der Bildansicht Die Bewegung der Maus mit gedrückter, mittlerer Maustaste führt zu einer Verschiebung der Ansicht in die entsprechende Richtung.

Zoomen der Bildansicht Bei einer Vorwärtsbewegung der Maus mit gedrückter, rechter Maustaste erfolgt ein Herauszoomen, während mithilfe einer Rückwärtsbewegung herangezoomt wird. So lässt sich das Wegschieben bzw. Heranziehen der Bilddaten repräsentieren.

Rotieren der Bildansicht Durch die Bewegung der Maus nach links bzw. rechts und das gleichzeitige Halten der linken Maustaste findet eine Rotation um die y-Achse statt, während Bewegungen nach oben bzw. unten zu einer Rotation um die x-Achse führen. Eine adäquate Übersetzung der Geste zur Rotation um die z-Achse ist hingegen nicht möglich, da sich die Maus nicht im dreidimensionalen Raum bewegt lässt. Dies ist jedoch zu vernachlässigen, da eine derartige Rotation durch die Kombination der anderen beiden Rotationen erreicht werden kann.

Navigation durch die 2D-Schichtdaten Die Navigation durch die 2D-Bilddaten erfolgt durch die Pfeiltaste nach unten (eine Schicht zurück) bzw. die Pfeiltaste nach oben (eine Schicht vor). Dabei sollen die Pfeile die entsprechende Aktion symbolisieren.

Rückgängig machen und Wiederholen von Aktionen Um eine Aktion rückgängig zu machen, ist die Pfeiltaste nach links zu betätigen, während die Pfeiltaste nach rechts Aktionen wiederholt. Dies scheint angemessen, da diese Tasten die beiden Funktionen sinnbildhaft repräsentieren.

5.5 Zusammenfassung

Im Verlauf des Kapitels sind die wichtigsten Details bezüglich der Umsetzung eines Prototyps zur Distanzmessung in medizinischen Bilddaten geschildert worden. So erfolgte die Implementierung mithilfe der Game Engine „Unity“, welche die schnelle Entwicklung von Prototypen anhand einer Kombination aus grafischem Editor und Visual Studio

Community 2015 ermöglicht. Dabei wird der Einsatz des Leap Motion Controllers durch die Verwendung des Leap Motion SDK 2.3.1 sowie der Leap Motion Core Assets 3.0 möglich. Außerdem erfolgt das Laden der DICOM-Bilddaten mithilfe eines auf EvilDICOM basierenden Frameworks.
Anschließend wurden die Algorithmen zur Detektion der verschiedenen Gesten beschrieben. Diese basieren hauptsächlich auf den vom Leap Motion Framework berechneten Positionsdaten der Hand und ihrer Gelenke. Darüber hinaus lässt sich die vordefinierte Faust-Geste ohne Einschränkungen nutzen, während für die Pinch-Geste sicherzustellen ist, dass diese zwischen Daumen und Zeigefinger stattfindet. Andere vom Leap Motion Framework vordefinierte Gesten können jedoch aufgrund unzureichender Qualität in der Detektion nicht genutzt werden. Ihre Umsetzung erfolgte daher eigenständig.
Des Weiteren sind die Mechanismen zur Nutzerunterstützung erläutert worden. So wird die Umsetzung der Filterung der Handpositionen mithilfe der exponentiellen Glättung beschrieben und die Wahl der dazugehörigen Parameter erörtert. Anschließend wurden die für das Snapping erforderlichen Verarbeitungsschritte beleuchtet. Dabei ist zunächst die Position des Pointers auf der Benutzungsoberfläche auf die Bilddaten zu projizieren sowie die Datengrundlage für die Positionierung der möglichen Snap-Punkte zu berechnen. Letzteres erfolgt in den 2D-Bilddaten anhand von, durch den Sobel-Operator ermittelten, Kantenbildern, während für das 3D-Modell die Eckpunktdaten des dazugehörigen Dreiecksnetzes genutzt werden. Basierend auf diesen Berechnungen lässt sich der zur Pointerposition nächstgelegene Snap-Punkt mithilfe von effizienten Quad- bzw. Octreestrukturen bestimmen. Anhand der so erhaltenen Informationen kann letztendlich entschieden werden, ob gesnappt werden muss oder nicht.
Den Abschluss bildete die Umsetzung der Maus- und Tastatursteuerung, die für eine zweite Version des Prototyps genutzt wird und die bestehende Freihandgestensteuerung entsprechend übersetzt. Die während der Implementierung entstandenen Versionen des Prototyps sind die Grundlage für die im kommenden Kapitel beschriebene Nutzerstudie zur Evaluierung der im Konzept erläuterten Interaktionstechniken. Dabei dient die zweite Version dem Vergleich zwischen der Steuerung über Freihandgesten und den klassischen Maus- und Tastatureingaben.

6 Evaluierung

Nachdem die Entwicklung des Prototyps zur Messung der Distanz in medizinischen Bilddaten abgeschlossen ist, kann dieser im Rahmen einer Nutzerstudie evaluiert werden. Auf diese Weise sollen nicht nur Stärken und Schwächen sowie Verbesserungsmöglichkeiten der implementierten Interaktionskonzepte identifiziert werden, sondern auch ein Vergleich zwischen der Freihandgestensteuerung und der Steuerung des Prototyps über Maus und Tastatur stattfinden. Letzterer dient dabei zur Schaffung von Referenzen für die Bewertung der gestenbasierten Interaktionstechniken.
Nachfolgend wird zunächst die für die Studie notwendige Planung in Form der Aufstellung der verfolgten Bewertungskriterien sowie der Auswahl der Probanden geschildert. Anschließend erfolgt die Beschreibung der Versuchsumgebung und des Ablaufes der Nutzerstudie, so dass im weiteren Verlauf die dabei erzielten Ergebnisse vorgestellt werden können. Diese sollen abschließend ausgewertet und diskutiert werden.

6.1 Planung

Der Durchführung der Nutzerstudie gehen einige Schritte der Planung voraus. Zu diesen gehört sowohl das Aufstellen der relevanten Bewertungskriterien als auch die Auswahl geeigneter Probanden. Beides soll im Folgenden näher beleuchtet werden.

6.1.1 Bewertungskriterien

Um eine gezielte Evaluierung durchführen zu können, sind Kriterien zu definieren, anhand derer der gesten- und der mausgesteuerte Prototyp bewertet bzw. verglichen werden sollen. Diese lassen sich aus den in Abschnitt 3.2 erläuterten, nicht-funktionalen Anforderungen ableiten und sollen nachfolgend geschildert werden.

Bearbeitungszeit

Die Messung der für die Erledigung einer Aufgabe benötigten Zeit bietet einen guten Einblick in die Effizienz der jeweiligen Interaktion [34]. Aus diesem Grund erfolgt die Implementierung eines Datenloggers, der die benötigte Zeit misst. Diese wird anschließend in einer Log-Datei zusammen mit der Aufgabennummer ausgegeben.

Genauigkeit

In Abschnitt 3.2.4 wurde bereits die Wichtigkeit einer hohen Genauigkeit bei der Vermessung in medizinischen Bilddaten geschildert. Daher soll diese auch während der Evaluierung berücksichtigt werden. Zu diesem Zweck wird ein weiterer Datenlogger umgesetzt. Er erhält als Eingabe eine Ziel-Distanzmessung, die daraufhin vom Probanden so genau wie möglich und mithilfe der entwickelten Vermessungsinteraktion nachgeahmt werden muss. Dabei misst der Logger sowohl die Abweichung der Messwerte an sich als auch die Abweichung zwischen den vorgegebenen Messpunkten und den vom Probanden gesetzten Start- und Endpunkten.

Usability

Da die entwickelten Interaktionskonzepte nicht nur eine hohe Effizienz und Genauigkeit bieten, sondern auch einen gewissen Grad an Benutzungsfreundlichkeit aufweisen sollen, gilt es die Einhaltung der Usability-Kriterien ebenfalls zu überprüfen. Hierzu existieren bereits einige, durch vorangegangene Arbeiten validierte Fragebögen. Einer von ihnen bewertet die Benutzungsfreundlichkeit von Systemen anhand der System Usability Scale (SUS) [108]. Der SUS-Fragebogen konnte sich dabei als robust sowie zuverlässig erweisen [108] und gehört „zu den etablierten und anerkannten Verfahren in der Nutzerforschung“ [109]. Daher soll er in Form eines abschließenden Fragebogens (siehe Anhang B) im Rahmen der Nutzerstudie zum Einsatz kommen.
Darüber hinaus schließt die Usability von Freihandgesten zusätzlich den Aspekt der Ermüdung durch die ausgeführten Gesten mit ein [34]. Dieser wird nicht vom SUS-Fragebogen abgedeckt und soll dem Fragebogen deshalb als gesonderte Fragestellung hinzugefügt werden.

6.1.2 Auswahl der Probanden

Bei der Auswahl der Probanden für die Nutzerstudie sind Ärzte und Medizinstudenten zu bevorzugen, da sie potenziell den engsten Bezug zu Systemen im medizinischen Bereich

besitzen und außerdem fachspezifische Einsichten liefern können. Da jedoch für das Setzen und Bearbeiten von Start- und Endpunkten zur Messung von Distanzen Fachkenntnisse nicht zwingend erforderlich sind, eignen sich auch Personen anderer Fachrichtungen zur Evaluierung der Prototyps.

6.2 Durchführung

Im Wesentlichen gliedert sich die Nutzerstudie in drei Teile, die im Anschluss an die Schilderung der verwendeten Versuchsumgebung erläutert werden sollen. Dabei ist zu beachten, dass während der Evaluierung nicht nur die für die Freihandgesten entwickelten Interaktionstechniken bewertet, sondern auch mit der klassisch gesteuerten Version des Systems verglichen werden sollen. Letzteres erfolgt, indem die Probanden in zwei Gruppen aufgeteilt werden, von denen die eine die Freihandgestensteuerung und die andere die Steuerung mit Maus und Tastatur nutzt. Diese Vorgehensweise wird Between Subject Design genannt und angewandt, da ein Within Subject Design, bei dem jeder Proband beide Versionen des Prototyps testet, aufgrund des großen Funktionsumfangs des Systems nicht praktikabel ist. Währenddessen lässt sich mit dem Between Subjects Design nicht nur die pro Proband benötigte Zeit reduzieren, sondern auch die Verzerrung der Ergebnisse durch den Lerneffekt zwischen den beiden Versionen vermeiden. Andernfalls wäre es beispielsweise denkbar, dass der Proband während der Nutzung der ersten Steuerung die für beide Prototypen gültige, grundlegende Funktionsweise erlernt und so bei der Bedienung der zweiten einen Vorteil besitzt. Nachteilig am Between Subjects Design ist jedoch, dass Unterschiede in den Evaluierungsergebnissen der beiden Versionen auch durch die verschiedenen Voraussetzungen der Testpersonen hinsichtlich ihrer Auffassungsgabe, Vorkenntnisse und Präferenzen bedingt sein können [11]. Dies führt dazu, dass die Ursache der unterschiedlichen Resultate nicht eindeutig auf die getesteten Interaktionstechniken zurückgeführt werden kann. Um diese Schwäche zu mildern, ist auf eine gleichmäßige Verteilung der Probanden in Bezug auf ihre Voraussetzungen zu achten (siehe Abschnitt 6.3.1).

6.2.1 Versuchsumgebung

Sowohl die Einführung in die Handhabung als auch die Phase der praktischen Anwendung finden an einem Sectra Table [110] in der in Abbildung 6.1 gezeigten Versuchsumgebung statt. Mit seinem 55-Zoll-Bildschirm [107] ermöglicht dieser das Arbeiten mit dem Prototypen auf einem ähnlich großen Ausgabegerät wie im Operationssaal des Universitätsklinikums Magdeburg, so dass realistischere Arbeitsbedingungen geschaffen werden. Angeschlossen ist der Bildschirm an einen Dell Precision T5610, welcher Windows 10 in

(a) (b)

Abbildung 6.1: Versuchsumgebungen für die Freihandgestensteuerung (a) und die Steuerung über Maus und Tastatur (b)

der 64-bit Version als Betriebssystem besitzt. Er ist außerdem mit einem Intel® Xeon® CPU E5-2650 v2 mit 2.60 Ghz sowie 16 GB Arbeitsspeicher ausgestattet.

6.2.2 Einführung in die Handhabung

Die Nutzerstudie beginnt mit einer Einweisung des Probanden in die Bedienung des jeweiligen Prototyps. Dies ist wichtig, um zu verhindern, dass die Bearbeitungszeiten an verschiedenen Stellen durch längere Einarbeitungszeiten verzerrt werden. Während der Einführung werden daher die unterschiedlichen Funktionalitäten zur Vermessung medizinischer Strukturen anhand des Systems erklärt. Anschließend können vom Probanden Fragen zum Verständnis der Handhabung gestellt und ggf. nicht verstandene Interaktionen kurz selbst am System ausprobiert werden. Auf eine längere Lernphase wurde dabei jedoch verzichtet, um den zeitlichen Umfang der Studie zu begrenzen und diese somit auch für Probanden mit wenig Zeit attraktiver zu gestalten.

6.2.3 Praktische Anwendung

Nachdem die Einweisung abgeschlossen ist, folgt die praktische Anwendung. In diesem Rahmen werden dem Probanden zunächst sechs Distanzmessungen vorgegeben, die er durch das Setzen der entsprechenden Start- und Endpunkte so genau wie möglich reproduzieren muss. Die Messungsvorgaben sind dabei so gewählt, dass sie sich sowohl gleichmäßig auf die 2D- und 3D-Bilddaten verteilen als auch verschiedene Positionierungen der Messpunkte abdecken. Letzteres zeichnet sich dadurch aus, dass sich jeweils zwei Messpunkte genau auf einer Kante, sehr nah neben einer Kante sowie sehr weit von einer Kante entfernt befinden. Auf diese Weise wird eine einseitige Evaluierung der

Interaktionstechniken zur Erstellung der Messungen vermieden, da an den Probanden stets unterschiedliche Anforderungen zur Positionierung der Punkte gestellt werden.
Darüber hinaus soll auch eine Überprüfung der Konzepte zur Bearbeitung der Messungen stattfinden. Dazu ist eine weitere, kurze Einweisung des Probanden in die entsprechende Handhabung notwendig. Diese wurde nicht zu Beginn der Nutzerstudie vorgenommen, um ein mögliches Vergessen der Vorgehensweise zu verhindern. Ist der Proband über die Bedienung informiert, wird er aufgefordert sowohl eine bestimmte Bearbeitung in 2D als auch eine Bearbeitung in 3D durchzuführen.
Während jeder Vermessung und Bearbeitung wird dabei die Zeit gemessen und bei der Positionierung der Messpunkte während der Messungserstellung die Genauigkeit aufgezeichnet.

6.2.4 Befragung

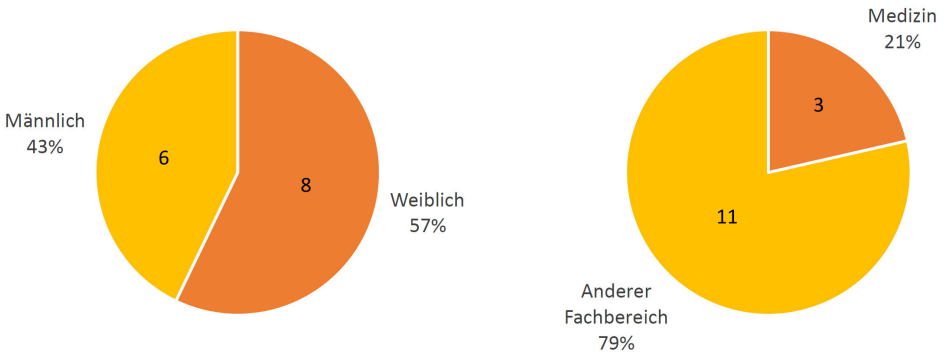
Im Anschluss an den praktischen Teil der Nutzerstudie wird dem Probanden der in Abschnitt 6.1.1 beschriebene Fragebogen vorgelegt. Dieser ist in Bezug auf den jeweils genutzten Prototypen auszufüllen und bildet den Abschluss der Studie.

6.3 Ergebnisse

Im Folgenden sollen die Ergebnisse der Nutzerstudie präsentiert werden. Dazu wird zunächst die Zusammensetzung der Probanden beschrieben und anschließend die Aufzeichnungen der Datenlogger dargestellt. Außerdem erfolgt die Darlegung der Resultate aus dem Usability-Fragebogen sowie der gemachten Beobachtungen.

6.3.1 Zusammensetzung der Probanden

Die Nutzerstudie erfolgte mit insgesamt 14 Probanden, welche nach Geschlecht und Fachrichtung unterschieden werden und sich entsprechend der Abbildung 6.2 zusammensetzen. Dabei ist zu erkennen, dass nur ein kleiner Teil der Testpersonen aus dem medizinischen Bereich stammt. Dies liegt darin begründet, dass sich die Erreichbarkeit von Medizinstudenten wesentlich schlechter gestaltete als die von Personen aus anderen Fachrichtungen.
Eine Unterscheidung zwischen Geschlecht und Fachrichtung während der Studie ist notwendig, um die Testpersonen möglichst gleichmäßig auf die beiden Versuchsgruppen für die verschiedenen Steuerungen verteilen zu können. Dabei spielt insbesondere die Trennung zwischen dem medizinischen und nicht medizinischen Bereich eine Rolle, da



(a) (b)

Abbildung 6.2: Zusammensetzung der Probanden unterteilt nach Geschlecht (a) und Fachrichtung (b)

Personen aus dem medizinischen Sektor andere Erfahrungen bezüglich fachlicher Hintergründe und entsprechender Software mitbringen können als Probanden aus anderen Fachrichtungen.

6.3.2 Quantitative Ergebnisse

Zu den quantitativ messbaren Ergebnissen gehören sowohl die Bearbeitungszeit und die Genauigkeit während den Vermessungsaufgaben als auch die Resultate des Fragebogens zur Usability. Diese werden nachfolgend dargestellt.

Bearbeitungszeit

Die Varianzanalyse (ANOVA) ($F(1, 12) = 40{,}860$; $p < 0{,}001$) der Bearbeitungszeiten während der Erstellung und Bearbeitung der Messungen ergibt einen statistisch signifikanten Unterschied zwischen der Handhabung des Prototyps mit der Freihandgestensteuerung und der Handhabung mithilfe von Maus und Tastatur. So benötigen Probanden mit der Gestensteuerung im Durchschnitt 6,6-mal so viel Zeit für die Aufgaben als Probanden, die die Maussteuerung nutzen. Auf diese Weise arbeiten sie durchschnittlich 1,81 Minuten länger an einer Vermessungsaufgabe.
Allerdings ist im Falle der Gestensteuerung bis zur fünften Messungserstellung eine stark sinkende Bearbeitungszeit zu verzeichnen, während diese bei der Steuerung mit Maus und Tastatur relativ konstant bleibt (siehe Abbildung 6.3). Demgegenüber lässt sich bei beiden ab der sechsten Aufgabe eine Steigerung der Zeit beobachten. Diese lässt sich einerseits dadurch erklären, dass für die Erstellung der sechsten Messung ein erhöhter Navigationsaufwand erforderlich ist, und andererseits dadurch, dass während der ersten

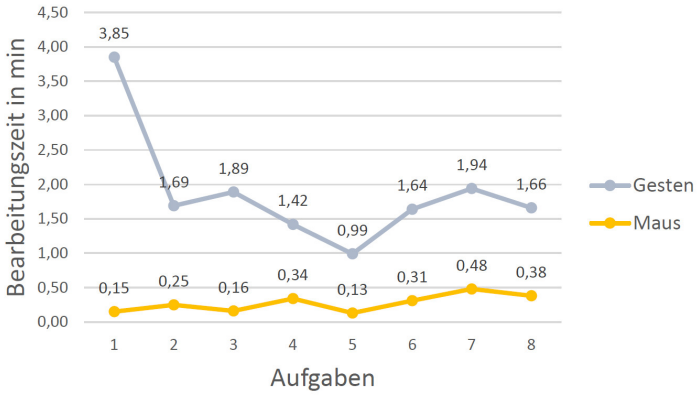
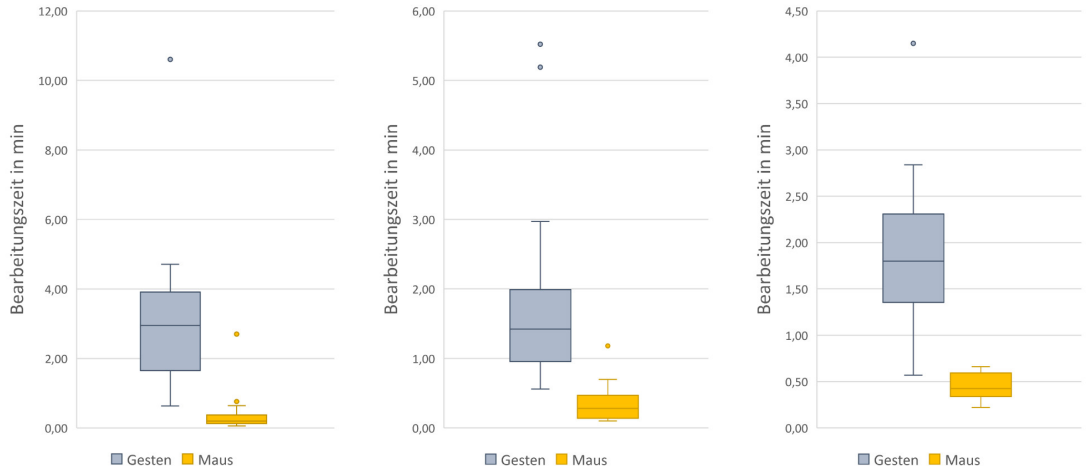


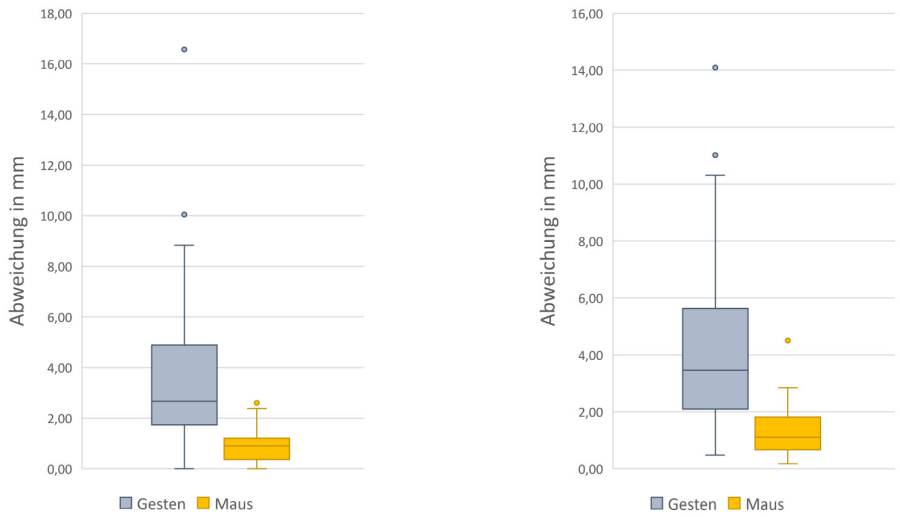
Abbildung 6.3: Entwicklung der Bearbeitungszeit während der verschiedenen Vermessungsaufgaben

Bearbeitungsaufgabe zunächst eine Gewöhnung an die entsprechend neue Interaktion erfolgen muss. Letzteres wird dadurch bekräftigt, dass die Bearbeitungszeit für beide Steuerungen ab der zweiten Messungsbearbeitung erneut sinkt.
Abbildung 6.4 zeigt außerdem die Verteilung der Bearbeitungszeiten für die Gesten- und Maussteuerung. Hier wird unter anderem ersichtlich, dass mit der Freihandgestensteuerung ebenfalls Bearbeitungszeiten unter einer Minute erzielt werden konnten, der Großteil der Probanden jedoch länger brauchte bzw. sich in seiner Arbeitszeit stark von der mit Maus und Tastatur erreichten Zeit unterschied.



(a) **(b)** **(c)**

Abbildung 6.4: Verteilung der Bearbeitungszeiten unterteilt nach Vermessungen in 2D (a), Vermessungen in 3D (b) und Bearbeitung von Messungen (c)



(a) **(b)**

Abbildung 6.5: Verteilung der Abweichungen von den vorgegebenen Messpunkten in den 2D-Schichtdaten (a) und im 3D-Modell (b)

Genauigkeit

Auch im Falle der Genauigkeitsanalyse ergab die ANOVA ($F(1, 10) = 18{,}610$; $p = 0{,}002$) einen statistisch signifikanten Unterschied zwischen Gesten- und Maussteuerung. Dabei erzielen die Probanden mit der Steuerung per Freihandgesten im Schnitt eine 3,82-mal größere Abweichung und differieren so durchschnittlich um 3,01 mm mehr von den vorgegebenen Messpunkten.
Des Weiteren sind sowohl bei der Maus- als auch bei der Gestensteuerung auf den 3D-Bilddaten größere Unterschiede zwischen den Vorgaben und den von den Probanden erstellten Messungen zu erkennen als in den 2D-Schichtdaten (siehe Abbildung 6.5). Dies lässt sich möglicherweise durch die zusätzliche Dimension erklären, die weniger leicht überschaubar ist als eine Ansicht in 2D. Dazu passt auch, dass die Genauigkeit im Verlauf der Messungen am 3D-Modell zum Teil stark zunimmt (siehe Abbildung 6.6), was durch eine Gewöhnung der Probanden an die zusätzliche Dimension begründet sein kann.
Eine weitere mögliche Erklärung für die Unterschiede in der Genauigkeit zwischen den 2D- und 3D-Daten hängt damit zusammen, dass die aktuelle Version des Prototyps die Messungen innerhalb des Unity-Koordinatensystems durchführt und noch keine automatischen Umrechnungen in das Koordinatensystem der Bilddaten vornimmt. Daher war es für die Aufbereitung der Ergebnisse der Genauigkeitsanalyse notwendig auf den verwendeten Bilddaten Referenzmessungen mithilfe einer anderen medizinischen Software vorzunehmen. Anhand dieser Referenzmessungen im Koordinatensystem der Bilddaten lassen sich anschließend die jeweiligen Maßstäbe für die Transformation zwischen den beiden Koordinatensystemen ermitteln. Während dabei für die 2D-Schichtdaten die pas-

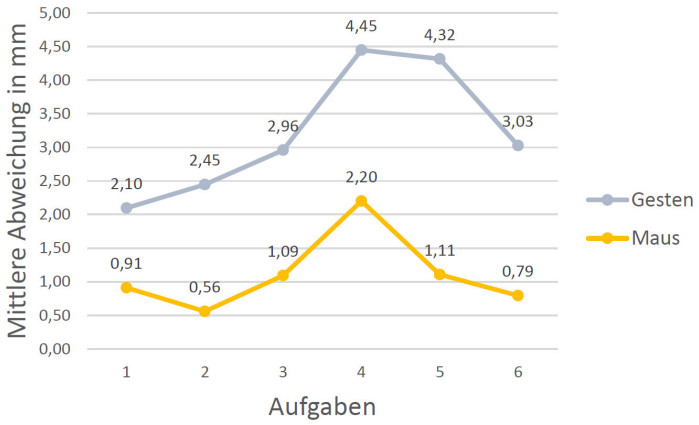


Abbildung 6.6: Entwicklung der Abweichungen während der Messungserstellung

senden DICOM-Daten vorliegen, ist dies für das verwendete 3D-Schädelmodell nicht der Fall. So konnten die Referenzmessungen für die 3D-Daten nur auf einem sehr ähnlichen Schädel vorgenommen werden. Aus diesem Grund wäre es denkbar, dass sich hier kleinere Fehler bei der Transformation der Ergebnisse ergeben haben.
Darüber hinaus zeigt Abbildung 6.6, dass die Genauigkeit während der Messungen mit dem gestengesteuerten Prototypen in den 2D-Schichtdaten abnimmt. Ein Grund hierfür könnte eine zunehmende Ermüdung sein, die durch die anfängliche, häufig lange und aufwendige Eingewöhnungsphase noch verstärkt wird. Darüber hinaus ist jedoch auch hier anzumerken, dass es einige wenige Probanden gibt, die mithilfe der Freihandgesten eine gute Genauigkeit mit Abweichungen unter einem Millimeter erzielen konnten.

Abbildung 6.7: Bewertungsskala für die durch die Beantwortung des SUS-Fragebogens erhaltenen Usability-Werte [109] mit den durchschnittlichen Ergebnissen der Freihandgestensteuerung und der Steuerung mit Maus und Tastatur

Usability

Anhand der Antworten auf die SUS-Fragebögen lässt sich ein Usability-Wert zwischen 0 und 100 berechnen und anschließend gemäß der in Abbildung 6.7 dargestellten Skala einordnen [109]. So erreicht der gestengesteuerte Prototyp im Durchschnitt einen Wert von 51,79 und die mausgesteuerte Version einen Wert von 68,93. Damit ist die Benutzungsfreundlichkeit der Freihandgestensteuerung „Okay“ mit einer Tendenz zu „Gut“, während sich die Maussteuerung im gelben Bereich von „Gut“ befindet und zu einer exzellenten Usability tendiert.
Abbildung 6.8 zeigt jedoch auch, dass sich im Falle der gestengesteuerten Version neben den guten bis exzellenten Ergebnissen auch einige mittelmäßige Resultate ergeben haben. Demgegenüber wurde die Steuerung per Maus und Tastatur nur in Ausnahmefällen mit einem Wert schlechter als 60 bewertet.
Darüber hinaus sind beide Steuerungen als weniger bis mittelmäßig ermüdend und nur in Einzelfällen als stark ermüdend eingestuft worden (siehe Abbildung 6.9). Dabei ist die Bedienung mit Maus und Tastatur anstrengender für die Finger, während die Freihandgesten zu mehr Ermüdungen in Handgelenken, Armen und Schultern führen. Außerdem besteht der einzige größere Unterschied zwischen der Maus- und der Gestensteuerung in der Belastung der Schultern.

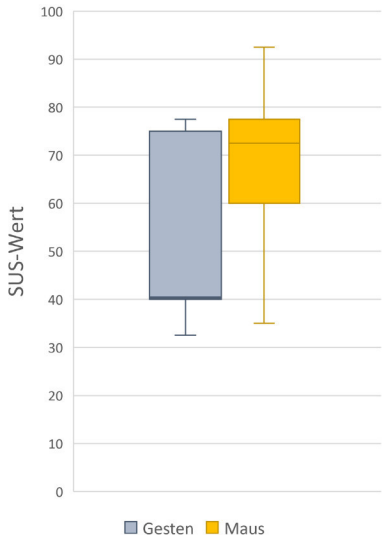


Abbildung 6.8: Verteilung der Usability-Werte von Maus- und Gestensteuerung

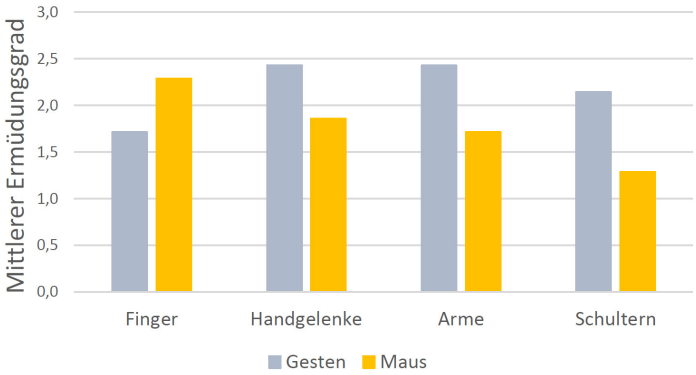


Abbildung 6.9: Vergleich des mittleren Ermüdungsgrades zwischen Maus- und Gestensteuerung

6.3.3 Qualitative Ergebnisse

Während der Nutzerstudie ließen sich neben den quantitativen Ergebnissen auch qualitative Erkenntnisse in Form von Beobachtungen sammeln. Diese sollen im Folgenden geschildert werden.

Robustheit der Gestenerkennung

In Zusammenhang mit der Erkennung der Gesten traten während der Studie die meisten Probleme auf. Die größten Schwierigkeiten gab es bei der Verwendung der Circle-Geste. So muss die Kreisbewegung exakt um den anhand der Startposition festgelegten Mittelpunkt erfolgen. Dabei kommt es häufig zu Problemen, wenn die Handposition anschließend noch einmal während der Grundhaltung verändert wird. In diesem Fall muss die Geste meist erneut begonnen werden, da sich der ursprüngliche Mittelpunkt nur schwer wiederfinden lässt.
Hinzu kommt, dass einige Probanden die Startposition der Hand nicht als untere Mitte des zu beschreibenden Kreises anzusehen schienen, sondern als die in Abbildung 6.10 gezeigte Position auf dem Kreis. Auf diese Weise wurde eine entsprechend falsche Kreisbewegung ausgeführt, die nicht als solche erkannt wurde.
Aufgrund dieser Schwierigkeiten gestaltete sich das Rückgängig machen und Wiederholen von Aktionen für die Probanden sowohl ermüdend als auch irritierend, da die Circle-Geste oft wiederholt werden musste und nicht klar war, wo der Fehler in der Ausführung lag.

Zu weiteren Problemen kam es bei der Detektion der Pinch-Geste. Diese wird häufig nur schwer erkannt, da die Hand sehr genau so gehalten werden muss, dass Daumen und Zeigefinger während der Ausführung leicht schräg nach unten zeigen. Im Gegensatz

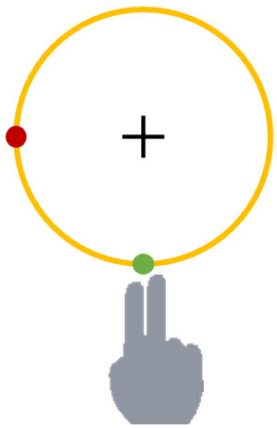


Abbildung 6.10: Vergleich der gewollten Startposition für die Circle-Geste (grün) und der häufig von den Probanden angenommenen Startposition der Kreisbewegung (rot)

dazu ist die Detektion in manchen Situationen auch sehr empfindlich und fehleranfällig. So kommt es zu Komplikationen, wenn sich Daumen und Zeigefinger permanent so nah beieinander befinden, dass der Schwellenwert für die Pinch-Geste sehr oft hintereinander über- und unterschritten wird. Dies fiel besonders bei der Erstellung der Messungen auf, da auf diese Weise häufig sehr viele ungewollte Messungen hintereinander erzeugt wurden.
Während die vielen, durch den Detektionsfehler versehentlich erstellten Messungen im Rahmen der Nutzerstudie stets bis auf die letzte fehlerhafte Messung vom Studienleiter entfernt wurden, um zu starke Zeitverzerrungen und Studiendauern zu verhindern, führte die zögerliche Erkennung der Pinch-Geste oft zu einer erschwerten Platzierung der Messpunkte. So war häufig ein erneutes Anvisieren der Punkte erforderlich, bevor ein neuerlicher Versuch der Positionsbestätigung unternommen werden konnte.

Darüber hinaus wurden häufiger unbeabsichtigte Wischgesten ausgeführt, wenn die Pinch-Geste falsch gehalten wurde oder eine unbewusste Handbewegung erfolgte. Dies führte ebenfalls zu größeren Zeitverzögerungen, da die jeweiligen Probanden auch im weiteren Verlauf oft Probleme mit der gezielten Ausführung der Wischgeste hatten. Dabei bestand das Problem neben unbewussten Handbewegungen darin, dass sie ein wenig zu weit in die falsche Richtung ausholten, bevor sie in die richtige Richtung wischten. Daraus lässt sich schließen, dass die Detektion der Wischgeste momentan noch zu empfindlich ist.

Des Weiteren existieren Schwierigkeiten mit den Freihandgesten zur Manipulation der Bildansichten. So führte das Ballen der Fäuste in einigen Fällen dazu, dass eine Manipulation erkannt wurde, obwohl zunächst nur die Grundhaltung eingenommen werden sollte. Demgegenüber stehen Situationen, in denen eine geringe und präzise Veränderung

angestrebt und ausgeführt, aber nicht detektiert wurde.
Im Allgemeinen scheint das Problem hier darin zu bestehen, dass die Schwellenwerte nicht genügend an die verschiedenen Situationen und Umstände angepasst werden, so dass die Detektion manchmal zu empfindlich und manchmal zu zögerlich erfolgt.

Außerdem treten generell Komplikationen in Zusammenhang mit den Lichtverhältnissen auf. Dabei werden vom Leap Motion Controller zum Teil gänzlich falsche oder verzerrte Daten geliefert. In diesem Rahmen traten bei manchem Probanden beispielsweise Probleme mit der Zeigegeste auf, weil neben dem ausgestreckten Zeigefinger ein weiterer, eigentlich nicht gestreckter Finger erkannt wurde. Dies führte dazu, dass trotz richtiger Ausführung der Geste keine entsprechende Detektion erfolgte, wodurch bei dem Probanden große Verwirrung hervorgerufen wurde.

Allein die Faustgeste mit ausgestrecktem Daumen wurde in fast allen Fällen fehlerfrei erkannt. Dies gilt jedoch nicht für die ursprünglich geplante Ausführung der Geste, bei der der Daumen an der Faust anliegt. Sie führte fast immer zu einer Fehlinterpretation, bei der es besonders häufig zu Verwechslungen mit der Pinch-Geste kam.

Handhabung des Systems

Bei der Bedienung des Systems fiel in erster Linie auf, dass es einige Anstrengung und Zeit erforderte, um mit dem Pointer einen bestimmten Punkt auf den Bilddaten zu treffen. So erfolgen dessen Bewegungen laut Aussagen der Probanden zu schnell und Zitterbewegungen wurden nicht in dem Maße ausgeglichen, als dass sich eine Position einfach und präzise anvisieren lässt.
Zusätzlich wurde das Setzen der Messpunkte dadurch erschwert, dass die Visualisierung des Pointers als Fläche keine markierte Mitte besitzt, mit der sich auf konkrete Punkte zielen lässt. Ein Proband gab dabei außerdem an, dass eine Visualisierung als Kreuz mit den entsprechenden Abmaßen der Kreisfläche für ihn natürlicher wäre.

Des Weiteren wurde die Handhabung des Radialmenüs von den Probanden als wenig intuitiv empfunden, da der durch den Zeigefinger hervorgerufene Pointer nicht weiterhin zum Zeigen genutzt werden kann.
Außerdem scheint für viele die Selektion eines Elements über das Öffnen der Pinch-Geste natürlicher. Eine derartige Vorgehensweise wurde ursprünglich vermieden, um zu verhindern, dass durch das Öffnen der Pinch-Geste die angezeigte Position verschoben und so ein falsches Menüelement selektiert wird. Diese Befürchtung stellte sich jedoch aufgrund der Größe des Ausgabegerätes als unbegründet heraus.

Darüber hinaus führte die Möglichkeit, die Zustandsschaltflächen in der Seitenleiste sowohl über Pinch-to-Click als auch über Wait-to-Click zu bedienen, in manchen Fällen zu doppelten Klicks und entsprechenden Verwirrungen seitens der Probanden, da diese sich nur eine der beiden Klick-Interaktionen gemerkt hatten. Auch kam es auf diese Weise zu Fehlern, da einige Testpersonen mit einem derartigen Doppelklick nicht rechneten und die Schaltfläche als aktiviert bzw. deaktiviert wähnten, obwohl sie es nicht war.

Zusätzlich riefen kleinere Fehler in der Implementierung des Prototyps in Einzelfällen ein unerwartetes Verhalten hervor. So kam es während der Bearbeitung von Messungen in der 3D-Ansicht hin und wieder zu einer falschen Berechnung des nächsten Messpunktes. Dies führte dazu, dass die Probanden den falschen Punkt selektierten und bewegten. Außerdem funktionierte die Anpassung der Größe des Pointers auf den 3D-Bilddaten aufgrund der perspektivischen Verzerrungen nicht korrekt.

Wirksamkeit der Mechanismen zur Nutzerunterstützung

Die im vorhergehenden Abschnitt beschriebenen Probleme mit der Ungenauigkeit des Pointers zeigen, dass die Wirksamkeit der Mechanismen zur Nutzerunterstützung bisher eher begrenzt ist. Insbesondere beim Snapping konnte während der Erledigung der Aufgaben durch die Probanden beobachtet werden, dass dieses zu sprunghaft reagiert und keine ausreichende Haftung an den gewünschten Positionen ermöglicht. Dies führte teilweise dazu, dass eine Bedienung mit starkem Zoom und ohne Snapping durch die berechenbareren Reaktionen als schneller und komfortabler empfunden wurde.
Außerdem ließen sich einige Fehler in der Implementierung des Snappings feststellen. Sie resultierten unter anderem in falschen Berechnungen des nächsten Snap-Punktes und in Problemen am Rand des 3D-Modells. Letztere zeigten sich dadurch, dass die dort detektierten Snap-Punkte oft nicht selektiert werden konnten bzw. ihre Berechnung solange nicht erfolgte, bis das 3D-Modell vom Pointer getroffen wurde. Auf diese Weise lässt sich ein Pointer, der sich neben dem 3D-Modell befindet, nicht an den Rand des Modells snappen.

Wirksamkeit des visuellen Feedbacks

Das visuelle Feedback schien nur von sehr wenigen Probanden wahrgenommen bzw. genutzt zu werden. So wurden häufig weder die auf dem Bildschirm bei Erkennung einer Geste angezeigten Icons beachtet noch die Darstellung der 3D-Modelle der Hände bei Problemen mit der Gestendetektion zu Rate gezogen. Dies zeigte sich insbesondere in Situationen, in denen bestimmte Freihandgesten nicht erkannt wurden, aber dennoch der Versuch stattfand, die entsprechende Handlung auszuführen. Auch bei Detektionsschwie-

rigkeiten, die durch den Empfang fehlerhafter Handdaten vom Leap Motion Controller entstanden, wurde nicht versucht anhand der angezeigten Handmodelle die Ursache für das Problem herauszufinden. Darüber hinaus erfolgte die Nutzung des Gesten-Guides ebenfalls nur selten, wenn der Proband eine Geste vergessen hatte.
Aus diesen Gründen war es oft notwendig nach einer längeren Phase der Ratlosigkeit des Probanden auf die entsprechenden Möglichkeiten erneut hinzuweisen. Während dies im Falle des Gesten-Guides häufiger zum Ziel und zur Erinnerung an die jeweilige Geste führte, wurden die restlichen Hilfen dennoch nur sporadisch wahrgenommen.

6.4 Auswertung

Nachdem im vergangenen Abschnitt die Ergebnisse der Nutzerstudie erläutert wurden, soll der folgende dazu dienen entsprechende Schlüsse aus den Resultaten zu ziehen. Diese werden dann als Stärken und Schwächen der entwickelten Interaktionskonzepte zusammengefasst. Außerdem erfolgt die Vorstellung möglicher Verbesserungen zur Minderung bzw. Behebung der beschriebenen Probleme.

6.4.1 Stärken

Aus den Ergebnissen der Studie können die nachfolgend beschriebenen Stärken der gestenbasierten Interaktionstechniken abgeleitet werden.

Gute Usability der grundlegenden Interaktionskonzepte Die Ergebnisse des SUS-Fragebogens lassen darauf schließen, dass die Konzepte der Benutzungsoberfläche sowie der Interaktionstechniken im Kern eine gute Usability aufweisen. So liegt der durchschnittlich erzielte SUS-Wert für die Freihandgestensteuerung nur sehr knapp unter den, für eine gute Usability erforderlichen, 52 Punkten. Dabei ist davon auszugehen, dass diese ohne die Erschwernisse durch die mangelhafte Robustheit bei der Detektion der Gesten erreicht worden wären.

Starke Leistungssteigerung nach kurzer Eingewöhnung Anhand der Entwicklung der Bearbeitungszeiten während der Erledigung der Vermessungsaufgaben wird deutlich, dass eine längere Eingewöhnungs- bzw. Lernphase zu einer höheren zeitlichen Effizienz führen kann. Dies ist insbesondere an der großen Differenz zwischen der ersten und zweiten Aufgabe zu erkennen. Des Weiteren lässt sich aus den aufgezeichneten Daten herauslesen, dass kurze Bearbeitungszeiten in den meisten Fällen nicht mit einer starken Ungenauigkeit verbunden waren.

6.4.2 Schwächen

Die während der Nutzerstudie erzielten Resultate lassen auch auf einige Schwächen des gestengesteuerten Protoyps schließen. Diese sollen im Folgenden näher beleuchtet werden.

Mangelnde Robustheit bei der Gestenerkennung Die in Abschnitt 6.3.3 geschilderten Probleme bei der Gestenerkennung zeigen, dass noch einige Nachbesserungen an den Algorithmen zur Detektion der Freihandgesten erforderlich sind. So funktioniert bisher nur die Faustgeste mit ausgestrecktem Daumen zuverlässig, während die Pinch- und Circle-Geste sowie die Gesten zur Manipulation der Kamera eine sehr genaue und kontrollierte Ausführung erfordern. Demgegenüber ist die Erkennung der Zeigegeste in den meisten Fällen in Ordnung, kann jedoch unter ungünstigen Lichtverhältnissen leiden.

Ungenauigkeiten beim Zeigen mit dem Pointer Anhand der Resultate der Genauigkeitsanalyse und den gemachten Beobachtungen lässt sich feststellen, dass auch bei der Genauigkeit des Pointers Verbesserungsbedarf besteht. Dessen Bewegungen erfolgen zu schnell und sind noch nicht vollständig frei von Zitterbewegungen.

Verbesserungswürdiges Snapping Abschnitt 6.3.3 beschreibt ebenfalls Schwierigkeiten mit dem Snapping, die eine fehlende Stabilität erkennen lassen. Außerdem existieren Implementierungsfehler, die zu Problemen mit der Berechnung des nächsten Snap-Punktes und dem Snapping am Rand der 3D-Bilddaten führen. Ersteres wurde dabei bereits kurz nach der Durchführung der Nutzerstudie größtenteils behoben.

Inkonsistenzen in der Handhabung des Radialmenüs Im Rahmen der Beobachtungen während der Erledigung der Vermessungsaufgaben konnte festgestellt werden, dass einige Probanden Schwierigkeiten mit der Bedienung der Radialmenüs hatten. Dies schien dabei größtenteils dadurch bedingt, dass dort, im Vergleich zur Handhabung der restlichen Benutzungsoberfläche, eine leicht veränderte Vorgehensweise zum Anzeigen und Selektieren der Schaltflächen erforderlich war.

6.4.3 Verbesserungsmöglichkeiten

Anhand der gemachten Beobachtungen lassen sich bereits einige Verbesserungsmöglichkeiten sowohl in Hinblick auf die Interaktionskonzepte als auch bezüglich der technischen Umsetzung ableiten. Diese werden nachfolgend erläutert.

Anpassung der Handhabung der Radialmenüs

Für eine verbesserte Interaktion mit den Radialmenüs sind zwei Lösungen denkbar. Einerseits könnte die Bedienung so verändert werden, dass die aufrufende Pinch-Geste nicht gehalten werden muss, damit die Menüs geöffnet bleiben. Auf diese Weise ließe sich anschließend die gewohnte Kombination aus Zeigegeste und Pinch-to-Click für die Selektion der Menüelemente nutzen und die Erwartungen der Probanden erfüllen. Dabei muss jedoch darauf geachtet werden, dass bei der Ausführung der Zeigegeste der Pointer initial mittig zum Radialmenü platziert wird und danach eine zu dieser Position relative Bewegung erfolgt. Dies ist wichtig, um dem Nutzer einen längeren Weg von der ursprünglichen Pointerposition zum Menü zu ersparen sowie die Vorteile bei der Verwendung von Radialmenüs weiterhin nutzen zu können (siehe Abschnitt 4.2.2). Außerdem erfordert diese Vorgehensweise einen neuen Mechanismus zum Schließen der Menüs. Dies könnte beispielsweise wie in herkömmlichen, mausgesteuerten Programmen erfolgen, indem ein Klick auf einen Punkt außerhalb des Radialmenüs erfolgt. Allerdings ist dieser Vorgang weniger schnell als das Schließen über das Öffnen der Pinch-Geste.
Andererseits besteht eine weitere Möglichkeit darin, die zweite, von den Probanden erwartete Vorgehensweise umzusetzen, so dass auch die Selektion der Menüelemente über das Öffnen der Pinch-Geste stattfindet. Auf diese Weise wird eine schnellere Interaktion mit den Radialmenüs ermöglicht. Die von einigen Probanden wahrgenommenen Inkonsistenzen lösen sich dabei jedoch nicht auf.

Anpassung der Aktivierung der Zustandsschaltflächen

Versehentliche, doppelte Aktivierungen bzw. Deaktivierungen der Zustandsschaltflächen könnten zukünftig vermieden werden, indem eine Entscheidung zwischen der Selektion über Pinch-to-Click und der Selektion über Wait-to-Click erfolgt. Alternativ wäre es denkbar für den primären Pointer nur noch die Aktivierung über Pinch-to-Click zuzulassen, während eine Selektion über den Sekundärpointer nur per Wait-to-Click möglich ist. Dies kann sinnvoll sein, da vor allem Letzterer in Szenarien zum Einsatz kommt, in denen die andere Hand bereits eine andere Geste ausführt und möglicherweise nicht in der Lage ist eine Pinch-Geste vorzunehmen. Allerdings ist diese Vorgehensweise mit Inkonsistenzen in der Handhabung der Zustandsschaltflächen verbunden, weshalb diese Alternative gründlich abzuwägen ist.

Erweiterung der Pointervisualisierung

Um ein besseres Anvisieren von konkreten Positionen und somit eine einfachere Platzierung von Messpunkten zu ermöglichen, sollte die Mitte des Flächenpointers markiert werden. Außerdem ist darüber nachzudenken, ob eine reduzierte Visualisierung als Kreuz

sinnvoller ist. So würde eine bessere Sichtbarkeit der darunter befindlichen Bilddaten gewährleistet, während mithilfe der durch das Kreuz aufgespannten Fläche weiterhin eine entsprechende Flächenselektion möglich wäre.

Erhöhung der Genauigkeit des Pointers

Eine Verbesserung der Präzision des Pointers kann unter anderem durch die Umsetzung einer adaptiven Pointergeschwindigkeit erzielt werden. Dies ist sowohl durch die Anpassung des Control-to-Display Ratio (C/D-Wert) an die Bewegungsgeschwindigkeit der Zeigegeste als auch mithilfe von pseudo-haptischem Feedback möglich [111, 112, 113, 114]. Unter dem C/D-Wert ist dabei das Verhältnis zwischen der Bewegung der Zeigegeste und der Bewegung des Pointers auf dem Bildschirm zu verstehen [39].
Bei der Anpassung des C/D-Wertes an die Bewegungsgeschwindigkeit wird davon ausgegangen, dass langsame Bewegungen mit dem Ziel hoher Eingabegenauigkeit erfolgen, so dass ein hoher C/D-Wert verwendet wird, um die Geschwindigkeit des Pointers zu verlangsamen. Umgekehrt führt eine schnelle Bewegung zur Verringerung des C/D-Wertes. Insbesondere das von Frees et al. [115] entwickelte PRISM-Verfahren scheint hier empfehlenswert, da es sich als präzise, schnelle und komfortable Methode zur Unterstützung des Zeigens via Raycasting herausstellte.
Dahingegen passt pseudo-haptisches Feedback die Pointergeschwindigkeit über den C/D-Wert an die zugrundeliegenden Bilddaten an. Dabei wird die Geschwindigkeit beim Passieren relevanter Strukturen, wie z. B. Kanten, verlangsamt und bei allen weiteren wieder erhöht. Diese Vorgehensweise sollte jedoch nur bei einer langsamen bis mittelmäßigen Bewegungsgeschwindigkeit zum Einsatz kommen, um eine Behinderung des Nutzers außerhalb der Vermessungsinteraktionen zu vermeiden.

Flexiblere und robustere Gestenerkennung

Ein Großteil der Probleme bei der Detektion der Circle-Geste entstanden dadurch, dass diese nicht flexibel genug gestaltet ist. Hier könnte Abhilfe geschaffen werden, indem sich der berechnete Mittelpunkt des zu beschreibenden Kreises dynamisch anpasst. Um dies zu erreichen ist es notwendig zu unterscheiden, ob eine Handbewegung der Ausführung der Kreisbewegung dient oder nicht. Kann dies entschieden werden, gilt es während einer nicht zur Kreisbewegung gehörenden Handbewegung den Mittelpunkt entsprechend anzugleichen. Darüber hinaus sollte eine Lösung gefunden werden, die ohne einen festen Startpunkt auskommt, so dass die flexible Erkennung verschiedener Kreisbewegungen ermöglicht wird.

Auch die Detektion der Gesten zur Manipulation der Bildansicht erfolgte bisher zu unflexibel. Helfen würde hier eine situationsabhängige Gestenerkennung, die die jeweiligen

Schwellenwerte abhängig von bestimmten Kriterien anpasst. Dabei könnte zunächst zwischen dem Einnehmen der Grundhaltung durch das Formen der Fäuste und der Manipulation an sich unterschieden werden, um das Auslösen einer Manipulation während des Schließens der Fäuste zu verhindern. Anschließend ist eine Orientierung der Schwellenwerte an der Bewegungsgeschwindigkeit denkbar, so dass diese bei sehr langsamen Geschwindigkeiten gesenkt und bei schnelleren Geschwindigkeiten erhöht werden. Auf diese Weise würden sowohl langsame und präzise als auch schnelle und grobe Manipulationen der Bildansicht ermöglicht, während Zitterbewegungen weniger Berücksichtigung fänden.

Kleinere Anpassungen betreffen die Faust-, Pinch- und Wischgeste. Erstere ließe sich verbessern, indem auf das ergonomisch weniger komfortable Ausstrecken des Daumens verzichtet wird. Dies ist möglich, wenn die Unterscheidung zwischen Faust- und Pinch-Geste nicht nur anhand der vom Leap Motion Controller gelieferten Schwellenwerte erfolgt, sondern weitere Kriterien hinzugezogen werden. So lässt sich die Faust von einem Pinch zusätzlich anhand der nicht ausgestreckten Finger abgrenzen.
Zur Verbesserung der Pinch-Geste würde dahingegen beitragen, wenn ein schnell aufeinander folgendes Über- und Unterschreiten des entsprechenden Schwellenwertes abgefangen wird. Währenddessen müssen die Schwellenwerte für die Wischgeste so erhöht werden, dass diese weniger oft versehentlich ausgeführt wird. In diesem Zusammenhang ist darüber nachzudenken, ob das Navigieren durch mehrere Schichten eher durch das längere Halten des Beugungswinkels für die Einzelnavigation ausgelöst werden sollte, anstatt über eine noch stärkere Beugung des Handgelenks. Diese verringert sowohl den Komfort der Geste als auch die Möglichkeiten für robustere Schwellenwerte. Letzteres geschieht, indem der Bereich erreichbarer Winkel auf zwei Schwellen aufgeteilt werden muss und so ein weniger großer Unterschied zwischen beiden möglich ist.

Schließlich könnte auch die Verwendung des neueren Orion SDKs des Leap Motion Controllers einige Verbesserungen in der Erkennung der Handdaten mit sich bringen [116]. So wird auf der Webseite des Herstellers angegeben, dass die Erkennung der Hände sowohl schneller als auch zuverlässiger und robuster gegenüber Verdeckungen und äußeren Einflüssen wie z. B. Licht erfolgt. Auch die Detektion der Faust- und Pinch-Geste sowie die Reichweite des Leap Motion Controllers sollen verbessert worden sein.
Ursprünglich wurde auf das Orion SDK verzichtet, da es keine vordefinierte Detektion der Wisch- und Circle-Geste bieten konnte. Dies kann jedoch in Zukunft vernachlässigt werden, da sich bereits während der Umsetzung des Prototyps herausstellte, dass die im V2 SDK vorgefertigte Erkennung dieser Gesten keine zufriedenstellenden Ergebnisse liefert. Es ist allerdings zu beachten, dass das Orion SDK für eine Anwendung in der Virtual Reality entwickelt wurde, bei der der Leap Motion Controller von einem HMD aus auf die Handrücken des Nutzers gerichtet wird. Daher sollte geprüft werden, inwiefern sich die Qualität der Handerkennung ändert, wenn die Hände über dem Gerät interagieren.

Anpassung des Snapping-Verfahrens

Die größten Schwierigkeiten beim Snapping bestanden während der Nutzerstudie darin, dass keine ausreichende Stabilität geboten werden konnte. Dies scheint vor allem in den 2D-Schichtdaten in der genutzten Datengrundlage bzw. ihrer Verarbeitung begründet. So wurde hier jedes Pixel, das zu einer ausreichend starken Kante gehört, als Snap-Punkt markiert. Dies kann dazu führen, dass eine Kante an einigen Stellen mehr als ein Pixel breit ist und daher bereits an dieser Position mehrere Snap-Punkte vorliegen. Auf diese Weise befinden sich im weiteren Verlauf der Kante viele Snap-Punkte nah beieinander. Dabei werden einerseits aufgrund der festgelegten Snapping-Reichweite, über die hinweg an dem aktuellen Snap-Punkt festgehalten wird, einige Punkte übersprungen. Andererseits visiert der Pointer, sobald die Snapping-Reichweite überschritten wurde, sofort den nächsten Snap-Punkt an. Daher muss eine bessere Unterscheidung zwischen Eckpunkten und Kanten stattfinden. Hier könnten Erstere diskret angesteuert werden, während an Letzteren ein kontinuierliches Entlangfahren ermöglicht wird. Mithilfe dieser Vorgehensweise würde sowohl das Auslassen bestimmter Kantenabschnitte vermieden als auch eine bessere Haftung an den einzelnen Eckpunkten und Kanten erzielt, da sich nicht ein einzelner Snap-Punkt an den nächsten reiht. Zusätzlich ist eine Verbesserung der Kantendetektion mithilfe des Scharr-Operators möglich, weil dieser die Rotationssymmetrie besser berücksichtigt als der verwendete Sobel-Operator [117].
Demgegenüber gestaltet sich die Stabilisierung des Snappings in 3D schwieriger, da hier eine große Abhängigkeit vom Aufbau des Dreiecksnetzes des Modells besteht. Dieses kann je nach dargestellter Struktur sehr feinmaschig sein, so dass die Eckpunkte entsprechend nah beieinander liegen und kaum Möglichkeiten für ein stabilen Snappings bieten. Daher ist es an dieser Stelle nur möglich zu versuchen das Dreiecksnetz ohne den Verlust relevanter Informationen zu vereinfachen und auf diese Weise weniger Eckpunkte zu erhalten.
Sowohl für die 2D- als auch für die 3D-Ansicht sollte jedoch geprüft werden, ob eine Anpassung der Schwellenwerte für die Snapping-Reichweite sowie die Sichtweite der Snap-Punkte die Qualität des Snappings verbessern kann.

Behebung von Implementierungsfehlern

Um die vollständige Funktionsfähigkeit der entwickelten Interaktionstechniken und Mechanismen zur Nutzerunterstützung gewährleisten zu können, sind auch die in Abschnitt 6.3.3 beschriebenen Implementierungsfehler zu beheben. Zu diesen gehören sowohl die Probleme des Snappings am Rand der 3D-Bilddaten als auch die falsche Berechnung der Pointergröße und der nächsten Messpunkte in der 3D-Ansicht.

6.5 Diskussion

Die Ergebnisse der Nutzerstudie lassen erkennen, dass der mit Freihandgesten gesteuerte Prototyp zur Messung von Distanzen in medizinischen Bilddaten bisher zum Teil deutlich schlechter abschneidet als seine mausgesteuerte Version. Besonders groß sind die Unterschiede bei der Effizienz und Genauigkeit, während der Ermüdungsgrad zwischen der Gestensteuerung und der Steuerung über Maus und Tastatur nicht allzu stark differiert und in einem akzeptablen Bereich liegt.
Dennoch ist in Bezug auf die Effizienz ein gewisses Potenzial der Freihandgestensteuerung zu erkennen. So sinken die Bearbeitungszeiten im Laufe der von den Probanden zu erledigenden Vermessungsaufgaben stark. Diese Tendenz könnte durch eine vorhergehende, auch mithilfe von praktischen Übungen erfolgende, Lernphase noch verstärkt werden. Gestützt wird diese Annahme dadurch, dass einige Probanden angaben sich durch den Umfang der Funktionen des Prototyps überfordert zu fühlen. Dies hätte vermutlich durch die Ergänzung um anwendungsorientierte, aufeinander aufbauende Übungsaufgaben vermieden werden können.
Des Weiteren handelt es sich bei den für die Steuerung mit Maus und Tastatur gemessenen Bearbeitungszeiten nur um jene, die allein für das Messen benötigt werden, während die Zeit, die zusätzlich für das erneute steril Machen des Mediziners bzw. dessen Kommunikation mit der Assistenz im Kontrollraum gebraucht wird, außer Acht gelassen wurde. Daher ist davon auszugehen, dass die für die Maussteuerung erforderliche Bearbeitungszeit unter realen Bedingungen höher liegt und sich somit der Unterschied zwischen dem gestengesteuerten Prototypen und seiner mausgesteuerten Version verringert. Um dabei jedoch genau bestimmen zu können, wie stark sich der zeitliche Unterschied verändert, ist ein Vergleich der beiden Steuerungen unter den entsprechenden Bedingungen im Operationssaal erforderlich.
Darüber hinaus scheinen die Probleme des gestengesteuerten Prototypen weniger in den grundlegenden Interaktionstechniken und Konzepten der Benutzungsoberfläche begründet als vielmehr durch Mängel bei der Gestenerkennung sowie den Mechanismen zur Unterstützung des Nutzers. So sollten neben der Korrektur der Handhabung des Radialmenüs hauptsächlich die Algorithmen zur Detektion und Verarbeitung der Gesten sowie die Nutzerunterstützung für eine höhere Genauigkeit verbessert bzw. erweitert werden.

7 Zusammenfassung

In dieser Arbeit erfolgte die Untersuchung der Frage nach der Eignung der Interaktion über Freihandgesten für die Vermessung medizinischer Strukturen während der Planung und Durchführung einer Intervention. Außerdem galt es herauszufinden, wie derartige, gestenbasierte Interaktionstechniken zu gestalten sind, um den Anforderungen dieses Anwendungsgebietes gerecht zu werden. Der Schwerpunkt lag dabei auf der Erhaltung der Sterilität bei gleichzeitiger Erfüllung der Kriterien der Benutzungsfreundlichkeit sowie der Überwindung der bei der Interaktion mit Freihandgesten auftretenden Herausforderungen.

So wurde zunächst das vorausgesetzte Hintergrundwissen im Bereich der Freihandgesten vermittelt und die bereits existierenden Untersuchungen zu den Themen Freihandgesten und Vermessung vorgestellt. Im Rahmen der anschließenden Anforderungsanalyse erfolgte die Präsentation der während einer Hospitation am Institut für Neuroradiologie des Universitätsklinikums Magdeburg gemachten Beobachtungen. Diese dienten zusammen mit Recherchen zur Usability von Freihandgesten und zur Vermessung in der Medizin als Basis für die Bestimmung der von der Freihandgestensteuerung zu erfüllenden Anforderungen.

Anhand der Anforderungsanalyse ließen sich schließlich die verschiedenen Interaktionskonzepte entwickeln. Dabei waren neben den unterschiedlichen Lösungen zur Vermessung von Distanzen, Winkeln, Flächen und Volumen auch grundlegende Interaktionen für das Zeigen und Selektieren sowie die Navigation in den medizinischen 2D- und 3D-Bilddaten zu erarbeiten. Des Weiteren wurden Konzepte zur Unterstützung des Nutzers entworfen. Diese umfassen sowohl spezielles, visuelles Feedback als auch Mechanismen, die die Genauigkeit der Gestensteuerung erhöhen sollen. Ersteres betrifft dabei die im jeweiligen Zustand verfügbaren Freihandgesteninteraktionen sowie die erkannten Gesten und Handdaten.

Die entwickelten Interaktionstechniken wurden schließlich in einem Prototyp umgesetzt. Da jedoch eine Implementierung aller konzipierten Interaktionen zur Vermessung den zeitlichen Rahmen der aktuellen Arbeit überstiegen hätte, erfolgte hier eine Beschränkung auf die Distanzmessung. Darüber hinaus wurde zusätzlich zu dem gestengesteuerten

Prototypen eine mit Maus und Tastatur gesteuerte Version implementiert, um die komparative Evaluierung zu ermöglichen.

Die mithilfe der beiden Prototypen durchgeführte Nutzerstudie bildete den Abschluss der Arbeit und sollte die Eignung der Freihandgestensteuerung im Vergleich zur Maussteuerung überprüfen. Zu diesem Zweck wurden beide Versionen des Prototyps im Rahmen eines Between Subject Designs von Probanden getestet und anschließend bewertet. Dabei erfolgte während der Erledigung der Vermessungsaufgaben durch die Probanden die Aufzeichnung der Bearbeitungszeit und der Genauigkeit der Messungen. Außerdem wurde die Usability anhand eines abschließenden Fragebogens erfasst sowie gemachte Beobachtungen festgehalten. Die so erzielten Ergebnissen konnten im Anschluss ausgewertet und diskutiert werden. Auf diese Weise ließen sich nicht nur Stärken und Schwächen der entwickelten Interaktionstechniken ableiten, sondern auch Verbesserungsmöglichkeiten herausarbeiten.

Anhand der Resultate der Studie zeigte sich, dass der gestengesteuerte Prototyp seiner mausgesteuerten Version hinsichtlich der Bearbeitungszeit und Genauigkeit unterlegen ist. Gleichzeitig führte die Handhabung der Freihandgesten nur zu einer geringfügig stärkeren Ermüdung in Händen, Armen und Schultern, während sie für die Finger geringer als bei der Steuerung über Maus und Tastatur ausfiel. Außerdem ließ sich für die Gestensteuerung trotz einiger Schwierigkeiten eine annähernd gute Usability feststellen, auch wenn diese schlechter bewertet wurde als bei der Maussteuerung.
Als größte Störfaktoren konnten dabei die mangelnde Robustheit der Gestenerkennung und die ungenügende Unterstützung des Nutzers für eine bessere Genauigkeit der Zeigegeste identifiziert werden. Dahingegen gab es nur wenige Probleme mit den grundlegenden Interaktionskonzepten.

7.1 Fazit

Zusammenfassend lässt sich festhalten, dass die Steuerung über Freihandgesten geeignete Interaktionskonzepte für die Vermessung in medizinischen Bilddaten während der Planung und Durchführung einer Intervention bieten kann, wenn ihr Potenzial durch entsprechende Fortschritte in der Robustheit der Gestenerkennung sowie der Genauigkeit der Eingaben weiter ausgeschöpft wird. Diese Art der berührungslosen Interaktion verringert durch das Vermeiden des Kontaktes mit dem nicht sterilen Eingabegerät außerdem das Infektionsrisiko.

7.2 Ausblick

Die aktuelle Arbeit bietet einen ersten Ansatz für die sterile Vermessung medizinischer Strukturen im Operationssaal mithilfe von Freihandgesten. Weiterführend steht jedoch noch Einiges an Arbeit bevor. So sollte zunächst die Umsetzung der in Abschnitt 6.4.3 beschriebenen Verbesserungsmöglichkeiten erfolgen. Diese betreffen in erster Linie eine flexiblere und robustere Gestaltung der Gestenerkennung sowie die Erhöhung der Genauigkeit und Stabilisierung des Pointers. Außerdem ist die Handhabung der Radialmenüs und Zustandsschaltflächen anzupassen, um Inkonsistenzen, unerwartetes Verhalten und Fehler zu vermeiden. Auch eine Optimierung der Pointervisualisierung sollte in Betracht gezogen sowie die aufgetretenen Implementierungsfehler behoben werden.
Ferner ist die Erweiterung des gestengesteuerten Prototyps um die Interaktionskonzepte für die Winkel-, Flächen- und Volumenmessung erforderlich, um diese ebenfalls hinsichtlich ihrer Eignung untersuchen zu können.

Darüber hinaus sind weitere Evaluierungen der entwickelten Interaktionstechniken erforderlich. Dabei könnte eine Erprobung des Prototyps unter realistischeren Arbeitsbedingungen weiteren Aufschluss geben, da Ärzte in der Regel nicht die Möglichkeit haben die Ellenbogen auf einem Tisch abzustützen und die Vermessungen im Stehen durchführen müssen.
Des Weiteren wäre ein Vergleich der Ergebnisse der Freihandgestensteuerung mit den Bearbeitungszeiten, Genauigkeiten und der Usability im Operationssaal interessant. Hier sollten verschiedene Szenarien, wie die Vermessung über einen Joystick oder über ein im Kontrollraum befindliches System, betrachtet werden. Auf diese Weise ließe sich beispielsweise in Erfahrung bringen, ob und inwiefern die direkte Bedienung mithilfe von Freihandgesten einen Vorteil gegenüber der indirekten Handhabung über einen Assistenten an einem klassisch gesteuerten System im Kontrollraum besitzt.
Außerdem ist zu überprüfen, ob eine längere und auch praktische Lernphase zur Eingewöhnung der Probanden eine Verbesserung der Ergebnisse bewirken kann. Auch eine umfangreichere Befragung der Testpersonen hinsichtlich der aufgetretenen Probleme bei der Bedienung des Prototyps sowie in Bezug auf Komponenten, die als besonders angenehm empfunden wurden, könnten aufschlussreich sein.

Anhang

Anhang A: Fragebogen für die Interviews am Universitätsklinikum Magdeburg

Fragen

1. Welche Messungen kommen zum Einsatz?
    a. Während der Planung einer Operation
    b. Während der Operation
    c. Bei der Ausbildung von angehenden Medizinern
2. Wie sehen die dazugehörigen Szenarien aus? Welche Teilaufgaben sind notwendig?
3. Wenn mehrere Arten von Messungen zum Einsatz kommen, wie häufig treten diese auf? (Priorisierung und absolute Häufigkeit)
4. Wie wichtig ist der Vergleich von Maßen? Unter welchen Umständen werden sie benötigt?
5. Welche Ein- und Ausgabegeräte kommen zum Einsatz?
6. Von welchen Personen werden die Messungen durchgeführt? Wer führt sie häufiger aus und wer nicht? Ist eine Zusammenarbeit notwendig?
7. Welche Einschränkungen bzw. Nachteile existieren bei den aktuellen Verfahren zur Messung?
8. Was sind die Vorteile der aktuellen Vorgehensweise und welche davon sollten unbedingt beibehalten werden?
9. Was ist für die Messung besonders wichtig? (z.B. das zur Verfügung stellen bestimmter Informationen)
10. Ändert sich etwas an der Vorgehensweise bei der Messung, wenn Zeitdruck herrscht? In welchem Szenario existiert überhaupt Zeitdruck?
11. Werden während der Messung noch weitere Tätigkeiten ausgeführt? Wenn ja, welche und wie oft?
12. Kommt es dazu, dass Messungen unterbrochen und kurze Zeit später fortgeführt werden müssen? Wenn ja, wie oft?
13. Müssen die Messungen immer gleich präzise sein? Wenn nein, bei welchen Maßen steht die Genauigkeit nicht an erster Stelle und warum?
14. Kommen Messungen in einem Kontext vor, der ohnehin schon ermüdend ist? Wenn ja, welche, wie häufig und in welchem Kontext?
15. Welche halbautomatisch berechneten Werte werden häufig benötigt? (z.B. minimaler Abstand)
16. Werden bestimmte Messungen nur auf bestimmten Bilddaten (CT, MRI, Röntgenaufnahmen usw.) ausgeführt? Wenn ja welche?
17. Was wird eher in 2D und was eher in 3D gemacht?
18. Hat der Mediziner nur eine Hand frei oder wäre auch der Einsatz von zwei Händen denkbar?
19. Wie viel Platz steht zur Ausführung von Gesten bereit?
20. Wo wäre die Steuerung der Messungen mit Freihandgesten vorstellbar?
21. Gibt es schon Erfahrung in der Verwendung von Freihandgesten? Sind diese gut oder schlecht?

Anhang B: Fragebogen für die Nutzerstudie

Fragebogen zur Evaluierung

Im Folgenden werden verschiedene Aussagen zum genutzten System getroffen. Bitte geben Sie zu jeder den Grad Ihrer Zustimmung an. Hierfür steht Ihnen eine Skala von 1 (Stimme gar nicht zu) bis 5 (Stimme voll zu) zur Verfügung.

	Stimme gar nicht zu				**Stimme voll zu**
	1	**2**	**3**	**4**	**5**
1. Ich kann mir sehr gut vorstellen, das System regelmäßig zu nutzen.	☐	☐	☐	☐	☐
2. Ich empfinde das System als unnötig komplex.	☐	☐	☐	☐	☐
3. Ich empfinde das System als einfach zu nutzen.	☐	☐	☐	☐	☐
4. Ich denke, dass ich technischen Support brauchen würde, um das System zu nutzen.	☐	☐	☐	☐	☐
5. Ich finde, dass die verschiedenen Funktionen des Systems gut integriert sind.	☐	☐	☐	☐	☐
6. Ich finde, dass es im System zu viele Inkonsistenzen gibt.	☐	☐	☐	☐	☐
7. Ich kann mir vorstellen, dass die meisten Leute das System schnell zu beherrschen lernen.	☐	☐	☐	☐	☐
8. Ich empfinde die Bedienung als sehr umständlich.	☐	☐	☐	☐	☐
9. Ich habe mich bei der Nutzung des Systems sehr sicher gefühlt.	☐	☐	☐	☐	☐
10. Ich musste eine Menge Dinge lernen, bevor ich mit dem System arbeiten konnte.	☐	☐	☐	☐	☐
11. Die Nutzung des Systems war ermüdend für …					
… die Finger.	☐	☐	☐	☐	☐
… die Handgelenke.	☐	☐	☐	☐	☐
… die Arme.	☐	☐	☐	☐	☐
… die Schultern.	☐	☐	☐	☐	☐

Literaturverzeichnis

[1] Ivo Rössling, Christian Cyrus, Lars Dornheim, Peter Hahn, Bernhard Preim, Andreas Boehm (2009): *Interaktive Visualisierung von Abständen und Ausdehnungen anatomischer Strukturen für die Interventionsplanung*. In: *Bildverarbeitung für die Medizin*, S. 381-385

[2] Ivo Rössling: *Vermessung von medizinischen Segmentierungen für die chirurgische Interventionsplanung und Dokumentation*. Dissertation, Otto-von-Guericke-Universität Magdeburg. 2015

[3] Bernhard Preim (2006): *Computerunterstützung für die bildbasierte medizinische Diagnostik und Therapieplanung*. In: *Magdeburger Wissenschaftsjournal*, 1, S. 19-26

[4] André Mewes, Patrick Saalfeld, Oleksandr Riabikin, Martin Skalej, Christian Hansen (2015): *A Gesture-Controlled Projection Display for CT-Guided Interventions*. In: *International Journal of Computer Assisted Radiology and Surgery*, 10, S. 1-8

[5] Patrick Saalfeld, André Mewes, Maria Lutz, Bernhard Preim, Christian Hansen (2015): *Comparative Evaluation of Gesture and Touch Input for Medical Software*. In: *Mensch und Computer*, S. 143-152

[6] Kenton O'Hara, Gerardo Gonzalez, Abigail Sellen, Graeme Penney, Andreas Varnavas, Helena Mentis, Antonio Criminisi, Robert Corish, Mark Rouncefield, Neville Dastur, Tom Carrell (2014): *Touchless Interaction in Surgery*. In: *Communications of the ACM 57*, 1 (Jan. 2014), S. 70-77

[7] Felix Ritter, Christian Hansen, Kjen Wilkens, Alexander Köhn, Heinz-Otto Peitgen (2009): *Benutzungsschnittstellen für den direkten Zugriff auf 3D-Planungsdaten im OP*. In: *Journal of i-com*, 8(1), S. 24-31

[8] Rose Johnson, Kenton O'Hara, Abigail Sellen, Claire Cousins, Antonio Criminisi (2011): *Exploring the potential for touchless interaction in image-guided interventional radiology*. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, S. 3323-3332

[9] Juan P. Wachs, Helman I. Stern, Yael Edan, Michael Gillam, Jon Handler, Craig Feied, Mark Smith (2008): *A Gesture-based Tool for Sterile Browsing of Radiology Images*. In: *Journal of the American Medical Informatics Association*, 15 (3), S. 321-323

[10] Marcio C. Cabral, Carlos H. Morimoto, Marcelo K. Zuffo (2005): *On the usability of gesture interfaces in virtual reality environments*. In: *CLIHC'05 Proceedings of the 2005 Latin American conference on Human-computer interaction*, S. 100-108

[11] Bernhard Preim, Raimund Dachselt: *Interaktive Systeme. Band 2. User Interface Engineering, 3D-Interaktion, Natural User Interfaces*. Springer-Verlag Berlin Heidelberg. 2015

[12] Werner Goebl: *Motion capture of piano performance*. URL: `http://iwk.mdw.ac.at/goebl/pianomocap.html` (letzter Abruf am 03.08.2016)

[13] SkywalkerPL: *Leap Motion Orion controller connected*. URL: `https://upload.wikimedia.org/wikipedia/commons/d/df/Leap_Motion_Orion_Controller_Plugged.jpg` (letzter Abruf am 03.08.2016)

[14] NetMediaEurope Deutschland GmbH, CBS Interactive: *Kinect 2 fuer Windows*. URL: `http://www.cnet.de/wp-content/uploads/2014/03/kinect-2-fuer-windows.jpg` (letzter Abruf am 03.08.2016)

[15] Iron Will Innovations Canada Inc.: *Peregrine Product*. URL: `http://theperegrine.com/product/` (letzter Abruf am 03.08.2016)

[16] Advanced Realtime Tracking: *Fingertracking*. URL: `http://www.ar-tracking.com/products/interaction/fingertracking/` (letzter Abruf am 03.08.2016)

[17] Julian Hettig: *Verwendbarkeit von myoelektrischer Gestensteuerung im Operationssaal*. Otto-von-Guericke-Universität Magdeburg. 2015

[18] Thalmic Labs Inc.: *Myo*. URL: `https://static.thalmic.com/sapphire/tenFootExperience/hero/myo.png` (letzter Abruf am 03.08.2016)

[19] Julian Hettig, André Mewes, Oleksandr Riabikin, Martin Skalej, Bernard Preim, Christian Hansen (2015): *Exploration of 3D Medical Image Data for Interventional Radiology using Myoelectric Gesture Control*. In: *Eurographics Workshop on Visual Computing for Biology and Medicine*, Chester, United Kingdom, S. 177-185

[20] Mohammad Riduwan, Ahmad Hoirul Basori, Farhan Mohamed (2013): *Finger-based gestural interaction for exploration of 3D heart visualization*. In: *Social and Behavioral Sciences*, 97, S. 684-690

[21] Luigi Gallo (2010): *A Glove-Based Interface for 3D Medical Image Visualization*. In: *Intelligent Interactive Multimedia Systems and Services*, S. 221-230

[22] Till Kipshagen, Mathis Graw, Volker Tronnier, Matteo Bonsanto, Ulrich G. Hoffmann (2009): *Touch- and marker-free interaction with medical software*. In: *IFMBE Proceedings*, 25 (6), S. 75-78

[23] Stefan Soutschek, Jochen Penne, Joachim Hornegger, Johannes Kornhuber (2008): *3-D Gesture-Based Scene Navigation in Medical Imaging Applications Using Time-Of-Flight Cameras*. In: *Computer Vision and Pattern Recognition Workshops*, S. 1-6

[24] Andrea Ferracani, Daniele Pezzatini, Alberto Del Bimbo (2014): *A Natural and Immersive Virtual Interface for the Surgical Safety Checklist Training*. In: *Proceedings of the 2014 ACM International Workshop on Serious Games*, S. 27-32

[25] Junchen Shen, Yanlin Luo, Zhongke Wu, Yun Tian, Qingqiong Deng (2016): *CUDA-based real-time hand gesture interaction and visualization for CT volume dataset using leap motion*. In: *The Visual Computer*, 32(3), S. 359-370

[26] Tolan Druckmiller, Tobias Poel: *Berührungslose Mensch-Computer-Interaktion (HCI) im Operationssaal*. Universität Bremen

[27] Andrea F. Abate, Maria De Marsico, Stefano Levialdi, Vincenzo Mastronardi, Stefano Ricciardi, Genoveffa Tortora (2008): *Gesture Based Interface for Crime Scene Analysis: A proposal*. In: *Computational Science and Its Applications – ICCSA 2008*, S. 143-154

[28] Bernhard Preim, Wolf Spindler, Karl J. Oldhafer, Heinz-Otto Peitgen (2001): *3d Distance Measurements in Medical Visualizations*. In: *Workshop on Interactive Medical Visualization and Analysis*, S. 31-36

[29] Bernhard Preim, Christian Tietjen, Wolf Spindler, Heinz-Otto Peitgen (2002): *Integration of Measurement Tools in Medical 3d Visualizations*. In: *Proceedings of the Conference on Visualization*, S. 21-28

[30] Bernhard Preim, Henry Sonnet, Wolf Spindler, Heinz-Otto Peitgen (2001): *Interaktive und automatische Vermessung in medizinischen 3d-Visualisierungen*. In: *SimVis*, S. 361-374

[31] John G. Hagedorn, Joy P. Dunkers, Steven G. Satterfield, Adele P. Peskin, John T. Kelso, Judith E. Terrill (2007): *Measurement Tools for the Immersive Visualization Environment: Steps Toward the Virtual Laboratory*. In: *Journal of Research of the National Institute of Standards and Technology*, 112 (5), S. 257-270

[32] Bernhard Reitinger, Dieter Schmalstieg, Alexander Bornik, Reinhard Beichel (2006): *Spatial Analysis Tool for Virtual Reality-based Surgical Planning*. In: *3D User Interfaces*, S. 37-44

[33] Can Ozmen, Selim Balcisoy (2008): *Bi-manual Interactive Tools for Cultural Heritage Researchers*. In: *International Conference on Cyberworlds*, S. 439-446

[34] Cristina Manresa-Yee, Esperanca Amengual, Pere Ponsa (2013): *Usability of Vision-Based Interfaces*. In: *Proceedings of CEDI 2013, Interaccion 2013 International Conference*, S. 113-118

[35] Jakob Nielsen: *Usability Engineering*. AP Professional. 1993

[36] Édimo Sousa Silva, Maria Andréia Formico Rodrigues (2014): *Design and Evaluation of a Gesture-Controlled System for Interactive Manipulation of Medical Images and 3D Models*. In: *SBC Journal on Interactive Systems*, 5 (3), S. 53-65

[37] Kenton O'Hara, Gerardo Gonzalez, Graeme Penney, Abigail Sellen, Robert Corish, Helena Mentis, Andreas Varnavas, Antonio Criminisi, Mark Rouncefield, Neville Dastur, Tom Carrell (2014): *Interactional Order and Constructed Ways in Seeing with Touchless Imaging System in Surgery*. In: *Computer Supported Cooperative Work*, 23 (3), S. 299-337

[38] Gang Ren, Eamonn O'Neill (2013): *3D selection with freehand gesture*. In: *Computers & Graphics*, 37, S. 101-120

[39] Bernhard Preim, Raimund Dachselt: *Interaktive Systeme. Band 1. Grundlagen, Graphical User Interface, Informationsvisualisierung*. Springer-Verlag Berlin Heidelberg. 2010

[40] Hans-Walter Fiedler: *Duplexsonographische Quantifizierung von Karotisstenosen auf der Grundlage des Geschwindigkeit-Zeit-Integrals*. Medizinische Hochschule Hannover. 2009

[41] Anette Römer: *Bewertung quantitativer sonografischer Gewebetypisierung und Längenbestimmung der Cervix uteri als neuer Prädiktor für Cervixinsuffizienz und Frühgeburtlichkeit*. Phillipps-Universität Marburg. 2007

[42] Heinz Handels: *Medizinische Bildverarbeitung. Bildanalyse, Mustererkennung und Visualisierung für die computergestützte ärztliche Diagnostik und Therapie*. Viweg+Teubner, GWV Fachverlage GmbH. Wiesbaden. 2009

[43] Bernhard Preim, Christian Tietjen, Milo Hindennach, Heinz-Otto Peitgen (2003): *Integration automatischer Abstandsberechnungen in die Interventionsplanung*. In:

Bildverarbeitung für die Medizin 2003, Algorithmen – Systeme – Anwendungen, Proceedings des Workshops vom 9.-11. März 2003 in Erlangen, Springer-Verlag Berlin Heidelberg, S. 259-263

[44] Sameer Ahmed, Stefan L. Zimmerman, Pamela T. Johnson, Hong Lai, Satomi Kawamoto, Karen M. Horton, Elliot K. Fishman (2014): *MDCT interpretation of the ascending aorta with semiautomated measurement software: Improved reproducibility compared with manual techniques*. In: *Journal of Cardiovascular Computed Tomography*, 8, S. 108-114

[45] Colin Ware: *Information Visualization. Perception for Design*. Elsevier Inc. 2004

[46] Debaleena Chattopadhyay, Davide Bolchini (2014): *Touchless Circular Menus: Toward an Intuitive UI for Touchless Interactions with Large Displays*. In: *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, S. 33-40

[47] Philip Kortum: *HCI Beyond the GUI. Design for Haptic, Speech, Olfactory and Other Nontraditional Interfaces*. Elsevier Inc. 2008

[48] Carmelo Adito, Maria Francesca Costabile, Hans-Christian Jetter (2014): *Gestures that people can understand and use*. In: *Journal of Visual Languages and Computing*, 25, S. 572-576

[49] Carole Graetzel, Terry Fong, Sebastien Grange, Charles Baur (2004): *A Non-Contact Mouse for Surgeon-Computer Interaction*. In: *Technology and Health Care*, 12(3), S. 245-257

[50] Sukeshini A. Grandhi, Gina Joue, Irene Mittelberg (2011): *Understanding Naturalness and Intuitiveness in Gesture Production: Insights for Touchless Gestural Interfaces*. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, S. 821-824

[51] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, Andy Cockburn (2013): *User-Defined Gestures for Augmented Reality*. In: *Human-Computer Interaction – INTERACT 2013*, S. 282-299

[52] Wim Fikkert, Paul van der Vet, Anton Nijholt (2010): *Gestures in an Intelligent User Interface*. In: *Multimedia Interaction and Intelligent User Interfaces*, S. 215-242

[53] Julien-Charles Lévesque, Denis Laurendeau, Marielle Mokhtari (2013): *An Asymmetric Bimanual Gestural Interface for Immersive Virtual Environments*. In: *Virtual*

Augmented and Mixed Reality. Designing and Developing Augmented and Virtual Environments, S. 192-201

[54] Ken Hinckley, Randy Pausch, Dennis Proffitt (1997): *Attention and Visual Feedback: The Bimanual Frame of Reference*. In: *Proceedings of the 1997 symposium on Interactive 3D graphics*, S. 121 – 126

[55] Ken Hinckley, Randy Pausch, Dennis Proffitt (1998): *Two-Handed Virtual Manipulation*. In. *ACM Transactions on Computer-Human Interaction*, 5(3), S. 260-302

[56] Ken Hinckley, Randy Pausch, John C. Goble, Neal F. Kassell (1994): *A Survey of Design Issues in Spatial Input*. In: *Proceedings of the 7th annual ACM symposium on User interface software and technology*, S. 213-222

[57] Yves Guiard (1987): *Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model*. In: *Journal of Motor Behavior*, 19 (4), S. 486-517

[58] Russel Owen, Gordon Kurtenbach, George Fitzmaurice, Thomas Baudel, Bill Buxton (2005): *When It Gets More Difficult, Use Both Hands – Exploring Bimanual Curve Manipulations*. In: *GI '05 Proceedings of Graphics Interface 2005*, S. 17-24

[59] Michael Nielsen, Moritz Störring, Thomas B. Moeslund, Erik Granum (2004): *A Procedure For Developing Intuitive And Ergonomic Gesture Interfaces For Man-Machine Interaction*. In: *Gesture-Based Communication in Human-Computer Interaction*, S. 409-420

[60] Gang Ren, Eamonn O'Neill (2013): *3D selection with freehand gesture*. In: *Computers & Graphics*, 37, S. 101-120

[61] Hani Karam, Jiro Tanaka (2014): *Two-Handed Interactive Menu: An Application of Asymmetric Bimanual Gestures and Depth Based Selection Techniques*. In: *Human Interface and the Management of Information. Information and Knowledge Design and Evaluation*, S. 187-198

[62] David Rempel, Matt J. Camilleri, David L. Lee (2014): *The design of hand gestures for human-computer interaction: Lessons from sign language interpreters*. In: *International Journal of Human-Computer Studies*, 72, S. 728-735

[63] Ferran Argelaguet, Carlos Andujar (2013): *A survey of 3D object selection techniques for virtual environments*. In: *Computers & Graphics*, 37(3), S. 121-136

[64] Brian S. Tani, Rafael S. Maia, Aldo von Wangenheim (2007): *A Gesture Inter-*

face for Radiological Workstations. In: *Proceedings of the IEEE Symposium on computer-based medical Systems*, S. 27-32

[65] Ravin Balakrishnan (2004): *„Beating" Fitts' law: virtual enhancements for pointing facilitation*. In: *International Journal of Human-Computer Studies*, 61(6), S. 857-875

[66] Jack Callahan, Dan Hopkins, Mark Weiser, Ben Shneiderman (1988): *An empirical comparison of pie vs. linear menus*. In: *CHI '88 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, S. 95-100

[67] Sören Lenman, Lars Bretzner, Björn Thuresson (2002): *Using Marking Menus to Develop Command Sets for Computer Vision Based Hand Gesture Interfaces*. In: *Proceedings of the second Nordic conference on Human-computer interaction*, S. 239-242

[68] Dustin B. Chertoff, Ross W. Byers, Joseph J. LaViola Jr. (2009): *An Exploration of Menu Techniques using a 3D Game Input Device*. In: *Proceedings of the 4th International Conference on Foundations of Digital Games*, S. 256-262

[69] Peter Brandl, Clifton Forlines, Daniel Wigdor, Michael Haller, Chia Shen (2008): *Combining and Measuring the Benefits of Bimanual Pen and Direct-Touch Interaction on Horizontal Interfaces*. In: *Proceedings of the working conference on Advanced Visual Interfaces*, S. 154-161

[70] Robert Walter, Gilley Bailly, Jörg Müller (2013): *StrikeAPose: Revealing Mid-Air Gestures on Public Displays*. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, S. 841-850

[71] Gustavo Rovelo, Donald Degraen, Davy Vanacken, Kris Luyten, Karin Coninx (2015): *Gestu-Wan – An Intelligible Mid-Air Gesture Guidance System for Walk-up-and-Use Displays*. In: *Human-Computer-Interaction – INTERACT 2015*, S. 368-386

[72] Debaleena Chattopadhyay, Davide Bolchini (2014): *Understanding Visual Feedback in Large-Display Touchless Interactions: An Exploratory Study*. IUPUI Scholar Works. Indiana University

[73] Paul Kabbash, William Buxton (1995): *The "Prince" Technique: Fitts' Law and Selection Using Area Cursors*. In: *Proceedings of CHI'95*, S. 273-279

[74] Tao Ni, Doug A. Bowman, Chris North, Ryan P. McMahan (2011): *Design and evaluation of freehand menu selection interfaces using tilt and pinch gestures*. In: *International Journal of Human-Computer Studies*, 69, S. 551-562

[75] Nadia Brancati, Guiseppe Caggianese, Maria Frucci, Luigi Gallo, Pietro Neroni (2015): *Touchless Target Selection Techniques for Wearable Augmented Reality Systems*. In: *Intelligent Interactive Multimedia Systems and Services*, S. 1-9

[76] Doug A. Bowman, Chadwick A. Wingrave, Joshua M. Campbell, Vinh Q. Ly, Christopher J. Rhoton (2002): *Novel Uses of Pinch Gloves for Virtual Environment Interaction Techniques*. In: *Virtual Reality*, 6(3), S. 122-129

[77] Peng Song, Wooi Boon Goh, William Hutama, Chi-Wing Fu, Xiaopei Liu (2012): *A Handle Bar Metaphor for Virtual Object Manipulation with Mid-Air Interaction*. In: *CHI 2012*, S. 1297-1306

[78] Daniel Mendes, Fernando Fonseca, Bruno Araújo (2014): *Mid-Air Interactions Above Stereoscopic Interactive Tables*. In: *IEEE Symposium on 3D User Interfaces 2014*, S. 3-10

[79] Joachim Grifka, Jürgen Krämer: *Orthopädie Unfallchirurgie*. Springer-Verlag Berlin Heidelberg. 2013

[80] Yupo Chan: *Location, Transport and Land-Use. Modelling Spatial-Temporal Information*. Springer-Verlag Berlin Heidelberg. 2005

[81] Rob J. Hyndman, George Athanasopoulos (2013): *Forecasting: principles and practice*. OTexts, Melbourne, Australia. URL: `http://otexts.org/fpp/` (letzter Abruf am 29.06.2016)

[82] Christian Tietjen: *Modifikation und Erweiterung von Techniken zur interaktiven und automatischen Vermessung in medizinischen 3D Visualisierungen*. Diplomarbeit, Otto-von-Guericke-Universität Magdeburg. 2002

[83] Elmos Semiconductor AG: *E909.06 HALIOS® 2-Inch-Slider. Application Note*. Elmos. 2015

[84] Thalmic Labs Inc.: *Gestures and Motion*. In: *Tech Specs*. URL: `https://www.myo.com/techspecs` (letzter Abruf am 15.03.2016)

[85] Senic GmbH: *NUIMO*. URL: `https://www.senic.com/` (letzter Abruf am 30.06.2016)

[86] Intel Corporation: *Intel RealSense Technology*. URL: `https://software.intel.com/de-de/realsense/home` (letzter Abruf am 30.06.2016)

[87] Leap Motion Inc.: *Leap Motion for Mac and PC*. URL: `https://www.leapmotion.com/product/desktop` (letzter Abruf am 15.03.2016)

[88] Microsoft: *Kinect für Xbox One*. URL: http://www.xbox.com/de-DE/xbox-one/accessories/kinect-for-xbox-one#fbid=60ENEfSYCrz (letzter Abruf am 18.03.2016)

[89] SoftKinetic Korea Ltd.: *DS525 Datasheet. Short Range Module*. URL: http://www.softkinetic.com/Portals/0/Users/234/06/16106/US_20130527_SK_DS525_Datasheet_Verso_V3.0WEB.pdf (letzter Abruf am 17.03.2016)

[90] SoftKinetic: *FAQ*. URL: http://www.softkinetic.com/Support/FAQ (letzter Abruf am 17.03.2016)

[91] Frank Weichert, Daniel Bachmann, Bartholomäus Rudak, Denis Fisseler (2013): *Analysis of the Accuracy and Robustness of the Leap Motion Controller*. In: *Sensors 2013*, 13, S. 6380-6393

[92] Joze Guna, Grega Jakus, Matevz Pogacnik, Saso Tomazic, Jaka Sodnik (2014): *An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking*. In: *Sensors 2014*, 14, S. 3702-3720

[93] Andrea Corti, Silvio Giancola, Giacomo Mainetti, Remo Sala (2016): *A metrological characterization of the Kinect V2 time-of-flight camera*. In: *Robotics and Autonomous Systems*, 75, S. 584-594

[94] SoftKinetic: *iisu 3.6 Datasheet. Software Development Kit*. URL: http://www.softkinetic.com/Portals/0/Documents/PDF/WEB_20130527_iisu%203.6%20Datasheet_V2.0.pdf (letzter Abruf am 17.03.2016)

[95] Leap Motion Inc.: *API Reference*. URL: https://developer.leapmotion.com/documentation/python/api/Leap_Classes.html (letzter Abruf am 16.03.2016)

[96] Alex Colgan (Leap Motion Inc.): *What can the mounted controller track?* In: *12 FAQ About the VR Developer Mount*. URL: http://blog.leapmotion.com/12-faqs-vr-developer-mount/ (letzter Abruf am 16.03.2016)

[97] Microsoft: *Body Tracking*. URL: https://msdn.microsoft.com/de-de/library/dn799273.aspx (letzter Abruf am 15.03.2016)

[98] Microsoft: *Kinect hardware*. URL: https://dev.windows.com/en-us/kinect/hardware (letzter Abruf am 15.03.2016)

[99] Xu Xu, Raymond W. McGorry (2015): *The validity of the first and second genera-*

tion Microsoft Kinect™for identifying joint center locations during static postures. In: *Applied Ergonomics*, 49, S. 47-54

[100] Icojam: *Unigrid Tools*. URL: https://www.iconfinder.com/iconsets/unigrid-tools (letzter Abruf am 24.06.2016)

[101] Liubou Yasiukovich: *Hand Maussymbol Zeiger*. URL: http://de.123rf.com/photo_34259672_stock-vector-hand-mouse-icon-pointer-finger-click-icon.html (letzter Abruf am 24.06.2016)

[102] greyjj: *Vector Hände Symbole gesetzt Fingerzählen, Stopp-Geste, Faust, Teufel Hörner Geste, in Ordnung Geste, Friedenszeichen*. URL: http://de.123rf.com/photo_23520716_stock-photo.html (letzter Abruf am 24.06.2016)

[103] PSD Graphics: *Right and wrong check marks*. URL: http://de.freepik.com/index.php?goto=41&idd=567818 (letzter Abruf am 15.08.2016)

[104] Klaus D. Tönnies: *Grundlagen der Bildverarbeitung*. Pearson Studium. München. 2005

[105] Gary Bradski, Adrian Kaehler: *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly & Associates. 2008

[106] Wilhelm Burger, Mark J. Burge: *Digital Image Processing. An Algorithmic Introduction Using Java. Second Edition*. Springer-Verlag. London. 2016

[107] Sectra AB: *Sectra Visualization Table*. URL: https://www.sectra.com/medical/sectra_table/solutions/pdf/product_leaflet_Sectra_Table_6.0.pdf (letzter Abruf am 04.08.2016)

[108] John Brooke: *SUS – A quick and dirty usability scale*. Digital Equipment Corporation. 1986

[109] Matthias Rauer (2011):*Quantitative Usability-Analysen mit der System Usability Scale (SUS)*. URL: https://blog.seibert-media.net/blog/2011/04/11/usablility-analysen-system-usability-scale-sus/ (letzter Abruf am 25.07.2016)

[110] Sectra AB: *Sectra Table – A tool for developing critical thinking*. URL: http://www.sectra.com/medical/sectra_table/solutions/education/index.html (letzter Abruf am 26.07.2016)

[111] Werner A. König, Jens Gerken, Stefan Dierdorf, Harald Reiterer (2009): *Adaptive*

Pointing – Design and Evaluation of a Precision Enhancing Technique für Absolute Pointing Devices. In: *Human-Computer Interaction – INTERACT 2009*, S. 658-671

[112] Scott Frees, G. Drew Kessler (2005): *Precise and Rapid Interaction through Scaled Manipulation in Immersive Virtual Environments*. In: *Proceedings Virtual Reality 2005*, S. 99-106

[113] Anatole Lécuyer, Jean-Marie Burkhardt, Laurent Etienne (2004): *Feeling Bumps and Holes without Haptic Interface: the Perception of Pseudo-Haptic Textures*. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, S. 239-246

[114] Wolfram Schoor, Felix Bollenbeck, Marc Hofmann, Rüdiger Mecke, Udo Seiffert, Bernhard Preim (2008): *Automatic Zoom and Pseudo Haptics to Support Semiautomatic Segmentation Tasks*. In: *16th International Conference in Central Europe On Computer Graphics, Visualization and Computer Vision, Wscg'2008 - in Co-Operation With Eurographics*, S. 81-88

[115] Scott Frees, G. Drew Kessler, Edwin Kay (2007): *PRISM interaction for enhancing control in immersive virtual environments*. In: *ACM Transactions on Computer-Human Interaction*, 14(1), Artikel Nr. 2

[116] Leap Motion Inc.: *Orion Beta*. URL: `https://developer.leapmotion.com/orion` (letzter Abruf am 09.08.2016)

[117] Hanno Scharr: *Optimale Operationen in der Digitalen Bildverarbeitung*. Dissertation, Ruprecht-Karls-Universität Heidelberg. 2000