

Institut für Visualisierung und Interaktive Systeme
Abteilung Mensch-Computer-Interaktion
Universität Stuttgart
Pfaffenwaldring 5a
D-70569 Stuttgart

Bachelorarbeit Nr. 71

Blickbasierte Interaktion mit 3D Displays

**Entwicklung einer Kommunikationsarchitektur
zur Evaluierung blickbasierter Interaktion**

Rufat Rzayev

Studiengang:	Informatik
Prüfer/in:	Prof. Dr. Albrecht Schmidt
Betreuer/in:	Prof. Dr. Florian Alt M.Sc. Stefan Schneegaß
Beginn am:	2. Mai 2013
Beendet am:	1. November 2013
CR-Nummer:	H.5.2

Kurzfassung

Die 3D-Display-Technologie ist heutzutage eine der wichtigsten Teilen der Unterhaltungsindustrie. Durch die 3D-Technologie können Filme, Computeranimationen und Spiele ein noch realistischeres visuelles Erlebnis bieten. Um mehr Benutzer anzusprechen ist es nicht ausreichend Applikationen nur visuell zu verbessern. Die Interaktionstechniken, die die Benutzung der 3D-Applikationen vereinfachen und das Navigieren und Manipulieren der Objekte in 3D-Raum ermöglichen, sollen auch definiert werden.

In dieser Arbeit wird blickbasierte Interaktion mit 3D-Displays vorgestellt. Das Ziel dieser Arbeit ist die Implementierung und die Evaluierung der vorgestellten Interaktionstechniken. Unter Zuhilfenahme von diesen Interaktionsmethoden können Objekte auf dem 3D-Display manipuliert werden. Außerdem wird untersucht, ob die Objekte, die in unterschiedlichen Tiefenebenen auf dem 3D-Display dargestellt sind, unter Verwendung des Blicks des Benutzers erkannt werden können. Dazu werden in der Arbeit zwei unterschiedliche Methoden vorgestellt, welche als Berechnungsgrundlage zum einen den Pupillendurchmesser und zum anderen die Distanz zwischen linker und rechter Pupille verwenden. Im Folgenden werden die Ergebnisse der durchgeführten Benutzerstudie zu Evaluierung dieser Methoden und der Vergleich der blickbasierten Interaktionsmethoden auf 2D- und 3D-Displays vorgestellt.

Abstract

The 3D display technology is nowadays one of the most important parts of the entertainment industry. By using 3D technology, films, computer animations, and games may provide an even more realistic visual experience. In order to attract more users, it is not sufficient to improve applications only visually. The interaction techniques that simplify the use of 3D applications and allow to navigate and manipulate objects in 3D space need also be addressed.

In this thesis, gaze-based interaction with 3D displays is explored. The goal of this thesis is the implementation and evaluation of the presented interaction techniques. With the aid of these interaction methods, objects can be manipulated on the 3D display. It also examines whether the objects shown at different depth levels can be identified using the user's gaze. Two different methods are presented in this thesis, which use the distance between the left and the right pupils or the pupil diameter to determine the gaze depth. In the following, the results of the user study and the comparison of the gaze-based interaction methods on 2D and 3D displays are presented.

Inhaltsverzeichnis

1	Einleitung	9
1.1	Motivation	9
1.2	Aufgabenstellung	10
1.3	Gliederung	11
2	Hintergrund	13
2.1	Das menschliche Auge	13
2.2	3D-Wahrnehmung	14
2.2.1	Okulomotorische Tiefenhinweise	14
2.2.2	Visuelle Tiefenhinweise	15
2.3	Stereoskopische Displaytechnologie	17
2.3.1	Farbe-Multiplex Ansatz	18
2.3.2	Polarisation-Multiplex Ansatz	19
2.3.3	Zeit-Multiplex Ansatz	19
2.4	Eyetracking	19
2.4.1	Augenbewegung	20
2.4.2	Eyetracking Technologie	21
3	Verwandte Arbeiten	23
3.1	Interaktion mit 3D-Displays	23
3.2	Verwandte Arbeiten im Bereich Eyetracking	24
3.2.1	Eyetracking als Interaktionstechnik	25
3.2.2	Eyetracking als Bewertungsmöglichkeit für Usability	29
4	Konzept	31
5	Entwurf	33
5.1	Der Ablauf des Spiels	33
5.2	Verwendete Programmen	34
5.2.1	Die Spiel-Engine	34
5.2.2	Die Programmen für Grafikbearbeitung	35
5.2.3	iView X™	35
6	Implementierung	37
6.1	Kommunikationsserver	37
6.1.1	ConnectionHandler	38
6.1.2	RemoteController	38

6.1.3	StreamHandler	39
6.1.4	Grafische Benutzeroberfläche	40
6.2	Spiel	41
6.2.1	Kalibrierung	42
6.2.2	Spielbare Szenen	43
6.2.3	Tiefenkalibrierung	44
6.2.4	Feedback	45
6.2.5	Datenerfassung	45
6.2.6	Erzeugung von stereoskopischem Inhalt	45
7	Evaluierung	47
7.1	Verwendete Hardware	47
7.1.1	Eye-Tracker: iView X RED	47
7.1.2	Laptop: Sony VAIO VPC-F22C5E	48
7.1.3	Laptop: Lenovo ThinkPad X200	48
7.1.4	Server	48
7.1.5	Display: Philips 3D-Fernseher: 55PFL7606K/02	48
7.2	Versuchsaufbau	48
7.3	Versuchsablauf	49
7.4	Teilnehmer	50
8	Ergebnisse und Diskussion	51
8.1	Die Ergebnisse der Fragebögen	51
8.2	Die Ergebnisse der Datenerfassung	53
8.3	Diskussion	55
9	Zusammenfassung und Ausblick	57
A	Ein Anhang	59
A.1	Technische Daten der verwendeten Hardware	59
A.2	Verwendete Fragebögen	60
	Literaturverzeichnis	65

Abbildungsverzeichnis

2.1	Schematische Darstellung des menschlichen Auges.	14
2.2	Monokulare Tiefenhinweise	16
2.3	Anaglyphen- und Polarisierungstechnik	18
2.4	Helle Pupille und Kornealreflex	22
3.1	Numerische Blickgesten für die PIN-Eingabe [DLWDo7]	28
3.2	Die Visualisierung der visuellen Aufmerksamkeit	29
5.1	Die Kommunikation zwischen dem Server, der Applikation und dem Eyetracker	34
6.1	Die Systemarchitektur	38
6.2	Grafische Benutzeroberfläche des Kommunikationsservers	41
6.3	Die Kalibrierungsszene	42
6.4	Eine Szene des ersten Teil des Spiels	44
6.5	Side-by-Side Ansicht	46
8.1	Die Ergebnisse von NASA-TLX	52
8.2	Durchschnittliche Geschwindigkeit von Meteoriten	53

Tabellenverzeichnis

8.1	SUS-Auswertung	51
8.2	Durchschnittlicher Spielpunktstand	54
A.1	Die technische Daten der verwendeten Laptops	59
A.2	Die technischen Daten des 3D-Displays	59

1 Einleitung

1.1 Motivation

Durch die Entwicklung der 3D-Technologie werden 3D-Displays in mehreren Bereichen verwendet. In den letzten Jahren wurden auf dem Markt viele 3D-fähigen Fernsehen, Projektoren, Laptops und Smartphones vorgestellt. 3D-Displays sind nicht nur ein beliebtes Werkzeug für die Visualisierung von Objekten in verschiedenen Forschungsgebieten wie im Bereich der Medizin, der mathematischen Analysis und vielen weiteren Bereichen, sie sind auch ein Bestandteil der Unterhaltungsindustrie. Viele Displays im Privatgebrauch sind durch immer billiger werdende Hardware 3D-fähig. Durch diese günstig gewordenen Displays kann man Filme oder Spiele zu Hause dreidimensional genießen.

Bei den Herstellern gibt es heutzutage starke Tendenz auf 3D-Geräte. Es wurde angenommen, dass der 3D-Display-Markt im Jahr 2019 weltweit auf 226 Million Geräte anwachsen wird ¹. Allerdings ist die 3D-Technologie nicht neu. Der erste öffentliche 3D-Film war "The Power of Love" im Jahre 1922 ². Um die 3D-Effekte zu realisieren, wurde die Anaglyphen-Technologie benutzt. Die Zuschauer trugen Brillen mit roten und blauen Gläser, um diese Effekte wahrnehmen zu können. Durch die Weiterentwicklung der 3D-Technologie und wachsende Rechenleistung wurden noch weitere Methoden, wie Polarisationstechnik, Shuttertechnik entwickelt. Durch die autostereoskopische Displays ist eine Realisierung der dreidimensionale Wahrnehmung auch ohne Brillen möglich.

3D-Bildschirme bieten gewisse Vorteile, gegenüber herkömmlichen Bildschirmen. Benutzer können durch die zusätzliche Tiefeninformation dreidimensionale Objekte besser verstehen. Damit könnten die Benutzer ein natürlicheres visuelles Erlebnis haben. Für die Unternehmen sind die 3D-Displays auch attraktiv. Beispielsweise könnten die Produkte eines Unternehmens auf einem autostereoskopischen, öffentlichen Bildschirm beworben werden. Die räumliche Darstellung dieser Produkte könnte die Aufmerksamkeit der Beobachter erregen.

Trotz des Vorteils und der Beliebtheit fehlen den meisten 3D-Bildschirmen die Interaktionsmöglichkeiten. Der Grund dafür ist, dass die Realisierung dieser Art von Interaktion problembehaftet ist. Außerdem gibt es für die Interaktion mit 3D-Displays bis jetzt keinen Standard [Bow13]. Die Benutzung der Elemente einer Bedienoberfläche eines Programms

¹Quelle: http://www.displaysearch.com/cps/rde/xchg/displaysearch/hs.xsl/120926_tv_to_drive_global_demand_for_3d_ready_devices.asp, letzter Zugriff am 20.10.2013

²Quelle: <http://www.visionnw.com/history-of-3d-technology.html>, letzter Zugriff am 20.10.2013

und ihre Interaktionsmetaphern einer 2D-Umgebung sind nicht in 3D-Applikationen genauso verwendbar [SHSKo8]. Die Interaktion mit Inhalten durch herkömmliche Eingabegeräte - wie der Maus oder der Tastatur - auf einem 3D-fähigen Display kann nicht genau wie im 2D-Fall erfolgen. Deshalb sollte die 2D-Eingabemethode für einen 3D-Raum erweitert werden. Außerdem ist die Entfernung zum Display bei der Benutzung einiger Eingabegeräte problematisch. Um ein gutes visuelles Erlebnis zu garantieren, ist ein bestimmte Abstand zum Display erforderlich. Dadurch verringert sich der Handlungsspielraum der Benutzer.

"Natürliche" Interaktionsmethoden können verwendet werden, um diese Probleme zu lösen. Meistens verfügen diese Methoden mindestens über drei Freiheitsgrade. Diese sind ausreichend, um im dreidimensionalen Raum zu navigieren. Außerdem muss der Benutzer sich nicht nah am Display befinden, um mit dessen Inhalt zu interagieren.

Das Ziel dieser Arbeit ist der Entwurf und die Implementierung der Interaktionstechniken für 3D-Displays. Durch die Benutzung der vorgestellten Interaktionsmethoden soll es dem Nutzer eines stereoskopischen Displays möglich sein, in einem gewissen Abstand, ohne die oben genannten Probleme mit dem Display interagieren zu können. Wie gut diese Interaktionstechnik für 3D-Displays geeignet ist, wird in dieser Arbeit evaluiert.

1.2 Aufgabenstellung

Diese Bachelorarbeit untersucht blickbasierte Interaktion auf einem stereoskopischen Display. Für die blickbasierte Interaktion wird der Blick des Benutzers verfolgt und als Eingabeparameter benutzt. Diese Art der Interaktion entspricht der natürlichen Augenbewegungen des Benutzers und diese brauchen keine außerordentlichen Kenntnisse, um mit einem solchen System zu interagieren [Jac95]. Der Benutzer befindet sich in einer bestimmten Entfernung zum Display und kontrolliert den Inhalt auf dem Bildschirm nur mit den Augen. Ein weiterer Vorteil ist, dass die blickbasierte Interaktion gut geeignet für behinderte und körperlich eingeschränkte Menschen ist. Für körperlich gesunde Benutzer kann die Blickinformation als zusätzlicher Eingabeparameter dienen, sodass beispielsweise die Hände für anderen Aufgaben benutzt werden können.

Die Blickerfassung wird durch die Verwendung von Eyetrackern durchgeführt. Außer der Blickkoordinaten kann dieses Gerät auch Informationen, wie die Pupillenposition und der Pupillendurchmesser zurückgeben. Auf einer zweidimensionalen Oberfläche funktioniert die Blickverfolgung verlässlich. In dieser Arbeit soll die Blickerfassung in einem dreidimensionalen virtuellen Raum evaluiert werden. Außerdem soll es untersucht werden, ob es möglich ist die Objekte, die in unterschiedlichen Tiefen auf einem stereoskopischen Display dargestellt sind, mit der Verwendung eines Eyetrackers zu erkennen.

Für die Arbeit soll ein Server entwickelt werden, der für die Fernsteuerung des Eyetrackingsystems zuständig ist. Durch die Benutzung eines solchen Servers sollen die wichtigen Steuerungsbefehle des Eyetrackers und die dazu gehörigen Parameter durch herkömmliche Methoden (z.B. Java-Methoden) vereinfacht werden. Dieser Server kann später in verschiedensten Applikation verwendet werden. Des weiteren soll eine Beispielapplikation entwickelt

werden, welche den Server nutzt, um den Eyetracker als Eingabemöglichkeit bereitzustellen. Am Ende soll die blickbasierte Interaktion mit 3D-Displays durch verschiedene Kriterien in einer Benutzerstudie evaluiert werden.

1.3 Gliederung

Die Arbeit ist in folgender Weise gegliedert:

Kapitel 2 – Hintergrund: In diesem Kapitel wird auf die grundlegenden Informationen über stereoskopische Display-Technologien und Blickerfassungsmethoden eingegangen. Dabei werden wichtige Themen, wie die Anatomie des menschlichen Auges, die 3D-Wahrnehmung und verschiedene Arten der Augenbewegungen behandelt.

Kapitel 3 – Verwandte Arbeiten: Hier werden verschiedene verwandte Arbeiten aus dem Bereich der Interaktion mit stereoskopischen Bildschirmen vorgestellt. Danach wird auf die verwandte Arbeiten eingegangen, in denen die blickbasierte Interaktion als Interaktionstechnik verwendet wurde. Zum Schluss wird die Anwendung des Eyetrackings bei der Usability-Evaluierungen kurz behandelt.

Kapitel 4 – Konzept: Hier wird auf das Konzept der Arbeit eingegangen. Zuerst wird eine grobe Beschreibung der vorgestellten Interaktionsmethoden gegeben. Danach werden mögliche Anwendungsfälle für diese Interaktionsmethoden behandelt.

Kapitel 5 – Entwurf: In diesem Kapitel wird ein Szenario unter Verwendung der Beispielapplikation für die Interaktion beschrieben. Zusätzlich wird eine Übersicht über die verwendeten Softwarekomponenten gegeben.

Kapitel 6 – Implementierung: Dieses Kapitel beinhaltet die Implementierung der einzelnen Komponenten - des Kommunikationsservers und der Beispielapplikation, die für die Evaluierung der vorgestellten Interaktionsmethoden entwickelt wurden.

Kapitel 7 – Evaluierung: In diesem Kapitel wird die durchgeführte Benutzerstudie beschrieben. Dabei werden Versuchsabbau, der Ablauf der Evaluierung und die Information über Studienteilnehmer detailliert vorgestellt.

Kapitel 8 – Ergebnisse und Diskussion: Hier werden die Ergebnisse der Evaluierung durch die Fragebögen und die Datenerfassung der Beispielapplikation vorgestellt. Zum Schluss werden diese Ergebnisse diskutiert.

Kapitel 9 – Zusammenfassung und Ausblick: Dieses Kapitel fasst die Arbeit zusammen und stellt Ideen für zukünftige Arbeiten vor.

Diese Arbeit entstand in Zusammenarbeit mit Jonas Auda. In der Arbeit sind mit * gekennzeichnete Abbildungen auch in der Arbeit von Jonas Auda verwendet worden.

2 Hintergrund

In dieser Arbeit soll untersucht werden, wie Anwendungen auf einem stereoskopischen Display dargestellt und mit der Blickbewegung kontrolliert werden kann. Da die Interaktion blickbasiert ist, sollte die Funktionsweise des menschlichen Auges verstanden werden. Außerdem sollen visuelle Wahrnehmungsaspekte untersucht werden, damit eine Applikation fehlerfrei funktioniert und der Benutzer sich während der Verwendung nicht unangenehm fühlt. Zusätzlich sollen die Blickerfassungsmethoden und die dafür benötigte Hardware untersucht werden.

In diesem Kapitel wird auf die grundlegenden Informationen über die räumliche Wahrnehmung und die Blickverfolgungstechnologie eingegangen. In dem ersten Teil wird kurz die Struktur des menschlichen Auges erklärt. Der zweite Teil stellt die Grundlagen der dreidimensionalen Wahrnehmung vor. Hier werden verschiedenen Tiefenhinweise vorgestellt, womit das Gehirn den Abstand zu Objekten in der Umgebung abschätzt. Im dritten Teil werden stereoskopische Displaytechnologien diskutiert. Das Eyetracking und die dazu gehörigen Blickerfassungsmethoden sind Thema des letzten Teil dieses Kapitels.

2.1 Das menschliche Auge

Das Auge, ein komplexes Sinnesorgan, besteht aus vielen Teilen, wie Muskeln, Nerven, Häuten. In Abbildung 2.1 ist ein vereinfachter Aufbau des menschlichen Auges dargestellt. Mit bloßem Auge sind die Iris, die Lederhaut des Auges und die Pupille sichtbar. Die durch ihre weiße Farbe erkennbare Lederhaut schützt das Auge. Auf der vorderen Seite des Auges kann die farbige Iris leicht unterschieden werden. Diese ist durch die transparente Hornhaut abgedeckt. Die Pupille, die den Innenteil der Iris bildet, sorgt für die Steuerung der Lichtmenge, die in das Auge fällt. Dafür kann die Größe der Pupille durch Muskeln geändert werden. Hinter der Pupille befindet sich eine Linse, die mithilfe von den Ziliarmuskeln der Akkommodation (siehe Unterabschnitt 2.2.1) dient.

In der Netzhaut des Auges liegen die Photorezeptoren, Zapfen und Stäbchen, die ermöglichen, im Hellen und Dunklen zu sehen. Die Netzhaut enthält ungefähr 120 Millionen Stäbchen, die für achromatisches und schwaches Licht empfindlich sind und somit der Nachtsicht dienen und circa 7 Millionen Zapfen, die für helles, chromatisches Licht empfindlich sind und das Sehen bei guten Lichtverhältnissen ermöglichen [Duco7].

Die höchste Dichte von Photorezeptoren befindet sich nahe dem Zentrum der Netzhaut, wo die Fovea liegt. Die Fovea enthält 147.000 Zapfen/mm² und nur wenig Stäbchen [Duco7]. Da die Zapfen das Gehirn meistens mit visuellen Informationen versorgen, haben Bilder,

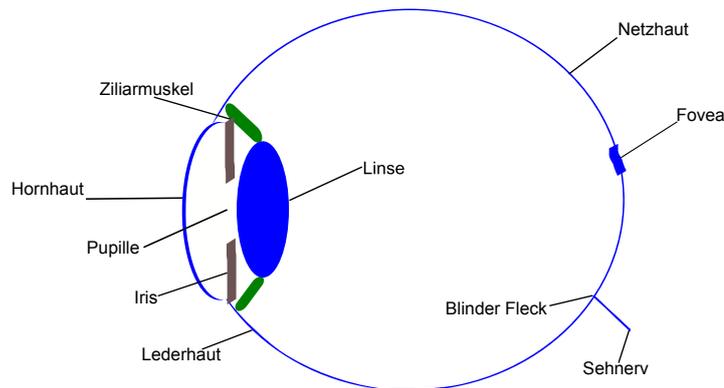


Abbildung 2.1: Schematische Darstellung des menschlichen Auges.

die ihren Fokus in der Fovea haben, höhere Sehschärfe als Bilder, die ihren Fokus in der umliegenden Fläche haben [Jac95].

2.2 3D-Wahrnehmung

Menschen haben zwei Augen, welche eine bessere Tiefenwahrnehmung ermöglichen. Stereoskopische Displays haben eine zweidimensionale Oberfläche. Aber wenn man diese anschaut, interpretiert das Gehirn den Inhalt auf dem Display als räumlich. Der Grund dafür sind die Tiefenhinweise. Es gibt viele Tiefenhinweise, die dem Gehirn ermöglichen, von den Bildern, die auf der Netzhaut abgebildet sind, Tiefeninformation zu bekommen. Es ist wichtig für die 3D-Anwendungsentwickler, Tiefenhinweise aktiv zu beachten, weil in bestimmten Situationen spezielle Tiefenhinweise zu Konflikten bei der Tiefenerkennung führen können [McA02]. Diese Hinweise können auf okulomotorische und visuelle Tiefeninformationen aufgeteilt werden. In unten stehenden Unterabschnitten werden verschiedene Tiefenhinweise vorgestellt. [Hol05] und [McA02] wurden als Quellen verwendet.

2.2.1 Okulomotorische Tiefenhinweise

Mithilfe der okulomotorischen Tiefenhinweise können die Tiefeninformationen durch die Konvergenz und durch die Akkommodation entstandene Spannung auf den Augenmuskeln erlangt werden. Diese Hinweise sind binokular.

Konvergenz

Falls man einen Gegenstand anschaut, der sich einem nähert, werden Pupilleneigenschaften beeinflusst. In diesem Fall nähern sich die Pupillen des rechten und des linken Auges

einander an. Dieser Fall heißt Konvergenz. Bei der Konvergenz kann das Gehirn die Tiefeninformationen einer Szene erlangen, indem die Unterschiede zwischen Bildern des linken und rechten Auge verwendet werden. Ist das beobachtete Objekt nah, dann wird der Unterschied zwischen Bildern größer sein [KT04].

Akkommodation

Falls man einem Gegenstand eines bestimmten Abstands fokussiert, wird der Hintergrund unscharf wahrgenommen. Umgekehrt wird man einen Gegenstand verschwommen sehen, wenn der Fokus auf dem Hintergrund liegt. Um Objekte in verschiedenen Abständen zu fokussieren, ändern sich die Brennweite der Augenlinsen. Diese Änderung heißt Akkommodation. Für diese Änderung ist der Ziliarmuskel zuständig, der für nähere Objekte die Linsenkrümmung entsprechend anpasst [RHFL10]. Dadurch gewinnt das Gehirn Informationen über die Distanzen von Objekten in der Umgebung.

2.2.2 Visuelle Tiefenhinweise

Im Unterschied zu den okulomotorischen Hinweisen wird nicht die Spannung der Augenmuskulatur einbezogen, sondern rein sichtbare Eigenschaften des beobachteten Objekts. Die visuellen Hinweise können in zwei Bereiche eingeteilt werden: monokulare und binokulare Tiefenhinweise.

Monokulare Tiefenhinweise

Mithilfe dieser Hinweise kann die Tiefe von Objekten wahrgenommen werden, sogar wenn diese nur mit einem Auge angeschaut werden. Diese nennt man die monokularen Tiefenhinweise. Diese Faktoren machen es möglich, in zweidimensionalen Bildern Tiefeneindrücke zu erzeugen. Diese Tiefenhinweise sind erfahrungsbezogen und der Beobachter lernt im Laufe der Zeit die physikalische Bedeutung der Objekte, die er sieht und die Beziehung zur anderen Objekten in der 3D-Umgebung [Holo5]. Es gibt statische und dynamische monokulare Tiefenhinweise. Die statischen Hinweise sind folgende:

- **Größenverhältnisse:**
Falls zwei verschiedene Gegenstände gleich groß erscheinen, obwohl der Erfahrung nach einer größer als der andere ist, kann man schlussfolgern, dass der größere Gegenstand weiter hinten ist (Vgl. Abbildung 2.2).
- **Verdeckung:**
Falls ein Gegenstand gesehen wird, der unvollständig durch einen anderen verdeckt ist, kann gefolgert werden, dass der verdeckte Gegenstand sich hinter dem anderen befindet (Vgl. Abbildung 2.2).

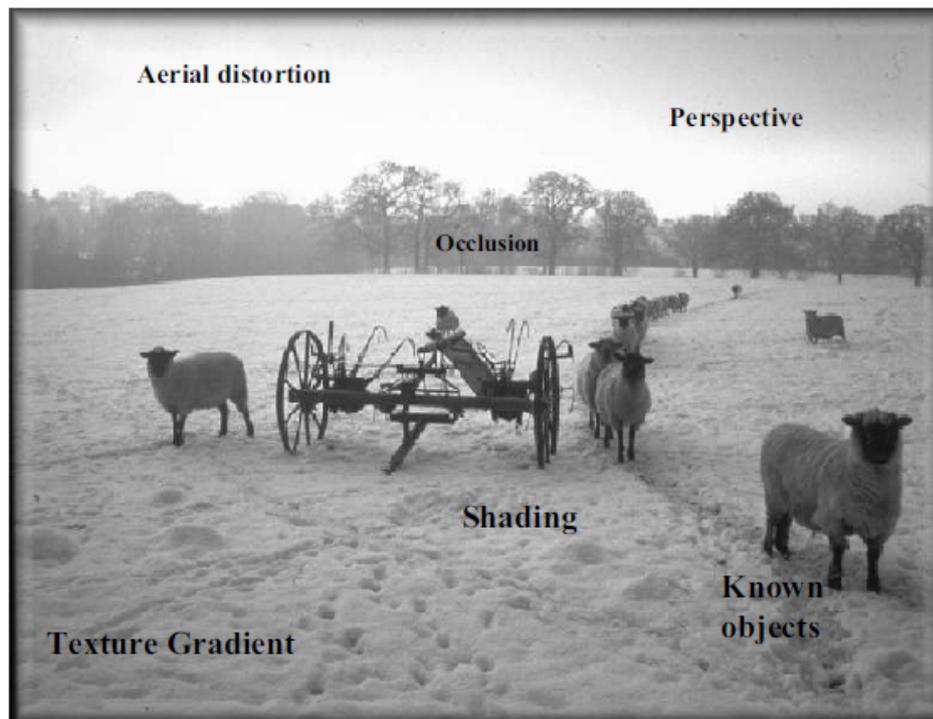


Abbildung 2.2: Monokulare Tiefenhinweise (Bildautor: David Burden) [Hol05]

- **Licht und Schatten:**
Mithilfe der Richtung des Lichts und der Platzierung der Schatten kann die Tiefe des Objekts erkannt werden.
- **Lineare Perspektive:**
Falls einer von zwei identischen Gegenständen größer wahrgenommen wird, ist dieser näher als anderer.
- **Texturgradient:**
Nähere Objekte können detailreiche wahrgenommen werden, als distanzierte Objekte (Vgl. Abbildung 2.2).
- **Atmosphärische Perspektive:**
Durch Staubpartikel in der Luft oder bei Regen kann das Licht durch die Atmosphäre beeinflussen werden [Hol05]. Dadurch werden entfernte Objekte verschwommen gesehen (Vgl. Abbildung 2.2).

Die dynamischen Kriterien werden im Folgenden vorgestellt:

- **Bewegungsparallaxe:**
Dieses Phänomen tritt bei sich bewegenden Objekten in der Szene oder bei einem sich bewegenden Beobachter auf [Hol05]. In Abhängigkeit des Abstands werden Objekte

unterschiedlich schnell wahrgenommen. Dies passiert beispielsweise wenn ein Beobachter von einem Fenster in seinem Haus aus ein vorbeifliegendes Flugzeug beobachtet. Trotz einer hohen Geschwindigkeit scheint es, als ob das Flugzeug langsam fliegt. Ein sich nahe befindliche Bus, der auf der Straße vor seinem fährt, wird trotz niedrigerer Geschwindigkeit, als das Flugzeug, vom Beobachter als sich schnelle bewegend wahrgenommen.

- **Dynamische Verdeckung:**
Bewegt man seinem Kopf, werden zwei Objekte, die sich in verschiedenen Abständen zum Beobachter befinden, so wahrgenommen, als ob sie sich relativ zueinander bewegen.

Binokulare Tiefenhinweise

Bei den binokularen Tiefenhinweisen kann die Tiefeninformation nur mit beiden Augen erkannt werden. Außer der Konvergenz und der Akkommodation, die die okulomotorische Kriterien darstellen, gibt es noch einen binokularen visuellen Tiefenhinweis - die binokulare Disparität. Die binokulare Disparität beschreibt der Unterschied zwischen den Bildern, die auf der Netzhaut der beiden Augen abgebildet sind [McAo2]. Der Abstand zwischen beiden Augen ermöglicht es, Bilder aus zwei gering unterschiedlichen Blickwinkeln aufzunehmen. Die Punkte, die vor oder hinter dem Fixationspunkt in der 3D-Umgebung liegen, werden auf verschiedenen Positionen auf die linke und rechte Netzhaut projiziert. Dieser Vorgang erzeugt die binokulare Disparität, welche das Gehirn des Beobachters mit stereoskopischen Tiefenhinweisen versorgt [Holo5].

2.3 Stereoskopische Displaytechnologie

Seit vielen Jahren wollen Menschen das Erlebnis des räumlichen Sehens auch in Bildern und Filmen genießen. Dafür wurden viele Technologien entwickelt, die die Imitation von dreidimensionalen Szenen ermöglichen. Bei Verwendung dieser Technologie werden zwei leicht unterschiedliche Bilder für linkes und rechtes Auge erzeugt und zum jeweiligen Auge geschickt. Das Gehirn nimmt die Tiefeninformationen von den Bildern, die durch die verschiedenen Tiefenhinweise entstanden sind und interpretiert anhand dieser Informationen die 3D-Objekte in einer 3D-Umgebung.

Das der Zuschauer mit dem linken und rechten Auge jeweils das für das entsprechende Auge vorgesehene Bild sieht, wird bei der Verwendung von stereoskopischen Displays eine spezielle Brille getragen. Es gibt auch autostereoskopische Displays, bei denen die Wahrnehmung von räumlichen Objekten ohne Brillen möglich ist. Auf 3D-Displays werden für das linke Auge die Bilder, welche für das rechte Auge bestimmt sind, ausgeblendet und umgekehrt. Um die Bilder für das linke und rechte Auge passend zu zuordnen, werden verschiedene Technologien für die stereoskopische Displays verwendet. Im Folgenden wird auf diese Technologien eingegangen.

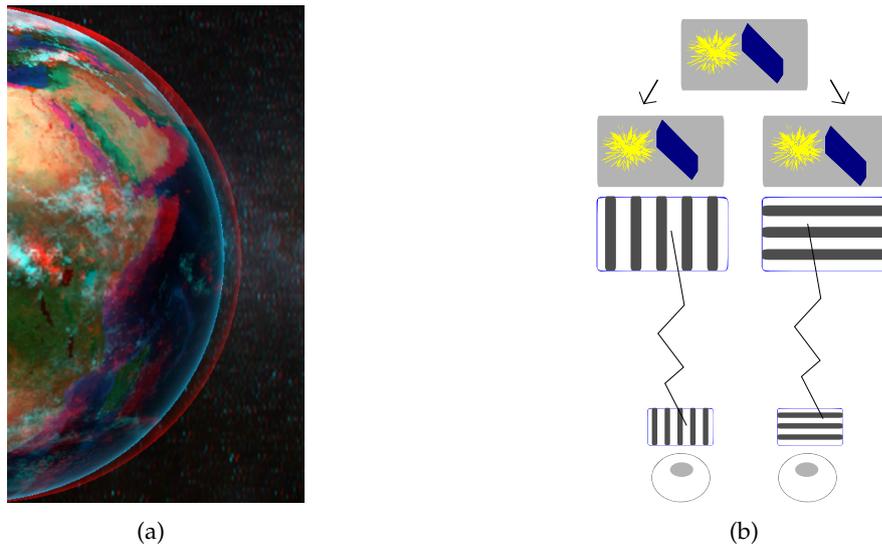


Abbildung 2.3: Anaglyphen- und Polarisierungstechnik. (a) Ein Beispiel für die Anaglyphen-Technik. Das Bild für das rechte und linke Auge wird überlappend, in Komplementärfarben dargestellt. (b) Das Konzept der linearen Polarisierungstechnik. Die senkrecht (waagrecht) polarisierten Lichtwellen werden nur durch die senkrecht (waagrecht) polarisierten Filtern gelassen.

2.3.1 Farbe-Multiplex Ansatz

Ein Beispiel für dieses Verfahren ist die Anaglyphen-Technik. Bei dieser Technik werden die Bilder für das linke und rechte Auge auf einem Display überlappend, in Komplementärfarben angezeigt. Dafür wird rote Farbe für die Bilder des linken Auges und grüne oder cyan Farbe für die Bilder des rechten Auges benutzt. Ein Beispiel dafür wird in Abbildung 2.3(a) dargestellt. Zuschauer müssen spezielle Brillen tragen, um die Bilder herauszufiltern. Diese Brillen haben entsprechende Farbfilter, wie die Bilder auf dem Display. Wegen der komplementären Farben werden die Bilder nur für das dazugehörige Auge sichtbar sein. Vorteile dieser Technik sind die einfache Handhabung und der Preis. Die großen Nachteile sind der Verlust der Farbinformationen und ein höheres Ausmaß von Bildüberschneidung [UCES11].

Ein andere Technik, die diesen Ansatz nutzt ist ColorCode 3D. Im Unterschied zu der Anaglyphen-Technik geht die Farbinformationen hier nicht verloren. In dieser Technik werden Brillen verwendet, die bernsteinfarbene und blaue Gläser haben. Mit dem bernsteinfarbenen Filter werden die Farbinformationen und mit dem blauen Filter Tiefeninformationen empfangen. Auf den Displays, die diese Technik verwendet kann "Ghosting" auftreten. Trotz dieses Nachteils findet ColorCode 3D in Filmen, Spielen und mobilen Inhalten Anwendung [UCES11].

2.3.2 Polarisation-Multiplex Ansatz

Eine andere Technik, die in stereoskopischen Displays verwendet wird, ist die Polarisations-technik. Bei dieser Methode werden Bilder auf einer speziellen Leinwand abgebildet. Dies geschieht durch zwei Projektoren. Beide Projektoren haben unterschiedliche Polarisationsfilter. Die Leinwand muss so gewählt werden, dass sie das darauf projizierte polarisierte Licht reflektiert. Durch die Benutzung von speziellen Brillen können die Zuschauer die Tiefeninformationen wahrnehmen. Diese Brillen verfügen über die gleichen Filtern, mit denen auch die Projektoren ausgestattet sind.

Beim Polarisation-Multiplex Ansatz wird lineare und zirkulare Polarisation benutzt. Senkrecht und waagrecht orientierten Polarisationsfiltern werden bei der lineare Polarisation verwendet, um Bilder für das linke und rechte Auge passend zum jeweiligen Auge zuzuordnen. In Abbildung 2.3(b) ist das vereinfachte Konzept der linearen Polarisation dargestellt. Im Unterschied zu der linearen Polarisation werden bei der zirkularen Polarisation im Uhrzeigersinn und entgegen dem Uhrzeigersinn orientierten Polarisationsfilter verwendet.

Im Gegensatz zur Anaglyphenmethode geht die Farbinformationen beim Polarisation-Multiplex Ansatz nicht verloren. Aber es kann bei diesem Verfahren auch zu Bildüberschneidungen kommen. Ein anderes Hindernis stellt die Kopfbewegung des Betrachters dar. Dann werden die Bildern nicht mehr ordnungsgemäß wahrgenommen. Dieses Problem tritt weniger auf, falls die zirkulare Polarisation verwendet wird.

2.3.3 Zeit-Multiplex Ansatz

Beim Zeit-Multiplex Ansatz werden Bilder für das rechte und linke Auge mit einer Shutterbrille getrennt, die mit zwei Flüssigkristall-Gläser ausgestattet ist. Diese Brillengläser können elektronisch sehr schnell ein- und ausgeschaltet werden. Im Gegensatz zu den obengenannten Methoden werden hier Bilder für rechte und linke Augen nicht gleichzeitig sondern nacheinander übertragen. Das Ein- und Ausschalten der Brillengläser muss für die Übertragung entsprechend synchronisiert sein. So wird beispielsweise bei der Übertragung eines Bildes, welches für das rechte Auge bestimmt ist, das linke Brillenglas undurchsichtig und das rechte Brillenglas transparent eingestellt. Analog geschieht dies auch für das linke Auge.

Der Wechsel der Bilder geschieht so schnell, dass das menschliche Auge diesen nicht erkennt. Dafür werden die Bilder für das linke und rechte Auge auf einem Display mit sehr hoher Bildwiederholrate angezeigt. Der Nachteil von der Shutter-Technik sind teure Brillen.

2.4 Eyetracking

Eyetracking kann man als die Verfolgung der Augenbewegungen definieren. Anders gesagt, Eyetracking ist die Ermittlung von Objekten oder Koordinaten, die ein Beobachter fokussiert.

Dafür werden Eyetracker benutzt. Die Eyetracker verwenden verschiedene Methoden, um Augenbewegungen zu verfolgen und somit festzustellen wohin ein Beobachter schaut.

Falls ein Beobachter Interesse an bestimmten Objekten hat, wird er auf diese Objekte seine Aufmerksamkeit richten. Mit der Verfolgung der Augenbewegungen ist es möglich, zu erkennen, wann und welche Objekte einen Beobachter interessieren. Außerdem findet Eye-tracking Anwendungen in vielen Bereichen, wie der Mensch-Computer-Interaktion, der Neurowissenschaft, der Psychologie und der Werbung. Eyetracking kann entweder als Interaktionstechnik oder für Analysezwecke verwendet werden. Wie dies geschehen kann wird später in Kapitel 3.2 beschrieben. Um Eyetracking genauer zu betrachten wird in dem nächsten Unterabschnitt eines der wichtigsten Prinzipien für das Eyetracking, die Augenbewegungen diskutiert. Im Folgenden wird die Eyetracking-Technologie kurz vorgestellt.

2.4.1 Augenbewegung

Bilder können nur dann scharf gesehen werden, falls das von ihnen ausgestrahlte Licht in der Fovea zusammen fällt. Dafür bewegen Beobachter ständig ihre Augen, dass Bilder genau in die Fovea abgebildet lassen. Die Augenbewegungen können in fünf Bereiche klassifiziert werden: Sakkaden, Folgebewegungen (smooth pursuit), Vergenzbewegungen, vestibulärer Nystagmus und optokinetischer Nystagmus [Duc07].

Falls man beispielsweise ein Objekt in der Umgebung sucht, bewegt man seine Augen von Objekt zu Objekt. Solange das gewünschte Objekt nicht gefunden wurde, wird diese Bewegung wiederholt. Diese "zügige" Augenbewegung nennt man Sakkade. Die Sakkaden sind sehr schnelle und ballistische Augenbewegungen, die unwillkürlich bei vestibulärem und optokinetischem Nystagmus und willkürlich bei der Beobachtung der Objekten in der Umgebung stattfinden können [Duc07]. Ballistisch bedeutet, dass diese Bewegungen vordefiniert sind und das Ziel und der Bewegungspfad nicht geändert werden können [Jac95]. Nach jeder Sakkade geschieht normalerweise eine Fixation. In dem obengenannten Beispiel entspricht die Fixation der Fokussierung auf ein Objekt. Trotz der ganz kleinen Bewegungen der Augen, wie Drift, Mikrosakkaden und Tremor (ein sich wiederholendes Augenzittern), während der Fixation, können die Objekte gut beobachtet werden.

Falls sich ein Objekt im Sichtfeld eines Beobachters langsam bewegt und der Beobachter dieses Objekt anschauen will, wird er mit beiden Augen, mit einer Geschwindigkeit relativ zu der Geschwindigkeit des Objekts, diesem folgen. Sobald das Objekt stoppt, stoppt auch diese Bewegung. Diese Bewegung heißt Folgebewegung (smooth pursuit). Im Unterschied zu Sakkaden sind diese Bewegungen glatt.

Bei den Vergenzbewegungen halten beiden Augen ein Objekt im Fokus, um die Tiefe dieses Objekts in der 3D-Umgebung zu ermitteln (siehe Unterabschnitt 2.2.1 auf Seite 14). In Abhängigkeit der Tiefe der Objekte können die Augen des Beobachters sich einander annähern oder zueinander stehen.

Ein andere wichtige Augenbewegung ist der Nystagmus. Diese Bewegung kann während der Kopfbewegung des Beobachters oder der Beobachtung eines sich bewegendes Objekts

stattfinden [Jac95]. Bei dem Nystagmus werden Augen Folgebewegung und Sakkaden in entgegengesetzter Richtung hintereinander durchführen. Im Gegensatz zum optokinetischen Nystagmus, in welchem Stimulus für Augenbewegung das Kompensieren der retinalen Verschiebung des Bilds ist, wird der vestibulärer Nystagmus durchgeführt, um Kopfbewegungen zu kompensieren [Duco7].

2.4.2 Eyetracking Technologie

Es gibt zahlreiche Technologien, die bei der Verfolgung der Augenbewegungen genutzt werden können. In einigen Eyetracking-Technologien werden Geräte verwendet, die Augen oder die Haut um die Augen herum berühren. Außerdem kann die Technologie für Bestimmung der Augenbewegungen noch anders klassifiziert werden: die Technologie, welche die Messung von Position der Augen relativ zu Kopf durchführt und die Technologie, welche die Orientierung der Augen im Raum bestimmt [Duco7, YS75]. Um Augenbewegungen zu bestimmen, werden verschiedene Ereignisse verfolgt, die bei diesen Bewegungen geschehen.

Zwischen der Hornhaut und der Netzhaut der Augen gibt es eine Potentialdifferenz bis zu 1 mV, die sich während der Rotation der Augen durch die Rotation des elektrostatischen Dipols ändert [YS75]. Auf Grundlage dieser Eigenschaft können Elektroden rund um die Augen platziert werden, um die Potentialdifferenz und damit die Augenbewegungen relativ zur Kopfposition zu messen. Diese Eyetracking-Methode heißt Elektro-Okulographie. Der große Nachteil dieser Methode ist die fehlende Genauigkeit [Jac95].

Im Unterschied zu der Elektro-Okulographie hat die Technik mit skleralen Kontaktlinsen eine höhere Genauigkeit. Wie der Name schon sagt, soll ein Proband beim Einsatz dieser Technik spezielle Kontaktlinsen tragen. Dafür gibt es zwei Verfahren. Bei dem optischen Verfahren werden Kontaktlinsen mit integrierten Spiegeln getragen. Sobald ein Proband seine Augen bewegt, wird das Licht, das durch die Kontaktlinsen einfällt, seinen Ausfallwinkel ändern. Durch das Aufzeichnen dieses Vorgangs können die Augenbewegungen verfolgt werden. Bei dem mechanischen Verfahren, werden anders als beim optischen Verfahren, Kontaktlinsen mit magnetischen Spulen verwendet. Diese Spulen bekommen eine Induktionsspannung von einem externen Magnetfeld, das während der Messung um den Kopf des Probanden herum erzeugt wird. Die Induktionsspannung in jeder Spule ändert sich mit dem Sinus des Winkels des Auges relativ zum Magnetfeld [YS75]. Durch die Messung dieser Spannung können die Augenbewegungen verfolgt werden. Trotz höherer Genauigkeit ist es sinnvoll diese Technik nur bei Laborversuchen zu benutzen, da das Tragen dieser Kontaktlinsen unbequem und gefährlich für die Augen ist [Jac95].

Trotz der komplexen Augenstruktur können einige Merkmale des Auges leicht verfolgt werden. Anhand dieser Merkmale können Augenbewegungen erfasst werden. Ein Beispiel dafür ist die Technik zur Verfolgung der Pupille oder des Limbus, die Grenze zwischen der Lederhaut und der Iris, bei einer Beleuchtung. Beispielsweise wird bei der Verfolgung des Limbus reflektiertes Licht gemessen, um die Augenposition festzustellen. Die Beleuchtung wird mit Infrarotlicht realisiert [YS75].

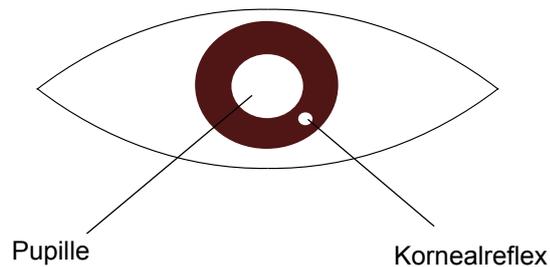


Abbildung 2.4: Helle Pupille und Kornealreflex. Mit Benutzung dieser beiden Daten kann videobasierter Eyetracker einige Berechnungen durchführen und die Blickposition eines Benutzers herausfinden.

Die modernen Eyetracker nutzen meist videobasierte Technologien, um die Bestimmung der Blickrichtung der Augen im Raum zu ermöglichen. Diese Eyetracker sind mit Videokameras und Lichtquellen ausgestattet. Bei dieser Technik werden aufgezeichnete Bilder der Augen mithilfe einer Bildverarbeitungssoftware untersucht und die Augenbewegungen ermittelt. Um die Blickrichtung zu bestimmen und kleine Kopfbewegungen auszugleichen, verfolgen videobasierte Eyetracker die Pupille und den Kornealreflex. Der Kornealreflex ist die Spiegelung des ins Auge einfallenden Lichts, das in der Außenfläche von der Hornhaut reflektiert wird (siehe Abbildung 2.4). Um den Benutzer nicht zu stören, sind diese Eyetracker mit infraroten Lichtquellen ausgestattet. Die Pupille kann mit verschiedenen Bilderkennungsalgorithmen erkannt werden, die die Eigenschaften der Pupillen in unterschiedlichen Beleuchtungen verwenden. Bei der Beleuchtung der Augen durch eine Infrarotlichtquelle, die sich neben der Videokamera befindetet, werden die Pupillen schwarz dargestellt und können nun leicht von der Iris unterschieden werden. Der Grund dafür ist die Absorption des Infrarotlichts durch die Pupillen. Außerdem nutzen einige Bilderkennungsanwendungen das Merkmal, dass die Pupille in einem Bild ganz hell erscheint. Dies kann durch Infrarotlicht realisiert werden, welches seinen Ursprung auf der gleichen Achse wie die Videokamera hat [YS75]. Da sich die Pupille und der Kornealreflex bei der Augenbewegungen unterschiedlich bewegen, kann der Mittelpunkt von Beiden gefunden werden und mithilfe dem Zusammenhang zwischen diesen zwei Punkten kann die Blickrichtung berechnet werden [Jac95].

Die meisten videobasierten Eyetracker müssen vor der Benutzung kalibriert werden [Duc07]. Bei der Kalibrierung soll ein Benutzer Punkte, die auf dem Display an verschiedenen Positionen erscheinen, fixieren. Bei jeder Fixation diese Punkte werden Pupillenposition und Kornealreflex durch den Eyetracker aufgezeichnet. Mithilfe dieser Daten werden Einstellungen vom Eyetracker für den jeweiligen Benutzer abgestimmt.

3 Verwandte Arbeiten

Dieses Kapitel befasst sich mit verschiedenen Arbeiten aus dem Themenfeld der Interaktion mit 3D-Displays und der blickbasierten Interaktion. Das Kapitel ist wie folgt aufgebaut: Der Abschnitt 3.1 fasst einige Arbeiten über die Interaktion mit 3D-Bildschirmen und damit einhergehenden Probleme zusammen. In Abschnitt 3.2 werden einige bisherige Arbeiten präsentiert, die sich mit Eyetracking beschäftigen. Der Unterabschnitt 3.2.1 befasst sich mit Arbeiten, in denen Eyetracking als Interaktionstechnik genutzt wird. Zum Schluss wird in Abschnitt 3.2.2 eine kurze Beschreibung gegeben, wie Eyetracking bei der Usability-Evaluierung verwendet werden kann.

3.1 Interaktion mit 3D-Displays

Für die Interaktion mit 3D-Bildschirmen werden in den verschiedenen Arbeiten unterschiedliche Interaktionstechniken verwendet. Die traditionellen Eingabegeräte, wie Maus oder Tastatur sind aufgrund der wenigen Freiheitsgrade für komplexe Aufgaben auf 3D-Displays nicht geeignet. In den letzten Jahren wurden auf dem Markt viele 2D-Displays vorgestellt, welche eine Steuerung durch Berührung unterstützen. In einigen Arbeiten wurde diese Interaktionstechnik für stereoskopische Bildschirme ebenfalls präsentiert. Hacket et al. haben in einer Arbeit direkte Multi-Touch-Interaktion mit 3D-Inhalt vorgestellt [HBCR11]. Der Prototyp besteht aus einem auf dem kopfstehenden stereoskopischen Bildschirm, einem halb-durchsichtigen Spiegel, welche unter dem Bildschirm angebracht ist und einem Touchscreen, der unter dem Spiegel liegt. Der durch den stereoskopischen Bildschirm erzeugte 3D-Inhalt wird auf dem Spiegel reflektiert. Head-Tracking wird benutzt, dass der erzeugte 3D-Inhalt für den Benutzer richtig angezeigt wird. Für die Interaktion legt ein Benutzer seine Hände auf den Touchscreen und steuert den stereoskopischen Inhalt, der über seinen Händen auf dem Spiegel dargestellt wird, durch die Benutzung 2D-Metaphern und Widgets. 2D-Metaphern und Widgets werden auf dem Touchscreen angezeigt. Die Evaluierung des Systems hat gezeigt, dass 3D-Inhalt und 2D-Widgets gleichzeitig angezeigt keine Probleme bei der Wahrnehmung machen.

Interaktion mit Berührung kann auf dem stereoskopischen Bildschirm zu unterschiedlichen Problemen führen. Damit eine solche Interaktion natürlich und intuitiv erscheint, braucht ein Benutzer ein haptisches Feedback. Die Darstellung der Objekte in verschiedenen Parallaxen macht es schwer einen Benutzer ein haptisches Feedback zu geben. Bei einer positiven Parallaxe wirkt es, als wären die Objekte hinten dem stereoskopischen Bildschirm, wodurch eine direkte Interaktion nicht möglich ist. Bei einer Berührung eines Objekts in positiver Parallaxe erhält der Benutzer ein haptisches Feedback, ohne dieses Objekt zu wirklich zu

erreichen. Objekte in negativer Parallaxe werden vor dem stereoskopischen Display positioniert wahrgenommen. Für ein haptisches Feedback muss der Benutzer den Bildschirm durch das Objekt hindurch berühren. Valkov et al. haben einen Ansatz vorgeschlagen, um dieses Probleme zu lösen [VGH12]: Bei der Berührung eines stereoskopischen Displays soll das Objekt auf die Bildschirmoberfläche so verschoben werden, dass seine Bewegung dem Benutzer nicht auffällt. Außerdem soll das 3D-Objekt während dieser Bewegung in Abhängigkeit zur Parallaxe seine Größe ändern, damit die Verschiebung nicht wahrnehmbar werden kann. Eine Evaluierung hat die Korrektheit dieses Ansatzes bewiesen. Unter Verwendung dieses Ansatzes kann die Touch-Interaktion auf einem stereoskopischen Display ohne extra Hardware realisiert werden.

Bruder et al. haben zwei unterschiedliche Touch-Interaktionsarten auf einer stereoskopischen Tabletop-Oberfläche verglichen [BSS13]: die 2D-Touch-Interaktion, bei der ein Benutzer durch ein Objekt, in negativer Parallaxe, hindurch die Oberfläche berührt und die dreidimensionale Interaktion, die frei über dem Raum des Gerätes stattfand. Es wurde festgestellt, dass die Interaktion mit der 2D-Berührung für Objekte in negativer Parallaxe bis zu 10 cm über der Tabletop-Oberfläche besser geeignet ist. Für Objekte, die noch höher wahrgenommen werden, ist die 3D-Touch-Interaktion zu bevorzugt.

Stereoskopische Bildschirme in einigen Arbeiten für die gemeinsame Benutzung verwendet. Özacar et al. haben für ein stereoskopisches Tabletop-Display, das die zeitgleiche, gemeinsame Benutzung unterstützt, eine Interaktionstechnik untersucht, die die direkte Manipulation von virtuellen Objekten ermöglicht [OTK13]. Die Manipulation wird durch Finger-basierten Gesten realisiert - in der Luft, ganz ohne Berührung. Die Änderungen, die durch einen Benutzer erfolgen, sind sofort für andere Benutzer erkennbar. Für die Interaktion wird optisches Finger-Tracking verwendet. Die Evaluierung dieser Interaktionstechnik hat gezeigt, dass direkte Manipulation der virtuellen Objekte auf dem gemeinsam nutzbaren stereoskopischen Display eine bessere User Experience bietet.

3.2 Verwandte Arbeiten im Bereich Eyetracking

Durch die Verfolgung der Augenbewegungen können einige Interaktionsmethoden entworfen werden, die neue Möglichkeiten der Interaktion zwischen Mensch und Computer bieten. Diese Interaktionsmethoden sind schnell und einfach zu benutzen [Jac95]. Außerdem ermöglichen solche Interaktionstechniken eine Steuerung der Geräte ohne physikalischen Kontakt.

Durch die blickbasierte Interaktion entwickeln sich neue Benutzerschnittstellen, die einige Unterschiede zu traditionellen Benutzerschnittstellen haben. Beispielsweise gibt ein Benutzer eines Systems basierend auf WIMP ¹ in einer bestimmten Zeit einen Befehl ein und der Rechner führt diesen aus. Interagieren Menschen mit Maschinen, bewegen sie dementsprechend ihre Augen, um eine spezielle Aufgabe zu lösen. Somit sind diese Augenbewegungen immer

¹WIMP steht für "Windows", "Icons", "Menus" und "Pointing Device".

vorhanden, wenn ein Mensch eine Maschine bedient [JKo3]. Somit muss ein Rechner nicht auf einen Befehl vom Benutzer warten, sondern kann den Blick des Benutzers analysieren, um zur richtigen Zeit den richtigen Befehl auszuführen. Jakob Nielsen nennt solche Benutzerschnittstellen "Nicht-Befehl(Noncommand) Benutzerschnittstellen" [Nie93]. Die virtuelle Realität ist ein anderes Beispiel für solche Schnittstellen, welche ein weiterer Bestandteil dieser Literatur sind.

Da die Augen bei der blickbasierten Interaktion sowohl als Ausgabe, als auch als Eingabe dienen, müssen die Augenbewegungen dahingehend unterschieden werden, ob Befehle auszuführen sind oder ob lediglich die Ausgabe angeschaut wird. Falls ein Benutzer beispielsweise einen Button anschaut, soll der Rechner fähig sein zu entscheiden, ob der Benutzer den Button anklicken will oder nur darauf blickt. Das ist ein Problem der blickbasierten Interaktion und wird "Midas Touch" Problem genannt [Jac95]. Die Applikationen, die mit dem Blick des Benutzers kontrolliert werden, müssen so entworfen sein, dass das Auftreten dieses Problems minimiert wird. Für die Lösung des "Midas Touch" Problems kann beispielsweise eine Verweilzeit, die Fixation auf einem Objekt eine bestimmte Zeit lang, um die dafür geeignete Aktion auszuführen, oder eine zusätzliche Eingabemöglichkeit - beispielsweise ein Mausklick - um eine Aktion für das durch den Blick fixierte Objekt auszuführen [ADSo5].

Wie oben bereits erwähnt, kann Eyetracking als Eingabemöglichkeit verwendet werden, um Applikationen zu steuern. Außerdem werden Eyetracker bei der Bewertung der Usability von Applikationen angewendet. Während der Benutzung der Applikation kann Blick des Benutzers verfolgt werden und damit kann ermittelt werden, wie gut die benutzte Schnittstelle für die Interaktion zwischen Mensch und Computer ist. Der folgende Unterabschnitt stellt einige Arbeiten vor, bei denen Eyetracker als Interaktionstechnik verwendet wurden. Die Bewertung der Usability mithilfe Eyetracking wird in dem letzten Unterabschnitt des Kapitels kurz ausgeführt.

3.2.1 Eyetracking als Interaktionstechnik

Die Augenbewegungen sind schnell und natürlich. Diese Vorteile machen es möglich Eyetracking als Eingabemöglichkeit zu nutzen. Robert J. K. Jacob and Keith S. Karn haben in [JKo3] jedoch einige Gründe angegeben, warum Eyetracking das traditionelle Eingabegerät - die Maus nicht komplett ersetzen kann. Mithilfe der Maus kann genau auf einem Punkt auf dem Bildschirm gezeigt werden. Beim Eyetracking ist das meist nicht möglich. Die ständigen Augenbewegungen und fehlende Genauigkeit von Eyetrackern erschweren dies. Zusätzlich ist das "Midas Touch" Problem in den meisten Fällen unvermeidbar. Mit einer Maus kann genau dann auf ein Objekt auf dem Display gezeigt werden, wenn sie darauf bewegt wird. Da die Augen ständig bewegt werden, zeigt der Blick des Benutzers immer auf neue Objekte. Anders als bei der Maus, wo Befehle durch die Tastendrucke ausgeführt werden, muss bei Befehlen, welche durch den Blick des Benutzers ausgeführt werden sollen anders vorgegangen werden. Das Blinzeln ist dafür keine Lösung, denn es ist ein natürlicher Reflex und passiert willkürlich. In einigen Projekten wird dafür eine Verweilzeit verwendet [Jac95]. In diesem Fall soll ein Benutzer eine bestimmte Zeit ein Objekt anschauen, damit auf dieses Objekt geklickt wird.

In verwandten Arbeiten wird Eyetracking meistens als Zeigemöglichkeit verwendet. Eine davon ist MAGIC (Mouse And Gaze Input Cascaded Pointing) [ZMI99]. Um zu "Zeigen" verwendet MAGIC sowohl Blickbewegungen, als auch die Maus. Durch die Verfolgung der Blickbewegungen wird ein Cursor grob auf ein Objekt gesetzt, das der Benutzer beobachtet. Danach kann er die Maus bewegen und mit dem Cursor ganz genau auf das Objekt zeigen. Der letzte Schritt, die manuelle Änderung des Cursors, bringt mehr Genauigkeit. Jedoch zählt dies dann nicht mehr zur freihändigen Interaktion.

Salvucci und Anderson haben im Jahr 2000 eine neue Schnittstelle, die IGO (Intelligent Gaze-added Operating System) Schnittstelle, vorgestellt [SA00]. Durch diese Schnittstelle wird die ganze Interaktion durch Blickbewegungen und einem Knopf, der analog zur linken Maustaste ist, durchgeführt. Beispielsweise soll ein Benutzer um einen Ordner zu öffnen auf den Ordner schauen und zwei mal der Knopf drücken. Das System benutzt einen intelligenten probabilistischen Algorithmus, um die Genauigkeit zu erhöhen. Durch die Benutzung dieses Algorithmus berechnet das System die Wahrscheinlichkeit, mit welcher ein Benutzer eine bestimmte Aktion ausführen möchte und entscheidet auf dieser Wahrscheinlichkeit basierend welche Aktion ausgeführt wird.

Lankford hat ein System vorgestellt, das die Verweilzeit verwendet, um verschiedene Aktivitäten mit dem Blick durchzuführen, die die der Maus ähneln [Lan00]. Ein Benutzer kann mit verschiedenen Verweilzeiten ein Objekt auf dem Display fixieren, um dieses auszuwählen, zu ziehen oder ein oder zwei mal darauf zu klicken. Während der Benutzung bekommt ein Benutzer ein visuelles Feedback, damit er weiß welche Verweilzeit aktuell ist. Weiterhin, um die Genauigkeit der Blicksteuerung zu verbessern, verfügt das System noch über eine Zoomfunktion. In diesem Modus wird der nach der ersten Verweilzeit fixierte Bereich vergrößert. In der zweiten Verweilzeit wird das Zielobjekt ausgewählt und ein extra Bildschirmmenü geöffnet, wo sechs Mausbefehle gezeigt werden. In der letzten Verweilzeit wird eine der Befehle ausgewählt und ausgeführt.

Um die Genauigkeit von Eyetracking zu erhöhen, haben Miniotas, Špakov und MacKenzie eine Technik verwendet, in der Zielobjekte statisch erweitert werden [MvM04]. Diese Erweiterungen sind für den Benutzer nicht sichtbar. Wegen der fehlenden Genauigkeit kann der Eyetracker nicht die Koordinaten des Objekts, auf das ein Benutzer schaut genau ermitteln. Um ein Objekt trotzdem manipulieren zu können, können auch Koordinaten, die leicht daneben liegen, verwendet werden. Falls die Blickkoordinaten im Umkreis des Zielobjekts liegen, wird immer noch der Befehl für das Zielobjekt ausgeführt, als wenn der Benutzer genau auf die exakten Koordinaten des Objekts fixiert hat. Die dynamische Erweiterung von Zielobjekten wurde durch Miniotas und Špakov evaluiert [vM05]. Im Unterschied zu statischen, ist die dynamische Erweiterung für den Benutzer sichtbar. Die Evaluierung hat gezeigt, dass mit der Verwendung dieser Technik die Genauigkeit der Selektion von Objekten zwar erhöht, aber die Ausführungsgeschwindigkeit etwas reduziert wird.

Ashmore et al. haben die Fisheye-Linsen-Technik verwendet, um ein durch ein Benutzer fixierten Bereich zu vergrößern und damit die Eyetracking-basierte Zielauswahl zu vereinfachen [ADS05]. Sie haben in einer Benutzerstudie vier Techniken verglichen: Interaktion ohne Fisheye-Linsen-Technik, mit einer immer aktiven Fisheye-Linsen-Technik und eine nur bei der Fixation auf einem Zielobjekt aktive Fisheye-Linsen-Technik. Bei der vierten Technik

ist die Fisheye-Linse erst nach der Fixation sichtbar. Nach der Aktivierung ist die Linse unbeweglich und ein Benutzer kann innerhalb der Linse Objekte anvisieren. Bei der vierten Technik und der Technik ohne Fisheye-Linse ist die Zielauswahl schwerer, als bei der anderen zwei Techniken gewesen. Die Studie hat gezeigt, dass die Interaktion durch die Fisheye-Linse, die während der Suche unsichtbar war und nur bei der Fixation aktiviert wurde, in den Punkten Geschwindigkeit und Genauigkeit besser, als eine immer sichtbare Fisheye-Linse oder die Interaktion ohne Fisheye-Linsen-Technik, war.

Kumar et al. haben EyePoint vorgestellt [KPW07]. Bei EyePoint wird ein durch einen Benutzer betrachteter Bereich vergrößert und extra Knöpfe benutzt. Mit diesen Knöpfen kann man verschiedene Mausaktivitäten, wie einmal oder zweimal die linke Maustaste klicken, die rechte Maustaste klicken oder Objekte ziehen, ausführen. In diesem System basiert die ganze Interaktion auf vier Schritten. Ein Benutzer schaut auf einem Zielobjekt und drückt einen Knopf für die gewünschte Aktion. Der fixierte Bereich wird vergrößert. Nun schaut der Benutzer ganz genau auf das gewünschte Objekt im vergrößerten Bereich und gibt den gedrückten Knopf frei, um eine Aktivität auszuführen. Es wurde gezeigt, dass die Leistung von EyePoint in einigen Fällen besser als die Leistung von der Maus war.

Blickbewegungen werden auch verwendet, um Eingabe zu tätigen. Isokoski hat eine Technik vorgeschlagen, die für die Texteingabe durch die Benutzung der Blickgesten realisiert wurde [Iso00]. Mit einem Eyetracker können die Augendaten auch erfasst werden, wenn ein Benutzer Punkte außerhalb des Displays fokussiert. Isokoski nutzt Koordinaten, welche außerhalb der Bildfläche liegen, um Eingaben zu machen. Hierfür wurden Bereiche außerhalb des Displays definiert, welche durch die Fixation die Eingabe verschiedenster Zeichen ermöglichen.

Die bei EyePassword vorgestellte Technik verwendet Eyetracking für die Passwordeingabe, um "Shoulder surfing" zu vermeiden [KGBW07]. Mit diesem System kann ein Benutzer ein Passwort durch die Fixation von Zeichen auf einem virtuellen QWERTY-Layout oder einem alphabetisch geordnetem Layout eingeben. Wegen der Sicherheit bekommt ein Benutzer während der Eingabe kein visuelles Feedback. Bei der Evaluierung des Systems wurde die Passwordeingabe mit Augenbewegungen und die Passwordeingabe unter der Verwendung der Tastatur verglichen. Die Ergebnisse haben gezeigt, dass die Fehlerrate in beiden Fällen fast gleich hoch sind. Aber die blickbasierte Passwordeingabe brauchte mehr Zeit als die Eingabe mit einer Tastatur. Eine subjektive Evaluierung hat gezeigt, dass die meisten Benutzer die blickbasierte Passwordeingabe, der Passwordeingabe mit einer Tastatur, vorziehen.

In einer ähnlichen Arbeit wurden für die PIN-Eingabe drei verschiedene Eyetracking-Methoden evaluiert [DLWD07]. Die erste Methode war Verweilzeit-basiert. In dieser Methode sollte ein Benutzer eine bestimmte Zeit eine Ziffer auf dem Bildschirm fixieren, um sie zu "drücken". Bei der "Look & Shoot" -Methode sollte eine spezielle Taste gedrückt werden, dass die gerade angeschaute Ziffer auf dem Bildschirm eingegeben wird. Für die dritte Methode wurden Blickgesten verwendet. Dafür wurden die in Abbildung 3.1 dargestellten Gesten entworfen. Eine Evaluierung hat gezeigt, dass obwohl für die Interaktion mit Blickgesten Benutzer mehr Zeit brauchen, die Fehlerrate bei dieser Methode kleiner war als bei den anderen Methoden.

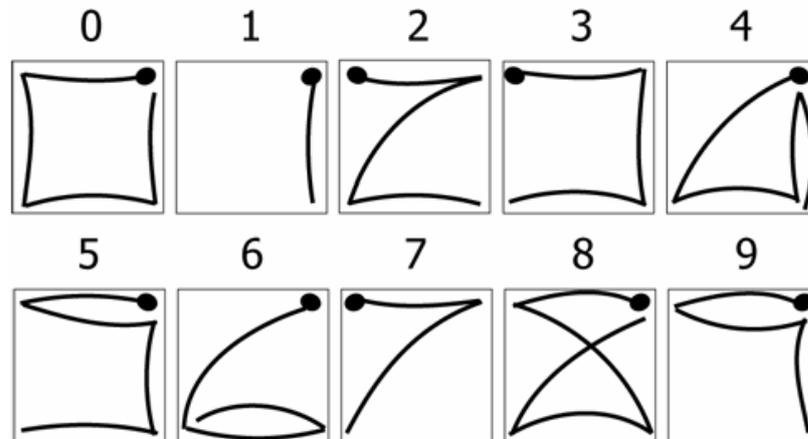


Abbildung 3.1: Numerische Blickgesten für die PIN-Eingabe [DLWD07]

Istance et al. haben Blickgesten für die Interaktion beim Spielen verwendet [IHI⁺10]. Die vorgestellte Technik verwendete fünf aktive Bereiche auf einem Bildschirm, um Gesten zu realisieren. Ein Bereich war die Mitte des Bildschirms und die andere vier lagen entweder in den Ecken oder auf den Seiten des Displays. Damit verschiedene Aktionen ausgelöst werden konnten, musste ein Benutzer zwei- oder dreiteilige Blickgesten tätigen, indem er mit einer bestimmten Reihenfolge jeweils auf zwei oder drei aktive Bereiche schaut. Die Gesten werden nur dann akzeptiert, falls Anfangs- und Endpunkt die Mitte des Bildschirms war. Die Experimente mit dieser Technik haben gezeigt, dass ein Benutzer durchschnittlich für zweiteilige Blickgesten 490 ms und für dreiteilige Blickgesten 880 ms brauchte. Außerdem können diese Gesten durch ein paar Übungen leicht erlernt werden.

In einigen Arbeiten wurde die blickbasierte Interaktion in Computerspielen demonstriert. Smith und Graham haben blickbasierte Interaktion und die Interaktion mit einer Maus für verschiedene Computerspielarten evaluiert [SG06]. Die Ergebnisse der Evaluierung zeigen, dass Benutzer stärker in die Spiel-Welt eintauchen konnten, wenn sie blickbasiert interagierten. Nacke et al. haben die "Gameplay-Erfahrung" untersucht, indem Benutzer ein Spiel mit dem Blick steuern sollten [NSSL09]. Die Ergebnisse zeigten, dass bei der blickbasierten Steuerung eines Computerspiels eine positive "Gameplay-Erfahrung" geboten werden kann.

Außerdem wurde in einigen Arbeiten die Kombination von blickbasierter Interaktion und anderen Interaktionsmethoden vorgestellt. Elepfandt und Grund haben als Lösung für das "Midas Touch" -Problem eine Kombination von Blick- und Sprachsteuerung vorgeschlagen [EG12]. Bei dieser Methode kann ein Benutzer Objekte auf einem Display mit einem Blick auswählen und daraufhin mit Sprachbefehlen verschiedene Aktionen auslösen. Solange kein Sprachbefehl vom Benutzer kommt, wird keine Aktion ausgeführt. Dadurch kann das "Midas Touch" -Problem in einem solchen System nicht auftreten.

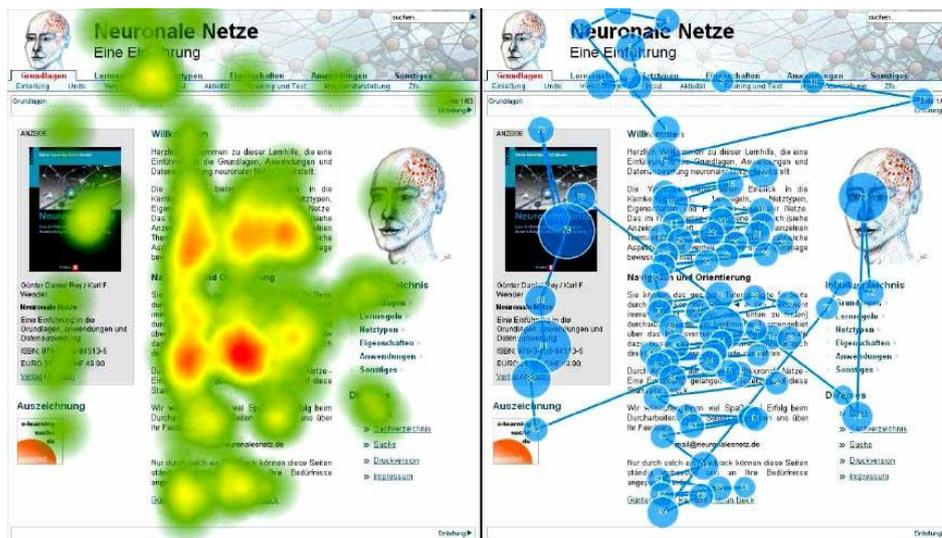


Abbildung 3.2: Die Visualisierung der visuellen Aufmerksamkeit eines Benutzers durch Heat-Map (links) und Gaze-Plot (rechts).

Quelle: http://www.elearning-psychologie.de/eyetracker_ii.html, letzter Zugriff am 20.10.2013

3.2.2 Eyetracking als Bewertungsmöglichkeit für Usability

Eyetracking wird nicht nur als eine Interaktionstechnik verwendet, sondern kann auch als ein Analyseverfahren für die Usability verwendet werden. Durch die Verwendung eines Eyetrackers kann ermittelt werden, wann und welche Bereiche einer Benutzerschnittstelle angeschaut werden oder welche Objekte auf einem Display am meisten fixiert werden. Die Aufzeichnungen dieser Blickinformationen stellen eine dynamische Spur von der visuellen Aufmerksamkeit eines Benutzers dar [PB05]. Diese Aufzeichnungen können mit verschiedenen Methoden visualisiert werden. Gaze-Plot ist eine Visualisierungstechnik, in der die Fixationen nummeriert und als Kreis gekennzeichnet werden (siehe Abbildung 3.2). In dieser Technik werden Sakkaden als Linien dargestellt, die Fixationen miteinander verbinden. Außerdem sagt die Größe des Kreises aus, wie Länge dieser Bereich angeschaut wurde. Diese Visualisierungstechnik ist für längere Aufnahmen von Blickdaten nicht geeignet, denn die Ansicht wird nach einer langen Aufnahme unübersichtlich [MZ05]. Dieses Problem tritt bei Heat-Maps (siehe Abbildung 3.2) nicht auf. Bei dieser Darstellungsform werden Bereiche, die fixiert werden, farblich angezeigt. Die Länge der Fixation in einem Bereich bestimmt die Intensität der Farbe für diesen Bereich. Bei Heat-Maps kann auch die Blickpositionen von mehreren Benutzern enthalten sein und in einer Ansicht dargestellt werden [MZ05]. Außer Heat-Maps und Gaze-Plots werden noch andere Visualisierungstechniken für Blickdaten wie "Area of Interest", "Clusters", "Bee-Swarm" verwendet [MVA⁺12].

Für die Bewertung der Usability werden verschiedene Metriken verwendet. Diese Metriken sind Gegenstand unterschiedlichster Literatur [JK03, PB05]. Die Metriken für die Bewertung

der Usability sollten so gewählt werden, dass sie für die Aufgabe passend sind. Metriken können für verschiedene Aufgaben unterschiedlich gedeutet werden. Zum Beispiel kann eine höhere Anzahl von Fixationen in einem Bereich in Abhängigkeit des Kontextes als Interesse an diesem Bereich oder als Unsicherheit bei der Erkennung eines Zielobjekts interpretiert werden [PB05].

4 Konzept

In dieser Arbeit sollte die Interaktion mit 3D-Displays untersucht werden, die auf dem Blick eines Benutzers basiert. Um die Interaktion mit 2D- und 3D-Displays zu vergleichen, wurde eine Beispielapplikation entwickelt, die sowohl eine 2D- als auch eine 3D-Darstellung ermöglichte. Hier wird die Tiefe des 3D-Inhalts bei der Blickfassung ignoriert. Gibt es ein Objekt an den aktuellen Blickkoordinaten, wird entsprechend eine passende Aktion ausgeführt. Da die x - und y -Koordinaten der Blickposition unter Verwendung eines Eyetrackers erlangt werden können, sollte die Implementierung solcher Interaktionsmöglichkeiten nicht sonderlich aufwändig sein.

Zusätzlich sollte eine blickbasierte Interaktionsmethoden entwickelt werden, die die Tiefe des 3D-Inhalts mitberücksichtigen. Unter der Verwendung dieser Interaktionsmethoden sollte es möglich sein, eine Aktion für ein Objekt auszuführen, das von einem anderen Objekt überdeckt wird. Diese 3D-Objekte sollten jedoch auf dem Display in signifikant unterschiedlichen Tiefen (z -Koordinaten) dargestellt werden. Durch eine Fixation eines solchen Objekts sollte dieses unter Verwendung des Blicks des Benutzers manipuliert werden, auch wenn sich dieses hinter einem anderen befindet. Im Unterschied zur ersten Interaktionstechnik sollten hier die Augendaten nur für das Zielen und eine zusätzliche Eingabemethode (z.B. ein Mausklick) für die Ausführung einer Aktion verwendet werden. Da die Objekte in den unterschiedlichen Tiefenebenen dargestellt werden, reicht hier die Blickposition, die der Eyetracker ermittelt, nicht aus, weil zwei Objekte an der selben Bildschirmkoordinate angezeigt werden, sich jedoch in verschiedenen Tiefenebenen befinden. Um die Tiefe des Objekts zu unterscheiden, sollten die Techniken erweitert werden. Des Weiteren sollten die während der Interaktion ausgeführte Aktionen nur einem Objekt gelten. Dafür wurde das vordere Objekt teilweise durchsichtig angezeigt, sodass das hintere Objekt für die Betrachter sichtbar war.

Durch immer günstiger werdende Eyetracking-Systeme und 3D-Hardware rücken auch die damit eine hergehende Interaktionsmöglichkeiten in der Vordergrund. Die zuvor angesprochenen Vorteile des Eyetracking (beispielsweise Geschwindigkeit, einfache Benutzung) können blickbasierte Interaktionsmethoden mit 3D-Displays ansprechender machen. Die blickbasierte Interaktion mit einem 3D-Display kann in unterschiedlichen Anwendungsfällen verwendet werden. Es könnten Benutzerschnittstellen für 3D-Displays entworfen werden, bei denen Schnittstellenelemente der Wichtigkeit nach in unterschiedlichen Tiefenebenen dargestellt werden. Beispielsweise könnten die Schnittstellenelemente, die häufig verwendet werden, in der vorderen Ebene und die anderen in einer hinteren Ebene angezeigt werden. Somit könnten die oft verwendete Elemente schneller erreichbar sein. Außerdem kann die visuelle Suche bei der Stereoskopie in verschiedenen Tiefenebenen parallel durchgeführt werden [NS86].

Des Weiteren könnte die blickbasierte Interaktion bei der dreidimensionalen Visualisierung komplexer Objekte verwendet werden. In diesen Fällen könnten die verschiedenen Bestandteile eines Objekts in unterschiedlichen Tiefenebenen dargestellt werden. Durch den Blick kann ein bestimmter Teil des Objekts ausgewählt und gesteuert werden. Ein weiterer Vorteil dieser Methoden ist die berührungslose Interaktion, welche durch die Verwendung eines Eyetrackers realisiert werden kann.

Die blickbasierte Interaktion mit 3D-Displays kann auch im medizinischen Bereich eingesetzt werden. Dreidimensionale Bilder findet Anwendungen in diesem Bereich [CSPo6]. Da es bei der blickbasierten Interaktion keinen physikalischen Kontakt mit dem Geräte gibt, ist dies eine sterile Interaktionsmöglichkeit, was im medizinischen Bereich sehr wichtig ist.

Die blickbasierte Interaktion mit Smartphones wurde in einigen Arbeiten evaluiert [DDLSo7]. Außerdem gibt es auf dem Markt mobile Geräte mit 3D-fähigen Displays, beispielsweise 3D-Smartphones von LG ¹. Durch die Gruppierung der Informationen in verschiedenen Tiefen des Displays könnten bei solchen Geräten mehr Information gleichzeitig angezeigt werden. Touch-Interaktion kann bei der geringeren Displaygröße dieser Geräte Schwierigkeiten bereiten. In diesem Fall könnten die blickbasierten Interaktionsmethoden die Interaktion mit 3D-Smartphones vereinfachen.

Die Verwendung von 3D-Displays im Auto erhöht die Anforderung an den Benutzer beim Autofahren nicht [BAS12]. Durch die Verwendung der blickbasierten Interaktionsmethoden könnte die Interaktion mit solchen Displays im Auto schnell - kurzes darauf Schauen oder Blickgesten erledigt werden, ohne die Hände von dem Lenkrad nehmen zu müssen.

Ein weiterer Anwendungsfall für die vorgestellte Interaktionstechniken ist die Interaktion mit öffentlichen Displays. Die blickbasierte Interaktion mit öffentlichen Displays wurde von San Agustin et al. vorgestellt [SAHT10]. Außerdem könnten Produkte durch Werbung auf autostereoskopischen, öffentlichen Bildschirmen die Aufmerksamkeit der Beobachter stärker erregen. Bei derartigen Displays könnte die blickbasierte Interaktion entweder allein oder zusammen mit anderen Eingabemodalitäten (beispielsweise Hand-Gesten) verwendet werden.

¹<http://www.lg.com/de/3D-Produkte/LG-3D-Smartphones.html>, letzter Zugriff am 20.10.2013

5 Entwurf

Für diese Arbeit wurde eine Beispielapplikation entwickelt, die auf einem stereoskopischen Display ausgegeben werden kann. Die Beispielapplikation ist ein 3D-Spiel mit einem Weltraumkontext. Eine Eigenschaft, die das Spiel interessant macht, ist die benutzte Interaktionstechnik. Die ganze Steuerung des Spiels beruht komplett auf Augenbewegungen. Um die Augenbewegungen zu erfassen, wird ein Eyetracker verwendet. Außer dem Spiel wurde ein Server implementiert, der die Kommunikation zwischen dem Spiel und dem Eyetracker realisiert. Der Server übernimmt die Rolle einer Schnittstelle zwischen einer 3D-Applikation, in diesem Fall das Spiel, und dem Eye-Tracker. Der Server wird während der Verwendung Befehle von dem Spiel empfangen und diese an die Eyetracking-Software weiterleiten. Umgekehrt erhält der Server Daten vom Eyetracker. Diese werden geparkt und zum Spiel weitergeleitet. Zum Spiel gesendete Daten sind normalerweise die Blickpositionen des Benutzers, die der Eyetracker erfasst. Die Augendaten dienen als Eingabe für das Spiel. Diese Daten werden über einen Steuerungslaptop, auf dem eine spezielle Software läuft, zum Server geschickt. Der Server analysiert diese Daten und sendet diese zum 3D-Spiel, das auf einem hochperformanten Rechner läuft. Im Spiel empfangene Augendaten können nun für Berechnungen genutzt werden. Die Ausgabe wird dann auf einem stereoskopischen Bildschirm dargestellt. Die komplette Interaktion zwischen dem Spiel, dem Server und dem Eyetracker kann Abbildung 5.1 entnommen werden. Die benutzte Software für das Eyetracking-System wird in dem Unterabschnitt 5.2.3 erläutert.

Der Rest dieses Kapitels beschreibt den Ablauf des Spiels und den Programmen, die zur Entwicklung der Beispielapplikation verwendet wurden.

5.1 Der Ablauf des Spiels

Um die blickbasierte Interaktion zu realisieren, wurde in dieser Arbeit ein videobasierter Eyetracker verwendet. Der Eyetracker muss vor der Verwendung kalibriert werden. Hierfür wird das Spiel mit einem kurzen Kalibrierungsprozess begonnen. Dieser Prozess wurde auch interaktiv aufgebaut und der Spieler soll die Kalibrierungspunkte fixieren und diese mit einem Klick auf die linke Maustaste akzeptieren. Nach diesem Schritt wird eine Test-Szene erscheinen, in welche ein Benutzer eine kleine Übung mit der verwendeten Interaktionstechnik machen soll. Nach Abschluss dieser Schritte, kann eigentliche Spielen beginnen. Solange keine großen Kopfbewegungen oder irgendeine Körperbewegung gemacht wird, kann das Spiel mit der gleichen Kalibrierung mehrmals gespielt werden. Jetzt kann ein Benutzer entscheiden, welchen Teil des Spiels, in welcher Darstellung (2D/3D) und mit welcher Zieltechnik er spielen will. Falls das 3D-Spiel ausgewählt wurde, muss der Benutzer eine

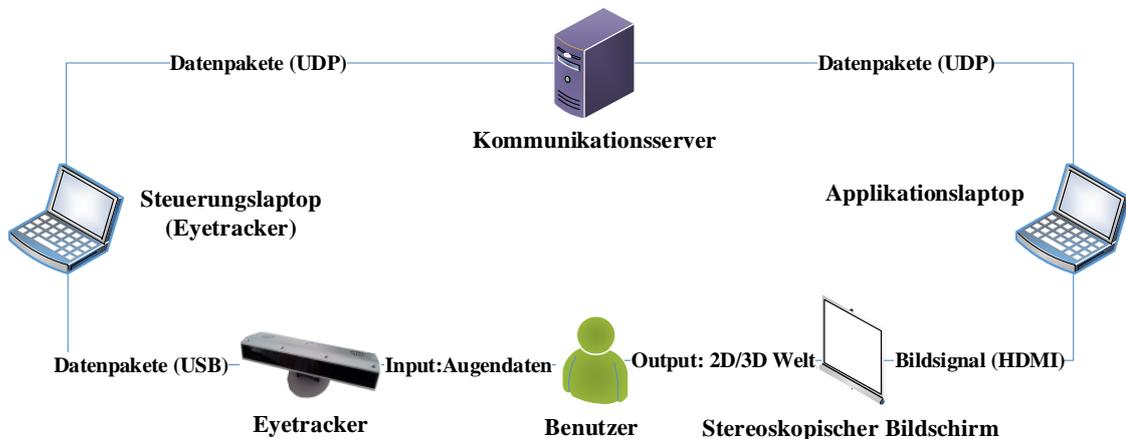


Abbildung 5.1: Die Kommunikation zwischen dem Server, der Applikation und dem Eye-tracker *

Polarisationsbrille tragen. Wenn ein Spiel beendet ist, wird dem Benutzer eine Spielstatistik mit den während des Spiels gesammelten Punkten angezeigt.

5.2 Verwendete Programmen

In diesem Kapitel werden verschiedene Programmen aufgezählt, die für die Entwicklung des Spiels und des Servers verwendet wurden. Als erstens werden die verwendete Software für das 3D-Spiel beschrieben. Diese sind die in Unterabschnitt 5.2.2 beschriebenen Programme für die graphische Bildbearbeitung und um 2D- und 3D-Inhalte zu gestalten, außerdem die in Unterabschnitt 5.2.1 vorgestellte Spiel-Engine um das Spiel zu entwickeln. Danach wird in Unterabschnitt 5.2.3 die Eyetracking-Software genauer erläutert.

5.2.1 Die Spiel-Engine

Zu Entwicklung des Spiels wurde die 3D-Engine Unity3D¹ verwendet. Damit entwickelte 3D-Spiele und 3D-Grafikanwendungen werden auf verschiedenen Plattformen unterstützt. Dazu zählen Windows, Linux, Mac OS X, Android, iOS, Windows Phone 8, Xbox 360, PlayStation 3 usw. Das Hauptfenster von Unity3D besteht aus vielen unterschiedlichen Bereichen. Im Szene-Tab kann die Spielszene aufgebaut werden. Dort können Kamera, Lichtquellen, und für 3D-Applikation passende verschiedene Spielobjekte in die Szene eingefügt und manipuliert

¹www.unity3d.com, letzter Zugriff am 20.10.2013

werden. Einfache Spielobjekte, wie Lichtquellen, Kameras, GUI-Texturen und einige 3D-Modelle werden durch Unity3D bereitgestellt und können direkt verwendet werden. Für komplexere 3D-Modelle unterstützt Unity3D verschiedene Dateiformate. Diese Modelle können in verschiedenen Grafikanwendungen erzeugt und dann in Unity3D benutzt werden. Sowohl 3D-Modelle, als auch durch andere Programmen erzeugte Videos, Musikstücke und Klänge können in Unity3D verwendet werden. Außerdem können für unterschiedliche Effekte Shader-Programme in Cg oder GLSL geschoben werden.

Für die Spiellogik und verschiedene Aktivitäten in der 3D-Applikation können Skripte geschrieben werden. Die Programmiersprachen wie C#, Javascript und Boo dienen in Unity3D diesem Zweck. Außerdem können in einem Spiel für ein Spielobjekt Skripte geschrieben werden, die das Verhalten dieser Objekte während des Spiels definiert. Die verschiedenen Tabs, die im Hauptfenster der Spiel-Engine sichtbar sind, erleichtern es verschiedene Komponenten der Spielobjekten zu konfigurieren. Z.B. um eine Skript auf ein Spielobjekt zu setzen, muss man lediglich das Skript mit der Maus aus dem Projekt-Tab ziehen und auf einem Spielobjekt in dem Hierarchie-Tab loslassen.

5.2.2 Die Programmen für Grafikbearbeitung

Das entwickelte Spiel enthält verschiedene 2D- und 3D-Modelle. Diese Modelle wurden in Blender ² erstellt. Blender ist eine Open-Source-Software für 3D-Grafikdesign. Diese freie Software bietet die Möglichkeit der Modellierung, Texturierung, Beleuchtung, Animation und des Renders eines 3D-Modells oder einer 3D-Szene. Blender kann auch als selbständige Spiel-Engine verwendet werden.

Einige Texturen für 3D-Modelle wurden in GIMP ³ (GNU Image Manipulation Program) vorbereitet. GIMP ist eine freie Open-Source-Software für Bildverarbeitung. Einige 2D-Objekte im Spiel wurden in Blender modelliert und nach dem Rendern als PNG-Bilddatei gespeichert. Danach wurden diese Bilder für das Spiel in GIMP bearbeitet. Die 3D-Modelle, die in Blender modelliert wurden, sind für die Verwendung in Unity3D in das Autodesk FBX-Format exportiert worden.

5.2.3 iView X TM

Um auf die Daten des Eyetrackers zuzugreifen, wurde das iView X TM System von SensoMotoric Instruments ⁴ verwendet. Dieses System erlaubt es verschiedene Studien mit Eyetrackern durchzuführen und enthält alle benötigten Programmteile für Kalibrierung, Verfolgung des Blicks, Identifizierung verschiedenen Augenbewegungen usw. Um den Blick des Benutzers zu erfassen, analysiert die iView X Software Bilder des Auges, die durch eine Videokamera und mithilfe infraroter Beleuchtung aufgezeichnet wurden. Die Software stellt

²www.blender.org, letzter Zugriff am 20.10.2013

³www.gimp.org, letzter Zugriff am 20.10.2013

⁴www.smivision.com, letzter Zugriff am 20.10.2013

dafür die im Bild dunkle Pupille und die Position des Kornealreflexes fest. Mithilfe dieser zwei Informationen kann die Software die Blickposition berechnen.

Das iView X™ System kann auch mit Remote-Befehlen gesteuert werden. Dafür muss ein Rechner mit der Netzwerkschnittstelle des iView X™ System verbunden sein. Dann können Remote-Befehle über Ethernet an iView X™ geschickt werden. Außerdem kann iView X™ über den gleichen Kanal Antworten vom Eyetracker zurücksenden. Die Schnittstelle für Remote-Befehle ist ein Betriebssystem-unabhängige, sprachunabhängige, Protokoll-orientierte Programmierschnittstelle [Sen09]. Mit diesen Befehlen können verschiedene Aktivitäten mit unterschiedlichen Parametern ausgeführt und damit der Eyetracker über das iView X™ System kontrolliert werden. Alle diese Remote-Befehle haben eine bestimmte Syntax. Diese fangen mit "ET_" an und enden mit einem Zeichen für einen Zeilenvorschub. Die Remote-Befehle sind in dem Handbuch für das iView X™ System aufgelistet [Sen09].

6 Implementierung

Das entwickelte System bietet die Möglichkeit mit den Augendaten zweidimensionale und stereoskopische Inhalt zu manipulieren. Das dafür implementierte Spiel kann sowohl auf einem herkömmlichen Bildschirm, als auch auf einem 3D-Fernseher gespielt werden. Die Augendaten werden durch einen Eyetracker erfasst. Für die Kommunikation zwischen dem Eye-Tracker und dem Spiel ist ein Kommunikationsserver zuständig.

In Abbildung 6.1 ist eine vereinfachte Systemarchitektur dargestellt. Sie besteht aus einem Eyetracking-System, einem Kommunikationsserver und einem Applikationsrechner. Auf dem Applikationsrechner läuft das Spiel und dieser ist durch ein HDMI-Kabel mit einem Display verbunden, das die Anzeige von 3D-Inhalten ermöglicht. Um Befehle an das Eyetracking-System zu senden, werden diese vorerst an den Kommunikationsserver geschickt. Der Server leitet diese an das Eyetracking-System weiter und empfängt entsprechende Daten als Antwort. Diese Daten können nun an den Applikationsrechner weitergeleitet werden. Der Server wurde so entwickelt, dass dieser mit zusätzlichen Remote-Befehlen für das Eyetracking-System leicht erweitert werden kann. Außerdem können andere Applikationen durch Benutzung des Kommunikationsprotokolls den Server auch für die Steuerung des Eyetracking-Systems verwenden.

Im restlichen Teil dieses Kapitels wird die Entwicklung des Kommunikationsservers und des Spiels beschrieben. In Unterkapitel 6.1 wird auf die Entwicklung des Kommunikationsservers eingegangen. Die Implementierung des Spiels, das in 2D- und 3D-Modus gespielt werden kann, wird in der Unterabschnitt 6.2 vorgestellt. In dieser Arbeit wurden die Beispielapplikation und der Kommunikationsserver zusammen mit Jonas Auda implementiert.

6.1 Kommunikationsserver

Um die Interaktion zwischen dem Eyetracker und dem Spiel zu realisieren, wurde ein Server entwickelt. Der Server wurde vollständig in der Programmiersprache Java implementiert. Dieser akzeptiert Befehle von dem Spiel und leitet diese an iView X weiter, welche wiederum den Eyetracker steuert. Danach bekommt der Server Augendaten als Rückmeldung der iView X Software, analysiert diese und schick sie zum Spiel zurück. Der Server besteht aus unterschiedlichen Klassen (siehe Abbildung 6.1), welche weiter unten detailliert vorstellt werden.

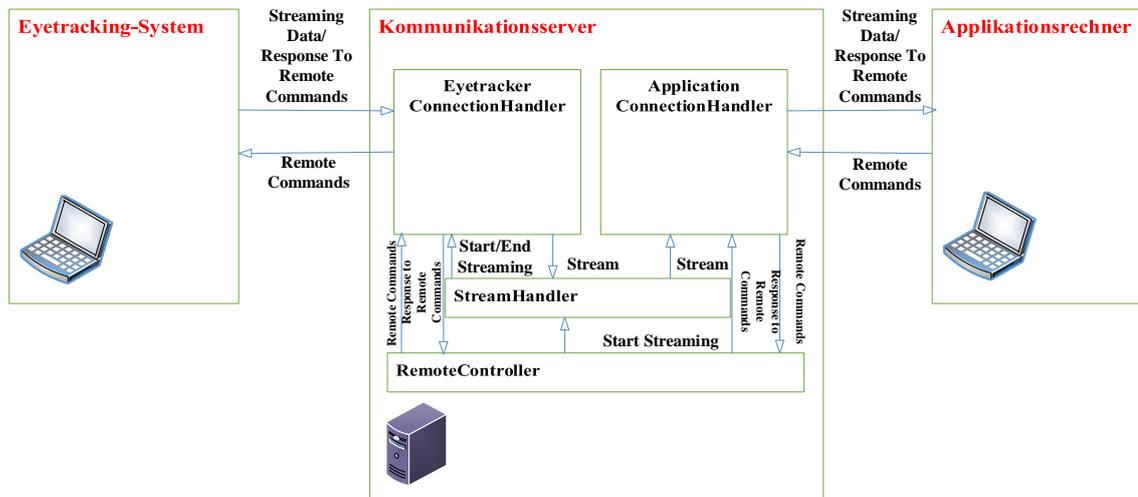


Abbildung 6.1: Die Systemarchitektur. Sie besteht aus einem Eyetracking-System, das Augendaten erfassen, einem hochperformanten Rechner, auf dem die Beispielapplikation läuft und der mit einem stereoskopischen Display verbunden und einem Kommunikationsserver, der die Kommunikation zwischen Beispielapplikation und dem Eyetracking-System ermöglicht. *

6.1.1 ConnectionHandler

Der ConnectionHandler unterstützt die Kommunikation zwischen dem Eyetracking-System und dem Applikationsrechner. Davon gibt es zwei Instanzen für die Kommunikation mit der iView X Software und dem Spiel. Die Netzwerkkommunikation wird mit einem UDP-Datagrammsocket aufgebaut. Dafür müssen die IP-Adressen und Portnummern angegeben werden. Des Weiteren bietet diese Java-Klasse noch Methoden, um Daten mit verbundenen Rechnern auszutauschen.

6.1.2 RemoteController

Der RemoteController kommuniziert mit den beiden Instanzen der ConnectionHandler-Klasse und enthält vordefinierte Methoden, welche das Senden von Remote-Befehlen an das Eyetracking-System ermöglicht. Jeder Befehl ist als Java-Methode implementiert. Auf dem Server wurden nur für die Befehle eine Methode implementiert, die auch im Spiel Verwendung finden. Diese Klasse kann mit weiteren Remote-Befehlen, die durch das verwendete Eyetracking-System unterstützt, erweitert werden. Diese Klasse bekommt die Anfragen für Remote-Befehle über die Instanz von der ConnectionHandler-Klasse, die für die Kommunikation mit der Beispielapplikation zuständig ist. Ein von der Beispielapplikation empfangener Befehl ruft eine Methode auf dem Kommunikationsserver auf, welche den passenden

Remote-Befehl an das Eyetracking-System schickt und als Rückgabewert die Antwort des Eyetracking-Systems auf diesen Remote-Befehl zurückliefert. In diesen Methoden werden Remote-Befehle, die eine bestimmte Syntax haben (siehe Unterabschnitt 5.2.3), über die Instanz der ConnectionHandler-Klasse für das Eyetracking-System an die iView X Software geschickt. Außer der Remote-Befehl, der das Streaming startet, werden alle Befehle auf die gleiche Weise ausgeführt. Dieser Befehl wird in einem zusätzlichen Thread ausgeführt. Nach der Ausführung dieses Befehls fängt die iView X Software an, die Augendaten in einem vorgegebenen Format und mit einer bestimmter Frequenz an den Server zu senden. Das Format für die Übertragung wird auch mit einer Methode definiert, die vor dem Startbefehl des Streams ausgeführt werden muss. Ist das Streaming gestartet, sendet das Eyetracking-System kontinuierlich Augendaten an den Kommunikationsserver.

Da die Netzwerkkommunikation mit den UDP-Datenpaketen realisiert wurde, können Datenpaketen verloren gehen oder in falscher Reihenfolge ankommen. Dieser Nachteile kann dem Kommunikationsprotokoll schaden. Beispielsweise falls während der Kalibrierung Pakete verloren gehen, kann die Kalibrierung nicht erfolgreich abgeschlossen werden. Da die Kalibrierung-Schritte teilweise in der Beispielapplikation geschehen, kann die Kalibrierung nicht lokal in iView X durchgeführt werden. Aufgrund dessen wurden die Methode für die relevanten Befehle so entwickelt, dass ein Paketverlust kein Problem darstellt und automatisch erkannt und beseitigt wird. Dieses Problem bereitet meistens bei der Kalibrierung Schwierigkeiten. Nach dem Starten der Kalibrierung soll jeder Kalibrierungspunkt akzeptiert werden. Dazu wird jedes mal eine zusätzliche Methode ausgeführt. Falls ein Paket während der Kalibrierung verloren geht, kommt es dazu, dass in dieser Methode für das Akzeptieren des Kalibrierungspunktes eine falsche oder gar keine Antwort zurückkommen. Wird dies vom Server nicht abgefangen, kann dies der Kalibrierung schaden und im schlimmsten Falle muss diese wiederholt werden. Im Server wurde diese Probleme behandelt. Eine detaillierte Beschreibung der Kalibrierung im Spiel wird in Unterkapitel 6.2.1 gegeben.

6.1.3 StreamHandler

Der StreamHandler ist zuständig für die Analyse und die Vermittlung der Augendaten, die das Eyetracker-System erfasst. Um Augendaten vom Eyetracker zu erhalten, muss eine kontinuierliche Übertragung der Daten gestartet werden. Dafür wird eine Methode in der RemoteController-Klasse aufgerufen, die die StreamHandler-Klasse als Thread startet. Die Daten werden in einem vordefinierten Format und mit einer bestimmten Frequenz verschickt. In der Arbeit wird das Format für die Augendaten mit folgendem Befehl bestimmt:

```
"ET_FRM \ "%ET,%SX#%SY,%DX#%DY,%PX#%PY# \ " \ n"
```

Damit schickt das Eyetracking-System Information über Augentyp, in diesem Fall binokular, X- und Y-Blickkoordinaten von linkem und rechtem Auge, Pupillendurchmesser und Pupillenposition für die beiden Augen an den Kommunikationsserver. Die Daten sind mit Kommata und Doppelkreuz-Zeichen getrennt, damit diese leicht geparst werden können. Das Streaming erfolgt mit einer Frequenz von 30 Hz. Ein Datenverlust eines einzelnen

Paketes kann die Übertragung nicht sonderlich stören, denn die Daten des nächsten Paketes können den Fehler kompensieren.

Die Daten eines Eyetracker sind nicht von Grund auf fehlerfrei, denn die erfassten Daten sind durch ein ständiges Augenzittern gestört. Um die Blickdaten zu glätten, wurde ein Filter implementiert. Der Filter ist optional und kann jederzeit an- oder ausgeschaltet werden. Für den Filter wurde der in der Arbeit von Kumar vorgestellte Algorithmus verwendet [Kum07]. Die Idee des Algorithmus ist die Unterscheidung von Sakkaden und der Blickkoordinate mit einem Rauschen. Der Algorithmus arbeitet mit zwei Gruppen von Blickkoordinaten. Diese sind schon ausgewählte Fixationspunkte und potentielle Fixationspunkte, für die noch entschieden werden muss, ob diese echte Fixationspunkte oder ein Rauschen sind. Falls eine neue Blickkoordinate nahe dem aktuellen Fixationspunkt im Umkreis von einem Schwellwert für eine Sakkade liegt, wird diese zu der Gruppe der aktuellen Fixationspunkte hinzugefügt und der Fixationspunkt nochmal berechnet und zurückgegeben. Ein Fixationspunkt von einer Gruppe wird mit Hilfe unten angegebene Formel berechnet. Mit dieser Formel bekommt der letzte zur Gruppe hinzugefügte Fixationspunkt höhere Priorität.

$$p_{mean} = \frac{p_0 + 2 * p_1 + 3 * p_2 + \dots + (n+1) * p_n}{1 + 2 + 3 + \dots + n + n + 1}$$

Falls diese Blickkoordinate nicht innerhalb des Schwellwerts liegt, werden die Einträge der Gruppe von potentiellen Fixationspunkten gelöscht und mit dieser Blickkoordinate neu initialisiert. In diesem Fall wird auch der aktueller Fixationspunkt zurückgegeben. Ist eine neue Blickkoordinate nahe dem potentiellen Fixationspunkt im Umkreis mit einem Schwellwert für eine Sakkade, wird dieser Punkt zur Gruppe von potentiellen Fixationspunkten hinzugefügt, alle Einträge von aktuellen Fixationspunkten gelöscht und die Gruppe von potentiellen Fixationspunkten wird die neue Gruppe der aktuellen Fixationspunkten. Der zurückgegebene Fixationspunkt wird dann der aktuelle Fixationspunkt sein. Falls der neue Punkt nicht im Umkreis von dem Schwellwert für eine Sakkade liegt, wird der Fixationspunkt aus den potentiellen Fixationspunkten berechnet und zurückgegeben, danach werden die Einträge aus dieser Gruppe entfernt und der neue Punkt wird der einziger Eintrag dieser Gruppe sein.

Ein Gruppe kann maximal zehn Einträge haben. Falls dieser Wert überschritten wird, werden alle Einträge der Gruppe der potentiellen Fixationspunkten und alle Einträge der Gruppe der aktuellen Fixationspunkten bis auf den letzten Eintrag entfernt. Diese sollte zu keinem Problem führen, denn mehr als zehn Einträge in der Gruppe bedeutet, dass ein Betrachter für einige Zeit eine bestimmte Stelle angeschaut hat. Dadurch unterscheiden sich neue Blickkoordinaten nicht sonderlich vom aktuellen Fixationspunkt. Als Schwellwert für Sakkaden wurden 40 Pixel ausgewählt.

6.1.4 Grafische Benutzeroberfläche

Die grafische Benutzeroberfläche wurde für den Kommunikationsserver entwickelt, um die Bedienbarkeit zu vereinfachen. In Abbildung 6.2 ist diese dargestellt. Im oberen Teil

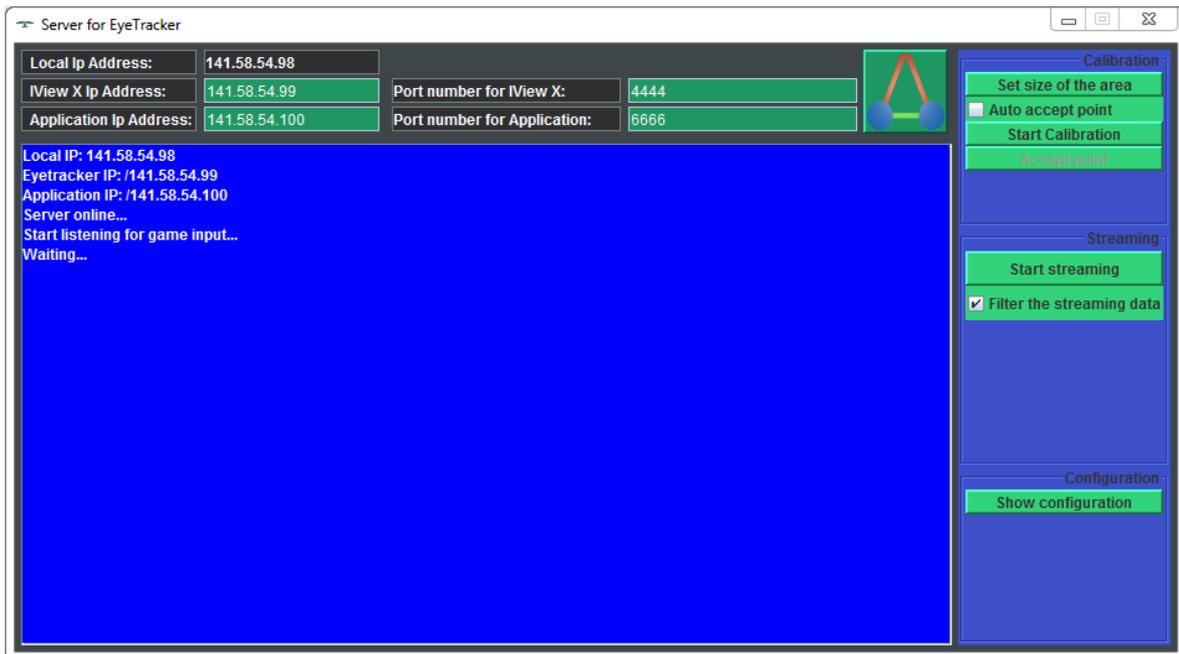


Abbildung 6.2: Grafische Benutzeroberfläche des Kommunikationsservers

der Benutzeroberfläche gibt es Eingabefelder, in denen die IP-Adressen und Portnummern von einem Applikationsrechner und dem Rechner des Eye-tracking-Systems eingegeben werden sollen. Der große Button auf der rechten Seite der Eingabefelder dient für das Ein- und Ausschalten des Servers. Außerdem gibt es auf der rechten Seite einige GUI-Elemente, die die manuelle Steuerung einiger Befehle ermöglicht. Der Filter kann auch durch ein GUI-Element jederzeit ein- oder ausgeschaltet werden.

Um die Kommunikation zwischen Applikation und iView X Software zu erstellen, braucht man, im Prinzip, nur die IP-Adressen und Portnummern einzugeben und den Server zu starten. Danach werden während der Kommunikation gesendete Nachrichten unten, in dem Anzeigebereich dargestellt. GUI-Elemente für andere Remote-Befehle sollten nur dann benutzt werden, falls irgendein Problem auftritt. Als Beispiel kann der Button für die Datenübertragung (Starten/Stoppen) genannt werden. Falls irgendein Problem im Spiel auftritt, sollte es neu gestartet und die Kalibrierung wiederholt werden. Dazu sollte die kontinuierliche Übertragung von Augendaten manuell ausgeschaltet werden, da sonst diese dem Kommunikationsprotokoll Schwierigkeiten bereiten.

6.2 Spiel

Als Beispielapplikation wurde ein Spiel entwickelt, das zwei Teile mit unterschiedlichen Spielzielen hat. In dem Spiel sollen Objekte abgeschossen werden. Um zu Zielen werden



Abbildung 6.3: Die Kalibrierungsszene. In der Mitte der Szene ist der erste Kalibrierungspunkt dargestellt. Die Nummern verdeutlichen die Reihenfolge des Erscheinens der Kalibrierungspunkte auf dem Bildschirm.

Blickdaten vom Benutzer verwendet. Das Spiel wurde komplett mit der Unity3D Spiel-Engine entwickelt. Das Spiel ist auf verschiedenen Szenen aufgebaut. Insgesamt gibt es sieben Szenen im Spiel: zwei Szenen für Kalibrierung, eine für das Hauptfenster, zwei Spielszenen und zwei Endszenen. In den Endszenen werden die Anzahl der gesammelten Spielstunden und andere Spielstatistiken angezeigt.

In dem Hauptfenster kann die IP-Adresse des Kommunikationsservers und der Spielname eingegeben werden. Außerdem kann hier das Verfahren für das Zielen und ein Spielteil ausgewählt werden, welche vom Spieler gespielt werden möchte. Von dieser Szene aus kann entweder eine Kalibrierung durchgeführt oder ein Teil des Spiels gestartet werden.

6.2.1 Kalibrierung

Vor dem Spielen muss der Eyetracker kalibriert werden. Dafür wurde ein lokaler Kalibrierungsprozess implementiert. Wenn der Button für Kalibrierung im Hauptfenster gedrückt wird, sendet das Spiel die Größe der Kalibrierungsfläche in Pixeln an den Server. Der Server führt einen Remote-Befehl aus, der das Eyetracker-System informiert, in welchem Bereich Blickkoordinaten erfassen werden sollen. Danach wird ein Remote-Befehl ausgeführt, der die Kalibrierung startet. In der Mitte der Kalibrierungsszene wird ein blaues Kreuz auf dunklem Hintergrund dargestellt. Dieses Kreuz repräsentiert einen Kalibrierungspunkt. Es befindet sich ein rotes Kreuz in der Mitte des blauen Kreuzes. Bei einer Kalibrierung muss ein Benutzer dieses rote Kreuz fixieren und den Kalibrierungspunkt mit einem Klick auf

die linke Maustaste bestätigen. Das Eyetracker-System akzeptieren daraufhin den Kalibrierungspunkt. Falls das Fixieren auf einen Punkt nicht erfolgreich war, muss der Benutzer wieder diesen Punkt fixieren und bestätigen. Nach erfolgreicher Bestätigung wird der nächste Kalibrierungspunkt angezeigt. Diese Schritte werden neun mal unterschiedlichen Kalibrierungspunkten wiederholt. Die Reihenfolge des Erscheinens der Kalibrierungspunkte wird in Abbildung 6.3 dargestellt. Sobald der Kalibrierungsprozess erfolgreich abgeschlossen wurde, wird der Kommunikationsserver einen Befehl zum Eyetracker-System schicken, um die Übertragung der Augendaten zu starten. Jetzt kann ein Nutzer die Qualität der Kalibrierung überprüfen und die Interaktion mit dem System üben. Hierzu wird ein Meteorit in der selben Reihenfolge wie zuvor die Kalibrierungspunkte erschienen sind, angezeigt. Ein Benutzer soll den dargestellten Meteorit anschauen, dadurch explodiert dieser. Nachdem die Übung erfolgreich abgeschlossen wurde, kann entweder der erste Teil des Spiels gespielt werden oder eine andere Kalibrierungsszene ausgewählt werden.

6.2.2 Spielbare Szenen

Das Spiel besteht auf zwei verschiedenen Teilen. Der erste Teil des Spiels kann sowohl auf einem herkömmlichen Bildschirm, als auch auf einem stereoskopischen Display gespielt werden. Dieser Teil wird komplett mit Blickdaten kontrolliert. In diesem Teil ist die Aufgabe des Spielers die Erde vor Meteoriten zu schützen. In der Mitte der Spielszene werden die Erde und der sie umkreisende Mond dargestellt. Die Meteoriten kommen von allen Richtungen auf die Erde zugeflogen. Bei einem Einschlag auf der Erde explodieren die Meteoriten. Falls das zehn mal passiert, wird das Spiel beendet. Der Benutzer soll die Meteoriten vor dem Einschlag abschießen. Dafür braucht er nur einen Meteorit anzuschauen, wodurch auf diesen gezielt und geschossen wird. Dadurch explodiert der Meteorit. In dem Spiel gibt es auch solche Meteoriten, welche an der Erde vorbeifliegen. Diese Meteoriten sind durch grüne Flecken gekennzeichnet. Falls dieser Art von Meteoriten abgeschossen wird, wandeln diese sich in zwei auf die Erde zufliegende Meteoriten um. Das Abschießen solcher Meteoriten sollte vermieden werden. Dieser Spielteil ist in Wellen eingeteilt. Die Nummer der Welle anspricht der Anzahl der in der Szene fliegenden Meteoriten. Eine Szene von diesem Teil des Spiels wird in Abbildung 6.4 dargestellt.

Im zweiten Teil des Spiels soll ein Benutzer feindliche Raumschiffe abschießen. Dieser Teil kann nur auf dem stereoskopischen Bildschirm gespielt werden. In der Spielszene werden Meteoriten und Raumschiffe in einer bestimmten Reihenfolge von links nach rechts fliegen. Die Meteoriten fliegen in dieser Szene vor der Bildfläche des 3D-Displays (negative Parallaxe), die Raumschiffe hinter der Bildfläche (positive Parallaxe). Jedes Raumschiff fliegt hinter einem Meteoriten. Die Meteoriten hinter denen ein Raumschiff fliegt, werden transparent dargestellt und mit grüner Farbe umrandet (siehe Abbildung 6.5). Das Ziel des Spielers ist das Abschießen der Raumschiffe ohne den davor fliegenden Meteoriten explodieren lassen. Hierfür soll sich ein Nutzer einige Zeit auf ein Raumschiff konzentrieren. Deswegen werden in diesem Teil des Spiels Blickdaten nur für das Zielen verwendet. Für das Schießen muss die linke Maustaste gedrückt werden. Falls ein Meteorit abgeschossen wird, wird das dahinter fliegende Raumschiff beschleunigt. Wenn es zehn Raumschiffe zur rechten

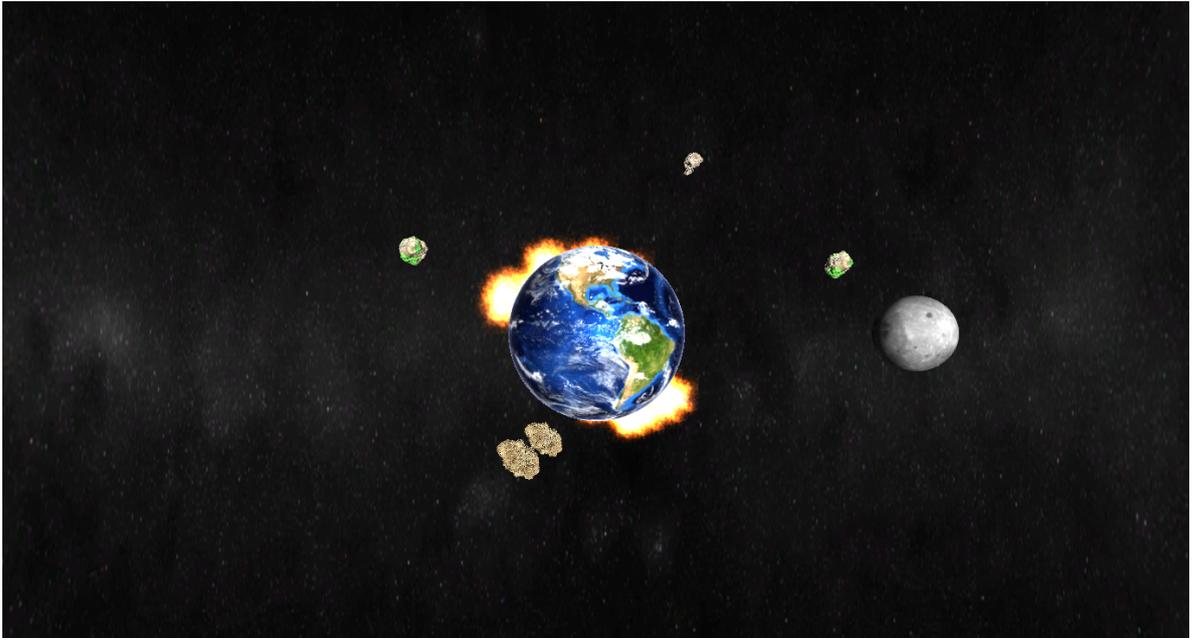


Abbildung 6.4: Eine Szene des ersten Teil des Spiels. Die Meteoriten fliegen von allen Richtungen auf die Erde.

Seite des Bildschirms schaffen, wird das Spiel beendet. Um diesen Teil des Spiels spielen zu können, muss eine Tiefenkalibrierung durchgeführt werden.

6.2.3 Tiefenkalibrierung

Die Tiefenkalibrierung kann nach der normalen Kalibrierung und auf einem stereoskopischen Bildschirm durchgeführt werden. Um die Tiefenkalibrierung zu machen, muss die Übertragung der Augendaten gestartet sein. Dafür wird erstmal ein 3D-Objekt mit negativ Parallaxe in der selben Reihenfolge wie die Kalibrierungspunkte dargestellt (Vgl. Abbildung 6.3). Ein Benutzer soll das angezeigte Objekt fixieren und dies mit einem Klick auf die linke Maustaste bestätigen. Dieser Schritt wird noch acht mal wiederholt. Danach wird der gleiche Prozess für ein 3D-Objekt wiederholt, das sich im Bereich der positiven Parallaxe befindet. Nach jeder Bestätigung wird der Pupillendurchmesser und der Abstand zwischen den Pupillen gespeichert und der Durchschnitt dieser Werte jeweils für positiv und negativ Parallaxe berechnet. Nun können diese vier Werte benutzt werden, um zu entscheiden, in welcher Parallaxe ein Objekt liegt. Beim Konzentrieren auf ein Objekt, das nahe zum Benutzer in negativ Parallaxe dargestellt wird, werden die Augen konvergieren (siehe Unterkapitel 2.2.1). Dadurch wird der Abstand zwischen beiden Augenpupillen kleiner, als beim Konzentrieren auf Objekte in positiver Parallaxe. Zusätzlich kommt es zur Verengung der Pupillen, wenn ein Nutzer sich auf einen nahen Gegenstand konzentriert [RHFL10]. Folgerichtig werden die Pupillendurchmesser beim Anschauen einem Objekt bei negativer Parallaxe kleiner sein.

Der Eyetracker erfasst den Pupillendurchmesser. Aber der Abstand zwischen den Pupillen wird vom Server aus den Positionen von den Pupillen berechnet.

Im zweiten Spielteil werden die Raumschiffe und die Meteoriten auf den Ebenen passend zur Tiefenkalibrierung fliegen - Meteoriten vorne und Raumschiffe hinten. Falls ein Benutzer ein Objekt abschießen will, wird erstmal die Blickposition erfasst und untersucht, ob es an dieser Koordinaten ein Objekt gibt. Falls ja, werden in Abhängigkeit vom verwendeten Verfahren für das Zielen, entweder der aktueller Pupillendurchmesser mit dem in der Tiefenkalibrierung berechneten Pupillendurchmesser oder der aktuelle Abstand zwischen beiden Pupillen mit von den in der Tiefenkalibrierung berechneten Pupillendistanzen verglichen. Ist der aktuelle Wert näher dem Wert, der bei der Tiefenkalibrierung für die positiv Parallaxe berechnet wurde, wird das an dieser Position fliegende Raumschiff zerstört. Ansonsten wird davor fliegender Meteorit zerstört.

6.2.4 Feedback

Während des Spiels bekommt ein Spieler auditives und visuelles Feedback. In beiden Spielteilen werden die Koordinate der Blickposition mit einem Zeichen markiert. Ohne dieses visuellen Feedback ist es schwer zu zielen, denn eine schlechte Kalibrierung kann ein Grund dafür sein, dass die echte Blickposition und die durch den Eyetracker erfasste Blickkoordinate nicht übereinstimmt. Mit einem solchen visuellen Feedback kann leicht erkannt werden, dass die Kalibrierung nicht gut war und ein Nutzer kann seine Kopfbewegung leicht ändern, um die erfasste Blickposition zu korrigieren oder noch mal kalibrieren. Außerdem wird dem Benutzer bei einer Explosion auch auditives und visuelles Feedback gegeben, was die Qualität des Spiels verbessert.

6.2.5 Datenerfassung

Um eine Evaluierung durchzuführen, werden die Ereignisse, die während des Spiels geschehen, als Daten erfasst. Aus diesen erfassten Daten kann das Verhalten des Spielers unter verschiedenen Spielumständen, Probleme bei der Augendatenübertragung vom Eyetracker-System und die gesammelten Spielpunkte abgelesen werden. Die Daten werden für beide Spielteile in unterschiedlichen Dateien im CSV-Format gespeichert. Bei jedem stattgefundenen Ereignis im Spiel wird eine neue Zeile der erfassten Daten in die Datei eingefügt. Zu den erfassten Ereignissen zählen Einschläge von Meteoriten auf der Erde, das Abschießen eines Meteoriten oder Raumschiffes. Mit jedem Ereignis werden auch der Zeitpunkt des Auftretens, die Art des Ereignisses, einige Augendaten gespeichert. Außerdem werden noch Anfangs- und Endzeitpunkt des Spiels, Spielernamen und erreichte Punktzahl aufgezeichnet.

6.2.6 Erzeugung von stereoskopischem Inhalt

Die 3D-Bildschirme benutzen binokulare Disparität (siehe Unterkapitel 2.2.2), um stereoskopische Effekte zu erzeugen. Dafür werden zwei Ansichten von einer Szene generiert,

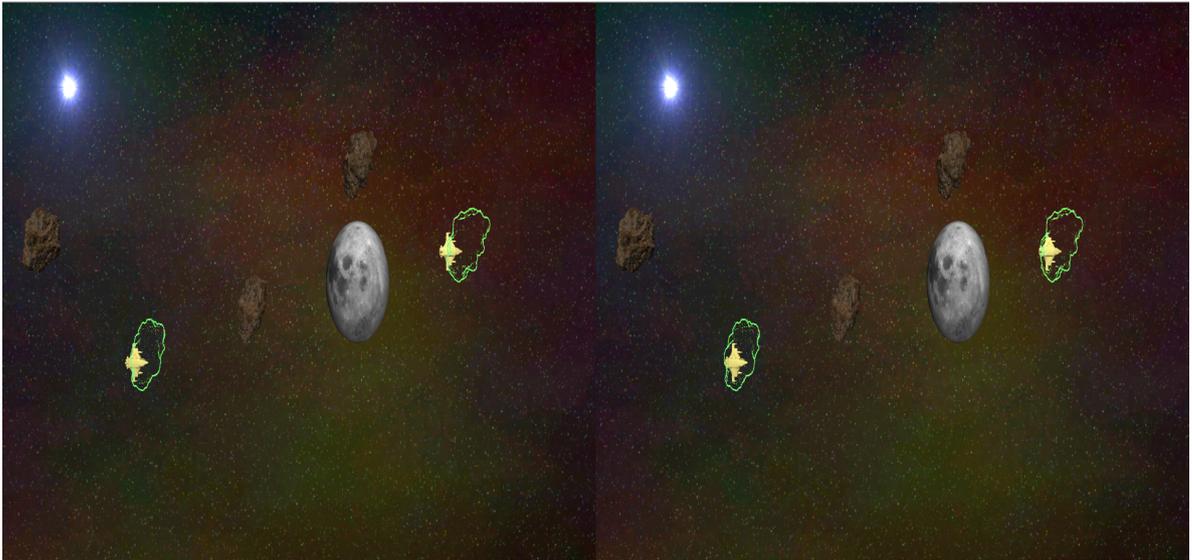


Abbildung 6.5: Side-by-Side Ansicht. Eine Szene des zweiten Teils des Spiels. Die Raumschiffe, die sich hinter den Meteoriten verstecken, fliegen von links nach rechts.

die sich leicht voneinander unterscheiden. Um diese zu realisieren, können entweder zwei Kameras in eine Szene gesetzt oder von einer Kamera aufgenommene Bilder in zwei Texturen gerendert werden.

Im Spiel werden Bilder im Side-by-Side-Format dargestellt. Dafür werden zwei leicht unterschiedliche Bilder für das linke und rechte Auge nebeneinander angezeigt (Vgl. Abbildung 6.5). Wenn der 3D-Modus des Bildschirms angeschaltet wird, werden diese Bilder überdeckt dargestellt. So können diese Bilder mithilfe spezieller Brillen stereoskopisch wahrgenommen werden. Um dem stereoskopischen Effekt in der Spiel-Engine zu erzeugen, wurde das frei verfügbare stereoskopix3D¹ Skript verwendet. Als 3D-Format wurde Side-by-Side mit der parallelen Kameras ausgewählt.

¹<http://forum.unity3d.com/threads/63874-stereo-3D-in-Unity-3D?p=416458#post416458>, letzter Zugriff am 20.10.2013

7 Evaluierung

Um die entwickelte Interaktionstechniken zu evaluieren, wurde eine Benutzerstudie durchgeführt. So wurde die Interaktionsmethoden unter verschiedenen Aspekten bewertet. Die eingeladenen Probanden sollten das in Unterkapitel 6.2.2 beschriebene Spiel im 2D- und 3D-Modus und unter der Verwendung unterschiedlicher Verfahren für das Zielen spielen. Das Spiel besteht aus zwei Teilen. In dem ersten Teil des Spiels sollen Meteoriten, die auf die Erde zufliegen, unter Verwendung des Blicks des Benutzers abgeschossen werden. Dieser Teil sollte im 2D- und 3D-Modus gespielt werden, um die blickbasierte Interaktion in einem 2D- und 3D-Display zu vergleichen. In dem zweiten Teil des Spiels sollen die Raumschiffe hinter Meteoriten abgeschossen werden, indem ein Benutzer diese fokussiert. Dieser Teil des Spiels sollte mit beiden Verfahren für das Zielen, also mit der Verwendung des Pupillendurchmessers und der Pupillendistanz, auf einem 3D-Display gespielt werden.

In diesem Kapitel wird die Benutzerstudie beschrieben. In Unterkapitel 7.1 wird auf die Hardware, die während der Evaluierung verwendet wurde, eingegangen. Der Aufbau der Benutzerstudie wird in Unterkapitel 7.2 beschrieben. In Unterabschnitt 7.3 wird der Ablauf des Versuchs vorgestellt. Zum Schluss wird in Unterkapitel 7.4 die Information über die Benutzerstudienteilnehmer gegeben. Die Evaluierung wurde zusammen mit Jonas Auda durchgeführt.

7.1 Verwendete Hardware

Für diese Arbeit wurde verschiedene Hardware genutzt. Diese sind ein 3D-fähige Display, ein hochperformanter Laptop für das Ansteuern des 3D-Displays, ein Laptop für den Server und ein Eyetracking-System für die Interaktion. Die technischen Daten der verwendeten Laptops und dem Display werden im Anhang A.1 angegeben. In diesem Abschnitt wird eine Übersicht über die während der Evaluierung verwendete Hardware gegeben.

7.1.1 Eye-Tracker: iView X RED

Für das Eyetracking wurde der iView X RED (Remote Eyetracking Device) verwendet. Der iView X RED ist ein stationärer, videobasierter Eyetracker, bei dem ein Benutzer keinen direkten Kontakt zum Gerät hat. Bei der Benutzung kann ein Benutzer entweder sitzen oder stehen, sollte sich aber nicht bewegen. Außer kleinere Kopfbewegungen kann jede Bewegung die Kalibrierung verschlechtern. Der iView X RED zeichnet die dunkle Pupille und den

Kornealreflex auf, um dem Blick des Benutzers zu verfolgen und kleine Kopfbewegungen auszugleichen.

7.1.2 Laptop: Sony VAIO VPC-F22C5E

Die Ausführung des Spiels, das für diese Arbeit entwickelt wurde, benötigt einen performanten Rechner. Für diesen Zweck wurde ein Sony VAIO Laptop verwendet. Dieser 16,4 Zoll Laptop verfügt über 8 GB Arbeitsspeicher, einen performanten Prozessor und eine leistungsfähige Grafikkarte.

7.1.3 Laptop: Lenovo ThinkPad X200

Während der Evaluierung wurde ein Lenovo ThinkPad verwendet, auf dem die iView X Software von SensoMotoric Instruments schon vorinstalliert war und das über ein USB-Kabel mit dem iView X RED Eyetracker verbunden war. Dieses Notebook ist nicht sonderlich performant, für unsere Aufgabe aber ausreichend. Außerdem ist dieses kleine Notebook ein Teil des verwendeten Eyetracking Systems.

7.1.4 Server

Um den Evaluierungsablauf zu beobachten und gegebenenfalls auftretende Probleme schnell zu beheben, läuft der Server auf einem externen Asus X83V Laptop. Der Server benötigt keinen performanten Rechner. Dieser muss lediglich mit dem Netzwerk verbunden sein.

7.1.5 Display: Philips 3D-Fernseher: 55PFL7606K/02

Für die stereoskopische Anzeige wurde ein 55 Zoll Philips 3D-Fernseher verwendet. Die Bilder vom Spiel-Laptop wurden über ein HDMI-Kabel übertragen. Während der Studie wurden 2D- und 3D-Modi des Bildschirms verwendet. Als Display-Auflösung wurde 1920 x 1080 bei 60 Hz ausgewählt. Um die stereoskopische Wahrnehmung in diesem Display zu ermöglichen, kann entweder die Side-by-Side- oder Over-and-Under-Technik verwendet werden. Die stereoskopischen Bilder können dann das Tragen einer Brille mit Polarisationsgläsern wahrgenommen werden.

7.2 Versuchsaufbau

Die Benutzerstudie wurde in einem Labor im Gebäude für Simulation Technology an der Universität Stuttgart durchgeführt. Der Laborraum wurde so ausgewählt, dass das Licht im Raum kontrolliert werden konnte. Dies war notwendig, um immer die gleiche Beleuchtung während der Benutzerstudie zu schaffen. Unterschiedliche Beleuchtung kann

dem Eyetracking Schwierigkeiten bereiten. In den Laborraum einfallendes Sonnenlicht wurde blockiert und der ganze Raum wurde durch vier Lampen gleichmäßig beleuchtet.

Die oben beschriebene Hardware wurde in den Laborraum gebracht und miteinander verbunden. Zusätzlich wurde noch eine Kopfstütze verwendet, damit die Probanden ihren Kopf während des Spielens gerade halten und nicht bewegen konnten. Die Probanden hatten nur physischen Kontakt mit der Maus, die mit dem Applikationsrechner verbunden war, um die Kalibrierung durchzuführen und den zweiten Spielteil zu spielen. Während des Versuchs sollten die Probanden nur auf den Display schauen. Die Laptops wurden so aufgestellt, dass die Bildschirme nur für die Studienleiter sichtbar waren. Der Abstand zwischen dem Display und dem Eyetracker betrug 171,5 cm. Bei einem kürzeren Abstand könnte der Eyetracker den Blick des Benutzers nicht erfassen, wenn er an den Rand des 55 Zoll Bildschirms schaute. Der Abstand zwischen Eyetracker und Proband betrug 60 cm. Im Übrigen mussten die Probanden während des Spielens eine Brille mit Polarisationsgläsern tragen, um das räumliche Sehen zu ermöglichen.

7.3 Versuchsablauf

Bei der Benutzerstudie hat jeder Proband etwa 45 Minuten an dem Versuch teilgenommen. Jeder Versuch begann mit einer kurzen Belehrung. Dazu wurde jedem Proband der Ablauf der Studie erklärt und es sollten demographische Informationen und die Erfahrung mit stereoskopischem Inhalt in einem Fragebogen angegeben werden (siehe Anhang A.2). Außerdem wurde verdeutlicht, dass der Benutzerstudie jederzeit abgebrochen werden konnte.

Nach der Belehrung wurde der Proband gebeten vor dem Eyetracker Platz zu nehmen, die Polarisationsbrille aufzusetzen und seinen Kopf auf der Kopfstütze abzulegen. Nun konnten die Studienleiter kontrollieren, ob die Augen des Probanden durch den Eyetracker gut aufgenommen werden konnten und den Kommunikationsserver in Bereitschaft versetzen. Vor dem Spielen wurden die in den Unterkapiteln 6.2.1 und 6.2.3 beschriebene Kalibrierungen durchgeführt. Falls die Kalibrierung für das Spielen nicht ausreichen genau war, konnte diese nochmal wiederholt werden.

Nach erfolgreicher Kalibrierung konnte die Studienteilnehmer spielen. Jeder Proband hat das Spielen mit unterschiedlichen Spielteilen begonnen. Hat der erste Teilnehmer den Versuch mit dem ersten Spielteil begonnen, hat der zweite Teilnehmer den Versuch mit dem zweiten Spielteil angefangen. Die Darstellungsart (2D/3D) für den ersten Spielteil und die Verfahren für das Zielen im zweiten Teil des Spiels wurden auch pro Studienteilnehmer abwechselnd genutzt. Hat ein Proband den zweiten Teil des Spiels mit der Benutzung des Pupillendurchmessers für die Tiefenerkennung eines Objekts begonnen, hat der nächste Proband den gleiche Spielteil unter der Verwendung der anderen Zielmethode gespielt. Außerdem wurden für die Benutzerstudie einige Änderungen am Spiel vorgenommen. Um Nutzerdaten sammeln zu können, wurde das Spiel nicht beispielsweise wie in Unterabschnitt 6.2.2 beschrieben nach zehn Einschlagen der Meteoriten auf die Erde beendet, sondern die Probanden konnten jedes Spiel drei Minuten lang spielen. Der Filter für die Augen Daten, der in Unterabschnitt 6.1.3 beschrieben wurde, wurde während der Evaluierung

nicht verwendet. Unter der Verwendung des Filters im ersten Spielteil ist es schwer, schnell auf die Meteoriten zu reagieren, wenn diese sich in verschiedenen Bereichen des Bildschirms befinden. Außerdem verlangsamt der Filter die Blickverfolgung des Benutzers. Durch die Glättung der Augendaten wird die zurückgegebene Blickposition nicht genau dem entsprechen, was ein Benutzer genau fokussiert. Im ersten Spielteil muss der Blick einmal einen Meteoriten treffen, dass dieser explodiert. Das Zittern bereitet hier keine Schwierigkeiten. Aber im zweiten Teil des Spiels ist es sinnvoll, den Filter zu benutzen. Denn die 3D-Objekte fliegen hier nur langsam von links nach rechts. Außerdem muss man sich auf die 3D-Objekt konzentrieren. Deswegen kann das Zittern in diesem Fall zu Problemen führen. Aber im zweiten Spielteil wurde auch der Filter ausgeschaltet, damit vergleichbare Daten gesammelt werden können.

Die Kalibrierung wurde normalerweise viermal für jeden Probanden durchgeführt, weil nach jedem Spielen unterschiedliche Fragebögen ausgefüllt werden sollten. Für den ersten Teil des Spiels wurden NASA-TLX (Task Load Index) [nas], SUS (System Usability Scale) [Bro96] und ein modifizierter PRC-Fragebogen (Product Reaction Cards) [BM02] verwendet. Der bei der Evaluierung genutzte PRC enthielt nur 52 Adjektive und einige davon sind nicht in originalen PRC enthalten. Nach dem Spielen des zweiten Spielteils wurden die Probanden gebeten nur den modifizierten PRC auszufüllen. Im Übrigen konnten die Studienteilnehmer Kommentare über die Interaktionsmethoden oder das Spiel in Form eines Freitextes geben. Die verwendeten Fragebögen sind in dem Anhang A.2 enthalten.

7.4 Teilnehmer

An der Benutzerstudie haben 17 freiwillige Probanden teilgenommen. Davon waren 16 männlich und ein Teilnehmer weiblich. Die Probanden waren im Alter zwischen 22 und 28 (*Durchschnitt* = 24,53, *Standardabweichung* = 1,62). Die Teilnehmer waren Studenten der Studiengänge Informatik, Softwaretechnik, Maschinenbau und Sozialwissenschaften. Nur ein Teilnehmer hatte bereits Erfahrung mit einem Eyetracker, aber alle Probanden hatten mindestens einmal mit stereoskopischen Inhalt zu tun. Neun Teilnehmer hatten kein beeinträchtigtes Sehvermögen, fünf Teilnehmer trugen eine Brille und ein Proband trug Kontaktlinsen. Die Probanden, die ein Brille trugen, trugen während der Benutzerstudie eine Brille mit Polarisationsgläser über ihrer einigen Brille. Ein Teilnehmer hatte eine Rot-Grün-Sehschwäche. 15 Teilnehmer haben das Experiment erfolgreich abgeschlossen. Die Daten von zwei Probanden wurden wegen Problemen bei der Kalibrierung und mögliche störender Umwelteinflüsse (falsche Raumbelichtung) nicht evaluiert. Aufgrund des gleichen Problems wurden die Daten eines Probanden des zweiten Spielteils ebenfalls nicht in die Auswertung hinein genommen.

8 Ergebnisse und Diskussion

In diesem Kapitel werden die Ergebnisse der Benutzerstudie vorgestellt. Außer Fragebögen werden auch die Daten, die während der Benutzerstudie erfasst wurden, ausgewertet. Die Resultate der Evaluierung der Interaktionstechniken durch Fragebögen und Datenerfassung sind jeweils in Abschnitt 8.1 und Abschnitt 8.2 gegeben. In Abschnitt 8.3 werden die Ergebnisse besprochen.

8.1 Die Ergebnisse der Fragebögen

Um die Usability der Interaktionstechniken zu bewerten, wurden die Probanden gebeten, nach jedem Spielen des ersten Teils des Spiels einen SUS-Fragebogen auszufüllen. Dieser Teil wurde zweimal - in 2D- und 3D-Modus - gespielt, damit die Usability der blickbasierten Interaktion mit einem herkömmlichen und einem stereoskopischen Display vergleichbar ist. Das Ergebnis ist in Tabelle 8.1 dargestellt. Durchschnittlich ist die SUS-Auswertung im 2D-Modus (*Durchschnitt* = 74,666, *Standardabweichung* = 11,796) etwas höher als im 3D-Modus (*Durchschnitt* = 74,333, *Standardabweichung* = 12,4809). Aber dieser Unterschied, -0,333, ist statistisch nicht signifikant, $t(14) = 0$, $p = 0,8463$. Dieses Ergebnis zeigt, dass die blickbasierte Interaktion mit einem 2D- und 3D-Fernseher im Bereich Usability in etwa gleich gut ist. Laut [BKM09] liegt die durchschnittliche SUS-Auswertung in akzeptablem Bereich.

	Minimum	Maximum	Durchschnitt	Median
2D	55	97,5	74,666	75
3D	57,5	95	74,333	75

Tabelle 8.1: SUS-Auswertung: Vergleich der Usability der blickbasierten Interaktion mit einem 2D- und stereoskopischen Bildschirm.

Außer des SUS-Fragebogens haben die Probanden noch den NASA-TLX-Fragebogen ausgefüllt, damit die subjektive Beanspruchung während der Benutzung der Interaktionstechnik bewertet wird. Während der Bewertung wurden keine Gewichte für die sechs Skalen verwendet und jede Skala kann Werte von 0 bis 20 annehmen. Die durchschnittliche Bewertung der subjektiven Beanspruchung bei der Verwendung der blickbasierten Interaktion mit 2D- und 3D-Bildschirmen werden in Abbildung 8.1 dargestellt. Als eine durchschnittliche Beanspruchung hat der NASA-TLX für die blickbasierte Interaktion mit 2D- und 3D-Bildschirmen keine signifikanten Unterschiede ergeben. Diese sind jeweils 53,4 und 53,33

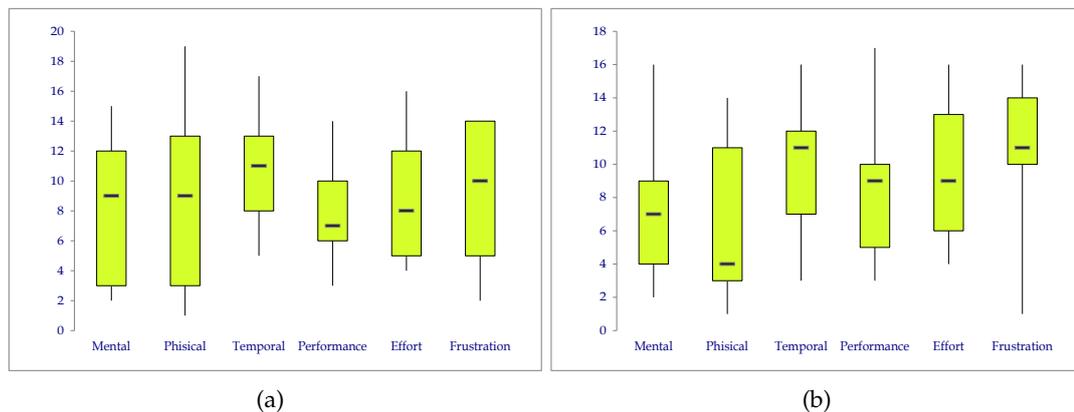


Abbildung 8.1: Die Ergebnisse von NASA-TLX. Die subjektive Beanspruchung während der Verwendung der blickbasierten Interaktion mit einem (a) 2D- und (b) 3D-Bildschirm.

von 120. Die drei Skalen des NASA-TLX-Fragebogens, geistiger und körperlicher Beanspruchung und Zeitdruck, stellen die Beanspruchungen dar, die auf einem Benutzer lastet [nas]. Diese Skalen sind bei der Interaktion mit einem 2D-Display höher als bei der Interaktion mit einem 3D-Display (*Durchschnitt* = 9,244, *Standardabweichung* = 1,538 in 2D und *Durchschnitt* = 8,177, *Standardabweichung* = 1,74 in 3D). Die restlichen drei Dimensionen sind Leistungsdruck, Mühe und Frustration, die die Beanspruchung im Zusammenhang mit der Interaktion eines Benutzers darstellen [nas]. Die Werte dieser Dimensionen sind höher bei der blickbasierte Interaktion mit einem 3D-Bildschirm (jeweils *Durchschnitt* = 8,555, *Standardabweichung* = 0,619 und *Durchschnitt* = 9,6, *Standardabweichung* = 1,006).

Nach jedem Spielen des ersten Teil des Spiels sollten die Probanden fünf Adjektive eines modifizierten PRCs auswählen, die die Erfahrung mit der Benutzung des Systems beschreiben. Ein PRC-Fragebogen enthält Adjektive mit positiver und auch negativer Bedeutung (Vgl. Anhang A.2). Bei der Evaluierung wurden einige Adjektive signifikant höher markiert. Viele Probanden haben das Spielen auf einem stereoskopischen Bildschirm unterhaltsamer (*Entertaining*) als auf einem 2D-Display gefunden. Das Adjektiv "schnell" (*Fast*) wurde mehrmals mit negativer Bedeutung ausgewählt. Der Grund dafür ist das Erscheinen eine hohen Anzahl an Meteoriten in höheren Wellen. Ein Proband musste in höheren Wellen schnellere Augenbewegungen machen, um alle Meteoriten abschießen zu können. Interessanterweise wurde dieses Adjektiv nach dem Spielen im 2D-Modus fünfmal, aber im 3D-Modus nur einmal ausgewählt. Wahrscheinlich wird die Geschwindigkeit eines 3D-Spiel auf einem stereoskopischen Bildschirm anders als auf einem 2D-Display wahrgenommen. Dies kann auch der Grund dafür sein, dass die Adjektive "kompliziert" (*Complex*) und "schwer zu benutzen" (*Difficult to use*) mehr nach dem Spielen im 2D-Modus und die Adjektive "einfach" (*Simple*) und "einfach zu benutzen" (*Easy to use*) mehr nach der Verwendung des Systems im 3D-Modus ausgewählt wurden. Außerdem war das System laut den ausgewählten Adjektiven im 3D-Modus optisch angenehmer (*Visually pleasant*).

Der gleiche PRC-Fragebogen wurde auch nach dem Spielen des zweiten Teil des Spiels mit unterschiedlichen Zieltechniken verwendet. Dieser Teil wurde auch mit den Wörtern "Spaß" (*Fun*) und "unterhaltsam" (*Entertaining*) hoch bewertet. Außerdem wurde dieser Teil mit beiden Zieltechniken als "anstrengend" (*Stressful*) beurteilt. Der Grund dafür kann sein, dass die Probanden für das Zielen sich auf die Raumschiffe, die hinter einem Meteoriten fliegen, konzentrieren sollten. Die Probanden haben die Zieltechnik, die den Abstand zwischen den Pupillen benutzt als kontrollierbarer (*Controllable*), als die Zieltechnik mit dem Pupillendurchmesser eingeschätzt.

8.2 Die Ergebnisse der Datenerfassung

Außer den Fragebögen wurden auch die während der Benutzerstudie erfassten Daten auf einige Kriterien evaluiert. Erstens wurde untersucht, ob es ein Zusammenhang zwischen der Anzahl der Meteoriten und Fehlerrate gibt. Die Fehlerrate wurde als die durchschnittliche Anzahl der Meteoriten, die auf der Erde eingeschlagen im Verhältnis zur Gesamtzahl an generierten Meteoriten pro Welle definiert. Pro Welle wurde die Anzahl der erscheinenden Meteoriten um eins erhöht. Um einen Zusammenhang zwischen der Fehlerrate und die Anzahl an Meteoriten pro Welle zu finden, wurde Spearman'sche Korrelationskoeffizient berechnet. Für den 2D- und 3D-Modus ergab die Berechnung einen positiven Zusammenhang (2D-Modus: $r_s = 0,615$, 3D-Modus: $r_s = 0,713$).

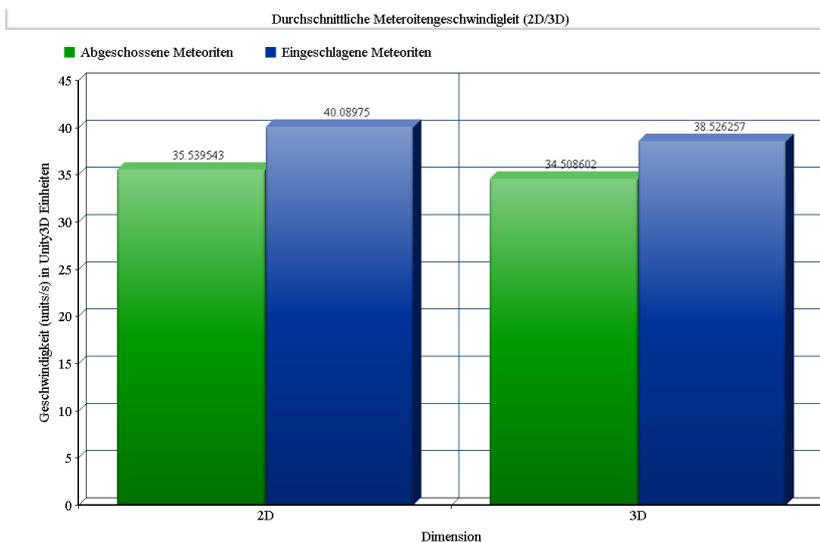


Abbildung 8.2: Durchschnittliche Geschwindigkeit von den auf der Erde eingeschlagenen und abgeschossenen Meteoriten.

Außerdem wurde überprüft, ob die Meteoriten, die auf der Erde einschlugen, durchschnittlich schneller als die abgeschossenen Meteoriten waren. In Abbildung 8.2 wird die durchschnittliche Geschwindigkeit der abgeschossenen und auf der Erde eingeschlagenen Meteoriten dargestellt. Die Geschwindigkeit ist in den Einheiten der verwendeten Spiel-Engine gegeben. Aus der Abbildung ist zu entnehmen, dass im 2D- und 3D-Modus die Meteoriten, die die Probanden nicht abschießen konnten, durchschnittlich schneller als die abgeschossenen Meteoriten waren.

Es wurden anhand der Daten die erreichten Spielpunkte verglichen, die im 2D- und 3D-Modus gesammelt wurden. Der durchschnittliche Spielpunktstand kann der Tabelle 8.2 entnommen werden. Die Probanden, die bei der Evaluierung teilgenommen haben, haben im 2D-Modus durchschnittlich besser als im 3D-Modus gespielt. Aber statistische signifikante Unterschiede konnten durch ein t Test nicht nachgewiesen werden, $t(14) = 0,907$, $p = 0,379$.

	Minimum	Maximum	Durchschnitt	Median	Standardabweichung
2D	22	96	56,13	55	19,36
3D	24	87	52,07	51	15,77

Tabelle 8.2: Durchschnittlicher Spielpunktstand: Vergleich des durchschnittlichen Spielpunktstands im 2D- und 3D-Modus.

In der Benutzerstudie wurde auch der zweiten Teil des Spiels evaluiert. Jeder Teilnehmer konnte den zweiten Teil des Spiels jeweils drei Minuten mit einer der beiden Zieltechniken spielen. Beobachtungen während des Spielvorgangs ließen darauf schließen, dass ein Benutzer der die Zieltechnik mit dem Pupillendurchmesser verwendet, einen bestimmten Lernprozess durchlaufen muss, um diese ordnungsgemäß zu benutzen. In der Benutzerstudie haben die Probanden durchschnittlich nach 102 Sekunden begonnen die hinteren Objekte (Raumschiffe) abzuschießen. Bis zu dieser Zeit konnten die Probanden meistens nur Meteoriten abschießen. Außerdem wurden die gesammelten Spielpunkte mit beiden Zieltechniken verglichen. Die durchschnittlichen Spielpunkte waren höher, wenn als Zieltechnik die Pupillendistanz verwendet wurde (jeweils *Durchschnitt* = 7,5, *Standardabweichung* = 5,945 und *Durchschnitt* = 2,285, *Standardabweichung* = 3,451). Die statistische signifikante Unterschiede konnte durch ein *Wilcoxon-Vorzeichen-Rang-Test* nachgewiesen werden, $Z = -2,134$, $p = 0,033$.

Bei der Verwendung der Zieltechnik, bei der die Pupillendistanz verglichen wurde, wurden einige Probleme festgestellt. Während der Benutzerstudie wurde entdeckt, dass beim Schauen auf eine Seite des Displays die berechnete Pupillendistanz klein und auf der anderen Bildschirmseite groß war. Falls ein Proband auf eine Seite des Bildschirms schaute, war die berechnete Distanz zwischen den Pupillen nah an einem Wert, der bei der Tiefenkalibrierung mit der in mit der positiver Parallaxe dargestellten Objekten berechnet wurde, was dazu führte, dass in diesem Bereich nur noch die Raumschiffe hinter den Meteoriten abgeschossen werden konnten, nicht aber die Meteoriten vor der Raumschiffe. Analog verhielt sich das

System auf dem anderen Bildschirmseite. Auf welchen Seiten nur noch die hintere Objekte abgeschossen werden konnte, verhielt sich für verschiedene Probanden unterschiedlich. Der Abstand zwischen den Pupillen wird nicht direkt durch den Eyetracker berechnet, sondern durch die Pupillenpositionen ausgerechnet, die der Eyetracker liefert. Es können andere Faktoren, wie die Helligkeit oder die verwendeten Polarisationsbrillen ein Grund sein, dass der Eyetracker keine korrekten Werte liefert.

8.3 Diskussion

Nach der Entwicklung einer Interaktionstechnik, die auf einem 2D- und stereoskopischen Bildschirm die Manipulation von Objekten durch den Blick ermöglicht, war es möglich Usability und die Anforderung an den Benutzer im 2D- und 3D-Modus zu bewerten und zu vergleichen. Durch die Evaluierung wurde gezeigt, dass die Benutzung der blickbasierten Interaktion mit einem 3D-Display die gleiche Usability wie mit einem 2D-Display haben kann. Außerdem waren die Anforderung an die Benutzer im 2D- und 3D-Fall nicht sonderlich unterschiedlich. Bei der Bewertung der User Experience haben Probanden die blickbasierte Interaktion mit einem stereoskopischen Display unterhaltsamer, optisch angenehmer und einfacher zu Benutzen als dies in einem herkömmlichen Bildschirm gefunden. Obwohl die Meteoriten sowohl im 2D-Modus, als auch im 3D-Modus die gleiche Geschwindigkeit hatten, haben die Nutzer dies in 3D-Fall langsamer als im 2D-Fall bewertet. Die Leistung (Spielpunktstand) der Probanden während der Spiels im 2D-Modus war höher als im 3D-Modus. Aber dieser Unterschied war nicht signifikant hoch.

Durch diese Erkenntnisse kann man sagen, dass die blickbasierte Interaktion mit einem stereoskopischen Bildschirm ähnlich verwendbar wie mit einem 2D-Display ist und keine zusätzliche Beanspruchung bei der Verwendung benötigt wird. Somit können die Aufgaben, die durch blickbasierte Interaktion mit einem 2D-Display bewältigt werden, ähnlich hohen Aufwand unter der Verwendung eines 3D-Bildschirms ausgeführt werden. Aber es ist wichtig, dass die Anzahl der interagierbaren Objekte und ihre Geschwindigkeiten angepasst werden. Bei der blickbasierten Interaktion mit einem stereoskopischen Display bereiten zu viele oder schnelle Objekte im 3D-Raum Schwierigkeiten. Außerdem wird die User Experience unter der Verwendung der blickbasierten Interaktion mit einem 3D-Display verbessert.

Im zweiten Teil des Spiels haben die Probanden eine bessere Leistung unter der Verwendung der Pupillendistanz gezeigt, als unter der Verwendung des Pupillendurchmessers. Aber wegen den Problemen, die in Unterkapitel 8.2 beschrieben wurden, kann diese Leistung nicht objektiv bewertet werden. Außerdem wurde entdeckt, dass für eine bessere Leistung unter der Verwendung des Pupillendurchmessers ein Nutzer einen bestimmten Lernprozess durchlaufend sollte, denn die Anzahl der erfolgreichen Abschüsse hat sich beim längeren Spielen erhöht. Wegen einem nicht stattgefundenen Lernprozess haben einige Probanden es nicht geschafft bei der Methode mit dem Pupillendurchmesser überhaupt einen Meteoriten abzuschießen. Im Übrigen war die Zielmethode mit dem Pupillendurchmesser zuverlässiger gegenüber Umwelteinflüssen, als die Methode, die den Pupillenabstand verwendete.

Während der Benutzerstudie wurde beobachtet, dass die meisten Probanden wegen der fehlenden Erfahrung mit einer solchen Interaktionsart, die Maus, die bei der Kalibrierung verwendet wurde, auf die Ziele bewegt haben. Es wurde auch bemerkt, dass die Bewegung der Maus während der Kalibrierung hilfreich sein kann. Vor der Bestätigung kann die Verschiebung der Maus auf die Kalibrierungspunkt die Genauigkeit der Kalibrierung erhöhen. In diesen Fällen haben die Probanden den Cursor genau über dem Kalibrierungspunkt bewegt, wenn diesen fixiert haben.

Einige Nutzer haben beklagt, dass die Blickverfolgung unpräzise war. Der Grund dafür ist, dass die Daten von dem Eyetracker nicht immer fehlerfrei sind. Außerdem konnten die benutzte Polarisationsbrillen auch der Blickerfassung Schwierigkeiten bereitet haben. Die Interaktion mit einem stereoskopischen Display war für einige Nutzer anstrengend. Zwei Probanden mit Sehhilfe war nach der Benutzerstudie schwindlig.

Viele Studienteilnehmer haben Rückmeldung gegeben, dass es schwer ist, sich auf eine Meteorit zu konzentrieren, wenn viele Meteoriten im Sichtfeld waren. In diesem Fall haben die Probanden versucht alle Meteoriten abzuschießen, indem sie alle beweglichen Objekte anschauten. Ist die Anzahl der Meteoriten im Sichtfeld klein, haben es alle Probanden geschafft, die vorbeifliegenden Meteoriten passieren zu lassen.

9 Zusammenfassung und Ausblick

Bis jetzt wurde in den meisten Arbeiten die blickbasierte Interaktion lediglich mit 2D-Bildschirmen untersucht. Das Ziel der hier vorgestellten Arbeit war die Implementierung und die Evaluierung der blickbasierten Interaktionsmethoden auf einem 3D-Display. Zu diesem Zweck wurde die grundlegende Theorie untersucht und eine Beispielapplikation - ein 3D-Spiel - für die Durchführung der Interaktion mit einem 3D-Display und einen Kommunikationsserver implementiert, der die Kommunikation zwischen der Beispielapplikation und dem Eyetracking-System realisiert. Bei der Evaluierung ergab sich, dass die vorgestellte blickbasierte Interaktion ohne zusätzliche Anforderungen an den Benutzer und mit einer ähnlichen Usability wie in einem 2D-Fall auf einem 3D-Display verwendet werden kann. Damit könnte man darauf schließen, dass die blickbasierte Interaktion mit einem 2D-Display auch auf einem 3D-Display durchgeführt werden kann. Somit können Benutzerschnittstellen für die blickbasierte Interaktion mit einem 3D-Display entwickelt werden, welche die Benutzer möglicherweise als optisch angenehmer und einfacher zu benutzen empfinden.

Außerdem wurde untersucht, ob es möglich ist, durch die blickbasierte Interaktion mit einem Objekt zu interagieren, das auf einer bestimmten Tiefebene auf dem Display dargestellt ist. Dafür wurden zwei Methoden entwickelt, deren Berechnungen auf dem Pupillendurchmesser und der Pupillendistanz basieren. Bei der Evaluierung ergab sich, dass die Methode mit der Pupillendistanz besser funktioniert als mit dem Pupillendurchmesser. Aber die Methode mit dem Pupillendurchmesser hat sich als zuverlässiger gegen Umweltstörfaktoren erwiesen, als die mit der Pupillendistanz. Unter Verwendung solcher Interaktionsmethoden könnten Schnittstellen für 3D-Anwendungen entwickelt werden, die auf Tiefebenen basieren. Im Kapitel 4 wurden einige Anwendungsfälle hierfür vorgestellt.

Ausblick

Der in dieser Arbeit vorgestellte Prototyp für die Interaktion mit 3D-Displays hat einige Einschränkungen. Der Versuchsraum für die Evaluierung (Benutzerstudie) wurde so gewählt, dass die Lichtverhältnisse im Raum gesteuert werden konnten. Denn die Änderung des Lichts im Raum könnte Schwierigkeiten bei der Blickfassung bereiten. Es lässt sich schlussfolgern, dass derartige Interaktionstechniken effektiv in solchen Umgebungen verwendet werden können, in denen die Lichtverhältnisse nicht stark variieren. Außerdem wurde für den Prototyp ein stereoskopischer Display mit Polarisationsbrillen verwendet. Diese Brillen können auch bei der Blickverfolgung Probleme bereiten. Für zukünftige Arbeit könnte der gleiche Prototyp unter Verwendung eines autostereoskopischen Displays evaluiert werden. Des Weiteren wurde bei der Studie eine Kopfstütze verwendet, um die Kopfbewegungen

der Probanden zu minimieren. Unter Verwendung von Head-Tracking könnte eine solche Kopfstütze vermieden werden.

Um die Tiefe von Objekten auf dem 3D-Display zu erkennen, wurden im Prototyp lediglich zwei Tiefenebenen verwendet. Für zukünftige Arbeiten könnten Prototypen entwickelt werden, welche die Anzahl der Ebenen erweitern, um diese Interaktionstechnik in komplexen Szenarien evaluieren zu können.

Danksagung

Zuletzt möchte ich mich an dieser Stelle bei allen bedanken, die mich bei dieser Bachelorarbeit unterstützt haben. Zunächst gilt mein Dank Florian Alt und Stefan Schneegaß für das Vertrauen und die Betreuung während der Arbeit. Außerdem möchte ich mich bei Jonas Auda bedanken für die sehr gute Zusammenarbeit und das Korrekturlesen meiner Arbeit. Außerdem gilt mein Dank Anna-Lena Janz, die ebenfalls meine Arbeit auf Fehlern untersucht hat. Auch allen Probanden möchte ich danken, die an der Benutzerstudie teilgenommen haben. Zum Schluss danke ich meine Eltern für die Unterstützung und Motivation während des Studiums.

A Ein Anhang

A.1 Technische Daten der verwendeten Hardware

Hier ist die technische Spezifikation der verwendeten Hardware angegeben. In Tabelle A.1 werden die technischen Daten des Applikationslaptops, des Kommunikationsservers und des Eyetracking-Laptops dargestellt. Tabelle A.2 stellt die technische Spezifikation des 3D-Fernsehers dar.

	Applikationslaptop	Kommunikationsserver	Eyetracking-Laptop
Hersteller	Sony	Asus	Lenovo
Bezeichnung	VPC-F22C5E	X83V	ThinkPad X200
Prozessor	8x2.0 GHz	2x2.0GHz	2,4GHz
Arbeitsspeicher	8 GB	4 GB	2 GB
Grafikkarte	NVIDIA GeForce GT 540M	NVIDIA GeForce 9300M GS	Mobile Intel 4 Series Express Chipset Family
Betriebssystem	Microsoft Windows 7	Microsoft Windows 7	Microsoft Windows XP

Tabelle A.1: Die technische Daten der verwendeten Laptops

	3D-Display
Hersteller	Philips
Bezeichnung	55PFL7606K/02
Bildschirmdiagonale	140 cm (55 Zoll)
Bildformat	16:9
3D	Easy3D
3D-Technologie	Polarisationstechnik
Display-Auflösung	bis zu 1920 x 1080 bei 60 Hz

Tabelle A.2: Die technischen Daten des 3D-Displays

A.2 Verwendete Fragebögen

Während der Evaluierung wurden die Probanden gebeten die in den folgenden Seiten dargestellten Fragebögen auszufüllen. Die Fragebögen sind in folgender Reihenfolge aufgelistet: Fragen zur Demographie und modifizierter PRC-Fragebogen (Product Reaction Cards) [BM02], SUS (System Usability Scale) [Bro96] und NASA-TLX (Task Load Index) [nas].



Fragebogen

A. Demographie

Geschlecht: männlich weiblich

Alter: _____ Beruf / Studiengang: _____

Besitzt du einen 3D-Fernseher?

Wenn ja, wie oft benutzt du die 3D-Funktionalität?

- täglich
- wöchentlich
- monatlich
- seltener / nie

B. User Experience

Choose five (5) words from the grid below which best describe your experience of using the system. If the word is followed by the sign (+/-), please mark whether you mean the word in a positive or negative sense.

Fast (+/-)	Useless	Useful	Inspiring
Rigid	Consistent	Too technical	Stressful
Clear	Uncontrollable	Businesslike (+/-)	Controllable
Unfamiliar (+/-)	Creative	Undesirable	Predictable (+/-)
Time-saving	Unpredictable	Familiar (+/-)	Time-consuming
Dated	Empowering	Slow (+/-)	Frustrating
Desirable	Unpleasant	Exciting	Entertaining
Dull	Fun (+/-)	Playful	Ordinary
Unapproachable	Inconsistent	Simple (+/-)	Restful
Responsive	Approachable	Serious	Complex
Difficult to use	Visually unpleasant	Innovative	Easy to use
Restrictive	Pleasant	Unclear	Boring
Visually pleasant	Poor quality	Novel	High quality

System Usability Scale

© Digital Equipment Corporation, 1986.

	Strongly disagree					Strongly agree
1. I think that I would like to use this system frequently	<input type="checkbox"/>					
	1	2	3	4	5	
2. I found the system unnecessarily complex	<input type="checkbox"/>					
	1	2	3	4	5	
3. I thought the system was easy to use	<input type="checkbox"/>					
	1	2	3	4	5	
4. I think that I would need the support of a technical person to be able to use this system	<input type="checkbox"/>					
	1	2	3	4	5	
5. I found the various functions in this system were well integrated	<input type="checkbox"/>					
	1	2	3	4	5	
6. I thought there was too much inconsistency in this system	<input type="checkbox"/>					
	1	2	3	4	5	
7. I would imagine that most people would learn to use this system very quickly	<input type="checkbox"/>					
	1	2	3	4	5	
8. I found the system very cumbersome to use	<input type="checkbox"/>					
	1	2	3	4	5	
9. I felt very confident using the system	<input type="checkbox"/>					
	1	2	3	4	5	
10. I needed to learn a lot of things before I could get going with this system	<input type="checkbox"/>					
	1	2	3	4	5	

Literaturverzeichnis

- [ADS05] M. Ashmore, A. T. Duchowski, G. Shoemaker. Efficient eye pointing with a fisheye lens. In *Proceedings of Graphics Interface 2005, GI '05*, S. 203–210. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. URL <http://dl.acm.org/citation.cfm?id=1089508.1089542>. (Zitiert auf den Seiten 25 und 26)
- [BAS12] N. Broy, E. André, A. Schmidt. Is stereoscopic 3D a better choice for information representation in the car? In *Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI '12*, S. 93–100. ACM, New York, NY, USA, 2012. doi:10.1145/2390256.2390270. URL <http://doi.acm.org/10.1145/2390256.2390270>. (Zitiert auf Seite 32)
- [BKM09] A. Bangor, P. Kortum, J. Miller. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. In *Journal of Usability Studies*, Band 4, S. 114–123. 2009. (Zitiert auf Seite 51)
- [BM02] J. Benedek, T. Miner. Measuring Desirability: New Methods for Evaluating Desirability in a Usability Lab Setting. In *Proceedings of UPA 2002 Conference*. 2002. (Zitiert auf den Seiten 50 und 60)
- [Bow13] D. A. Bowman. *3D User Interfaces*. In: Soegaard, Mads and Dam, Rikke Friis (eds.). "The Encyclopedia of Human-Computer Interaction 2nd Ed.", Aarhus, Denmark, 2013. URL http://www.interaction-design.org/encyclopedia/3d_user_interfaces.html. (Zitiert auf Seite 9)
- [Bro96] J. Brooke. SUS: A quick and dirty usability scale. In P. Jordan, B. Thomas, B. Weerdmeester, I. McClelland, Herausgeber, *Usability evaluation in industry*, S. 189–194. Taylor & Francis, London, 1996. (Zitiert auf den Seiten 50 und 60)
- [BSS13] G. Bruder, F. Steinicke, W. Sturzlinger. To touch or not to touch?: comparing 2D touch and 3D mid-air interaction on stereoscopic tabletop surfaces. In *Proceedings of the 1st symposium on Spatial user interaction, SUI '13*, S. 9–16. ACM, New York, NY, USA, 2013. doi:10.1145/2491367.2491369. URL <http://doi.acm.org/10.1145/2491367.2491369>. (Zitiert auf Seite 24)
- [CSP06] L. H. Cevidanes, M. A. Styner, W. R. Proffit. Image analysis and superimposition of 3-dimensional cone-beam computed tomography models. *Am J Orthod Dentofacial Orthop.*, 129(5):611–618, 2006. doi:10.1016/j.ajodo.2005.12.008. (Zitiert auf Seite 32)

- [DDLS07] H. Drewes, A. De Luca, A. Schmidt. Eye-gaze interaction for mobile phones. In *Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology, Mobility '07*, S. 364–371. ACM, New York, NY, USA, 2007. doi:10.1145/1378063.1378122. URL <http://doi.acm.org/10.1145/1378063.1378122>. (Zitiert auf Seite 32)
- [DLWDo7] A. De Luca, R. Weiss, H. Drewes. Evaluation of eye-gaze interaction methods for security enhanced PIN-entry. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces, OZCHI '07*, S. 199–202. ACM, New York, NY, USA, 2007. doi:10.1145/1324892.1324932. URL <http://doi.acm.org/10.1145/1324892.1324932>. (Zitiert auf den Seiten 7, 27 und 28)
- [Duc07] A. T. Duchowski. *Eye tracking methodology: Theory and practice*. Springer - Verlag London, 2. Auflage, 2007. (Zitiert auf den Seiten 13, 20, 21 und 22)
- [EG12] M. Elepfandt, M. Grund. Move it there, or not?: the design of voice commands for gaze with speech. In *Proceedings of the 4th Workshop on Eye Gaze in Intelligent Human Machine Interaction, Gaze-In '12*, S. 12:1–12:3. ACM, New York, NY, USA, 2012. doi:10.1145/2401836.2401848. URL <http://doi.acm.org/10.1145/2401836.2401848>. (Zitiert auf Seite 28)
- [HBCR11] M. Hachet, B. Bossavit, A. Cohé, J.-B. de la Rivière. Toucheo: multitouch and stereo combined in a seamless workspace. In *Proceedings of the 24th annual ACM symposium on User interface software and technology, UIST '11*, S. 587–592. ACM, New York, NY, USA, 2011. doi:10.1145/2047196.2047273. URL <http://doi.acm.org/10.1145/2047196.2047273>. (Zitiert auf Seite 23)
- [Holo5] N. Holliman. 3D Display Systems. 2005. (Zitiert auf den Seiten 14, 15, 16 und 17)
- [IHI⁺10] H. Istance, A. Hyrskykari, L. Immonen, S. Mansikkamaa, S. Vickers. Designing gaze gestures for gaming: an investigation of performance. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, ETRA '10*, S. 323–330. ACM, New York, NY, USA, 2010. doi:10.1145/1743666.1743740. URL <http://doi.acm.org/10.1145/1743666.1743740>. (Zitiert auf Seite 28)
- [Iso00] P. Isokoski. Text input methods for eye trackers using off-screen targets. In *Proceedings of the 2000 symposium on Eye tracking research & applications, ETRA '00*, S. 15–21. ACM, New York, NY, USA, 2000. doi:10.1145/355017.355020. URL <http://doi.acm.org/10.1145/355017.355020>. (Zitiert auf Seite 27)
- [Jac95] R. J. K. Jacob. Eye tracking in advanced interface design. In W. Barfield, T. A. Furness, III, Herausgeber, *Virtual environments and advanced interface design*, S. 258–288. Oxford University Press, Inc., New York, NY, USA, 1995. URL <http://www.cs.tufts.edu/~jacob/papers/barfield.pdf>. (Zitiert auf den Seiten 10, 14, 20, 21, 22, 24 und 25)

- [JK03] R. J. K. Jacob, K. S. Karn. Eye Tracking in Human–Computer Interaction and Usability Research: Ready to Deliver the Promises. In J. Hyönä, R. Radach, H. Deubel, Herausgeber, *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research*, S. 573–605. Elsevier Science, Amsterdam, The Netherlands, 2003. (Zitiert auf den Seiten 25 und 29)
- [KGBW07] M. Kumar, T. Garfinkel, D. Boneh, T. Winograd. Reducing shoulder-surfing by using gaze-based password entry. In *Proceedings of the 3rd symposium on Usable privacy and security*, SOUPS '07, S. 13–19. ACM, New York, NY, USA, 2007. doi:10.1145/1280680.1280683. URL <http://doi.acm.org/10.1145/1280680.1280683>. (Zitiert auf Seite 27)
- [KPW07] M. Kumar, A. Paepcke, T. Winograd. EyePoint: practical pointing and selection using gaze and keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, S. 421–430. ACM, New York, NY, USA, 2007. doi:10.1145/1240624.1240692. URL <http://doi.acm.org/10.1145/1240624.1240692>. (Zitiert auf Seite 27)
- [KT04] F. L. Kooi, A. Toet. *Visual Comfort of Binocular and 3-D Displays*, Kapitel 254, S. 1–14. 2004. doi:10.1081/E-EOE-120024169. URL <http://www.tandfonline.com/doi/abs/10.1081/E-EOE-120024169>. (Zitiert auf Seite 15)
- [Kum07] M. Kumar. GUIDe Saccade Detection and Smoothing Algorithm. Technical report cstr 2007-03, Stanford University, Stanford, 2007. (Zitiert auf Seite 40)
- [Lan00] C. Lankford. Effective eye-gaze input into Windows. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, ETRA '00, S. 23–27. ACM, New York, NY, USA, 2000. doi:10.1145/355017.355021. URL <http://doi.acm.org/10.1145/355017.355021>. (Zitiert auf Seite 26)
- [McA02] D. F. McAllister. Display Technology: Stereo & 3D Display Technologies. *Wiley Encyclopedia on Imaging, Jan.*, S. 1327–1344, 2002. (Zitiert auf den Seiten 14 und 17)
- [MVA⁺12] O. Mealha, A. Veloso, S. Almeida, R. Rodrigues, L. Roque, R. Marques, C. Mantigueiro. Eye Tracking Data Representation and Visualization: on Information and Communication studies at CETAC.MEDIA. *Journal of Eye tracking, Visual Cognition and Emotion*, 2(1):65–75, 2012. (Zitiert auf Seite 29)
- [MvM04] D. Miniotas, O. Špakov, I. S. MacKenzie. Eye gaze interaction with expanding targets. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '04, S. 1255–1258. ACM, New York, NY, USA, 2004. doi:10.1145/985921.986037. URL <http://doi.acm.org/10.1145/985921.986037>. (Zitiert auf Seite 26)
- [MZ05] M. Manhartsberger, N. Zellhofer. Eye tracking in usability research: What users really see. In *Empowering Software Quality: How Can Usability Engineering Reach These Goals? In Usability Symposium 2005*, Band 198, S. 141–152. OCG publication, 2005. (Zitiert auf Seite 29)

- [nas] NASA Task Load Index (NASA-TLX): A paper and pencil package, Version 1.0. Human Performance Research Group, NASA Ames Research Center, Moffett Field, CA. (Zitiert auf den Seiten 50, 52 und 60)
- [Nie93] J. Nielsen. Noncommand user interfaces. *Commun. ACM*, 36(4):83–99, 1993. doi:10.1145/255950.153582. URL <http://doi.acm.org/10.1145/255950.153582>. (Zitiert auf Seite 25)
- [NS86] K. Nakayama, G. H. Silverman. Serial and parallel processing of visual feature conjunctions. *Nature*, 320(6059):264–265, 1986. doi:10.1038/320264a0. URL <http://dx.doi.org/10.1038/320264a0>. (Zitiert auf Seite 31)
- [NSSLo9] L. Nacke, S. Stellmach, D. Sasse, C. A. Lindley. Gameplay Experience in a Gaze Interaction Game. In *The 5th Conference on Communication by Gaze Interaction – COGAIN 2009: Gaze Interaction For Those Who Want It Most*, S. 49–54. The COGAIN Association, Lyngby, Denmark, 2009. (Zitiert auf Seite 28)
- [OTK13] K. Özacar, K. Takashima, Y. Kitamura. Direct 3D object manipulation on a collaborative stereoscopic display. In *Proceedings of the 1st symposium on Spatial user interaction, SUI '13*, S. 69–72. ACM, New York, NY, USA, 2013. doi:10.1145/2491367.2491374. URL <http://doi.acm.org/10.1145/2491367.2491374>. (Zitiert auf Seite 24)
- [PB05] A. Poole, L. J. Ball. Eye Tracking in Human-Computer Interaction and Usability Research: Current Status and Future Prospects. In C. Ghaoui, Herausgeber, *Encyclopedia of Human Computer Interaction*, S. 211–219. Idea Group, Pennsylvania, 2005. (Zitiert auf den Seiten 29 und 30)
- [RHFL10] S. Reichelt, R. Häussler, G. Fütterer, N. Leister. Depth cues in human visual perception and their realization in 3D displays. In *SPIE Defense, Security, and Sensing*, S. 76900B–76900B. International Society for Optics and Photonics, 2010. doi:10.1117/12.850094. (Zitiert auf den Seiten 15 und 44)
- [SA00] D. D. Salvucci, J. R. Anderson. Intelligent gaze-added interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems, CHI '00*, S. 273–280. ACM, New York, NY, USA, 2000. doi:10.1145/332040.332444. URL <http://doi.acm.org/10.1145/332040.332444>. (Zitiert auf Seite 26)
- [SAHT10] J. San Agustin, J. P. Hansen, M. Tall. Gaze-based interaction with public displays using off-the-shelf components. In *Proceedings of the 12th ACM international conference adjunct papers on Ubiquitous computing - Adjunct, Ubicomp '10 Adjunct*, S. 377–378. ACM, New York, NY, USA, 2010. doi:10.1145/1864431.1864444. URL <http://doi.acm.org/10.1145/1864431.1864444>. (Zitiert auf Seite 32)
- [Sen09] SensoMotoric Instruments. *iView XTM System Manual*, 2009. (Zitiert auf Seite 36)
- [SG06] J. D. Smith, T. C. N. Graham. Use of eye movements for video game control. In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology, ACE '06*. ACM, New York, NY, USA, 2006. doi:10.1145/

- 1178823.1178847. URL <http://doi.acm.org/10.1145/1178823.1178847>. (Zitiert auf Seite 28)
- [SHSKo8] F. Steinicke, K. Hinrichs, J. Schöning, A. Krüger. Multi-touching 3D data: Towards direct interaction in stereoscopic display environments coupled with mobile devices. In *Advanced Visual Interfaces (AVI) Workshop on Designing Multi-Touch Interaction Techniques for Coupled Public and Private Displays*, S. 46–49. Citeseer, 2008. (Zitiert auf Seite 10)
- [UCES11] H. Urey, K. Chellappan, E. Erden, P. Surman. State of the Art in Stereoscopic and Autostereoscopic Displays. *Proceedings of the IEEE*, 99(4):540–555, 2011. doi:10.1109/JPROC.2010.2098351. (Zitiert auf Seite 18)
- [VGH12] D. Valkov, A. Giesler, K. Hinrichs. Evaluation of depth perception for touch interaction with stereoscopic rendered objects. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces, ITS '12*, S. 21–30. ACM, New York, NY, USA, 2012. doi:10.1145/2396636.2396640. URL <http://doi.acm.org/10.1145/2396636.2396640>. (Zitiert auf Seite 24)
- [vM05] O. Špakov, D. Miniotas. Gaze-based selection of standard-size menu items. In *Proceedings of the 7th international conference on Multimodal interfaces, ICMI '05*, S. 124–128. ACM, New York, NY, USA, 2005. doi:10.1145/1088463.1088486. URL <http://doi.acm.org/10.1145/1088463.1088486>. (Zitiert auf Seite 26)
- [YS75] L. Young, D. Sheena. Survey of eye movement recording methods. *Behavior Research Methods & Instrumentation*, 7(5):397–429, 1975. doi:10.3758/BF03201553. URL <http://dx.doi.org/10.3758/BF03201553>. (Zitiert auf den Seiten 21 und 22)
- [ZMI99] S. Zhai, C. Morimoto, S. Ihde. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems, CHI '99*, S. 246–253. ACM, New York, NY, USA, 1999. doi:10.1145/302979.303053. URL <http://doi.acm.org/10.1145/302979.303053>. (Zitiert auf Seite 26)

Alle URLs wurden zuletzt am 20. 10. 2013 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift