

Virtual Reality für Assistenzsysteme in der Bildauswertung

MASTERARBEIT

KIT - KARLSRUHER INSTITUT FÜR TECHNOLOGIE
FRAUNHOFER IOSB - FRAUNHOFER-INSTITUT FÜR OPTRONIK,
SYSTEMTECHNIK UND BILDAUSWERTUNG

Julien Christophe René Hoffmann

27.04.2017

Erstgutachter: Prof. Dr.-Ing. Jürgen Beyerer
Zweitgutachter: Prof. Dr. rer. nat. H. Steusloff
Betreuender Mitarbeiter: Dipl.-Inf. Alexander Streicher

Zusammenfassung

Gegenstand dieser Arbeit ist die Konzeption und Entwicklung eines Assistenzsystems für die Bildauswertung in der virtuellen Realität (VR). Die Herausforderungen hierbei sind die nicht-standardisierten Interaktionskonzepte, die Gestaltung der Immersion, die Vermeidung des Auslösens von Übelkeit durch die Nutzung von VR (VR-Krankheit), die fragwürdige subjektiv wahrgenommene Bildqualität von aktuellen VR-Systemen und der Umgang mit hochaufgelösten Luftbildaufnahmen in VR. Aus diesem Grund wurde ein systematisches Experiment zur Untersuchung der auftretenden visuellen Artefakte und zur Einschätzung der subjektiv wahrgenommenen Bildqualität vorgenommen. Anschließend wurden Interaktionskonzepte von existierenden VR-Anwendungen untersucht und eigene Interaktionskonzepte erstellt. Letztlich wurden mit Hilfe vom *Web Map Service*-Protokoll die Darstellung von hochaufgelösten Luftbildaufnahmen in VR ermöglicht. Aufbauend auf den gewonnenen Erkenntnissen wurde im weiteren Verlauf der Prototyp *Virtual Reality Image Exploitation Workspace (VIEW)* in der Spiel-Engine *Unity3D* erstellt. Dieser wurde in einer Studie mit 16 Teilnehmern evaluiert. Hierbei lagen die Schwerpunkte auf den Untersuchungen zu der erzielten Immersion, Intuitivität und Gebrauchstauglichkeit. Obwohl einige Aspekte der Implementierung noch verbessert werden sollten, konnte gezeigt werden, dass *VIEW* für die Bildauswertung eine gebrauchstaugliche, intuitive und weitestgehend immersive Umsetzung ist.

Abstract

The subject of this thesis is the conception and development of an assistance system for image exploitation in virtual reality (VR). The challenges consist of the non-standardized interaction concepts, the design of the immersion, the avoidance of the nausea caused by VR (VR-sickness), the questionable subjectively perceived image quality of current VR systems and the handling of high-resolution aerial imagery in VR. For this reason, a systematic experiment was carried out to investigate the visual artefacts and to assess the subjectively perceived image quality. Subsequently, interaction concepts of existing VR applications were examined and new interaction concepts were designed. Finally, using the *Web Map Service* protocol, high-resolution aerial images were brought to VR. Building upon the gained knowledge the prototype *Virtual Reality Image Exploitation Workspace (VIEW)* was build in the game engine *Unity3D*. The prototyp was evaluated in a study with 16 participants. The focus was the investigations of the achieved immersion, intuitiveness and usability. Although some aspects of the implementation should be improved, it has been shown that VIEW is a useful, intuitive and to a great extent immersive tool for image exploitation.

Erklärung der Selbstständigkeit

Ich versichere hiermit wahrheitsgemäß, die Arbeit bis auf die dem Aufgabensteller bereits bekannte Hilfe selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderung entnommen wurde.

Karlsruhe, den 27.04.2017

Julien Christophe René Hoffmann

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Projektumfeld	2
1.3	Problemstellung und Herausforderung	2
1.4	Lösungsansätze	2
1.5	Ziele dieser Arbeit	3
1.6	Umfang und Rahmenbedingungen	3
1.7	Gliederung	4
2	Stand der Forschung und Technik	5
2.1	Einsatzgebiete von VR	5
2.2	Assistenzsysteme und Bildauswertung	7
2.3	Mensch-Maschine Interaktionen in der virtuellen Realität	7
2.4	Menschliche Faktoren - Immersion und VR-Krankheit	11
3	Grundlagen	15
3.1	VR-Technologien im Überblick	15
3.2	Immersion und Mensch-Maschine Interaktion	19
3.3	Assistenzsysteme für die Bildauswertung	22
3.4	Wahrgenommene Qualität von Bildausgabegeräten	24
3.5	Software, Frameworks und Engines	27
4	Bildqualität in VR	29
4.1	Effekte und Probleme	29
4.2	Vergleich der aktuellen Hardware	31
4.3	Bewertung zur wahrgenommenen Bildqualität in VR	33
4.4	Diskussion	37
4.5	Fazit	37
5	Gestaltung von Immersion und Interaktion in VR	39
5.1	Immersion in die virtuelle Realität	39
5.2	Interaktionskonzepte für die Bildauswertung in VR	43
6	VR-Assistenzsystem für die Bildauswertung: VIEW	51
6.1	Vorgehensmodell für die Realisierung	51
6.2	Genutzte Software und Schnittstellen	52
6.3	Umgang mit hochaufgelösten Luftbildaufnahmen	54
6.4	Gestaltung vom virtuellen Raum	56
6.5	Umsetzung der Interaktionskonzepte	63
6.6	Anwendungsbeispiel	68

7	Evaluierung von VIEW	69
7.1	Versuchsaufbau	69
7.2	Versuchsdurchführung	82
7.3	Bewertung	84
7.4	Diskussion	94
7.5	Fazit der Evaluation	97
8	Fazit und Ausblick	99
	Literaturverzeichnis	101
	Abbildungsverzeichnis	109
	Tabellenverzeichnis	111
	Appendix A: Fragebögen der Evaluation	113
1	Fragebogen zur Person	113
2	NASA Raw TLX	114
3	Presence	115
4	SUS	117
5	Zusammenfassung der Hypothesen	118
	Appendix B: Studien Ergebnisse	119
1	Korrelationen und Regressionsanalyse	119

Glossar

Abkürzung	Begriff	Erklärung
FOV	Field Of View	Winkel bezüglich der Breite des Sichtfelds
FPS	Frames Per Second	Bilder pro Sekunde - gibt an wie oft ein neues Bild an ein Anzeigegerät gesendet wird
Hz	Hertz	Einheit, äquivalent zu <i>pro Sekunde</i> - gibt beispielsweise an, wie oft ein Anzeigegerät eines neues Bild darstellen kann
LCD	Liquid Crystal Display	Anzeigegerät welches bipolare Filmscheiben aus Glas nutzt. Sie sind leicht und transmissiv oder reflektierend und werden oft in HMDs eingesetzt
UI	User Interface	Benutzerschnittstelle
GUI	Graphical User Interface	Grafische Benutzerschnittstelle
WIMP	Windows Icons Menus Pointers	WIMP ist ein Interaktionskonzept, bei welchem Fenster, Symbole, Menüs und Zeiger genutzt werden
-	Usability	Benutzbarkeit - auch Gebrauchstauglichkeit - in DIN EN ISO 9126 definiert
UX	User Experience	Die Gesamterfahrung, die eine Person bei der Nutzung eines Produktes erfährt
-	Propriozeption	Eigenempfindung, welche die Wahrnehmung eigener Körperbewegungen oder der Lage einzelner Körperteile zueinander, ermöglicht
-	Controller	Eingabegerät, welches unterschiedliche Formen und Funktionen haben kann
HMD	Head Mounted Display	Typischerweise ein Helm, mit zwei integrierten Anzeigegeräten, welche direkt vor den Augen platziert sind. Erlaubt das wahrnehmen von stereoskopischen Bildern
VR	Virtual Reality	Das Konzept der Erstellung einer <i>betretbaren</i> virtuellen Welt
-	Head tracking	Ein VR-System welches Kopfbewegungen in die virtuelle Welt übernimmt

-	Positional tracking	Ein VR-System welches Körperbewegungen in die virtuelle Welt übernimmt
-	Room-scale VR	Ein VR-System, welches <i>head tracking</i> , <i>positional tracking</i> und mehr, erlaubt
-	Seated VR	Eine eingeschränkte Form von room-scale VR. Es werden nur Bewegungen in einem bestimmten Bereich erkannt.
-	Presence	Das Gefühl in der virtuellen Umgebung präsent zu sein
-	Motion Sickness	Übelkeit, welche durch Bewegungen in der virtuellen Welt ausgelöst werden können
-	Simulator Sickness	Allgemeiner Begriff welcher, durch Computer Simulationen ausgelöste unangenehme Nebeneffekte, beschreibt
-	VR-Sickness	VR-Krankheit: Allgemeiner Begriff welcher, durch VR ausgelöste unangenehme Nebeneffekte, beschreibt

-	MipMapping	Beim Mipmapping handelt es sich um ein Verfahren, bei dem hochaufgelöste Bilder in mehrere gleichgroße Kacheln zerlegt werden werden
LOD	Level of Detail	Wurde bei einem Bild Mipmapping, oder ein ähnliches Verfahren durchgeführt, so werden mit dem LOD die generierten Schichten unterscheidet
-	Tile	Eine Tile ist ein rechteckiger Ausschnitt eines größeren Bildes
-	Slippy Map	Slippy Map ist eine Darstellungsart von Karten, bei denen Tiles der Karte dynamisch geladen werden
	Georeferenzierung	Georeferenzierung ist der Prozess bei dem reale Koordinaten zu jedem Pixel eines Bildes referenziert werden
WMS	Web Map Service	WMS ist ein standardisiertes Protokoll für die Bereitstellung von georeferenzierten Aufnahmen
OSM	Open Street Map	Im Gegensatz zu WMS werden Karten nicht als Rastergrafik geliefert, sondern in Form von Vektordaten

VIEW	Virtual Reality Image Exploitation Workspace	Bezeichnung für den bei dieser Arbeit entwickelten Prototypen
-	InteractionCube	Teil von VIEW: Würfel-förmiges Objekt welches dem Benutzer Zugriff auf mehrere Befehle gibt
-	MapDisplay	Teil von VIEW: Bezeichnung für die Leinwand auf der die Karte angezeigt wird

Einleitung

Obwohl Virtual Reality (VR) bereits seit Jahren in der Forschung bekannt ist, blieb diese Technologie für die Allgemeinheit außer Reichweite. Laut Palmer Luckey, dem Gründer der US-amerikanischen Computertechnologiefirma *Oculus VR Inc.*, war die Herausforderung für den Massenmarkt der Mangel an Rechenkraft der derzeitigen Computer. Luckey gibt zudem an, dass die weniger präzisen Bewegungserkennung und der äußerst teure Herstellungs- und Entwicklungsprozess der VR-Headsets dazu beigetragen haben [Gre14].

Trotz dieser Herausforderungen werden sowohl im Militär als auch in der Medizin bereits seit Jahren Anwendungsfälle für VR entwickelt. Im Militär wird VR bereits seit längerem zum Trainieren von Soldaten genutzt [VRSb]. Dies beinhaltet Training in Flug-, Kampf- und Fahrzeugsimulatoren. Außerdem wird VR in der Medizin zum Training des Umgangs mit Robotersystemen für die Operation genutzt [Bri16].

Durch Firmen wie *HTC*, *Valve*, *Oculus*, *Sony*, *Google* und *Samsung* wird das Phänomen VR nun erschwinglich für die Massen angeboten.

1.1 Motivation

Besonders interessant an VR ist die Art, wie virtuelle Räume dreidimensional dargestellt werden und wie diese erkundet werden können. Mit Hilfe von Head- und Hand-Tracking werden unterschiedlichen Bewegungen des Benutzers direkt in den virtuellen Raum übernommen. Die US-amerikanische Softwarefirma *Valve* gehen zusammen mit dem taiwanesischen Konsumentenelektronik-Hersteller *HTC* mit ihrer *Vive* [HTC] noch einen Schritt weiter. Sie erlaubt neben der Erkennung von Kopf- und Handbewegungen auch die Positionserkennung im Raum. Die Gehbewegungen des Benutzers werden somit als solche im virtuellen Raum übernommen. Diese Freiheit wird als *room-scale* VR bezeichnet [Nic16].

Prashanth Shanmugam beschreibt diese Weiterentwicklung als Teil des VR-Prozesses [Sha15] (siehe Abb. 1.1). Mit der Steigerung der Interaktionsmöglichkeiten steigt auch die Immersion, die der Benutzer empfindet. Eine erhöhte Immersion erlaubt Benutzern, ihre Aufmerksamkeit auf die Durchführung ihrer Aufgabe zu richten [Igr15].



Abbildung 1.1: Der »VR Prozess« nach Prashanth Shanmugam(UXNESS - 2015) [Sha15]

1.2 Projektumfeld

Das Projektumfeld der Arbeit ist das Fraunhofer IOSB-Projekt *Intelligente bildgestützte Aufklärung in verteilten Sensorsystemen* (IBIS). IBIS wurde mit einem weiteren Projekt namens *enViRonment* kombiniert. Hierbei ist die Vision der »Bildauswerteplatz in VR«. Teilprojekte wie das Serious Game *Lost Earth 2307* und die Assistenzsysteme *DigLT.VR* und *ViLand* sollen in VR zusammengeführt werden, um somit ein komplettes VR-Erlebnis zu bieten, welches es dem Bildauswerter erlaubt, seine Aufgabe im virtuellen Raum auszuführen.

Der Fokus dieser Arbeit liegt auf der Konzeption von *ViLand* für VR. *ViLand* ist ein Hilfsmittel für Bildauswerter, welches die Betrachtung von Luftbildaufnahmen ermöglicht und zusätzliche Funktionen zum Anpassen von Bildwerten und zum Annotieren von Bildbereichen bietet.

1.3 Problemstellung und Herausforderung

Bei der Kombination von Bildauswertung und Virtual Reality gibt es eine Reihe von Schwierigkeiten, auf die geachtet werden muss. Eine Hauptproblematik liegt darin, dass bei der Bildauswertung ein sehr hoher Anspruch auf die Bildqualität gesetzt wird. Dahingegen liegt eine der Hauptschwächen von VR zurzeit darin, die subjektiv wahrgenommene Bildqualität deutlich niedriger ist als die, die bei herkömmlichen Monitoren wahrgenommen wird. Die VR-Bildausgabegeräte sind zwar bezüglich Auflösung und Pixelfarbtiefe ähnlich performant wie Desktopmonitor, da diese jedoch deutlich näher vom Benutzer betrachtet werden, fallen sogar die Abstände zwischen den einzelnen Pixeln merklich auf.

Außerdem können hochaufgelöste Aufnahmen nicht beliebig als Textur in VR-Anwendungen importiert werden. Derzeitige *Game Engines* und aktuelle Grafikkarten erlauben eine maximale Texturauflösung von 8k x 8k (bzw. 16k x 16k in einigen Fällen). Luftbildaufnahmen können jedoch diese Werte deutlich übertreffen.

Zudem kommt die Problematik, dass für VR derzeit noch keine festgelegten, universell genutzten Interaktionskonzepte existieren. Jeder Entwickler muss für sich herausfinden, welche Konzepte für seine Anwendung angebracht sind. Hierbei ist zu beachten, dass obwohl es keine festgelegten Interaktionskonzepte gibt, dennoch Paradigmen existieren, die zu befolgen sind. Besonders bei der Lokomotion – also der Fortbewegungsart des Benutzers im virtuellen Raum – sind eine Reihe von Faktoren zu beachten. Wird diese unachtsam durchgeführt, so kann es beim Benutzer zu Symptomen der VR-Krankheit kommen. Dies beinhaltet Schwindelgefühle, Übelkeit oder sogar Erbrechen.

1.4 Lösungsansätze

In erster Linie werden, im Hinblick auf die Interaktion, unterschiedliche Interaktionskonzepte entwickelt und miteinander verglichen. Zudem wird untersucht, wie die Interaktion in heutigen Anwendungen und Spielen umgesetzt werden.

Anschließend wird die subjektiv wahrgenommene Bildqualität von der HTC Vive untersucht. Hier liegt der Fokus auf der Untersuchung der potentiellen Artefakte die auftreten können.

Außerdem wird untersucht, wie hochaufgelöste Aufnahmen in VR realisiert werden können. Hierbei ist ein möglicher Lösungsansatz die Nutzung von sogenannten *MegaTexture-Frameworks*, welche das Laden von extrem hochaufgelösten Texturen erlaubt. Ein weiterer Lösungsansatz ist die Nutzung von MapServer-Software. Hierbei werden die hochaufgelösten Aufnahmen in Teilbereiche – sogenannten Tiles – geschnitten. Diese werden dann nach Bedürfnis dynamisch geladen.

1.5 Ziele dieser Arbeit

Bei dieser Arbeit soll analysiert und getestet werden, ob und wie Virtual Reality und Bildauswertung zueinander finden können. Des Weiteren soll untersucht werden, welche Vor- und Nachteile eine solche Art der Interaktion bieten kann. Hierbei wird zum einen untersucht, wie die Faktoren Immersionssteigerung, Interaktion und Gebrauchstauglichkeit ins Geschehen einspielen. Zum Anderen werden die Grenzen der VR-Bildqualität für die Bildauswertung genauer untersucht. Darauf aufbauend, werden die zur Zeit bekannten VR-Paradigmen auf die Bildauswertung angewandt und es wird untersucht, ob weitere intuitive Interaktionsmöglichkeiten umsetzbar sind.

Eine zentrale Fragestellung dabei ist, was die Grenzen bei der Darstellung von Bildern und Karten in VR sind. Des Weiteren wird untersucht, wie die einzelnen Komponenten der derzeitigen Desktop-Anwendung für die Kartenauswertung (ViLand) nach VR portiert werden können und welche Interaktionskonzepte dabei angewandt werden sollen. Wichtig ist hierbei zu betrachten, was die gängigen Vorgehen sind und was vermieden werden soll. Eine Schwierigkeit, die umgangen werden soll, ist beispielsweise die Gefahr, dass es den Benutzern schwindelig oder sogar übel werden kann, wenn nicht bestimmte Regeln eingehalten werden.

Die Erkenntnisse, welche während dieser Arbeit erörtert werden, wurden in der Form eines Prototyps umgesetzt. Dieser Prototyp, genannt *Virtual Reality Image Exploitation Workspace* (VIEW), wird während einer Evaluation geprüft. Bei dieser Studie liegen die Schwerpunkte auf der Untersuchung der Intuitivität, der Immersion und der Gebrauchstauglichkeit. Hierbei ist das Postulat:

*VIEW bietet dem Bildauswerter eine effektive und immersive Arbeitsunterstützung.
Dies wird ermöglicht durch die Nutzung von intuitiven Interaktionskonzepten.*

1.6 Umfang und Rahmenbedingungen

Bei dieser Arbeit sollen in erster Linie die Herausforderungen bei der Umsetzung der Bildauswertung für VR untersucht werden. Dies bedeutet, dass bei der Umsetzung von VIEW zunächst die Untersuchung der zentralen Fragestellungen im Vordergrund steht. Daher werden nur die wichtigsten Teile der ViLand-UI und -Logik umgesetzt. Da der Fokus dieser Untersuchungen auf der Konzeption von ViLand in VR liegt, ist eine komplette Portierung der gesamten Funktionalität nicht nötig. Wichtig ist es, dass der Prototyp genug ausgereift ist, dass er für die Evaluation genutzt werden kann.

Zusätzlich wird nur Unterstützung für Luftbilddaufnahmen gegeben. Videos werden demnach nicht unterstützt. Des Weiteren befinden sich die Aspekte *E-Learning* und *Serious Games* außerhalb vom Umfang dieser Arbeit. Dies heißt genauer, dass der Prototyp sich nicht adaptiv an das Lernverhalten des Bildauswerter anpassen wird, wie es mit dem ELAI Framework [Str15] möglich wäre. Dies würde es erlauben den Nutzer im *Flow-Kanal* zu halten. Dieser *Flow* [Nak14] kann durch eine Ausbalancierung der Fähigkeiten und Herausforderungen, die dem Benutzer geboten werden, erreicht werden.

Letztlich wird bei der Konzeption und Evaluierung nicht nach dem *Technology Acceptance Model* (TAM) [Dav86] vorgegangen. Bei TAM liegt der Fokus auf der Entwicklung eines Systems, welches vom Nutzer *akzeptiert* wird und auf der Untersuchung der Motivation des Nutzers, dieses System zu benutzen [Str16].

1.7 Gliederung

Nachdem in diesem Kapitel der Einleitung nun die Problemstellung und Zielsetzung dieser Arbeit geschildert wurden, werden in einem nächsten Schritt die Grundlagen für dieses Forschungsgebiet in Kapitel 2 und Kapitel 3 dargelegt.

In Kapitel 4 und 5 werden die Aspekte der Bildqualität und der Mensch-Maschine Interaktion vertieft und analysiert. Zudem werden mögliche Lösungsansätze diskutiert.

Aufbauend auf den gewonnenen Erkenntnissen wird in Kapitel 6 erläutert, wie der Prototyp erstellt wurde. In Kapitel 7 wird erklärt, wie die Studie zu diesem Prototyp aufgebaut wurde und welche Resultate hierbei erreicht wurden.

Letztlich wird in Kapitel 8 ein Fazit zu dieser Arbeit gezogen und ein Ausblick auf mögliche Herausforderungen geboten und Ideen für zukünftige Arbeiten diskutiert.

In Abbildung 1.2 werden die Kapitel und Unterkapitel zusammengefasst. Aufbauend auf den Kapiteln *Grundlagen* und *State of the Art* (in orange), stützen sich (in gelb) die Gebiete *Bildqualität*, *Der Mensch und die Maschine* und *Assistenzsysteme*. Die in diesen Kapiteln gewonnenen Erkenntnisse werden benötigt zur Erstellung und Evaluierung des Prototyps (in grün).

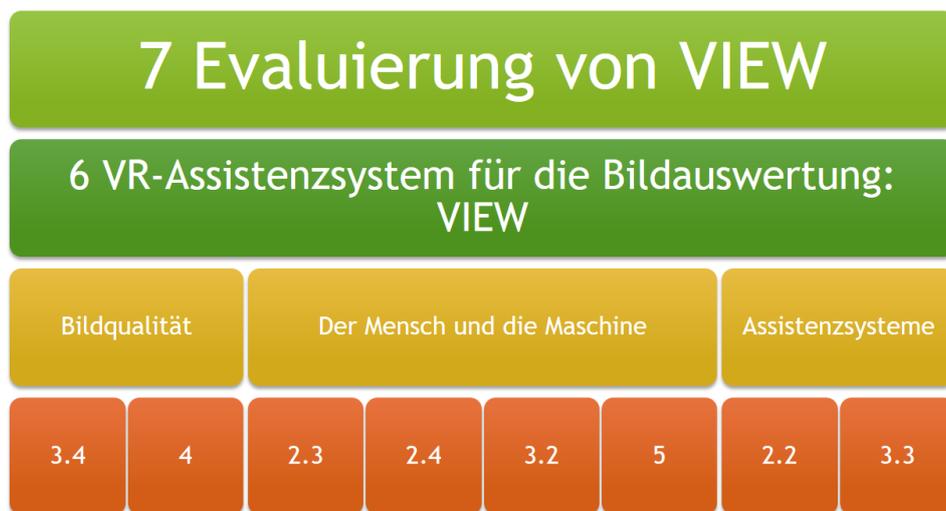


Abbildung 1.2: Übersicht zum Aufbau der Kapitel

2

Stand der Forschung und Technik

In diesem Kapitel wird die Problematik des Gebiets dieser Arbeit genauer untersucht. Es wird aufgezeigt, welche Arbeiten die unterschiedlichen Teile bereits behandelt haben und deren verwendete Lösungsansätze diskutiert. Das Kapitel ist in vier Themengebiete unterteilt:

Einsatzgebiete von VR

Im ersten Unterkapitel werden allgemeine Lösungsansätze für VR in der Industrie aufgezeigt.

Assistenzsysteme und Bildauswertung

Anschließend werden Einsatzgebiete und Lösungsansätze für die Nutzung von Assistenzsystemen und Bildauswertung dargelegt.

Mensch-Maschine Interaktionen in der virtuellen Realität

Im dritten Teil werden bereits existierende Interaktionskonzepte für VR geschildert.

Menschliche Faktoren - Immersion und VR-Krankheit

Zum Schluss werden Arbeiten zur Immersion und VR-Krankheit zusammengefasst.

2.1 Einsatzgebiete von VR

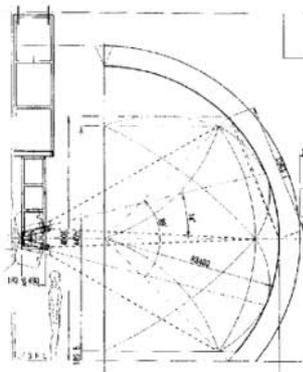
Jonathan Steuer beschreibt die virtuelle Realität als eine Ansammlung von Maschinen wie Computern, *head-mounted* Displays (HMD), Kopfhörern und bewegungserkennenden Handschuhen [Ste92]. Der Begriff VR wurde 1987 von Jaron Lanier eingeführt [VRSa].

Neben der Begeisterung, die VR in jüngster Zeit in der Spielwelt erhält [Pau15], gibt es bereits seit längerem produktive Anwendungsgebiete für diese Technologie.

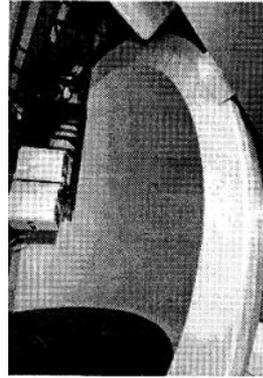
2001 haben Nomura et al. [Nom01] ein Review zur Nutzung von VR-Anwendungen in der Industrie veröffentlicht. Hierbei werden vier Arten von Anwendungsbeispielen untersucht.

- VR-System zum Relaxieren des Benutzers
- Pferderei-Simulator als therapeutisches Mittel
- Küchen-Planungs-System
- Stadt-Planungs-System

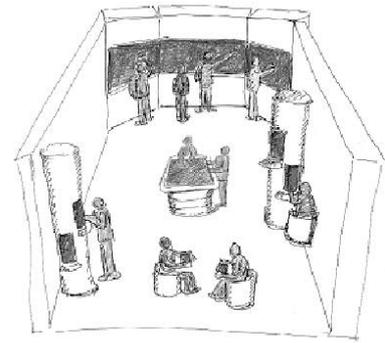
Interessant für diese Arbeit ist die Gestaltung des Dom-förmigen Displays, welches für die Planung von städtischen Gegenden genutzt wurde [Nom01] (siehe Abb.2.1a). Sechs Projektoren strahlen auf diese gewölbte Fläche eine VR-Umgebung. Dieses Display deckt ein Sichtfeld von 180° horizontal



(a) Seitenansicht des Dom-förmigen VR-Zimmers [Nom01]



(b) Foto des Dom-förmigen VR-Zimmers [Nom01]



(c) Skizze der interaktiven i-Land Umgebung [Str99] [Str01]

Abbildung 2.1: Visionen von Arbeitsumgebungen der Zukunft

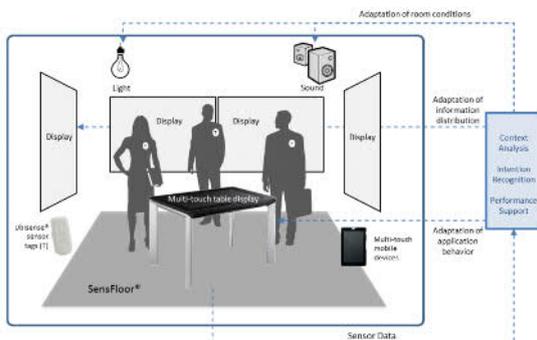
und 90° vertikal. Diese Form wurde als Vorlage für den ersten Entwurf des *MapDisplays* in unserem Prototyp genutzt.

Eine weitere Inspiration war die sogenannte *i-Land* Umgebung, welche in den Arbeiten von Streitz et al. [Str99] [Str01] entworfen wurde. Es wird geschildert, dass die schnellen Fortschritte der Technik eine große Herausforderung für die Gestaltung von Mensch-Maschine Interaktionen bieten. In diesen Arbeiten wird diskutiert, wie der Alltag in einer Firma der Zukunft aussehen könnte. Genauer wird diskutiert, wie die Architektur für Räume und für die Firma selbst aussehen soll. In einer Skizze von 1997 wird zusammengefasst, wie die sogenannte *i-Land* Umgebung aussehen sollte. Sie besteht aus einer interaktiven Wand (DynaWall®), einem interaktiven Tisch (InteracTable®), und Stühlen (CommChairs®), welche weitere Interaktionen erlauben (siehe Abb. 2.1c). Die Aufteilung des Raums und einige der Funktionalitäten sind auch in unserem Prototyp wieder zu finden.

Ein ähnlicher Aufbau, mit interaktivem Tisch und großer Wandprojektion, ist auch im *SmartControlRoom* (siehe Abb. 2.2) vorzufinden [Aeh13] [Voi].

Diese und viele weitere Arbeiten zeigen, dass die Nutzung von 3D-fähigen Bildausgabegeräten und VR, ein nützliches Mittel zur Unterstützung des Menschen auf dem Arbeitsplatz bietet.

Dies wird einerseits begründet durch die immersive Faktoren von VR, andererseits durch die Möglichkeit intuitivere Interaktionskonzepte realisieren zu können. Des Weiteren erlaubt VR eine effektivere Darstellungsweise von vielen unterschiedlichen Arten von Daten.



(a) Aufbau des SmartControlRooms [Aeh13]



(b) Foto des des SmartControlRooms [Voi]

Abbildung 2.2: Überblick zum SmartControlRoom

2.2 Assistenzsysteme und Bildauswertung

Das Ziel eines Assistenzsystems ist es, zusammen mit dem Menschen ein gemeinsames Ziel zu erreichen, eine Aufgabe gemeinsam zu lösen. Hierbei ist darauf zu achten, wie die Aufgaben zwischen Mensch und Maschine aufgeteilt werden [Rol13]. Laut Deuse et al. führt der Einsatz von Assistenzsystemen mit intuitiven Benutzerschnittstellen zu einer Entlastung des Nutzers. Dadurch, dass die Automatisierung die monoton belastenden Aufgaben übernimmt, wird der Nutzer flexibler in die Steuerungs- und Regelungsprozesse eingebunden [Deu15].

Im Bereich der Geowissenschaften, Fernerkundung und Bildauswertung wird VR vor allem zur Visualisierung von Rohdaten genutzt. Da bei diesen Arbeiten die Interaktionskonzepte von höherer Relevanz sind, werden sie erst in Unterkapitel 2.3 behandelt.

Bereits 1998 haben Lin et al. aufgezeigt, wie VR als Alltags-Werkzeug auf dem Arbeitsplatz von Bildauswertern genutzt werden könnte [Lin98].

In der Arbeit *A geoscience perspective on immersive 3D gridded data visualization* von Billen et al. wird die Software *Visualizer* beschrieben [Bil08]. Hierbei handelt es sich um eine VR-Anwendung, welche die Visualisierung von großen Datensätzen in VR erlaubt. Schwerpunkt dieser Arbeit ist die Performanceoptimierung, die vorgenommen werden mussten, um hohe Bildwiederholraten zu erreichen und das Hervorheben der Vorteile der Nutzung von VR für diese Art der Visualisierung von Daten.

Es wird aufgezeigt, dass auch bereits in anderen Arbeiten ([Fur04] und [Ohn07]) festgestellt wurde, dass die Geowissenschaften von den direkten Interaktionsmöglichkeiten in VR profitieren.

Im ersten Teil werden die Vorteile der Nutzung von 3D-fähigen Anzeigegeräten motiviert. Sie geben an, dass in der dreidimensionalen Welt viele Informationen gegeben werden, die über zweidimensionale Darstellung nur schwierig übermittelt werden können. Zu diesen gehören die Form und Tiefe von Objekten, welche in 3D anhand von Bewegungen, Schatten und Referenzgrößen besser erkannt werden können. Dazu geben sie an, dass immersive Umgebungen den Wissenschaftlern die Möglichkeit bietet, das Potential ihres Gehirns, beim Interpretieren von visuellen Informationen, auszuschöpfen [Hub05].

Zusammen erlauben diese Faktoren die Erstellung einer natürlichen Umgebung und eine effektivere Art der Erkundung von dreidimensionalen Daten.

2.3 Mensch-Maschine Interaktionen in der virtuellen Realität

Yasser Malaika gibt in seiner Rede *Interaction Design in VR: The Rules Have Changed (Again)* vom GDC Vault 2015 eine sehr ausführliche Zusammenfassung zu den Paradigmen, die bei der Entwicklung von VR-Anwendungen beachtet werden sollen [Mal].

Er erklärt, dass bei heutigen Konsolen-Spielen die Art der Steuerung, also die Interaktionskonzepte, weitestgehend standardisiert sind (siehe Abb. 2.3a). Die Controller der derzeitigen Konsolengeneration bestehen meist aus zwei Joysticks, einem Steuerkreuz, vier beschrifteten Knöpfen, vier Schultertasten und drei Menü-Tasten. Hierbei wird der linke Joystick zur Lokomotion (Bewegen, Steuern, ...), der rechte zur Blickwinkel-Änderung (umher Blicken) und die vier Tasten zur Aktion (Springen, Aktivieren, Schlagen, ...). Diese Art der Standardisierung existiert in VR zur Zeit noch nicht. Yasser erklärt, dass jeder Entwickler für seine Anwendung entdecken muss, welche Konzepte angebracht sind – dabei jedoch auf eine Reihe von wichtigen Paradigmen achten muss.

Interface Gestaltung In der Arbeit von Lin et al. [Lin98] wird das CWI (Closed world interaction) Konzept vorgestellt. Da die Anzahl an physikalischen Knöpfen der Controller limitiert ist, werden zusätzlich *virtuelle* Knöpfe genutzt. Es wird angegeben, dass es wichtig ist, diese virtuellen Interaktionen so simpel und natürlich wie möglich zu gestalten, damit ein *durchsichtiges* Interface erreicht

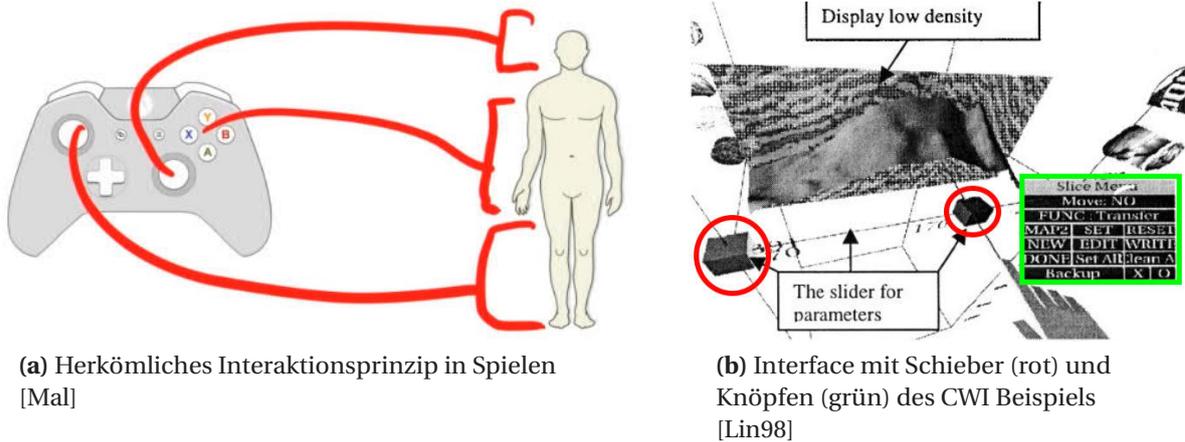


Abbildung 2.3: Interaktionskonzepte

werden kann. Mit *durchsichtig* ist in diesem Kontext gemeint, dass der Benutzer nicht mehr auf die Computer Hardware achten muss, sie wird ubiquitär.

Das CWI besteht aus einer Bounding Box, welche als Koordinaten-Rahmen dient. An den Kanten dieser Box sind Schieber angebracht, welche genutzt werden können, um Variablen anzupassen. Zusätzlich werden um die Box herum Knöpfe angezeigt, die weitere Interaktionsmöglichkeiten erlauben. In Abbildung 2.3b ist dieses Interface zu sehen. Zur besseren Erkennbarkeit, habe ich die Schieber rot und die Knöpfe grün markiert.

Diese Knöpfe basieren zum Teil auf dem 2D WIMP (windows, icons, menus, pointer) Paradigma, welches von Desktop-Computern bekannt ist. Die Autoren gehen davon aus, dass dies es dem Benutzer vereinfacht sich an die neue Umgebung zu adaptieren. Jedoch hat die Studie gezeigt, dass die Probanden Probleme dabei hatten, sie zu betätigen.

Im Jahr 2000 wurde das Projekt weitergeführt [Lof00]. Lin et al. betonen nochmal, wie vorteilhaft die Nutzung von VR bei der Visualisierung von Daten ist und zeigen auf, dass die größte Hürde die Verbesserung von VR Interaktionskonzepten ist.

Für diese Arbeit wurde erneut das WIMP-Paradigma genutzt und es wurde erneut gezeigt, dass die Nutzung dieser zweidimensionalen im Raum schwebenden Menüs problematisch ist. Dies sollte also bei unserer Arbeit unbedingt vermieden werden.

In weiteren Verlauf werden mehrere fragwürdige Designentscheidungen geschildert. So wird zum Beispiel ein einzelner Klick auf dem Controller zum Anwählen von Objekten genutzt - jedoch ein Doppelklick zum sofortigen Löschen dieser Objekte.

Hoppe überträgt in seiner Masterarbeit die WIMP-Metapher vom Desktop in VR [Adr16]. Hierbei präsentiert er vier Lösungsansätze:

Klassische UI Das K-UI projiziert das bekannte Desktop-Bild auf eine Fläche

Natürliche UI Die N-UI lässt den Benutzer Objekte in der realen Welt greifen und bildet diese Aktionen dann in der virtuellen Welt um

Metaphern UI Die Kategorie M-UI verwendet alternative Metaphern, welche fast nicht in WIMP-Applikationen vorzufinden sind

Unsichtbare UI Bei der U-UI werden vor allem Gesten erkannt. Es sollen nicht benötigte Interaktionsfenster vermieden werden

Es werden mehrere Interaktionsprinzipien des Desktops übertragen - unter anderem *Drücken eines Knopfes*, *Scrollen*, *Platzieren von Elementen* und *Lesen eines Textes*.

Bei der Studie hat sich gezeigt, dass alle Lösungsansätze grundsätzlich nutzbar sind, es jedoch Unterschiede bei der Bedienbarkeit, Geschwindigkeit und Anstrengung gibt. So wirkte die N-UI für die Probanden sehr intuitiv, verlangte jedoch ein hohes Maß an Bewegungen. Probanden gefielen am besten Interaktionen die *schnell, intuitiv* und *einfach* ausführbar waren. Interaktion die ein hohes Maß an Bewegung oder Komplexität enthielten, wurden negativ von den Probanden bewertet.

Zudem gibt Hoppe an, dass auf die Lesbarkeit von Texten geachtet werden muss. Der Entwickler von VR-Anwendungen muss sicherstellen, dass der Benutzer eine Möglichkeit hat sich dem Text zu nähern, damit dieser besser lesbar wird.

Bei der Gestaltung von Interaktionen in VR sind die Unterschiede zwischen direkter und indirekter Manipulation von Daten zu beachten. Bei der zuvor erwähnten Software *Visualizer* [Bil08] wird beispielsweise direkte Manipulation durchgeführt, beim Greifen und Drehen von Objekten. Kreylos et al. beschreiben die direkte Manipulation als *Berührung der Daten* [Kre06].

Indirekte Manipulation wird hingegen durchgeführt bei der Eingabe von Werten zur Anpassung der Rotation oder durch Betätigen von Knöpfen.

Kreylos's Co-Autoren glauben, dass die direkte Manipulation eines der Hauptmerkmale von der effektiven Interaktion in VR ist und dass es benötigt wird, um die Akzeptanz von VR als wissenschaftliches Werkzeug sicherzustellen.

Auch Forsberg et al. [For97] streben die direkte Interaktion an. Sie geben an, dass das *ultimate Ziel* ein 3D-Interface ist, welches so transparent wirkt, wie die Affordance, die durch ein Bleistift und ein Stück Papier gegeben ist.

Lokomotion Als Lokomotionsmittel empfiehlt Yasser [Mal] eine Kombination von *room-scale* und *Point-And-Teleport* (siehe Abb. 2.4a). Das Konzept des *Point-And-Teleport* hat sich für die Tester als sehr einnehmend für die Propriozeption erwiesen. Das heißt, den Testern war ganz klar, wo sie sich im virtuellen Raum relativ zu anderen Objekten befanden. Er berichtet, dass nach kurzer Zeit die Benutzer den Zeiger hinter sich gezielt haben, ohne hinzuschauen, um sich so schnell nach hinten zu teleportieren. Dies wird auch in der Arbeit von Bozgeyikli et al. [Boz16] bestätigt. Resultat dieser Studie war es, dass *Point-And-Teleport* für die Probanden Spaß machte und benutzerfreundlich ist. Wohingegen Bewegungen mit Hilfe vom Analog-Joystick die User Experience reduziert.

Von der Umsetzung der Lokomotion mit Hilfe von einem Joystick rät Yasser ab [Mal]. Dies verursacht einen Widerspruch, zwischen dem was der Benutzer fühlt und, dem er wahrnimmt, was als Folge Übelkeit verursachen kann. Genauer hierzu in Unterkapitel 2.4.

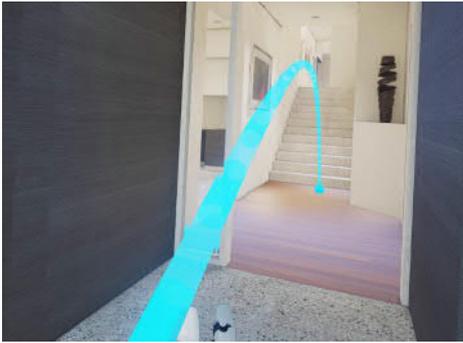
Des Weiteren gibt er an, dass durch Kopf- und Nacken-Bewegung ausgelöste Eingaben nicht als Interaktionsmethode genutzt werden soll. Die Nacken und Augenmuskel des Menschen sind nicht dazu bestimmt, derartige Interaktionen auszulösen und es ist für den Benutzer unbefriedigend, so Yasser in seiner Rede.

Jon Martindale gibt in seinem Artikel an, dass die Art wie Lokomotion in VR gestaltet werden sollte noch sehr umstritten ist [Mar16]. Bei vielen Benutzer löst die künstliche Art der Fortbewegung – beispielsweise Bewegung mit Joystick – sofortige Übelkeit aus. Andererseits geben mehrere Entwickler an, dass *Point-And-Teleport* nicht genutzt werden soll. Der VR-Entwickler Goetgeluk gibt im Interview an, dass es die Immersion und den Spielspaß *zerstört*.

Yasser gibt im Kontext der Lokomotion außerdem an, dass bei der Teleportation die Orientierung des Benutzers nicht verändert werden soll [Mal].

Zudem hilft es ein Referenzpunkt zu geben, damit der Benutzer sich einfacher orientieren kann. Um dies umzusetzen, kann beispielsweise das Cockpit des Fahrzeugs genutzt werden. Ein Beispiel hierfür ist die VR-Umsetzung vom Spiel *Elite: Dangerous*, zu sehen in Abbildung 2.4b.

Eine andere Möglichkeit, ist es den sichtbaren Bereich zu reduzieren. Fernandes et al. konnten 2016 in ihrer Arbeit zeigen, wie die dynamische Veränderung des sichtbaren Bereichs das Vorkommen von Übelkeit reduzieren kann [Fer16]. Dies wird im nächsten Unterkapitel 2.4 genauer aufgeführt.



(a) Point-To-Teleport in VR

Quelle: Imsopov
(Forums.UnrealEngine.com)



(b) Cockpit des Raumschiffs im Spiel Elite: Dangerous VR

Quelle: VR-Gaming.co.uk

Abbildung 2.4: Lokomotion in VR

Auf jeden Fall sollen Veränderungen der Beschleunigung vermieden werden. Das heißt, dass die Teleportation augenblicklich geschehen soll und nicht als eine Art schwebende Plattform, welche sich immer schneller zum Ziel bewegt [Mal].

Taxonomie Basierend auf den Arbeiten von Yi et al. [Yi07] und Wolf et al. [Wol11] hat Valve die Taxonomie von VR Eingaben erstellt (siehe Tabelle 2.1). Hier wird festgestellt, welche Art von Interaktionen mit welchen Arten von Aktionen verbunden werden sollten. Die Inhalte dieser Tabelle werden in Kapitel 5 näher betrachtet.

Taxonomy	Works best within	Example
6 DOF Manipulation	Within reach	Picking-up and moving objects directly in your reach
Pointing	Reach and beyond	Teleportation
Hand Controls	Actuation, Swiping	Buttons, triggers, wheels, touchpads
Ambient Invocation	Infrequent, Indirect	Voice control
Subconscious	Background	Position and acceleration in space

Tabelle 2.1: Taxonomy of VR input [Mal]

In VR muss genau so wie bei typischen Desktop-Anwendungen Fitt's Law [Fit54] beachtet werden. Bei der Festlegung der Größe von Knöpfen und sonstigen Zielobjekten in VR muss auf die Distanz und den Winkel zwischen diesem Objekt und dem Benutzer geachtet werden [Cha13]. Yasser betont in diesem Kontext, dass Benutzer bei der Interaktion im dreidimensionalen Raum mehr Zeit benötigen, weil sie Vergleich zu Desktop-Anwendungen eine zusätzliche Dimension berücksichtigen müssen [Mal]. Zudem kommt es öfter vor, dass Benutzer sich drehen oder Bewegen müssen, um in Reichweite von Zielobjekten zu gelangen. Im Gegensatz zu Desktop-Anwendungen, sind VR-Anwendungen nicht durch den Rand des Monitors limitiert.

Yves Guiard's Modell der bimanuellen Geschicklichkeit teilt bei der Interaktion mit zwei Händen, die dominante Hand für die Detailarbeit und die nicht-dominante Hand für die weniger genauen Aufgaben ein [Gui87].

Google demonstriert eine passende Umsetzung dieses Modells in ihrer 3D-Zeichenanwendung Tilt Brush [Goo16]. In der linken Hand hat der Benutzer eine Farbpalette. Mit der rechten Hand kann der Benutzer die verschiedenen Werkzeuge und Farben auf dieser Palette anwählen. Das Interessante hierbei ist, dass anstelle davon, dass der Benutzer den rechten Controller bewegt, um die Elemente anzuwählen, er auch den linken Controller bewegen kann, bis er unter dem gewünschten Ziel ist. Dies kehrt das Grundprinzip von Guiard's Modell um. Yasser bezeichnet dies als die *Bimanual*

Pointer/Target Ambiguity und gibt an, dass dies in der Praxis in Ordnung ist. Unabhängig von der gewählten Interaktionsart, fühlt es sich für den Benutzer natürlich und unkompliziert an [Mal].

Haptisches Feedback ist ein zusätzliches Hilfsmittel, welches dem Benutzer Informationen überliefern kann. So kann er mit einer kurzen Vibration darauf hingewiesen werden, wenn ein Objekt manipulierbar ist, oder wenn er es mit dem Controller berührt. Weitere Arten von Feedback, wie Gewicht, Temperatur oder Textur könnten in Zukunft noch mehr Leben in simulierte Objekte bringen [Bur96]. Die Vorteile von haptischen Feedbacks werden auch für Roboter-assistierte Operationen untersucht. 2009 wurde eine Studie durchgeführt, welche den Effekt hiervon untersucht [vdM09]. Es wurde aus der Studie geschlossen, dass der Zusatz von haptischem Feedback zu einer Reduzierung der chirurgischen Fehler führen kann.

Grundsätzlich ist also festzulegen, dass es derzeit noch keine Standards für die Interaktion in VR gibt. Dennoch gibt es eine Reihe von Richtlinien und Paradigmen, die beachtet werden müssen, um eine angenehme Nutzung zu erlauben.

2.4 Menschliche Faktoren - Immersion und VR-Krankheit

Bei der Entwicklung von VR-Anwendungen sollen Fähigkeiten und Bedürfnisse des Menschen stets beachtet werden. Die drei Faktoren *Interaktion*, *Immersion* und *VR-Krankheit* sind miteinander verbunden und beeinflussen sich gegenseitig.

Nachdem die Arbeiten im Gebiet der Interaktion zusammengefasst wurden, werden im folgenden Arbeiten zu den Gebieten VR-Krankheit und Immersion aufgelistet.

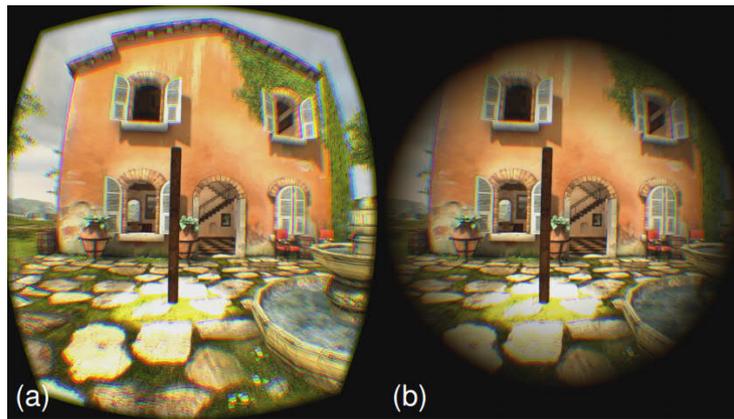
VR-Krankheit Wie bereits im vorherigen Unterkapitel erwähnt wurde, ist es wichtig bei der Lokomotion darauf zu achten, das Auslösen von der sogenannten VR-Krankheit zu vermeiden. Hierbei handelt es sich um das Vorkommen von Übelkeit, Schwindelgefühl oder sogar Erbrechen bei oder nach der Nutzung von VR-Anwendungen.

Die Anfälligkeit variiert von Mensch zu Mensch. Es ist jedoch wichtig eine Reihe von Paradigmen zu folgen, um den Effekt zu reduzieren.

Fernandes et al. erklären, dass die Reduzierung des sichtbaren Bereichs, den Effekt reduzieren kann – dies jedoch zu Kosten der Immersion [Fer16]. Daher haben sie ein System der dynamischen Anpassung des FOVs entwickelt. Bei der Studie wurde gezeigt, dass das Ausmaß der VR-Krankheit gesenkt wurde, ohne dass die Probanden eine Verminderung der Presence vermerken konnten. Zudem wurde die Auffälligkeit dieser Intervention minimiert.

Eine weitere Möglichkeit der Reduzierung von VR-Krankheit ist, dem Benutzer einen festen Rahmen zu geben. Dies ist beispielsweise möglich, in dem das Cockpit des Fahrzeugs zu sehen ist. Außerdem ist es hilfreich, einen unabhängigen statischen Hintergrund zu zeigen [Duh04]. Whittinghill et al. haben 2016 ein Patent auf das Anzeigen eines fixierten Objekts zur Reduzierung von VR-Krankheit veröffentlicht [Dav16]. Sie haben als Beispiel für ein solches Objekt eine *virtuelle Nase* genutzt und geben an, dass Nutzer deutlich weniger Übelkeit verspüren [Hec16]. Beispiele zu den unterschiedlichen Methoden sind in Abbildung 2.5 zu sehen.

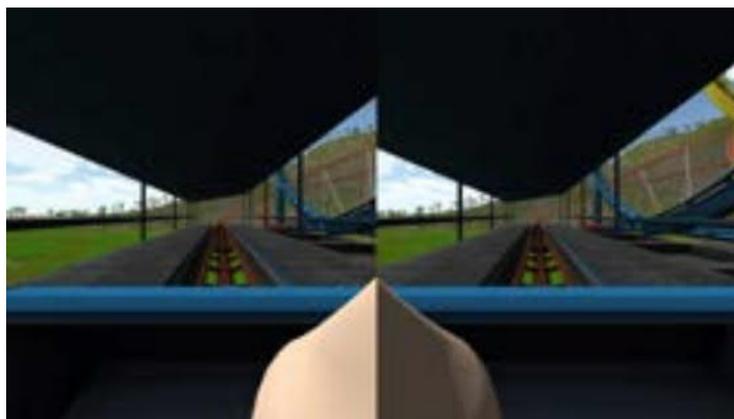
Grundsätzlich kann VR bei jedem Nutzer Übelkeit hervorrufen. Es muss darauf geachtet werden, wie der Benutzer bewegt wird und was er dabei sieht. Die nicht vom Benutzer ausgeführten Bewegungen müssen mit konstanter Geschwindigkeit ausgeführt werden und vorhersehbar sein [Kau15]. Zudem muss unbedingt vermieden werden, den virtuellen Kopf des Benutzers ohne dessen Eigeninitiative, zu bewegen [Pal17]. Außerdem ist darauf zu achten, dass ein längerer Aufenthalt in VR zu erhöhtem Vorkommen von VR-Krankheit führt – jedoch öfter Nutzung von VR die Anfälligkeit hierfür reduziert [Ken00].



(a) Limitiertes Sichtfeld reduziert VR-Krankheit [Fer16]



(b) Referenz Rahmen, der im Spiel Euro Truck Simulator 2 VR gegeben wird
Quelle: EuroTruckSimulator2.com



(c) Das Anzeigen einer virtuellen Nase, kann auch das Vorkommen von VR-Krankheit reduzieren [Hec16]

Abbildung 2.5: Möglichkeiten der Reduzierung von VR-Krankheit

Die Anzahl an *Best Practice Guides* von unterschiedlichen Firmen wie Oculus [Yao14], Epic Games [Epi], Google [Goo15], Unity [Uni] und vielen mehr ist bemerkenswert. Neben den visuellen Faktoren müssen auch technischen Faktoren beachtet werden.

So hat die Bildwiederholrate einen hohen Einfluss auf das Empfinden des Benutzers. Zu niedrige FPS (Frames per Second) können auch VR-Sickness führen. Außerdem haben St. Pierre et al. gezeigt, dass zu hohe Varianz in der Bildwiederholrate zu VR-Krankheit führt [St.15].

Der Konsens ist, dass bei interaktiven VR-Anwendungen eine Bildwiederholrate von mindestens 90 FPS erreicht werden muss – bei nicht interaktiven Szenen reichen 60 FPS. Dies bezieht sich nicht nur auf die Performance des Rechners und die Optimierung der Anwendung, sondern auch auf die genutzte Hardware. In Hinsicht auf die Bildausgabegeräte muss die Bildschirmwiederholrate (gemessen in Hertz) auch diese Mindestwerte erreichen können.

Immersion und Presence Im Kontext der Immersion ist das Konzept der *Presence* von Interesse. Sie beschreibt das Gefühl, das Benutzer verspüren können, wenn sie sich durch eine Technologie mit einer Welt außerhalb des eigenen Körpers (z. B. virtuelle Welt) verbunden fühlen.

Michael Abrash gibt während den Steam Dev Days 2014 [Abr14] an, dass *Presence* weitaus mehr, als *nur* Immersion ist. Laut Schubert et al. handelt es sich bei der Immersion um eine Variable der Technologie, welche objektiv beschrieben werden kann. Bei der *Presence* handelt es sich hingegen um eine Variable, die subjektiv vom Benutzer gefühlt wird. Daher werden zur Messung von *Presence* Fragebögen genutzt, bei welchen der Benutzer selbst sein Empfinden angibt [Igr].

In Abbildung 2.6 wird der Ablauf von der Immersion bis zur *Presence* beschrieben. Die Immersion vertieft die Sinne und den Körper des Benutzers in die virtuelle Welt. Die kognitiven Prozesse des Menschen erstellen dann ein mentales Modell des Körpers in der virtuellen Welt. Infolgedessen verspürt der Mensch die *Presence*, das Gefühl, sich in der virtuellen Welt zu befinden.

Presence gehört zu den zentralen Komponenten bei der Erstellung von VR-Anwendungen. In einigen Fällen wurde gezeigt, dass *Presence* wichtig für das Lernen und Trainieren in VR ist und für die Empfindungen von Spaß, Unterhaltung und realen Emotionen. Benutzer, die diese *Presence* empfinden, fokussieren ihre gesamte Aufmerksamkeit auf diese virtuelle Welt und empfinden sie nahezu als die *echte* Welt [Igr15]. In einer Studie von Lackey et al. [Lac16] tendierten Probanden, welche ein positives VR Erlebnis angaben, zu niedrigeren Stress- und Arbeitsbelastungswerten.

Nach der *Hierarchie der Bedürfnisse für VR* [Cro15] (siehe Abb. 2.7) wird der unterste Faktor *Comfort* durch die Berücksichtigung der VR-Krankheit Paradigmen befriedigt. Der Faktor *Interpretability* hängt davon ab, ob die Umgebung *Sinn* ergibt. Auf diese Schicht muss besonders geachtet werden, falls dem Benutzer die Fähigkeit des Fliegens verliehen wird und sonstige physikalisch-unsinnige Szenarien präsentiert werden. Der Faktor *Usefulness* wird gegeben durch die Gebrauchstauglichkeit der Anwendung. Der Faktor *Delight* hängt vom Gesamtsystem ab und gibt an, ob der Benutzer das System erneut nutzen würde.

Das Ausmaß der subjektiv empfundenen *Presence* wird von vielen Faktoren beeinflusst. In Kapitel 4.3 und 7.3.7 werden die von Michael Abrash [Abr14] beschriebenen technischen Aspekte genauer behandelt und in Kapitel 7.1 werden die subjektiven Faktoren beschrieben. In Kapitel 5 wird die Immersion und die Interaktion detaillierter untersucht.

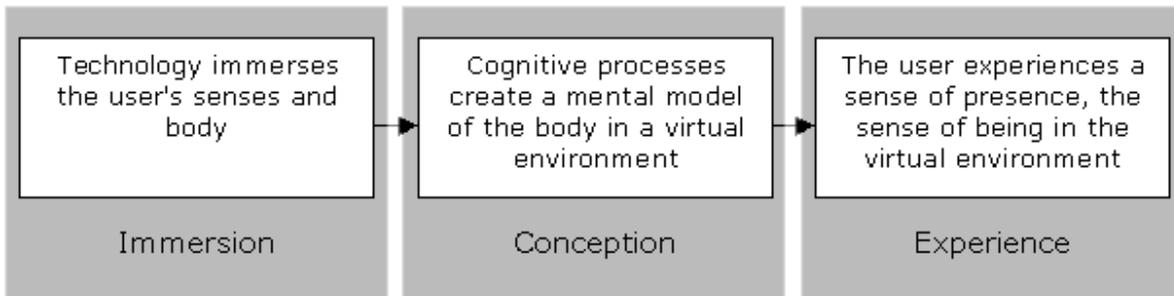


Abbildung 2.6: Prozess von der Immersion zur Presence [Igr]

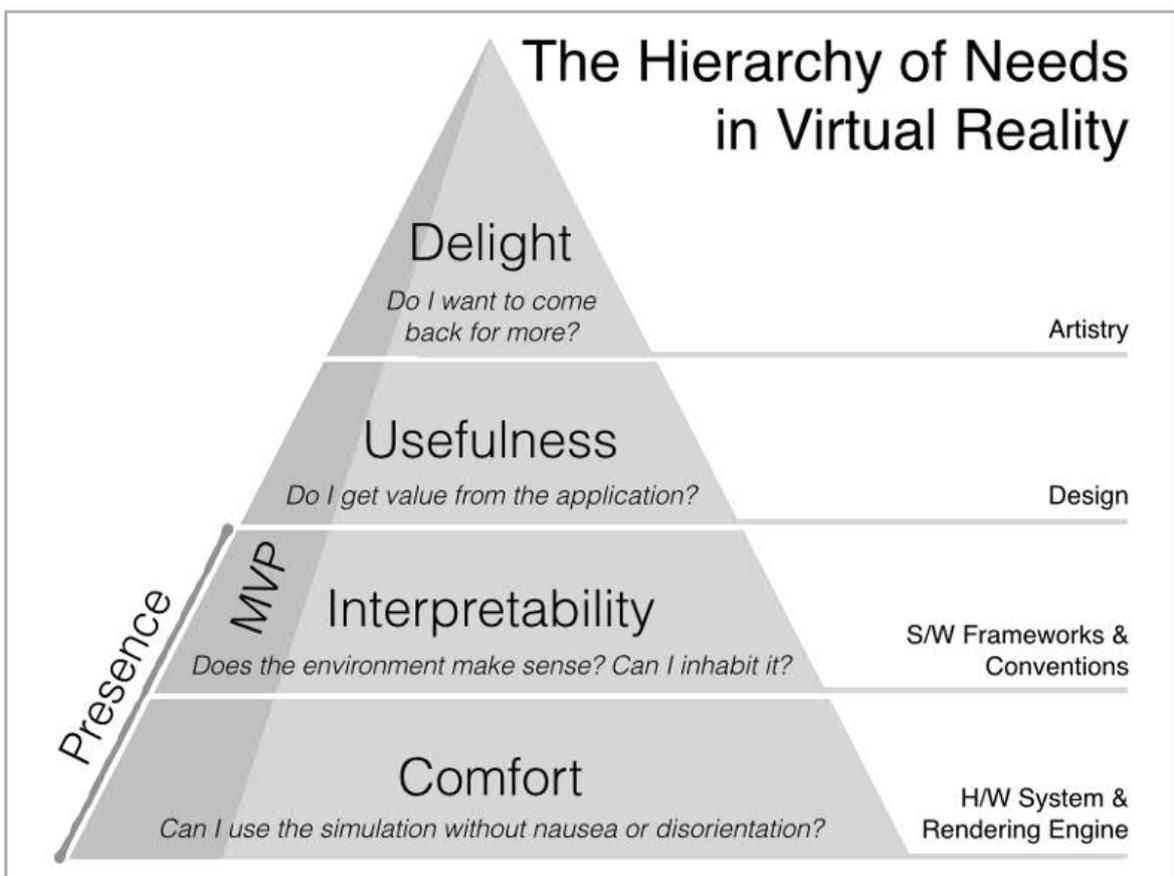


Abbildung 2.7: The hierarchy of needs in virtual reality nach Cronin [Cro15]

3

Grundlagen

In diesem Kapitel wird ein grundlegender Überblick zu den, in dieser Arbeit behandelten Themen geboten. Eine Zusammenfassung der häufig genutzten Abkürzungen und Begriffen befindet sich im Glossar. In Unterkapitel 3.1 werden die aktuellen Virtual Reality Hard- und Software Umgebungen erklärt. In Unterkapitel 3.2 wird eine Übersicht zu den Themen der Interaktion, Immersion und VR-Krankheit geschildert. In Unterkapitel 3.3 wird aufgezeigt, was Assistenzsysteme sind. Zudem wird die Bedeutung der Bildauswertung zusammengefasst. In Unterkapitel 3.4 werden Begriffe des Themengebiets Bildqualität erläutert. Schlussendlich werden in Unterkapitel 3.5 die in dieser Arbeit genutzten Frameworks, Softwares und Engines aufgelistet.

Der Zweck dieses Kapitels ist es dem Leser das, für das Verständnis dieser Arbeit, benötigte Grundwissen zusammenzufassen.

3.1 VR-Technologien im Überblick

Bei Virtual Reality (VR) handelt es sich um eine Technologie, welche ein Headset-artiges Gerät – ein sogenanntes Head-Mounted-Display (HMD) – benutzt, um dem Benutzer mit Hilfe von Bildern und Geräuschen, die Möglichkeit zu bieten in eine virtuelle Welt einzutauchen. Typischerweise besitzen diese Headsets zwei gleichgroße Displays, welche es dem Benutzer erlauben, die Bilder stereoskopisch – also dreidimensional – wahrzunehmen. Im Gegensatz zu herkömmlichen stereoskopischen Displays erlauben VR-HMDs nicht nur die Ausgabe von Informationen, sondern auch die Aufnahme/Eingabe von Informationen. Dreht der Benutzer seinen Kopf, so bewegt sich auch in der virtuellen Welt sein Blickfeld in diese Richtung. Dieses Zusammenspiel von Ein- und Ausgabe erlaubt die Erschaffung von höchst immersiven virtuellen Welten. In diesem Unterkapitel werden aktuelle Hard- und Software Angebote dieses Themengebiets geschildert, um einen Überblick zu den Möglichkeiten und Grenzen von VR zu schaffen.

3.1.1 Hardware mit integriertem Display

Virtual Reality war bereits in den 80er und 90er erhältlich [BEN16]. Die Technologie war damals jedoch wegen der teuren Herstellungs- und Entwicklungskosten und der derzeitigen Rechenkraft der Computer nur sehr limitiert gebräuchlich [Gre14]. Die US-amerikanische Computertechnologiefirma *Oculus VR Inc.* hat dies 2012 mit ihrem ersten für den Konsumentenmarkt erhältlichen HMD – Oculus Rift DK1 – geändert¹.

Dieser Prototyp war, trotz seiner Mängel, wie niedriger Auflösung (640 x 800 Pixel pro Auge) und niedriger Bildschirmwiederholrate (60 Hz), ein großer Fortschritt für die VR-Industrie [RIK14].

¹ Xinreality - Oculus Rift DK1: https://xinreality.com/wiki/Oculus_Rift_DK1



Abbildung 3.1: Oculus Rift HMDs

Über die Jahre hat Oculus mehrere Prototypen und die finale Rift veröffentlicht (zu sehen in Abb. 3.1). Zudem wurden sie 2014 von Facebook gekauft. Währenddessen, ist die Konkurrenz von Oculus gewachsen. Die taiwanesische Firma *HTC Corporation* hat Mitte 2016 zusammen mit der US-amerikanischen Firma *Valve Corporation* das VR-System *HTC Vive* auf den Markt gebracht.

Das HTC Vive erlaubt neben der Erfassung von Kopfbewegungen auch die Erfassung von Körperbewegungen. Die Bewegungen des Benutzers werden von zwei *Lighthouse*-Stationen erkannt und in die virtuelle Welt übertragen. Zusätzlich werden zwei kabellose Controller geliefert, deren Bewegungen auch in die virtuelle Welt übernommen werden. Dies erlaubt es dem Benutzer sich nahezu frei in der virtuellen Welt zu bewegen und zu agieren. Diese Art von VR wird als *room-scale* bezeichnet (siehe Abb. 3.2).

Das HTC Vive bietet zudem eine eingebaute Kamera, welche zu jedem Moment per Knopfdruck aktiviert werden kann. Dies ist hilfreich, wenn der Benutzer sich von den Kabeln lösen will, die sich häufig in den Beinen verfangen können. Eine weitere Abhilfe bietet das TPCAST. Dieser Aufsetzer wird Ende April 2017 für rund 250 € erhältlich sein und ermöglicht die kabellose Nutzung des HTC Vive HMDs. Das HTC Vive Kit kostet zur Zeit rund 1000 € und benötigt zusätzlich einen leistungsstarken Computer (zusätzlich mindestens 1200 €), welcher die Ausgaben berechnet.

Sony hat dieses Problem erkannt und Ende 2016 die Playstation VR herausgebracht. Das PS-VR Kit ist in Europa für 399 € erhältlich und benötigt zusätzlich eine Playstation 4 Pro, welche auch für 399 € erhältlich ist (siehe Abb. 3.3). PS-VR funktioniert auch mit der Standard PS4, dies wird jedoch nicht empfohlen. Obwohl die PS-VR nicht das volle *room-scale* Erlebnis des HTC Vive Systems erlaubt, hat sie wegen des verhältnismäßig niedrigen Preises an hoher Popularität gewonnen.



Abbildung 3.2: Überblick zum HTC Vive System



(a) Playstation 4 Pro
Quelle: Forbes.com

(b) Sony PS-VR Kit
Quelle: Playstation.com

Abbildung 3.3: Benötigte Komponenten für die Nutzung von PS-VR

3.1.2 Hardware mit Smartphones als Display

Neben den HMDs mit integrierten Displays, können auch die meisten Smartphones genutzt werden, um VR zu erleben (siehe Abb. 3.4). 2014 hat ein Team von Google das Google Cardboard präsentiert. Hierbei handelt es sich um eine Halterung aus Pappe mit zwei Linsen. Der Benutzer schiebt sein Smartphone seitlich in das Gehäuse und kann dann unterschiedliche VR-Anwendungen, welche zwei verschiedene Blickwinkel auf dem Display darstellen, erleben. Hierbei werden dann, mit Hilfe der im Smartphone vorhandenen Gyroskopen, die Kopfdrehungen erkannt. Das Google Cardboard kostet nur rund 20 €. Ziel dieses Projekts war es, das Interesse der Öffentlichkeit an der Erkundung und Entwicklung von VR-Anwendungen zu erwecken.

Samsung hat Ende 2015 seinerseits das GearVR herausgebracht. Dieses ist nur mit Samsung Smartphones der Reihe S6 und S7 nutzbar und kostet rund 100 €. Im Gegensatz zum Google Cardboard bietet das GearVR eine MicroUSB-Verbindung zum Smartphone, Eingabe Optionen und einen integrierten Bewegungserkennung.

2016 hat Google das Projekt Daydream vorgestellt und erhältlich gemacht. Google Daydream ist die Weiterentwicklung von Google Cardboard. Es ist in Android 7.1 Nougat eingebaut und ist nur für *Daydream-ready* Smartphones, wie das Google Pixel erhältlich. Neben der Halterung befindet im Daydream Kit ein Controller. Die Bewegungen dieses Controllers werden vom Headset erkannt und in die virtuelle Welt übertragen.

In Kapitel 4 werden die Auflösung, Bildqualität, Farbtiefe, Latenzen und ähnliche Faktoren dieser VR-Optionen genauer verglichen und untersucht.



(a) Google Cardboard
Quelle: Google.com

(b) Samsung GearVR
Quelle: Samsung.com

(c) Google Daydream
Quelle: Google.com

Abbildung 3.4: VR Headsets welche Smartphones als Display nutzen

3.1.3 Software

Virtual Reality findet immer mehr Anwendungsgebiete. Vor allem im Militär [VRSb] und in der Chirurgie [Bri16] wird es als Trainingsmethode genutzt. Weitere Anwendungsgebiete sind zum Beispiel die Therapie von unterschiedlichen Krankheiten [Par08], in der Automobilindustrie ein Mittel, den potentiellen Käufern einen tieferen Blick in das Fahrzeug zu gewährleisten [JOJ16], und beim Tourismus ein Weg, den Reisenden einen Vorgeschmack für unterschiedliche Zielorte zu bieten [Jul17]. Zudem trifft VR in der *Gaming*welt auf viel Begeisterung.

Gleich am Anfang dieser Arbeit war *Virtual Desktop* (siehe Abb. 3.5a) eine große Inspiration. Diese Anwendung erlaubt es den eigenen Desktop in VR zu nutzen. Im virtuellen Raum werden die geöffneten Fenster, Programme und 2D-Spiele auf eine unsichtbare Zylinder/Kugel-artige Oberfläche projiziert. Der Benutzer befindet sich innerhalb dieser Form und kann die VR-Controller als Maus benutzen. Der Hintergrund kann zu einer beliebigen Umgebung verändert werden.

Google Earth VR erlaubt seit Ende 2016 die Erkundung der gesamten Erde in VR (siehe Abb. 3.5b). In größeren Städten sind hierbei nicht bloß die flachen Satellitenaufnahmen – wie sie von *Google Maps* bekannt sind – verfügbar, sondern auch dreidimensional gerenderte Gebäude. Der Benutzer kann entweder durch Betätigen des Touchpads über die Welt fliegen oder mit Hilfe des Zeigers die Welt unter sich vorbeiziehen.

Google Tilt Brush lässt sich am besten Beschreiben als *Microsoft Paint* in VR (siehe Abb. 3.5c). Die Zeichnungen schweben im dreidimensionalen virtuellen Raum. Bei Tilt Brush ist vor allem das Interaktionskonzept für diese Arbeit von Interesse. In der linken Hand des Benutzers befindet sich eine würfelförmige Oberfläche, welche dem Benutzer Zugriff auf Optionen, wie die Farbpalette, Zeichenwerkzeuge und Speicher-Funktionen, bietet.



(a) Virtual Desktop
Quelle: VRDesktop.net



(b) Google Earth VR
Quelle: VR.Google.com



(c) Google Tilt Brush
Quelle: TiltBrush.com

Abbildung 3.5: Relevante Virtual Reality Anwendungen

3.2 Immersion und Mensch-Maschine Interaktion

Die Interaktivität spielt für die Immersion in VR eine große Rolle. Besonders bei *room-scale* und *seated* VR Anwendungen ermöglicht das direkte Greifen und Bewegung von virtuellen Objekten eine starke Verbindung zwischen der realen und der virtuellen Welt (siehe Abb. 3.6). Dieses Unterkapitel legt die Grundlagen für diese Themen, in Kapitel 5 werden sie genauer behandelt.



(a) Mini Golf in VR



(b) Mädchen spielt mit mechanischem Tier

Abbildung 3.6: VR verbindet die reale mit der virtuellen Welt

Quelle: Valve (Youtube.com)

3.2.1 Interaktion

Wegen der Neuartigkeit des Konsumenten-VR gibt es keine standardisierten Interaktionskonzepte für jeden Aspekt von VR. Universell vorzufindende Konzepte wie *Strg + C = Kopieren* existieren für VR nicht. VR-Entwickler müssen zurzeit noch für sich selbst entdecken, welche Interaktionen für ihre Anwendungen oder Spiele am angebrachtesten sind.

Es gibt einige Konzepte, die mittlerweile häufiger vorzufinden sind, wie zum Beispiel das *Arc-Teleport* für die Fortbewegung über weite Distanzen. Hierbei wird beim Betätigen eines bestimmten Knopfes ein Bogen vom Controller Richtung Boden projiziert. Lässt der Benutzer den Knopf los, so wird er augenblicklich auf die gezielte Position teleportiert. Trotz dessen werden in den meisten Spielen und Anwendungen andere Konzepte genutzt, wie das Anhalten eines Knopfes, um nach vorne zu Schweben oder das Ziehen des Bodens, um die Welt unter sich zu Bewegen.

Zudem gibt es noch viele Unterschiede zwischen den VR-Hardwareangeboten der zahlreichen Hersteller (siehe Abb. 3.7). Die Hersteller bieten entweder ihre eigenen Eingabegeräte an oder geben lediglich an, Konsolen Controller, wie den der Xbox One zu nutzen. Es gibt jedoch keine festgelegten Standardformen, wie sie beispielsweise bei herkömmlichen Tastaturen oder Mäusen bekannt sind.

Weiterhin unterstützt nicht jede VR-Plattform die gleichen Freiheitsgrade. Die verschiedenen Freiheitsgrade reichen von komplettes *room-scale VR* (beispielsweise mit dem HTC Vive) über *seated VR* (Oculus Rift und PS-VR) bis zu nur Erkennung der Kopfdrehung (Google Cardboard). Diese Einschränkungen führen wiederum dazu, dass unterschiedliche Interaktionskonzepte aufgestellt werden müssen.

Room-scale VR erlaubt die Bewegung im gesamten realen Raum. Jeder Schritt, jede Hand- und jede Kopfbewegung des Benutzers wird direkt in die virtuelle Umgebung übernommen.

Seated VR ist ähnlich, jedoch eingeschränkter. In *seated VR* werden lediglich Hand- und Kopfbewegungen übernommen.

VR-Plattformen wie das Google Cardboard erlauben nur noch das Erkennen von *Kopfdrehungen*. Das heißt genauer, dass der Benutzer zum Beispiel nicht mit Hilfe einer Kopfbewegung an einem Hindernis vorbeischaun könnte.



Abbildung 3.7: Beispiele unterschiedlicher VR Eingabegeräte

3.2.2 Immersion

Bei der Immersion bzw. der Presence handelt es sich um das Gefühl, sich in der virtuellen Welt physisch präsent zu fühlen. Es gehört zu den wichtigsten Faktoren, die bei VR erreicht werden können und sollen. Anders als bei einem herkömmlichen Computer-Displays, werden die Seh- und Hörsinne des Benutzers komplett auf die virtuelle Welt eingeschränkt.

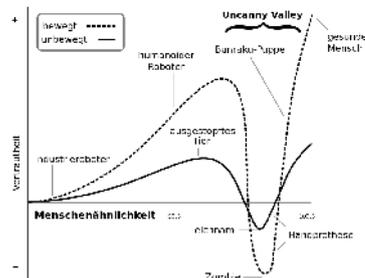
Das Bedürfnis nach der Gestaltung von Immersion ist dadurch begründet, dass der Benutzer seine gesamte Aufmerksamkeit auf die virtuelle Welt und die sich darin befindlichen Aufgaben fokussiert. Bei Spielen wird dadurch versucht den Spielspaß zu erhöhen. In der Industrie wird versucht die Produktivität durch diese Fokussierung zu steigern.

Hierbei ist es wichtig, dem Benutzer eine überzeugende Welt zu präsentieren. Dies bedeutet jedoch nicht gezwungenermaßen, dass sie realistisch sein muss. Sie muss für den Benutzer glaubwürdig wirken. In der Robotik existiert ein Effekt mit der Bezeichnung *Uncanny Valley*. Beim Versuch einen Roboter immer Menschen-ähnlicher zu gestalten, wird ein Punkt erreicht, bei dem der Realismus stark sinkt. In Abbildung 3.8a ist Professor Hiroshi Ishiguro (rechts) mit seinem humanoiden Roboter *Zwilling* (links) zu sehen. Sein *Zwilling* wirkt beinahe echt. Er gibt dem Betrachter jedoch ein unwohles Gefühl, was auslöst, dass die Akzeptanz des Betrachters stark sinkt. Dieses Sinken der Akzeptanz wird als *Uncanny Valley* bezeichnet (siehe Abb. 3.8b).

Dieser Effekt existiert auch in VR. Viele Spiele, wie beispielsweise *Job Simulator*, umgehen dies mit Hilfe einer *comichaften* Darstellung der virtuellen Welt. Sie erlauben, trotz des simplen Aussehens, einen hohen Grad an Immersion (siehe Abb. 3.8c).



(a) Hiroshi Ishiguro mit seinem humanoiden *Zwilling*
Quelle: Telegraph.co.uk



(b) Darstellung des Uncanny-Valley Effekts
Quelle: Tobias K. (de.Wikipedia.org)



(c) *Comichaftes* Aussehen des Spiels *Jobsimulator*
Quelle: JobSimulatorGame.com

Abbildung 3.8: Das Uncanny Valley in VR

3.2.3 VR-Krankheit

Ähnlich wie bei der Schifffahrt die Seekrankheit gibt es in Virtual Reality die VR-Krankheit. Ausgelöst wird diese durch eine Abweichung zwischen der Propriozeption (der körperlichen Selbstwahrnehmung) und dem was der Benutzer visuell wahrnimmt. Die Symptome der VR-Krankheit sind unter anderem Übelkeit, Schwindelgefühl und Kopfschmerzen. Vor der starken Verbreitung von VR war dieses Phänomen auch bekannt als Simulatorkrankheit. Die Anfälligkeit für das Vorkommen der Symptome variieren von Person zu Person.

Bei der Gestaltung von Interaktionskonzepten in VR muss sehr stark darauf geachtet werden, dass diese keine Übelkeit auslösen. Wird beispielsweise der virtuelle Kopf des Benutzers bewegt ohne, dass er selbst diese Bewegung macht, kann dies zum sofortigen Verlust des Gleichgewichts führen. Sogar, wenn als Lokomotionskonzept das Betätigen eines Knopfes oder das Bewegen des Joysticks festgelegt wird, kann dies den gleichen Effekt haben. Google Earth VR unterdrückt das Schwindelgefühl dadurch, dass das Sichtfeld des Benutzers eingeschränkt wird. Dies hat zur Folge, dass der Benutzer trotz der Bewegung einen festen Anhaltspunkt (den verdeckten Bereich) hat.

»And for god's sake, keep the horizon level!
The shortcut to puke-town
is to tip and tilt the horizon irrespective of reality«
*Jesse Schell, VRDC '16*¹

Youtube Nutzer *Super Bunnyhop* beschreibt das Phänomen in seinem Video² mit einer sehr greifbaren Schilderung. Bewegungen in VR werden in vielen Fällen nur visuell wahrgenommen. Das Gleichgewichtsorgan im Ohr, das dafür zuständig ist, die physischen Bewegungen wahrzunehmen, erhält eine widersprüchliche Information, da sich der Benutzer im gegebenen Fall nicht selbst bewegt. Diese Inkonsistenz der unterschiedlichen Wahrnehmung führt dazu, dass das Gehirn daraus schließt, dass es möglicherweise halluziniert. Die naheliegendste Ursache dafür ist, dass die letzte Mahlzeit vergiftet wurde. Dieses *nicht-existierende Gift* wird schließlich versucht durch Übelkeit auszustoßen. Google liefert, in seiner Anleitung zur Entwicklung für das Google Cardboard³, eine ähnliche Erklärung.

Neben den Interaktionskonzepten muss bei VR Anwendungen auch auf die technischen Aspekte geachtet werden. VR-Krankheit kann auch durch ein zu niedriges oder zu hohes Blickfeld, eine zu niedrige Bildwiederholrate und ähnlichem ausgelöst werden.

Eine tiefere Untersuchung der Interaktions-, Immersions- und VR-Krankheits-Aspekte folgt in Kapitel 5.

¹ Jesse Schell, VRDC '16: <http://www.gdcvault.com/play/1023671/Lessons-Learned-from-I-Expect>

² Motion Sickness in VR: https://www.youtube.com/watch?v=x3mMc_0_UtU

³ Designing for Google Cardboard: <https://www.google.com/design/spec-vr/designing-for-google-cardboard/physiological-considerations.html>

3.3 Assistenzsysteme für die Bildauswertung

In dieser Sektion werden die Begriffe Bildauswertung und Assistenzsystemen erklärt. In diesem Zusammenhang wird auch das Bildauswerteprogramm *ViLand* präsentiert.

Assistenzsysteme Bei Assistenzsystemen handelt es sich um Programme (oder auch ganze Systeme), welche den Menschen bei der Durchführung einer Arbeit unterstützen. Allgemein bekannt sind Systeme, wie beispielsweise Fahrerassistenzsystem, welche als Ziel haben die Sicherheit und den Fahrkomfort zu steigern. Hierzu gehören Techniken, wie Einparkhilfen, automatische Abstandswarner, Spurverlassenswarnung und unterschiedliche Notbremssysteme.

Der Benutzer wird somit von der Last der Aufgaben, welche computerisiert durchgeführt werden können, befreit. Er kann sich dadurch besser auf seine Aufgaben – welche nicht automatisiert werden kann – konzentrieren.

Bildauswertung Für diese Arbeit ist der Bildauswertungsbereich der bildgestützten Aufklärung von Interesse. Hierbei handelt es sich um das Verfahren zur Gewinnung von Informationen anhand von Luftbildaufnahmen. Bei der Bildauswertung werden nicht nur optische Bilder analysiert, sondern auch Bilder, die von nahen Infrarot-, Radar- oder Mikrowellensensoren aufgenommen wurden (siehe Abb. 3.9). Diese erlauben unterschiedliche Vorteile. Beispielsweise können Radarsensoren Wolken durchdrängen. Pflanzen absorbieren Wellen des sichtbaren Wellenbereichs für ihre Photosynthese. Ihre Zellenstruktur reflektiert Wellen des nahen Infrarotbereichs jedoch stark. Aus diesem Grund sind NIR-Aufnahmen hilfreich, wenn die Dichte von Vegetation untersucht werden soll.

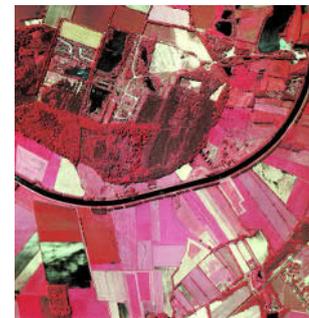
Die Aufgabe des Bildauswerters ist es, die Aspekte der Bildauswertung durchzuführen, die nicht zuverlässig von Computern automatisiert werden können.



(a) Optische Luftbildaufnahme
des Fraunhofer IOSB
Quelle: Maps.Google.com



(b) TerraSAR-X-Aufnahme der
Weihnachtsinsel
Quelle: DLR.de



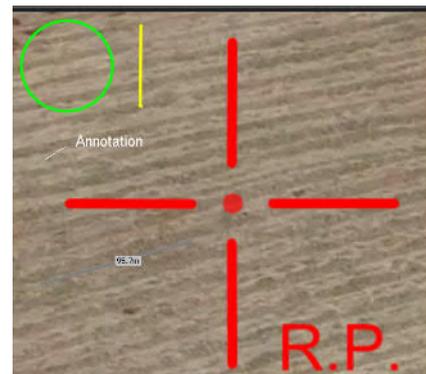
(c) CIR-Luftbild des
Elbe-Lübeck-Kanals
bei Siebeneichen
Quelle: GeoDZ.com

Abbildung 3.9: Unterschiedliche Arten von Luftbildaufnahmen

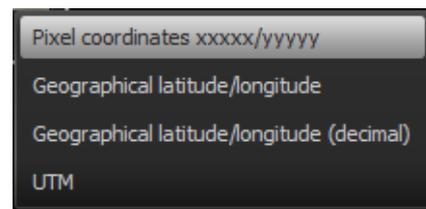
ViLand - Assistenzsystem für die Bildauswertung Bei *ViLand* handelt es sich um ein vom Fraunhofer IOSB entwickeltes Assistenzsystem für die Bildauswertung (siehe Abb. 3.10). Mit Hilfe von *Annotationen*, kann der Bildauswerter Markierungen auf strategisch wichtige Punkte legen, beziehungsweise die Größe von Objekten einschätzen (siehe Abb. 3.11). Zusätzlich lassen sich Werte wie Farbhelligkeit, -kontrast und -intensität der Aufnahmen anpassen. *ViLand* unterstützt zudem E-Learning Aspekte zur Ausbildung von Bildauswertern.



Abbildung 3.10: Ausschnitt des ViLand Interfaces im Überblick

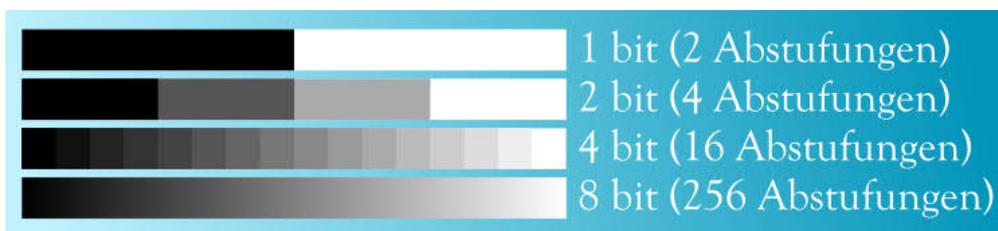


(a) Unterstützte Annotationsmöglichkeiten

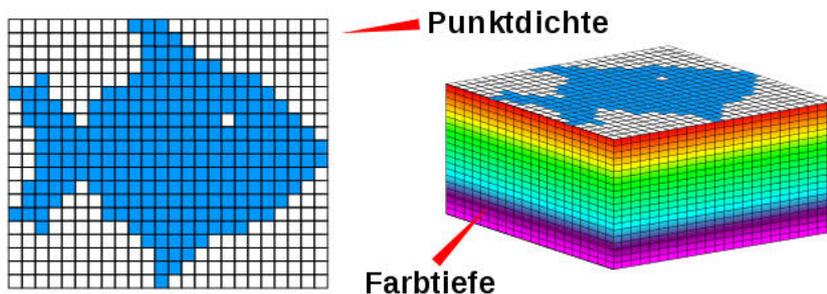


(b) Anzeige-Formate der Geokoordinaten

Abbildung 3.11: Ausschnitte aus dem Bildauswertungs-Programm ViLand



(a) Unterschiedliche Farbtiefen Abstufungen von Grauwerten
Quelle: Thomas R. Schwarz (de.Wikipedia.org)



(b) Die Farbtiefe gibt zusammen mit der Pixeldichte die maximale theoretisch erreichbare Bildqualität an
Quelle: Friedrich Graf (de.Wikipedia.org)

Abbildung 3.12: Qualitätsmerkmale von Bildausgabegeräten

3.4 Wahrgenommene Qualität von Bildausgabegeräten

In dieser Sektion werden die unterschiedlichen Faktoren, welche die Qualität der Darstellung und die Qualität der Betrachtung von Bildern beeinflussen, erklärt. Zudem wird ein Überblick zu den wichtigsten Softwarepaketen, welche für diese Arbeit genutzt wurden, gegeben.

3.4.1 Bilddarstellungsqualität

Bildauflösung Die Auflösung eines Monitors gibt an, wie viele Pixel (in der Breite und in der Höhe), dieser Monitor darstellen kann. Zeigen zwei unterschiedlich große Monitore die gleiche Anzahl an Pixel an, so hat der kleinere Monitor eine höhere *Pixeldichte*. Es werden also für einen gegebenen Bereich – zum Beispiel 1 x 1 Zoll – mehr Pixel angezeigt. Bei Bildausgabegeräten, welche auf weite Distanz betrachtet werden, ist dieses Maß nicht von größerer Bedeutung. Geräte, die jedoch auf sehr nahe Distanz betrachtet werden, benötigen eine hohe Pixeldichte. Zu diesen Geräten gehören Smartphones und auch VR-HMDs.

Gamut und Farbtiefe Das Gamut gibt die Anzahl an darstellbaren Tönen eines Bildausgabegerätes an. Handelsübliche Monitore unterstützen sRGB. Viele Faktoren beeinflussen, wie *gut* ein Monitor unterschiedliche Farben darstellen kann. Monitore die beispielsweise RGB LED Technologie unterstützen, haben eine breitere Farbpalette alleine dadurch, dass sie eine reinere Lichtquelle nutzen. Für uns von Interesse ist vor allem die Farbtiefe (siehe Abb. 3.12). Sie gibt an, wie viele unterschiedliche Helligkeits- und Farbwerte voneinander differenziert werden können. Diese wird in *bits* gemessen. Handelsübliche Monitore unterstützen 24-bit – also 8-bit für jeweils grün, rot und blau. In der Medizin und auch in der Bildauswertung werden oft höhere Standards gesetzt und Monitore, die bis zu 16-bit pro Farbe unterstützen, erwartet.

3.4.2 Aspekte des Betrachters

Neben den soeben erklärten Spezifikationen, gibt es weitere Faktoren, die das VR-Erlebnis beeinflussen. Diese werden im Folgenden dargelegt.

Field Of View - FOV Das FOV gibt in Winkel-Maße an, wie groß der sichtbare Bereich eines Betrachters ist. Zu beachten ist, dass Augenbewegungen das FOV nicht beeinflussen. Dieses Maß unterscheidet sich erheblich zwischen den unterschiedlichen Lebewesen (siehe Abb. 3.13a). Wird das FOV also vom VR-HMD eingeschränkt, ist auch ein Teil der möglichen Immersion eingeschränkt.

Ein höheres FOV kann entweder dadurch erreicht werden, dass die Linsen des HMDs näher an die Augen gebracht werden, oder dadurch, dass die Dicke der Linsen erhöht wird. Dies hat jedoch den Nebeneffekt, dass das Bild verzerrt wird. Diese Verzerrung ist vor allem an den Randbereichen der Linsen zu erkennen. Hersteller wie Oculus und HTC versuchen ein Gleichgewicht zwischen Dicke und Distanz zu erreichen, in dem sie Fresnel-Linsen nutzen. Fresnel-Linsen besitzen eine spezielle, deutlich volumen- und massenreduzierende Bauform.

Interpupillary distance - IPD Der Pupillen Abstand (interpupillary distance - IPD) unterscheidet sich von Person zu Person. Eine Reihe von HMDs unterstützen die Feinjustierung der Linsen, damit sie auf die individuellen Distanzen anpassbar sind. Wird diese Justierung nicht vorgenommen, so kann das Bild verzerrt wirken, oder der 3D-Effekt verloren gehen, oder der Benutzer erhält das Gefühl, er würde *schielen*.

Fovea centralis Das menschliche Auge erfasst bloß rund um das Zentrum des fixierten Blickpunkts (dem *Fovea centralis*) Informationen mit voller Schärfe. Gegenstände die sich nicht in diesem Bereich befinden, gehören zum peripheren Sehen und werden nur indirekt wahrgenommenen. Aus diesem Grund bietet es sich in der Computergrafik an, weniger Rechenleistung in das Rendering von Objekte, die außerhalb der Fovea centralis liegen, zu investieren. Dieses Verfahren hat den Namen *Foveated imaging*. Zurzeit wird diese Technik in VR nur eingeschränkt unterstützt. Da Augenbewegungen mit aktuellen VR-HMDs noch nicht erkennbar sind, können lediglich Kopfbewegungen genutzt werden um die unterschiedlichen Bereiche des Sichtfelds zu priorisieren. Nvidia unterstützt dieses eingeschränkte Vorgehen bereits nativ auf den neusten Karten.

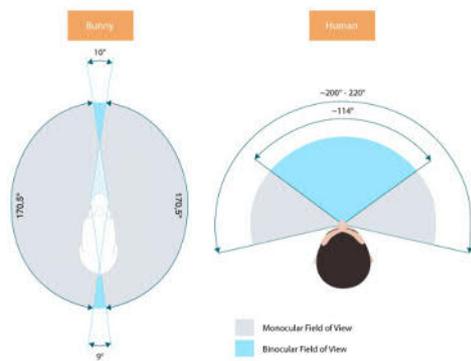
Zukünftige VR-HMDs, die Eye Tracking unterstützen, werden dieses Phänomen noch tiefgreifender ausnutzen können (Prototyp in Abb. 3.13b).

3.4.3 Begriffe der Computergrafik

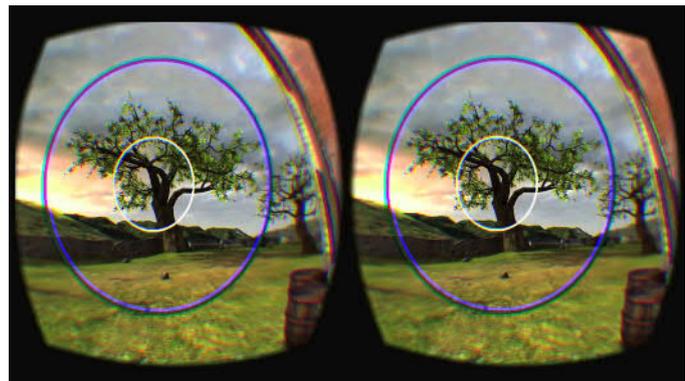
Mipmapping Beim *Mipmapping* handelt es sich um ein Verfahren, bei dem hochaufgelöste Bilder in mehrere gleichgroße Kacheln zerlegt werden. Dies wird für mehrere Stufen rekursiv durchgeführt. Jede dieser Stufe hat die Hälfte der Auflösung der vorherigen Stufen (siehe Abb. 3.14a). Dies wird fortgeführt, bis nur noch 1 x 1 Texel vorhanden ist. Bei einem Texel handelt es sich um ein einzelnes Pixel einer Textur. Zweck dieses Verfahrens ist es, Rauschen zu minimieren und Rechenkraft einzusparen. Abhängig von der Distanz zum texturierten Objekt, werden unterschiedliche Schichten (LOD) der Mipmap geladen.

Level of Detail - LOD Wurde bei einem Bild *Mipmapping* oder ein ähnliches Verfahren durchgeführt, so werden mit dem LOD die generierten Schichten unterscheidet. Die Quelldaten haben hierbei typischerweise das höchste LOD (LOD 0).

Filterung Neben Mipmapping ist Filterung eine zusätzliche Möglichkeit der Minimierung von Artefakten bei der Darstellung von Texturen. Wird keine Filterung vorgenommen (in manchen Fällen als *Point Sampling* oder *Point Filtering* beschrieben), so wird bei der Generierung von Bildern jedes Texel direkt als Pixel für das finale Bild übernommen. Dies kann zu vielen unerwünschten Effekten (z.B. Moirée-Muster, Aliasing, Rauschen, ...) führen. Diese Artefakte können mit Hilfe von Filterung minimiert werden (siehe Abb. 3.14b). Anstelle, dass jedes Texel direkt übernommen wird, werden die Nachbar-Textel zusätzlich betrachtet. Bei der bilinearen Filterung werden bis zu vier Texel betrachtet und gemittelt. Beim trilinearen Filter-Verfahren werden vier Pixel der ersten Mipmap Schicht betrachtet und vier der zweiten Schicht. Bei der Anisotropen Filterung werden bis zu 128 Texel in die Rechnung mit einbezogen.



(a) Das FOV eines Hasen und das eines Menschen
Quelle: vr-lens-lab.com

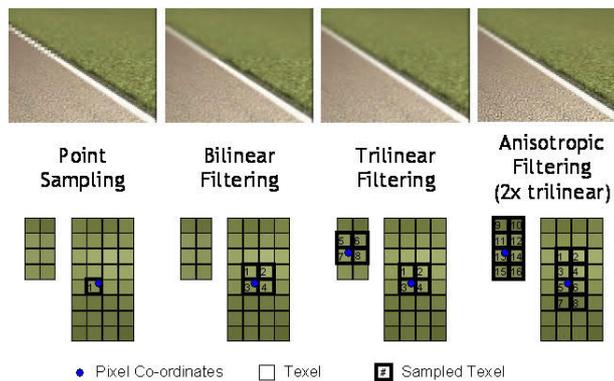


(b) SMI-Prototyp mit *Foveated Rendering*
Quelle: SensoMotoric Instruments (SMI) (Youtube.com)

Abbildung 3.13: Das Blickfeld in der Computergrafik



(a) Mipmapping bei einer Pflasterstein Textur
Quelle: TomsHardware.com



(b) Vergleich unterschiedlicher Filter-Verfahren
Quelle: cdrinfo.com

Abbildung 3.14: Computergrafik Verfahren

3.5 Software, Frameworks und Engines

Zur Entwicklung von VR-Anwendung bieten eine Reihe von Software-Paketen, Engines und Frameworks Abhilfe. Im Folgenden wird eine Übersicht zu den relevanten Hilfsmitteln geboten.

3.5.1 Game Engines

Unity3D¹ wird von Unity Technologies entwickelt und ist zur Zeit bei Version 5. Sie kann genutzt werden, um Spiele und Anwendungen für Computer, Konsolen, Smartphones und Webseiten zu erstellen. Die Grundbausteine eines in Unity entwickelten Programms sind die darin enthaltenen *GameObjects* und *Scripts*. *GameObjects* sind Objekte, welche in der Szene dargestellt werden können, wie zum Beispiel ein Haus. *Scripts* geben diesem Objekt Fähigkeiten, wie zum Beispiel das Öffnen eines Fensters. *Scripts* können in C# oder JavaScript geschrieben werden. Bei Unity3D handelt es sich um eine proprietäre Anwendung.

Unreal Engine² wird von *Epic Games* entwickelt und ist zurzeit bei Version 4. Genau wie Unity, unterstützt auch Unreal Engine eine lange Liste von unterschiedlichen Plattformen. Der wesentliche Unterschied zwischen beiden Engines sind die in Unreal genutzten Blueprints. Diese Blueprints erlauben die schnelle Entwicklung von Spiellogiken mit Hilfe von visueller Programmierung. Als Programmiersprache wird C++ genutzt. Unreal Engine ist eine proprietäre Anwendung.

3.5.2 Frameworks

SteamVR³ wurde von Valve und HTC als Unterstützung für Entwickler – zusammen mit dem HTC Vive System – veröffentlicht. Es bietet eine Reihe von API-Aufrufen und vorgefertigten Spiele Objekten, welche die Entwicklung von VR-Anwendungen zugänglicher gestaltet. Die Nutzung von SteamVR ist frei unter ihrer Steamworks VR API Code Lizenz⁴.

Virtual Reality Toolkit⁵ (VRTK) wurde von *TheStoneFox* auf Github als eine zusätzliche Unterstützung für Unity VR-Projekte veröffentlicht. Neben den in SteamVR schon erhältlichen API-Aufrufen, bietet VRTK zusätzlich vorgefertigte Schablonen für Lokomotion, Interaktion, Physik und Controller-Management. Zudem erlaubt VRTK eine hardwareunabhängigere Entwicklung von VR-Anwendungen. Die Nutzung von VRTK ist frei unter der MIT Lizenz.

1 Unity3D Engine: <https://unity3d.com/>

2 Unreal Engine: <https://www.unrealengine.com/>

3 SteamVR: <http://store.steampowered.com/steamvr>

4 SteamWorks VR API™ Code License:

<https://github.com/ValveSoftware/steamworks-vr-api/blob/master/LICENSE>

5 VRTK: <https://github.com/thestonefox/VRTK>

3.5.3 Weitere Anwendungen und genutzte Technologien

Blender¹ ist ein *OpenSource* Programm zur Gestaltung von kreativen Inhalten. Dies beinhaltet 3D-Objekt Modellierung, Simulation, Animation, Rendering, Motion Tracking und vieles mehr. Für diese Arbeit war besonders die 3D-Modellierung von Nutzen. Die Nutzung von Blender ist kostenlos unter der GNU General Public License (GPL).

Web Map Service² (WMS) ist ein Standardprotokoll zur Überlieferung von Karteninhalten. Typischerweise handelt es sich um georeferenzierte Luftbildaufnahmen, die ähnlich wie beim Mipmapping, mit mehreren LODs in einer Datenbank gespeichert sind. Die Daten werden als Rastergrafik (PNG, GIF oder JPEG) übermittelt. Die Nutzung des WMS ist frei, mit Kopierschutz³.

OpenStreetMap⁴ (OSM) ist ein kollaborative Projekt, welches als Ziel hat, die gesamte Erde als frei erhältliche Karte zur Verfügung zu stellen. Im Gegensatz zu WMS werden von OSM die Karten ähnlich wie Vektor-Daten behandelt. Unterschieden wird zwischen Knoten, Wegen, Verbindungen und Tags. Die Nutzung der von OSM gelieferten Daten ist frei unter der *Open Data Commons Open Database* Lizenz (ODbL).

1 Blender: <https://www.blender.org/>

2 OpenGIS® Web Map Service Interface Standard: <http://www.opengeospatial.org/standards/wms>

3 OGC Document Notice: <http://www.opengeospatial.org/ogc/document>

4 OpenStreetMap: <http://www.openstreetmap.org>

Bildqualität in VR

In diesem Kapitel wird die Bildqualität von aktuellen VR Geräten untersucht. Hierzu werden im ersten Teil visuelle Artefakte beschrieben, die bei der Nutzung von VR-HMDs auftreten können. Als Nächstes werden im zweiten Teil die Spezifikationen aktueller VR-System verglichen und diskutiert. Im Anschluss wird ein systematisches Experiment beschrieben, bei dem die subjektiv wahrgenommene Bildqualität des HTC Vive-HMDs in verschiedenen Testszenarien untersucht wurde.

Der Fokus dieses Kapitels liegt – wie in Abb. 4.1 blau markiert – beim Übergang von einem vorbereitetem Bild bis zu den Augen des Betrachters.

Hierbei ist zu beachten, dass Faktoren wie die Einstellungen der Darstellungssoftware (z. B. Game Engine), die Qualität des Bildausgabegeräts (z. B. HTC Vive) und weitere Beeinträchtigungen die subjektiv wahrgenommene Bildqualität beeinflussen können.

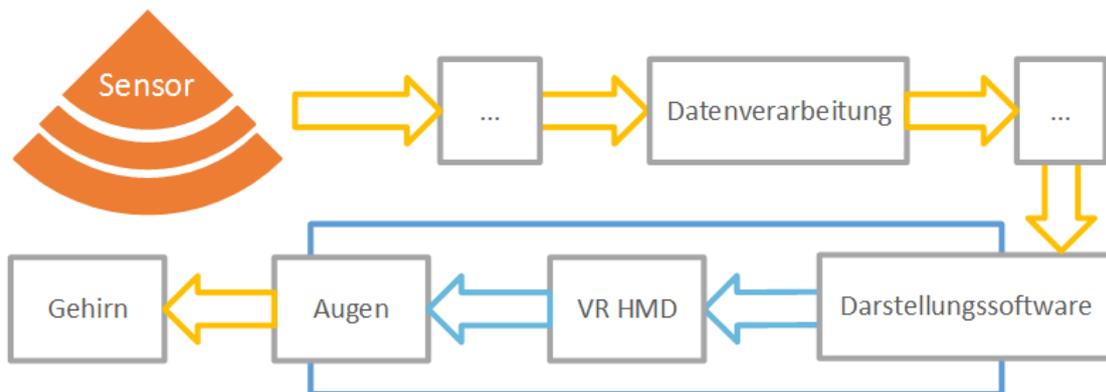


Abbildung 4.1: Vereinfachter Ablauf einer Aufnahme vom Sensor zum Gehirn des Betrachters

4.1 Effekte und Probleme

Dadurch, dass die Displays bei der Nutzung von Virtual Reality nur wenige Zentimeter von den Augen entfernt sind, treten eine Reihe von Effekten auf, die bei der typischen Nutzung von Bildausgabegeräten nicht auffallen. In dieser Sektion werden die häufiger auftretenden visuelle Artefakte beschrieben und illustriert.

4.1.1 Fliegengitter Effekt

Auf Englisch wird dieser Effekt *Screen-Door Effekt* bezeichnet. Bei diesem Effekt handelt es sich um ein visuelles Artefakt, welches erkennbar ist, wenn die Abstände zwischen Pixeln sichtbar sind (siehe Abb. 4.2a). Auf heutigen Monitoren tritt dieser Effekt kaum mehr auf; problematisch ist er jedoch noch immer bei Beamern (siehe Abb. 4.2b). In VR tritt dieses Problem verstärkt auf, wegen der kurzen Distanz zwischen Augen und Display. Zusätzlich wird der Effekt in VR dadurch verstärkt, dass die Linsen das Bild *verzerren*, um somit ein höheres FOV zu erreichen. Eine höhere Auflösung oder eine höhere Pixeldichte kann diesen Effekt vermindern. In Abbildung 4.2c wird der Fliegengitter Effekt mit Hilfe einer überspitzten Illustration deutlicher dargestellt.

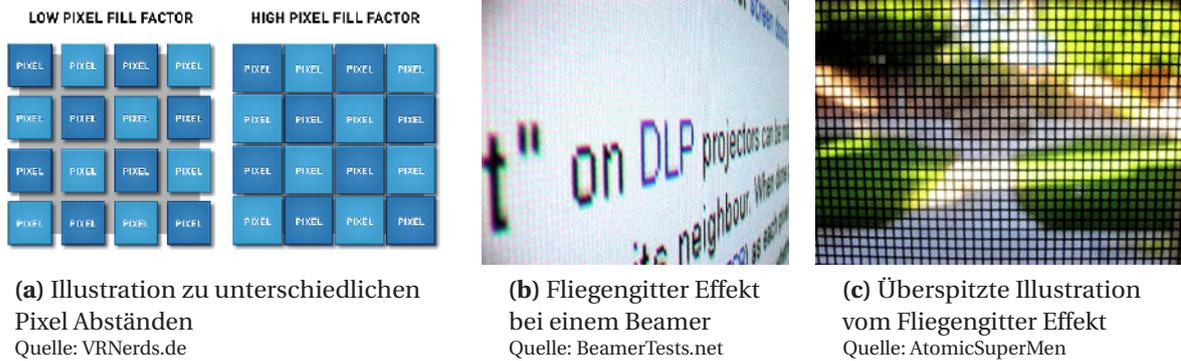


Abbildung 4.2: Fliegengitter Effekt

4.1.2 Ring Effekt

Auch *halo effect*, *glare* oder *lens flare* genannt. Dieses Artefakt entsteht durch die Oberfläche der Fresnel-Linsen. Der Ring Effekt ist üblicherweise vor allem an den Rändern des sichtbaren Bereichs zu erkennen. Deutlich zu erkennen ist dies in Abbildung 4.3b. Trotz dieses ungewollten Effekts, werden diese Linsen für VR genutzt, da sie eine deutlich volumen- und massenreduzierte Bauform besitzen. In Abbildung 4.3c wird der Querschnitt einer üblichen Linse, einer Fresnel-Linse und einer Mikrofresnel-Linse mit gleicher Brennweite verglichen.

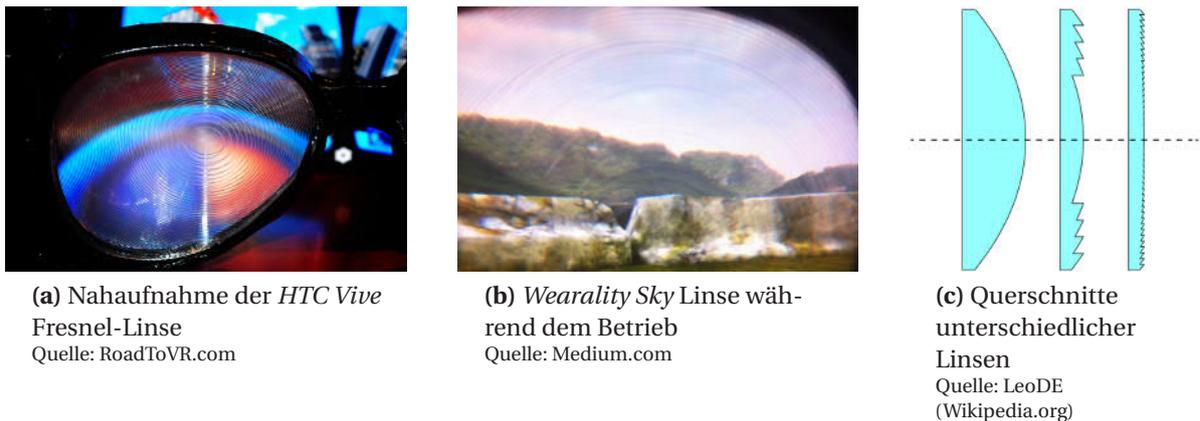


Abbildung 4.3: Ring Effekt

4.1.3 Light Rays

Dieser Effekt tritt am deutlichsten bei hohen Kontrastunterschieden auf. Einerseits verursacht dieser, dass dunkle Bereiche des Bildes, welche nahe an hellen Bereichen liegen, von diesen *gebleicht* werden. Zur Verdeutlichung siehe Abbildung 4.4a, welche diesen Effekt überspitzt darstellt. In Abbildungen 4.4b und 4.4c ist der tatsächliche Effekt zu sehen. Es scheint, als ob ein Lichtkegel den dunklen Bereich überdeckt. In Standbildern ist dieser Effekt nicht deutlich erkennbar. Ich empfehle, das Vorkommen des Effekts in einem Video zu betrachten (Beispielsweise: <https://vimeo.com/165662545>).



(a) Überspitzte Illustration der Light Rays
Quelle: RoadToVR.com

(b) Light Ray Effekt auf der Oculus Rift
Quelle: VRalf (Vimeo.com)

(c) Light Ray Effekt zu erkennen unter dem Fenster
Quelle: VRalf (Vimeo.com)

Abbildung 4.4: Light Ray Effekt

4.2 Vergleich der aktuellen Hardware

Zur Zeit sind sehr viele, unterschiedliche HMDs verfügbar und es scheint, als ob fast wöchentlich neue Hardware auf den Markt kommt. In dieser Sektion werden die Bildqualität der vier gängigsten HMDs untersucht.

In Tabelle 4.1 befindet sich eine Zusammenfassung der für uns wichtigsten Spezifikationen. In der Tabelle 4.2 werden die Hardware Spezifikationen der *nicht-Smartphone* gestützten System zusammengefasst. Für diese Zusammenfassung wurden die Wikipedia Einträge der einzelnen Teile und mehrere andere Webseiten (siehe Fußnote ¹) genutzt.

Die PS-VR unterscheidet sich in Punkt Displaytechnologie stark von den drei anderen Systemen. Der Unterschied zwischen herkömmlichen RGB Display und dem PenTile Display ist der Aufbau der Subpixeln (siehe Abb. 4.5). PenTile stützt sich hierbei auf die Funktionsweise des menschlichen Auges. Wird die Anzahl an grünen Subpixeln erhöht, so wird die Bildqualität nur kaum merklich reduziert [Oi]. Wegen des Aufbaus der PenTile Displays, tritt der Fliegengitter-Effekt verstärkt auf.

Obwohl die Auflösung des GearVR mit 2560×1440 deutlich höher ist, als die der anderen Systeme, ist die darstellbare Qualität deutlich schlechter. Dies liegt einerseits an der limitierten Rechenkraft

¹ Genutzte Quellen für die Tabellen:

https://xinreality.com/wiki/Pixel_density
<http://newatlas.com/best-vr-headsets-comparison-2016/45984/>
<http://ps4daily.com/2015/03/project-morpheus-vs-htc-vive-vs-oculus-rift/>
<http://www.digitaltrends.com/virtual-reality/oculus-rift-vs-htc-vive/>
<http://www.digitaltrends.com/virtual-reality/playstation-vr-vs-htc-vive-spec-comparison/>
<http://www.digitaltrends.com/virtual-reality/oculus-rift-vs-samsung-gear-vr/>
<https://www.vive.com/us/ready/>
<https://www.oculus.com/oculus-ready-pcs/>
<https://www.playstation.com/en-gb/explore/ps4/tech-specs/>

des Smartphones und andererseits am *Upscalen* das durchgeführt wird. Beim Upscaling wird ein niedrigauflösendes Bild groß gestreckt.

Sowohl die HTC Vive als auch die Oculus Rift können die 90 Hz ihrer Displays bei der Nutzung der empfohlenen Hardware voll unterstützen. Die für die PS-VR angegebenen maximal 120 Hz können nur durch *Reprojektion* erreicht werden. Bei der Reprojektion werden fehlende Frames nachgerechnet und eingesetzt. Die Vive- und die Rift-HMDs nutzen diese Technik auch, falls die 90 FPS zeitweise nicht erreicht werden können. Zu dem wird in diesen Fällen *Time Warp* aktiviert. Beim Time Warp handelt es sich um eine Technik, die genutzt wird, um die Latenz zu reduzieren. Das verarbeitete Bild wird verzerrt, um sich der neusten Kopfbewegung anzupassen.

	HTC Vive	Oculus Rift	PS-VR	Samsung GearVR mit Galaxy S6
Display Technologie	PenTile OLED	PenTile OLED	RGB OLED	PenTile OLED
Auflösung	2160 x 1200	2160 x 1200	1920 x 1080	2560 x 1440
Auflösung pro Auge	1080 x 1200	1080 x 1200	960 x 1080	1280 x 1440
Pixel-dichte	~456 PPI (pro Auge)	~456 PPI (pro Auge)	386 PPI	518 - 577 PPI
Aktualisierungsrate	90 Hz	90 Hz	60 - 120 Hz (mit Reprojektion)	60 Hz
Field of view	110 deg	~110 deg	100 deg	~101°
Gewicht	454 g	440 g	610 g	456 g
Tracking Latenz	22 ms	13 ms	18 ms	< 20ms

Tabelle 4.1: Zusammenfassung der Spezifikationen aktueller HMDs

	HTC Vive	Oculus Rift	PS-VR
Prozessor	Intel™ Core™ i5-4590 oder AMD FX™ 8350	Intel™ Core™ i5-4590 oder AMD FX™ 8350	x86-64 AMD "Jaguar", 8 cores
Grafikkarte	NVIDIA GeForce™ GTX 1060 oder AMD Radeon™ RX 480	NVIDIA GeForce™ GTX 970 oder AMD Radeon™ 290	4.20 TFLOPS, AMD Radeon™ based graphics engine
RAM	4 GB RAM	8GB	8GB

Tabelle 4.2: Zusammenfassung der empfohlenen Hardware Anforderungen

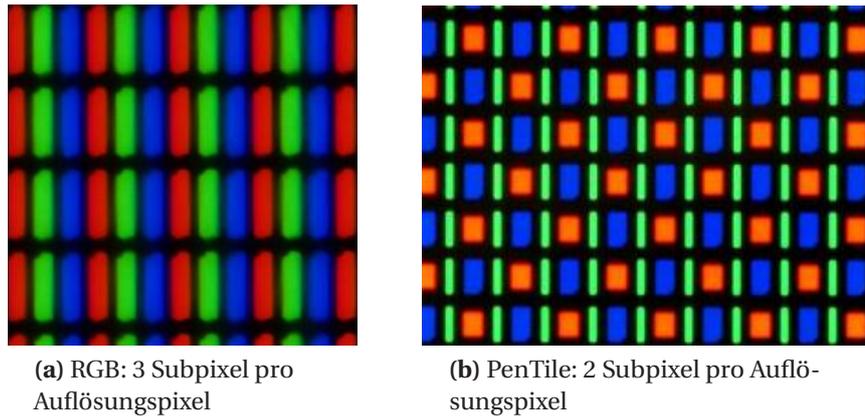


Abbildung 4.5: Vergleich der Subpixel-Muster
Quelle: oled-info.com/pentile

4.3 Bewertung zur wahrgenommenen Bildqualität in VR

Bei der Bildauswertung werden hohe Erwartungen an die Bildausgabegeräte gesetzt. In dieser Sektion wird untersucht, ob die derzeitige Generation von VR-HMDs diese Anforderungen erfüllen können. Bei der Darstellung den unterschiedlichen Sensoraufnahmen (optisch, Radar, NIR, ...) in VR birgt sich die Gefahr, dass die visuellen Artefakte, die zu Beginn dieses Kapitels beschrieben wurden, verstärkt auftreten. Des Weiteren ist fraglich, ob unterscheidet zwischen Farben, Formen und Kontrasten ausreichend erkennbar sind.

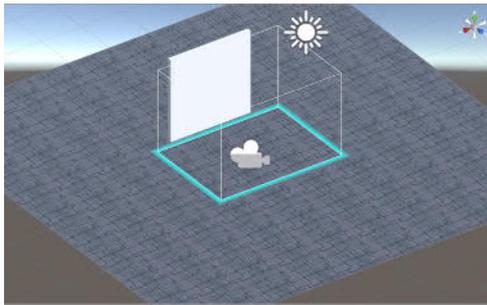
Aus diesem Grund wurden anhand von einem wissenschaftlichen Experiment eine Bewertung zur subjektiv wahrgenommenen Bildqualität durchgeführt. Diese untersucht das Ausmaß dieser Effekte und deckt eventuelle Gegenmaßnahmen auf. Besonders wurde hierbei darauf geachtet, ab welchem Abstand die Effekte auftauchen und ob noch weitere, nicht vorhergesehenen Effekte auftreten. Die Hypothese ist, dass obwohl die aktuellen VR-Displays nicht perfekt sind, die Qualität dennoch ausreichend zufriedenstellend ist für die Bildauswertung.

4.3.1 Versuchsaufbau

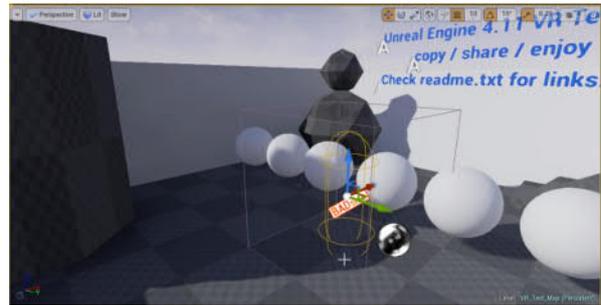
In diesem Teil wird die genutzte Software und Hardware dargelegt. Zusätzliche werden die importierten Texturen und deren Quellen angegeben.

Genutzte Hardware und Software Die Tests wurden mit Hilfe der HTC Vive auf einem Computer mit der Grafikkarte *GTX 1070* ausgeführt. Zur Bedienung wurden die dazu gehörigen Controller genutzt. Die Versuche wurden sowohl auf Unity3D (Version 5.4.2f2) als auch auf der Unreal Engine (Version 4.13.0) durchgeführt. Die Video-Ausgabe wurde mit *OBS Studio* aufgenommen. Hierbei wurde die Encoder-Einstellung *Nvidia NVENC* gewählt, welches nur einen minimalen Einfluss auf die Performance hat, da die Enkodierung direkt auf der Grafikkarte stattfindet.

Szenen Aufbau Für diese Untersuchung wurde jeweils eine Test-Szene für Unity und Unreal erstellt. Die Szenen enthielten lediglich einen Boden und eine Wand, auf die unterschiedlichen Texturen projiziert wurden. Somit konnten die Texturen durch einfaches Überschreiben der vorherigen Textur ersetzt und anschließend untersucht werden. Wichtig beim Szenen-Aufbau ist eine gleichmäßige Beleuchtung, da ansonsten Schatten die Ergebnisse beeinflussen könnten.



(a) Ausschnitt der Unity3D Szene



(b) Ausschnitt der Unreal Engine Szene

Abbildung 4.6: Überblick zu den beiden erstellten Szenen

Für die Unreal Engine Szene wurde ein Template¹ genutzt, da meine Erfahrung mit dieser Game Engine weitaus limitierter ist. Ausschnitte der beiden Szenen sind in Abb. 4.6 zu sehen.

Texturen Eine Vielzahl an Effekten wurden untersucht, daher war es wichtig, sehr unterschiedliche Texturen zu betrachten. So wurden beispielsweise Farbunterschiede und hohe Auflösungen mit farbenreichen, hochauflösten Bildern untersucht (Siehe Abb. 4.7a und 4.7b). Das Bild aus Abbildung 4.7a zeigt häufige direkte Übergänge zwischen Farben und schwarz und ruft unter anderem bei näherer Betrachtung die Effekte *light rays* und verschmierte Pixel hervor. Die Texturen aus dem Musterpaket *Pixel Gothic Lace* bieten beim Anzeige-Modus *tilde* eine hohe Entropie, da es viele Übergänge zwischen schwarz und weiß gibt. Hier würden Effekte wie *light rays* und *Fliegengitter* auftreten. Des Weiteren, lassen sich hier die unterschiedlichen Auswirkungen von den verschiedenen Filter-Modi (kein Filter, bilinear und trilinear) der beiden Engines untersuchen.

Die Quellen der unterschiedlichen Texturen befinden sich in der Fußnote².

¹ Unreal Engine 4.11 VR Template: <https://www.youtube.com/watch?v=Layx291UziQ>

² Quellen der genutzten Texturen aus Abb.: 4.7:

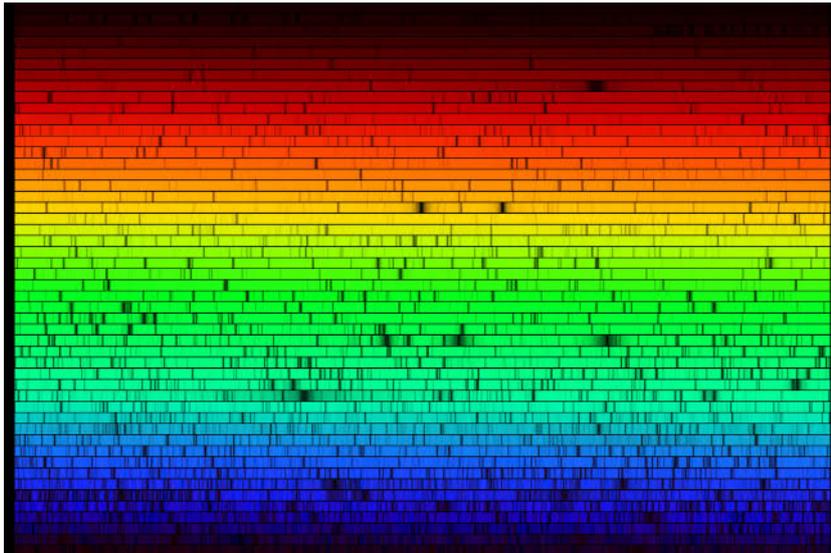
Spektrum der Sonne: https://www.noao.edu/image_gallery/html/im0600.html

Yosemite National Park - Bridalveil Fall: <https://wall.alphacoders.com/big.php?i=598721>

Pixel Gothic Lace Pattern Pack: <http://mirz123.deviantart.com/art/Pixel-Lace-Pattern-Pack-187559054>

Simplified RGB Color wheel: <http://www.psdgraphics.com/backgrounds/color-spectrum/>

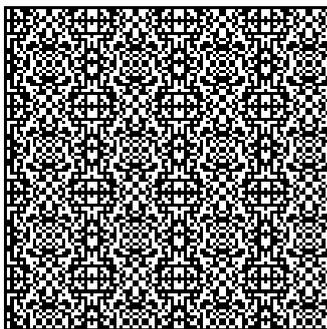
RGB Color Wheel: <http://printingcode.runemadsen.com/lecture-color/>



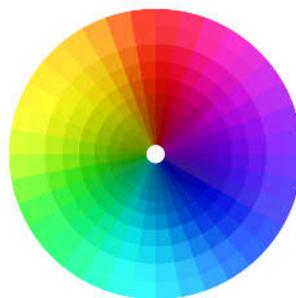
(a) Spektrum der Sonne



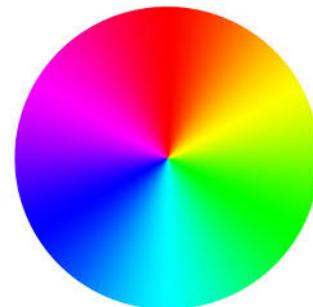
(b) Yosemite National Park – Bridalveil Fall



(c) Pixel Gothic Lace
Pattern Pack



(d) Color Spektrum



(e) Simplified RGB Color wheel

Abbildung 4.7: Beispiele der Texturen, die bei diesem Experiment genutzt wurden

4.3.2 Beobachtung

Die Textur aus Abb. 4.7b hat eine Auflösung von 10800 x 7200 Pixel. Sie konnte sowohl in Unity als auch in Unreal erst nach mehreren Anpassungen der Einstellungen in der höchstmöglichen Auflösung von 8192 x 8192 Pixeln importiert werden.

Sparkles Ursprünglich wurde Point-Filtering (also kein Filter) genutzt, damit keine Pixel bei der Filterung verschmiert werden. Dies führte jedoch bei mehreren Texturen zum sogenannten *Sparkles*-Effekt. Hierbei wechseln die Pixel sehr oft ihre Farbwerte, da bei der Berechnung ihrer Werte immer wieder ein anderes Texel getroffen wird. Die Lösung zu diesem Problem war die Nutzung von Mipmapping. Hierbei werden die Texturen mehrmals in immer kleinerer Form als unterschiedliche Level of Detail (LOD) gespeichert.

Eine weitere Möglichkeit zur Vermeidung des Sparkles-Effekt ist die Nutzung von Filtern (z. B. bilinear oder trilinear).

Mip Mapping Bei der Nutzung der Textur aus Abbildung 4.7b ist zu erkennen, ab welchen Distanzen die unterschiedlichen LODs der Mipmap geladen werden (siehe Abb. 4.8). Dieser Effekt konnte nur mit sehr hochauflösten Texturen repliziert werden. Texturen, mit niedriger Auflösung, welche mehrmals hintereinander angezeigt – *getiled* – wurden, wiesen diesen Effekt nicht auf.



Abbildung 4.8: Unterschiedliche LOD eines Bildausschnittes

Light Ray Der Light Ray Effekt konnte nicht so deutlich repliziert werden, wie er von den Benutzern der ersten Oculus Rift Dev-Kits berichtet wurde. Dennoch war er auffälliger, wenn das HMD nicht fest genug am Kopf angepasst wurde.

Fliegengitter-Effekt Der Fliegengitter-Effekt war bei der Betrachtung von Text auf längere Distanzen sehr auffällig. Nähert man sich jedoch dem Text, so wird er deutlich lesbar. Der Effekt tritt bei Betrachtung von Texturen nicht so auffällig auf, wie erwartet.

Ring-Effekt Der Ring-Effekt ist besonders an den Rändern der Linsen zu erkennen. Es empfiehlt sich also zum umher schauen nicht die Augen zu bewegen, sondern den Kopf zu drehen.

IPD Eine falsche Einstellung des IPD-Rädchen am HMD löste zudem den Effekt aus, dass unterschiedliche Objekte im dreidimensionalen Raum schwer erkennbar werden, abhängig von ihrer Distanz zum Betrachter. Der Betrachter hat das Gefühl, er würde *schielen*.

Unterschiede zwischen Engines Beide Engines liefern sehr ähnliche Resultate, sofern die gleichen Einstellungen vorgenommen werden. Ein auffälliger Unterschied ist, dass in Unreal die Texturen bei sehr naher Betrachtung *plastischer* also echter wirken.

4.3.3 Auswertung

Die Effekte, die im ersten Teil dieses Kapitels erklärt wurden, traten bei dem systematischen Experiment nicht so auffällig auf wie erwartet.

In den meisten Fällen ließen sie sich durch korrekte Einstellung der Engines vermeiden. Besonders bei der Betrachtung von hochaufgelösten Texturen muss stark auf die Filterung und das Mipmapping geachtet werden. An diesem Punkt vermerke ich des Weiteren, dass die Schwierigkeiten des Mipmapping für VIEW weniger stark eingreifen, da wir für dessen Implementierung SlippyMaps benutzen, welche ähnlich wie Mipmaps funktionieren.

Zusätzlich ist zu beachten, dass die Art, wie der Benutzer das HMD aufsetzt, einen großen Einfluss auf die wahrgenommene Bildqualität hat. Ist das HMD zu locker befestigt, so tritt besonders der *Light Ray*-Effekt sehr deutlich auf.

Der Fliegengitter-Effekt ist besonders bei der Betrachtung von Texten auf ferner Distanz aufgetreten. Es ist in dieser Hinsicht ratsam, entweder die Texte genügen groß darzustellen oder dem Benutzer eine Möglichkeit zu bieten, näher an die Texte heran zu treten.

Zudem ist es, bei der Nutzung von VR-Anwendungen wegen der Bauform der Linsen und dem nicht komplett umgesetzten Foveated Rendering, ratsam nur mit Hilfe von Kopfbewegungen den Raum zu betrachten. Die Ränder der Fresnel-Linsen verursachen Verzerrungen, die für den Nutzer störend wirken können.

4.4 Diskussion

In diesem Kapitel wurden die visuellen Effekte, die bei der Nutzung von VR-System auftreten können, untersucht. Anschließend wurden die Spezifikationen der aktuell-erhältlichen VR-HMDs verglichen. Die unterschiedlichen Systeme bieten unterschiedliche Vorteile. Die PS-VR bietet dank der FullRGB Displaytechnologie ein etwas besseres Bild, unterstützt dabei jedoch nur limitierte Rechenkraft. Die HTC Vive bietet gegenüber der Oculus Rift bessere Interaktionsmöglichkeiten, hat jedoch eine höhere Latenz. Im Anschluss wurde eine Bewertung der wahrgenommenen Bildqualität mit dem HTC Vive Headset durchgeführt. Hierbei wurde geschlossen, dass die visuellen Artefakte nicht so deutlich, wie erwartet aufgetreten sind. Zudem wurde erkennbar, dass die korrekte Einstellung der Engine und die Befestigung des HMDs am Kopf einen großen Einfluss auf die wahrgenommene Qualität hat.

4.5 Fazit

Das Fazit bei der Untersuchung der Bildqualität von aktuellen HMDs ist, dass obwohl die subjektiv wahrgenommene Qualität noch nicht auf dem Stand der aktuellen Desktopmonitore ist, sie dennoch gebrauchstauglich ist. Zu beachten ist außerdem, dass das Gesamtziel dieser Arbeit die Hardware-unabhängige Konzeption der Bildauswertung in VR ist. Das heißt genauer, dass, auch wenn die Bildqualität zur Zeit nicht ausreichend wäre, dies nicht den Fortschritt der Arbeit hindern würde. In Zukunft werden HMDs mit weitaus besseren Spezifikationen auf den Markt kommen, wie z. B. das 8K HMD von PiMax¹.

¹ PiMAX's 8k Headset: <http://www.roadtovr.com/hands-pimaxs-8k-headset-proves-high-fov-vr-coming/>

5

Gestaltung von Immersion und Interaktion in VR

In Kapitel 2 und 3 wurde bereits dargelegt, dass es bei der Gestaltung von VR-Anwendungen wichtig ist, auf die menschlichen Faktoren einzugehen. In diesem Kapitel werden diese Faktoren und die dazugehörigen Paradigmen untersucht.

Im ersten Teil wird erklärt, aus was Immersion besteht und welche Anforderungen bei der Gestaltung von VR-Anwendungen beachtet werden müssen.

Im zweiten Teil wird der Aspekt der Interaktion in VR behandelt. Dazu gehört eine Untersuchung der vorhandenen Interaktionskonzepte und die Gestaltung und Zusammensetzung von neuen Interaktionskonzepten für die Bildauswertung in VR.

5.1 Immersion in die virtuelle Realität

Die Immersion wird von mehreren Autoren in unterschiedliche Kategorien eingeteilt. Zu Beginn dieses Kapitels wird aufgezeigt, welche Konzepte zur Kategorisierung existieren. Anschließend wird dargelegt, welche Anforderungen an die Hardware und an die Entwickler bei der Gestaltung von VR-Anwendungen gestellt werden. Zum Schluss wird erklärt welche Konzepte bei der Umsetzung der VR-Umgebung für den Prototyp gestaltet werden.

5.1.1 Arten von Immersion

Das Konzept der Immersion lässt sich laut Ernet W. Adams in Gamasutra [Ada04] in drei – und laut Staffan Bjork und Jussi Holopainen in Patterns in Game Design [Bjo04] in vier – verschiedene Kategorien einteilen. Ich habe die Tabelle 5.1 erstellt, um diese Kategorien zusammenzufassen und eine kurze Beschreibung zu deren Bedeutung zu geben:

Ernest W. Adams	Staffan Björk und Jussi Holopainen	Beschreibung
Taktische Immersion	Sensor-motorische Immersion	Diese Arte der Immersion lässt sich mit dem <i>Flow</i> vergleichen, den Spieler verspüren, wenn sie eine schnelle Aufgabe mit Hilfe von Geschick oder Können bewältigen.
Strategische Immersion	Kognitive Immersion	Die strategische Art ist ähnlich wie die taktische Immersion, diese wird jedoch eher erfahren, wenn der Spieler eine mental herausfordernde Aufgabe löst.
Narrative Immersion	Emotionale Immersion	Diese Form der Immersion wird verglichen mit dem Gefühl, das verspürt wird, wenn jemand stark in die Geschichte eines Buches vertieft ist.
-	Räumliche Immersion	Die Räumliche Immersion wird wahrgenommen, wenn die simulierte Welt überzeugend wirkt - wenn sie echt wirkt.

Tabelle 5.1: Arten von Immersion laut Ernest W. Adams [Ada04], Staffan Bjork und Jussi Holopainen [Bjo04]

Für unser Projekt und die dazugehörige Studie sind hierbei die Kategorien zwei und vier von Interesse. Die taktische Immersion wäre von Interesse, wenn der Aspekt der *Gamification* zum Umfang der Arbeit gehören würde. Im Prototypen sind keine derartigen Features vorhanden. Sie wäre vor allem dann angesprochen, wenn die Aufgaben schnell in einer gewissen Zeitspanne vollbracht werden müssten. Nach mehreren Versuchen könnten die Spieler die Aufgaben flüssiger durchführen und möglicherweise in den *Flow* gelangen.

Gleichermaßen könnte die narrative Immersion von der *Gamification* profitieren. Wird die Story des Spiels überzeugend genug erzählt, kann auch diese narrative Immersion erreicht werden. Obwohl bei der Studie einer Story erzählt wird, liegt dieser Aspekt jedoch außerhalb des geplanten Umfangs der Arbeit.

Im Kapitel 2 wurde bereits erläutert, dass laut Schubert et al. [Sch01] die Immersion ein Faktor der Technik ist und dass, wenn von der subjektive wahrgenommenen Immersion gesprochen wird, eigentlich die Presence gemeint ist. Diese wird laut ihnen von drei Faktoren beeinflusst: *Spatial Presence*, *Involvement* und *Realness*. *Spatial Presence* gibt hierbei das Gefühl an, sich tatsächlich im virtuellen Raum zu befinden. Das *Involvement* gibt an, wie vertieft der Nutzer in die Aufgabe ist. Die *Realness* sagt aus, wie real die virtuelle Welt auf ihn wirkt. Für unsere Arbeit sind in diesem Fall die Faktoren *Spatial Presence* und *Involvement* von Interesse.

Thornson et al. sprechen von einer *Tendanzion zur Presence* und teilen diese in sechs ähnliche Unterkategorien ein (*Spatial Orientation*, *Ability to Construct Mental Models*, *Active Cognitive Involvement*, *Passive Empathy Cognitive Involvement*, und *Introversion*) Von Interesse sind hierbei die *Spatial Orientation* und das (active und passive) *Cognitive Involvement*.

Es zeigt sich, dass es viele Arten gibt, die Immersion/Presence zu beschreiben. Was schlussendlich für diese Arbeit von Relevanz ist, die Existenz des Unterschieds zwischen Immersion und Presence. Es handelt sich laut Barfield um das subjektiv-wahrgenommene Gefühl, sich in einer virtuellen Welt *präsent* zu fühlen [Bar95].

In Kapitel 7 wird erklärt, wie die Aspekte der Presence während der Studie gemessen wurden.

5.1.2 Anforderungen

Zur Erreichung der Presence, wurden von Forschern bei Valve, Laut Abrash [Abr14] folgende ideale Anforderungen an die VR-Hardware gestellt:

- Mindestens 80° Sichtfeld
- Auflösung von 1080p oder mehr
- Low pixel persistence (3 ms or less) – Dies bedeutet, dass das Updaten jedes einzelnen Pixels maximal 3 Millisekunden dauern darf.
- Die Bildschirmwiederholrate muss über 60 Hz sein. Abrash gibt an, 95 Hz sind genug, wenig könnten auch reichen.
- Ein globales Display, bei dem alle Pixel gleichzeitig beleuchtet werden.
- Linsen: Maximal 2 pro Auge – ideale Linsen sind mit der aktuellen Technologie noch nicht umsetzbar.
- Optische Kalibrierung (z. B. IPD-Rädchen an der HTC Vive).
- Nahezu perfektes Tracking – Übertragung mit Genauigkeit im Millimeterbereich, Drehung im mit viertel Grad Genauigkeit oder besser und Erkennung von Bewegungen von 1,5 Meter in jede Richtung.
- Niedrige Bild Latenz – 20-25 ms Motion-To-Last-Photon – Dies bedeutet, dass nach einer Bewegung nur 20 ms verstreichen dürfen, bis das dazugehörige Bild bei den Augen des Benutzers ankommt.

Einige dieser Anforderungen sind mit der heutigen Technologie noch nicht umsetzbar. In Tabelle 5.2 habe ich zusammengefasst, welche Anforderungen von der HTC Vive erfüllt werden (grün) und welche nicht (orange). Wie bereits von Abrash angesprochen wurde, ist die Verwirklichung von den idealen Linsen mit der heutigen Technologie noch nicht möglich. Alle weiteren Anforderungen wurden erfolgreich erfüllt.

Spezifikation	Anforderung	HTC Vive
FOV	≥ 80° Sichtfeld	110°
Auflösung (pro Auge)	≥ 1080p	1080 x 1200
Pixel Persistence	≤ 3 ms	1.9 ms ¹
Bildschirmwiederholrate	min. 60-95 Hz	90 Hz
Global Illuminiertes Display	-	Gegeben ²
Linsen	Ideal	Nicht ideal
Niedrige Latenz	max. 20-25 ms	22 ms
Optische Kalibrierung	-	IPD und Fokus anpassbar ³
Tracking	Millimeter genau, 0,25 ° Drehung, 1,5 m Bereich	1-2 mm RMS ⁴ , 0.1 ° Drehung, 4.5 x 4.5 m Bewegung

Tabelle 5.2: Hardware Anforderungen für VR Geräte

Google setzt mit ihrem Quality Guide für Daydream Apps¹ eine Reihe von Anforderungen, die auch auf VIEW anwendbar sind. Anforderungen, die nicht anwendbar sind, wurden aus der folgenden Liste ausgelassen:

- Dem Benutzer muss die Möglichkeit geboten werden, sich auf Elemente zu fokussieren und Text zu lesen.
- Die App muss das Head-Tracking beibehalten, es darf nicht unterbrochen werden. Zu dem dürfen keine virtuellen Objekte an den Kopf des Benutzers angebracht werden, da dies die Immersion stört
- Die Horizont-Linie muss stabil bleiben. Der Boden darf nicht geneigt werden und die Kamera darf nicht geschüttelt werden
- Der Benutzer muss stets die Controller über die Kamerabewegungen behalten.
- Der Controller muss wie ein Laser Pointer bei der Interaktion mit UI Elementen genutzt werden. Zudem sollte der Controller in VR sichtbar sein und das Anzeigen eines Laserstrahls ist ratsam.
- Der Cursor muss die gleiche Tiefe annehmen wie die Objekte, auf die gezielt werden

Neben dem Ziel, die Presence sicherzustellen, führt das Einhalten dieser Anforderungen auch zu einer angenehmeren User Experience. Google versucht für ihre Daydream-Apps eine Reihe von Konzepten zu standardisieren.

5.1.3 Gestaltung der Umgebung

In diesem Abschnitt wird vor erklärt, wie die Umgebung von VIEW entworfen wurde und welche Konzepte verworfen wurden.

Leinwand für die Kartendarstellung Die ersten Entwürfe zur die Leinwand bestanden aus einer kompletten und einer halben Kugel-förmigen *Umhüllung* des Nutzers. Ein zweiter Entwurf reduzierte die Umhüllung auf eine Kapsel-artige Form. Das Problem dieser beiden Herangehensweisen war, dass sie die Abschnitte an den Rändern und Krümmungen verzerrt haben. Dies ist für die Bildauswertung nicht angebracht.

Der nächste Entwurf bestand aus einer gewölbten Leinwand, ähnlich den *curved* HD-Fernsehern, die heutzutage erhältlich sind. Diese Form bietet eine große Übersicht für das gesamte Blickfeld des Nutzers, ohne jedoch die Darstellung zu verzerren.

Umgebung Zur Erhöhung der potentiellen Immersion und zum Einbetten von VIEW in das Gesamtprojekt, wurde als Umgebung das Modell genutzt, welches auch für das Serious Game *Lost Earth 2307* genutzt wurde. Neben den visuellen Objekten, wurden Ideen zu akustischen Komponenten entworfen. Im Hintergrund sollte das typische Rauschen einer Raumstation zu hören sein. Zudem bietet es sich an, die Sound-Effekte der verschiedenen Computer im Raum abzuspielen. Wichtig ist es hierbei, dass die Lautstärke dieser Geräusche abhängig von der Distanz zum Benutzer sind.

¹ Google Daydream Design Requirements: <https://developers.google.com/vr/distribute/daydream/design-requirements>

Weitere Konzepte Einige Konzepte konnten aus zeitlichen Gründen nicht umgesetzt werden. Hierzu gehörten die Implementierung von vorbei fliegenden Raumschiffen. Ein weiteres Konzept, war die Positionierung des Raumschiffs, in dem sich der Spieler befindet, über der Erde. Beim Navigieren der Karte, fliegt das Raumschiff rund um die Erdkugel bis zum relevanten Ort.

Konzeptuell wurde berücksichtigt, dass der Auswertetisch auf der Lost Earth Brücke in späteren Projekten als Tisch für die Nutzung mit dem digitalen Lagetisch in VR dienen soll. Es würde sich anbieten in späteren Projekten den DigLT in Kombination mit dem in dieser Arbeit entwickelten Prototypen zu nutzen. Der Bildauswerter könnte beispielsweise den Tisch als Übersichtskarte nutzen, um den ungefähren auszuwertenden Ort anzuwählen. Anschließend würde er zur Leinwand gehen, um dort seine Aufgabe zu erledigen. Aus diesem Grund wurden keine Funktionalitäten für diese Bereiche der Brücke *reserviert*.

In Kapitel 6 wird genauer beschrieben, wie diese sogenannte *MapDisplay*, die Lost Earth Brücke und weiteren Faktoren implementiert wurde.

5.2 Interaktionskonzepte für die Bildauswertung in VR

In der VR-Spiele und -Anwendungsindustrie gibt es zur Zeit noch keine standardisierten Gesten, welche über alle Bereiche genutzt werden. Dies wird sich wahrscheinlich erst in den Jahren ändern. Daher muss jeder, der eine VR-Anwendung entwickeln will, selbst entscheiden, welche Interaktionskonzepte für seine Anwendung am angemessensten sind. Bekannte Konzepte können nicht lediglich von Desktop Computern nach VR portiert werden.

»Designing for VR should not mean transferring 2D practices to 3D,
but finding a new paradigm.«

Jonathan Ravasz - Design Practices in Virtual Reality [Rav16]

In diesem Unterkapitel wird im ersten Teil aufgezeigt, welche Interaktionskonzepte aktuell häufig verwendet werden. Im zweiten Teil werden Konzepte vorgestellt, die ich für die Umsetzung von der Bildauswertung in VR, erstellt habe.

5.2.1 Existierende Konzepte

Als Vorbereitung vor Beginn meiner Master Arbeit habe ich eine Reihe von VR Spielen getestet und weitere online nachgeschaut. Aufgefallen ist mir dabei, dass es kaum Spiele mit sehr ähnlichen Interaktionsmöglichkeiten gibt. Teilweise existieren es sogar Unterschiede zwischen Spielen, die vom gleichen Entwickler stammen.

In Tabelle 5.3 werden die häufig genutzten Konzepte zusammengefasst und jeweils ein Beispiel für eine Anwendung, welches dieses Konzept nutzt, gegeben.

Bei einer Reihe von Spielen ist mir aufgefallen, dass sie VR-Krankheit-auslösende Konzepte nutzen. *Subnautica* und *Onward* nutzen beispielsweise die Eingabe von Joysticks für die Lokomotion. In mehreren Foren und Artikeln wird berichtet, dass diese Spiele oft sofortigen Übelkeit auslösen, sobald der Spieler bewegt. Bei der Nutzung von *The Lab* befand ich mich in einer Holzhütte, hier habe ich versucht unter einen Tisch zu blicken. Das Spiel wollte mich hierbei assistieren und hat die Kamera etwas weiter gesenkt - wahrscheinlich, damit ich mich nicht so tief bücken muss. Dies hat ausgelöst, dass ich durch die resultierende Desorientierung gefallen bin.

Obwohl ich viele der Aktionen, die für die Bildauswertung benötigt werden, nicht in den untersuchten Spielen finden konnte, kann diese Zusammenfassung als Inspiration bei der Gestaltung von Interaktionskonzepten für die Bildauswertung dienen. Ich konnte beispielsweise keine Anwendung finden, welche es erlaubt Bilder in VR zu rotieren.

Kategorie	Unterkategorie	Umsetzung	Anwendung
Lokomotion	Mit Teleportation	Point-To-Teleport	Vivecraft
		Teleportation mit Sphere	The Lab
	Ohne Teleportation	Keine Teleportation	Keep Talking and Nobody Explodes
		Schweben durch Drücken oder Berühren des Touchpads	Google Earth VR
		Gehen mit Hilfe eines Touchpads oder Controller	Subnautica
Mit Hilfe eines Fahrzeugs	EVE: Valkyrie VR		
Interaktion mit Bildern	Zoom	Durch Kopfbewegungen	Virtual Desktop
		Stufenweise durch Auswahl der erwünschten Bereiche	World of Comenius
	Pan (Bewegen)	Durch Ziehen mit beiden Fingern auf dem Touchpad	Batman VR
		Trigger drücken und Strahler ziehen	Google Earth VR
		Durch Kopfbewegungen	Virtual Reality Cross View
Sonstige	Erweiterte Funktionen	Eine Hand aktiviert Funktionen, die auf der anderen Hand angezeigt werden	Tilt Brush
		Separate Konsole im Raum	Subnautica
	Text Eingabe	Spracherkennung	Starship Commander VR
		Große virtuelle Tastatur	Onward
	Drehen von Objekten	Durch Greifen und Bewegungen der Hand	Nvidia Funhouse

Tabelle 5.3: Zusammenfassung gängiger Interaktionskonzepte für VR-Anwendungen

5.2.2 Gestaltung von Interaktionskonzepten

In diesem Unterkapitel werden die Interaktionskonzepte, die für VIEW erstellt wurden präsentiert. Bei der Gestaltung dieser Interaktionskonzepte war es wichtig, Hardware-unabhängig zu bleiben. Die Eingaben wurden auf Touchpad, Hand-Bewegungen, Trigger und einen Knopf limitiert, da diese Eingaben von den gängigen VR-Systemen unterstützt werden.

In Kapitel 6.5 wird geschildert, wie die gewählten Interaktionskonzepte für die HTC Vive in Unity3D umgesetzt wurden. Aufgrund von der zeitlichen Limitierung konnten nicht alle untersuchten Konzepte implementiert werden. Im Rahmen dieser Arbeit wurden Interaktionskonzepte, welche für die Durchführung der Studie benötigt wurden, priorisiert.

Mit Hilfe dieser Informationen und den aktuell vorgeschlagenen Paradigmen wurde entschieden, welche Interaktionen, für welche Aktionen am angebrachtsten sind. Die in Kapitel 2 aufgezeigte Tabelle der *Taxonomy of VR input* war dabei hilfreich, die unterschiedlichen Interaktionskonzepte den verschiedenen Aktionen zuzuordnen. Aktionen die auf Objekten in weiter Entfernung durchgeführt werden sollen, werden mit *Zeigen* angewählt. Objekte die in der direkten Umgebung liegen, werden mit direkter Berührung angewählt.

Zudem wurden kleinere Prototypen erstellt mit den unterschiedlichen Interaktionskonzepten, um zu testen, wie gebrauchstauglich sie sind und ob die Bedienung sich intuitiv anfühlt.

Lokomotion Bei der Gestaltung der Lokomotion ist es wichtig die Paradigmen zu Vermeidung der VR-Krankheit, die in Kapitel 2 und Kapitel 3 bereits erklärt wurden, zu beachten. Kurzgefasst ist zu beachten, dass der Benutzer die Kontrolle seiner Kopfbewegungen immer behalten muss. Die Anwendung darf nicht das Blickfeld des Benutzers – ohne eigene Aktion – verändern. Wird der Benutzer durch eine eigene Aktion bewegt, so ist zu beachten, dass dies mit einer konstanten Geschwindigkeit passieren muss. Des Weiteren, muss der Boden fest sein, in dem Sinne, dass er nicht geneigt werden darf.

Aus diesen Gründen wird eine Lokomotion durch Streichen über das Touchpad abgelehnt. Ich habe es dennoch an einem kleinen Prototyp ausprobiert und sofort Übelkeit empfunden.

Das Konzept eines schwebenden *Fahrzeugs* für die Lokomotion ist in Abbildung 5.1c zu sehen. Die Idee war es, sich mit Hilfe eines futuristischen Gefährts frei durch den Raum zu bewegen. Das Problem dieser Idee ist, dass sie die Bewegung im Raum unnötig komplex macht. Zudem ist die Verwirklichung schwierig, da sie anfällig für das Hervorrufen von Symptomen der VR-Krankheit ist.

Eine oft genutzte Art der Lokomotion ist das Point-To-Teleport. Hierbei kann der Strahl unterschiedliche Formen annehmen. Es wurden zwei Konzepte ausprobiert: *Linie* 5.1a und *Bogen* 5.1b.

Die Nutzung der Bogen-Form erwies sich als angenehmer und visuelle ansprechender. Des Weiteren bietet sich bei der Nutzung von room-scale VR an, den begehbaren Bereich am Zielort der Teleportation anzuzeigen. Dies hilft einerseits bei der Entscheidung, wohin teleportiert werden soll und andererseits wird dem Benutzer ein Größenverhältnis geboten, damit er leichter erkennt, wie weit entfernt der Zielort ist.

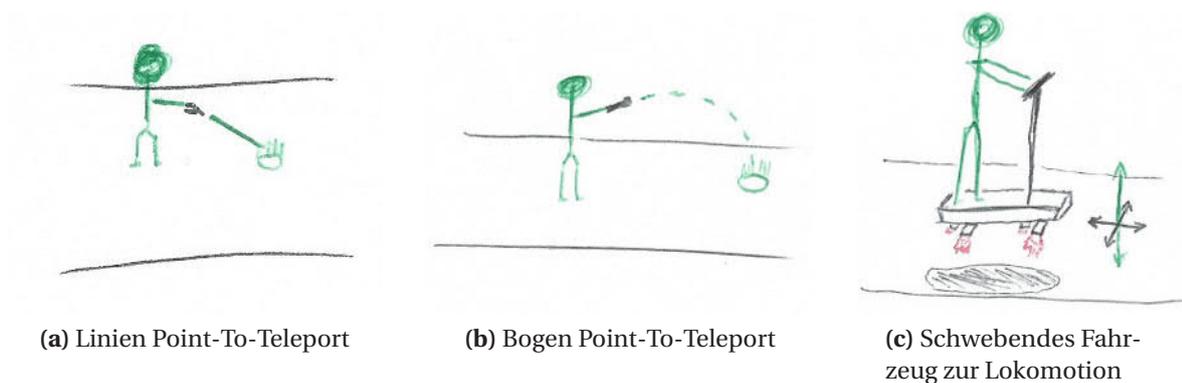


Abbildung 5.1: Skizzen für Lokomotionskonzepte

Karten Interaktion Für die Interaktion mit der Karte bietet sich die Smartphone-Metapher. Bei Smartphones werden zwei Finger auf dem Display auseinander gezogen, um das Bild zu vergrößern. Dies heißt für VR genauer, dass die Interaktion mit der Karte auf Distanz ähnlich mit Laser-Pointern durchgeführt wird.

Zum Schieben der Karte reicht die Nutzung eines einzelnen Laser-Pointers. Dies ähnelt dem Konzept, das in *Google Earth VR* genutzt wird.

Für das Zoomen, also Vergrößern und Verkleinern der Karte, werden beide Controller genutzt. Werden die (für diese Interaktion gewählten) Knöpfe betätigt, so wird die Distanz zwischen den beiden Strahlen genutzt, um die Karte zu skalieren (siehe Abb. 5.2b). Dies ähnelt dem Pinch-To-Zoom Konzept auf heutigen Smartphones.

Eine weitere Möglichkeit des Zooms ist das Vor- und Zurückziehen eines oder beider Controller bei gedrücktem Knopf (siehe Abb. 5.2a). Der Benutzer würde die Karte so-zu-sagen an sich heran ziehen, um sie zu vergrößern.

Zum Rotieren der Karte, könnten entweder beide Controller genutzt werden, um ähnlich wie auf Smartphone die Drehung Karte relativ zu der Drehung der beiden Controller zueinander anzupassen. Die Rotation der Karte könnte jedoch auch mit nur einem Controller durchgeführt werden, in dem der Benutzer den Knopf drückt und dann seinen Arm dreht.

Die Interaktion mit der Karte über das Touchpad wurde abgelehnt. Hier hätte sich angeboten, das Streichen über das Touchpad zum Bewegen der Karte zu nutzen. Andererseits würden kreisförmige Bewegungen das Rotieren auslösen. Zoomen müsste mit Hilfe des zweiten Controllers durchgeführt werden. Entweder durch gleichzeitiges Betätigen beider Touchpads oder durch die exklusive Bindung des Zooms auf das andere Touchpad. Dieses Konzept ist schwierig umzusetzen, da es schnell zu Missinterpretationen der auszuführenden Aktionen führen kann.

Des Weiteren abstrahiert dieses Konzept die Verbindung zwischen Benutzer und Karte – sie fühlt sich weniger *taktil* an.

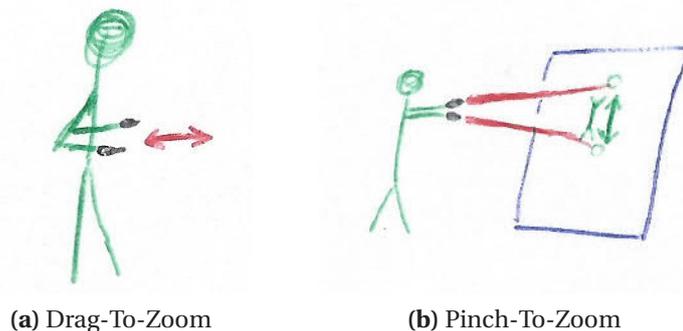


Abbildung 5.2: Skizzen für Zoomkonzepte

MapCards Für eine bessere Übersicht während der Erkundung der Luftbildaufnahme, wurde das Konzept der *MapCards* erstellt. Hierbei handelt es sich um kleine, Postkarten-große Ausschnitte der Aufnahme, die als eine Art Lesezeichen dienen sollen. Der Benutzer markiert mit den Laser-Pointern einen Bereich, den er behalten will und speichert in somit in einer MapCard. Die MapCard kann er entweder in der Hand behalten oder sie vor sich an eine Art unsichtbare Pinnwand heften (siehe Abbildung 5.3b). Dies erleichtert beispielsweise den Vorgang, bei dem zwei verschiedene Fahrzeuge verglichen werden sollen, da der Benutzer zwei MapCards für die jeweiligen Fahrzeuge erstellen kann und sie direkt nebeneinander hängen kann. Der Bereich, aus dem die MapCard geschnitten wurde, wird beim auf der Leinwand markiert (siehe Abbildung 5.3b). Zusätzlich lässt sich der markierte Bereich im Nachhinein mit Hilfe der Laser-Pointer anpassen, um ihn entweder zu vergrößern oder zu verkleinern. Dies wird erreicht, in dem mit dem Laser-Pointer auf die Ecken gezielt wird und diese dann verschoben werden.

Sofern es zu dem gegebenen Bereich 3D-Modell von beispielsweise dem Höhenmodell der Umgebung oder von Gebäuden in der Umgebung gibt, bietet es sich hier an, diese auch dreidimensional auf der MapCard anzuzeigen. Der Vorteil dieser Darstellungsweise ist, dass der Benutzer die Karte nach Belieben drehen kann, um somit das Modell von mehreren Winkeln zu betrachten (siehe Abbildung 5.3c).

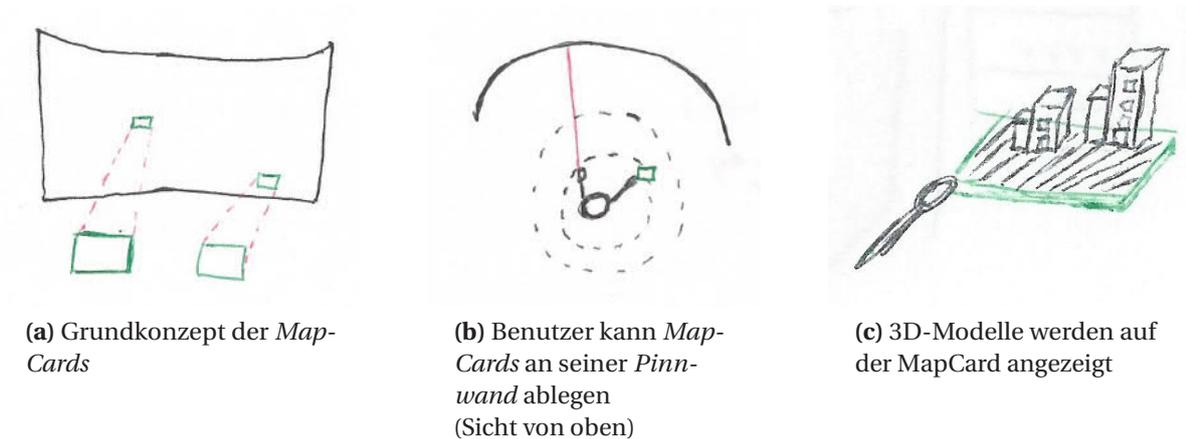


Abbildung 5.3: Übersicht zu den MapCards

Touchpad Shortcuts Während den ersten Versuchen bei der Umsetzung des Prototyps, wurden aktive Zonen des Touchpads genutzt, um bestimmte Funktionen, wie das Platzieren eines Markers, das Wechseln zwischen Schichten und das Zurücksetzen der Spielerposition genutzt (siehe Abb. 5.4). Hierbei wurde das Touchpad in vier Bereiche eingeteilt. Durch Klicken des jeweiligen Bereiches wurden die dazugehörigen Funktionen ausgeführt.

Das Problem bei dieser Umsetzung ist, dass die Möglichkeiten der erweiterten Interaktionen auf vier limitiert sind. Die Anzahl an aktiven Zonen könnte zwar erhöht werden, jedoch wäre es dann schwierig, die Übersicht zu behalten und es könnten zu Missinterpretationen bei der Erkennung der *Klick-Zonen* führen.

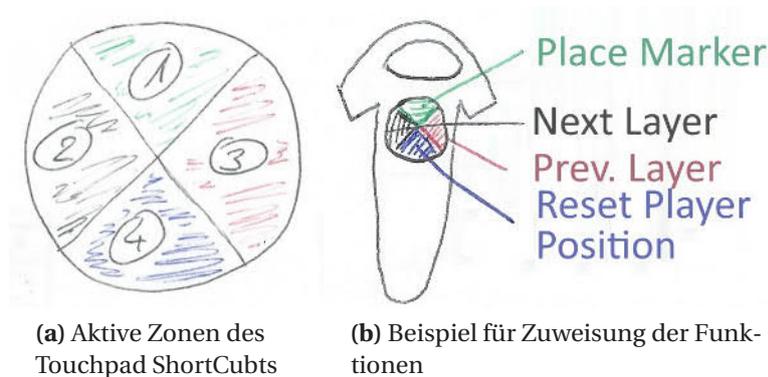


Abbildung 5.4: Touchpad Shortcuts Konzeptzeichnungen

InteractionCube Der sogenannte InteractionCube wurde inspiriert von der Farbpalette die in Google's *Tilt Brush* genutzt wird, um Werkzeuge auszuwählen (siehe Abb. 5.5). In *Tilt Brush* wird über dem linken Controller eine vier-seitige Box angezeigt. Jede Seite bietet eine Reihe von Funktionen wie das Wechseln des Zeichenwerkzeugs, das Auswählen der Farbe usw. Zum Auswählen einer Option zeigt der Benutzer mit dem anderen Controller auf die gewünschte Option und betätigt den Trigger.

Bei der Suche nach einer Lösung für das Problem der limitierten Aktionen der Touchpad Shortcuts, ist für VIEW ein ähnliches Konzept entstanden. Genau wie in *Tilt Brush* wird bei diesem Konzept ein interaktiver Würfel über dem linken Controller angezeigt (siehe Abb. 5.6a). Die unterschiedlichen *Panels* werden nach Kategorie eingeteilt. Diese Panels bieten eine Reihe von Möglichkeiten der Interaktion (siehe Abb. 5.6b). Es können unterschiedliche Knöpfe (3), Schieberegler (2), Toggles (5)

und Farbpaletten (6) auf jedes Panel platziert werden. Listen von Optionen könnten mit Hilfe einer Trommel-artigen Darstellung angezeigt werden (4). Diese wird gedreht, bis die erwünschte Position erreicht ist.

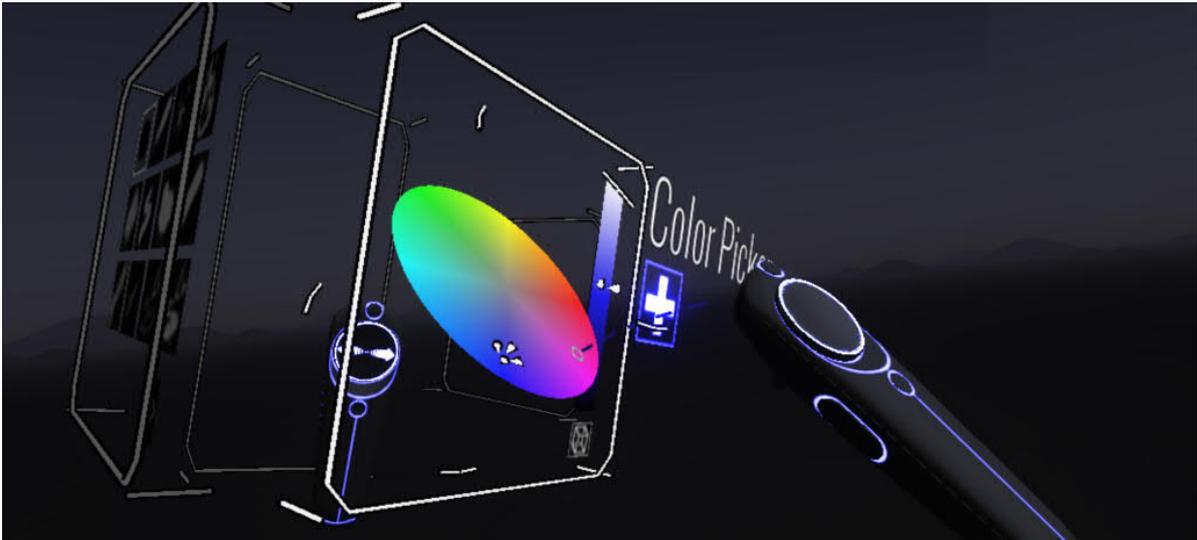
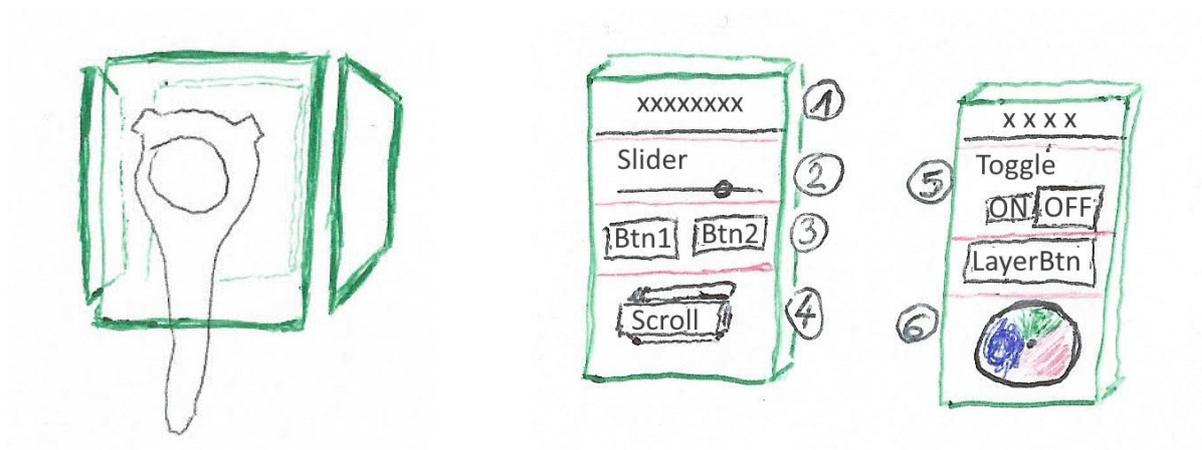


Abbildung 5.5: Tilt Brush 3D-Palette zur Interaktion
Quelle: UIGarage.net



(a) Grundkonzept des *InteractionCubes*

(b) Vorlagen für Interaktion mit den einzelnen *Panels*

Abbildung 5.6: InteractionCube Konzeptzeichnungen

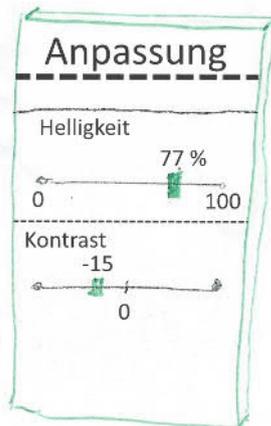
Annotation Markierungen auf der Karte können ähnlich wie bei einer Dartscheibe auf Distanz mit Hilfe des Laser-Pointer *geschossen* werden. Befindet sich der Benutzer sehr nahe an der Leinwand, so können die Markierung auch in die Karte gestochen werden. Beide Anwendungsmöglichkeiten können gleichzeitig implementiert werden.

Weitere Annotationen wie das Zeichnen einer Linie, eines Kreises oder eines Kastens sollten ähnlich wie in Microsoft Paint mit Hilfe der Festlegung eines Start- und Endpunkts erfolgen. So würde der Benutzer beispielsweise das gewünschte Werkzeug auf dem InteractionCube anwählen und dann mit Hilfe der Laser-Pointer den Start- und Endpunkt auf der Leinwand markieren.

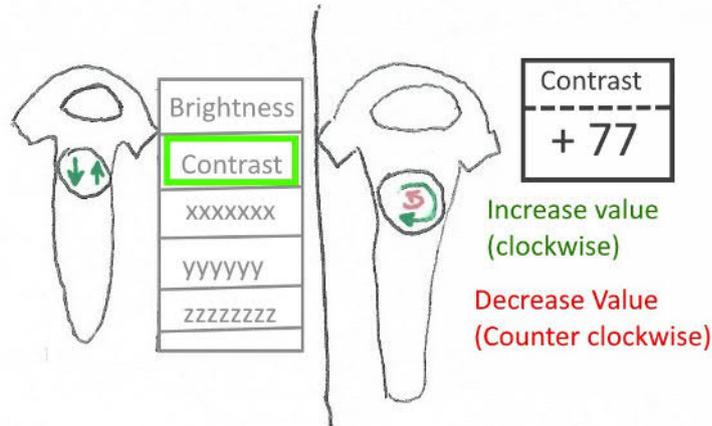
Bei der Benennung der Marker bietet es sich an, dies entweder über Spracheingabe zu realisieren oder mit Hilfe einer übergroßen virtuellen Tastatur.

Anpassung von Bildwerten Die Anpassung von Bildwerten wie Kontrast, Helligkeit usw. kann als ein Panel des InteractionCube eingefügt werden (siehe Abb. 5.7a). Hier könnten die Werte mit Hilfe von Slidern angepasst werden und durch betätigen eines Knopfes direkt neben diesen Slidern zurückgesetzt werden.

Ein weiteres Konzept ist in Abbildung 5.7b. Hier wird neben (oder auf) dem Controller eine Liste von Optionen angezeigt. Der Benutzer scrollt mit Hilfe des Touchpads durch diese Liste und bestätigt die Auswahl durch klicken des Touchpads. Anschließend kann der Benutzer durch kreisförmiges streichen über das Touchpad die Werte erhöhen oder senken. Der aktuelle Wert wird neben dem Controller angezeigt. Ein weiteres Klicken bestätigt die Änderung. Entfernt der Benutzer seinen Finger vom Touchpad, so wird die Aktion nach einigen Sekunden abgebrochen.



(a) Anpassung von Bildwerten mit Panels



(b) Alternative Art der Anpassung von Bildwerten

Abbildung 5.7: Anpassung von Bildwerten wie Kontrast und Helligkeit

InstructionScreen Im Kontext der Studie muss dem Proband die Aufgabe überlieft werden. Der Proband trägt während der Nutzung Kopfhörer, um von der akustischen Immersion zu profitieren. Aus diesem Grund müssen die Instruktionen innerhalb von der Anwendung geben werden. In diesem Fall wird ein Display neben der Leinwand angezeigt, auf dem die Instruktionen stehen. Der Proband kann durch Betätigen der *Zurück*- und *Weiter*-Knöpfe nach Belieben zwischen den verschiedenen Seiten der Instruktion blättern und sofort die nächsten Schritte der Aufgabe ausführen.

Hierbei ist *Fitt's Law* zu beachten. Der Text muss genügend groß sein, damit er aus dem erwarteten Winkel und der erwarteten Distanz gelesen werden kann. Das gleiche gilt auch für die Größe der Knöpfe. Diese müssen lesbar und leicht *anzielbar* sein.

3D Mode Des Weiteren wurde für die Studie der *3D Mode* implementiert. Aktiviert der Benutzer den 3D-Modus auf dem InteractionCube, so kann er Gebäude auf der Aufnahme mit dem Laser-Pointer anwählen und dadurch (sofern vorhanden) das 3D-Modell dieses Gebäudes im virtuellen Raum anzeigen lassen. Da das Objekt im Raum viel größer angezeigt wird, wie es bei den MapCards möglich wäre, kann der Benutzer das Modell einfacher betrachten. Zudem kann er das Objekt durch Greifen rotieren.

6

VR-Assistenzsystem für die Bildauswertung: VIEW

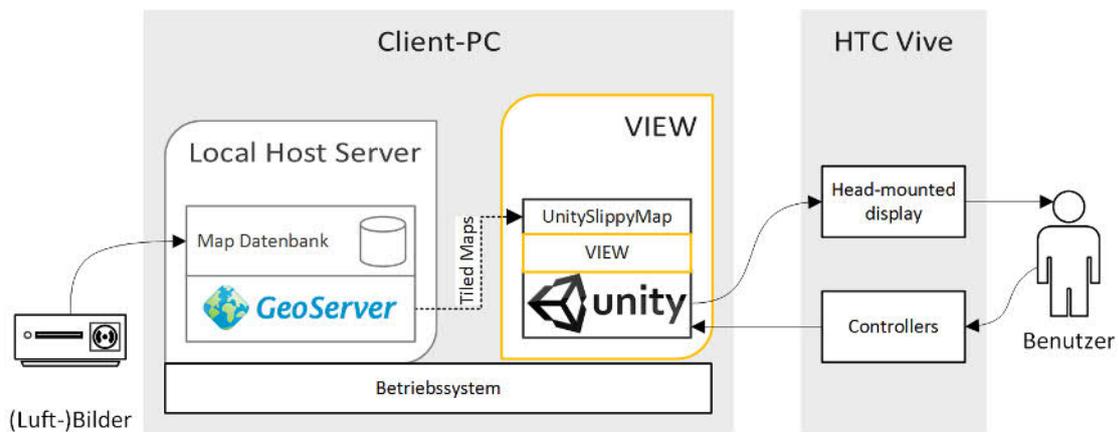


Abbildung 6.1: Aufbau der VIEW Umgebung. Gelb-markiert ist die Eigenentwicklung

Nachdem in den vorherigen Kapiteln das Fundament für die Erstellung des Prototyps erklärt wurde, wird in diesem Kapitel dargelegt, wie VIEW umgesetzt wurde. Hierzu gehört in erste Linie das Vorgehensmodell, welches in Kapitel 6.1 zusammengefasst wird.

In Abbildung 6.1 wird die finale Architektur des VIEW-Systems zusammengefasst. Georeferenzierte Luftbildaufnahmen werden auf den GeoServer geladen. Diese liefert die Tiles über das WMS-Protokoll an die UnitySlippyMap Schnittstelle in VIEW. Die HTC Vive bildet hierbei die Schnittstelle zwischen dem Benutzer und VIEW.

6.1 Vorgehensmodell für die Realisierung

Zu Beginn der Arbeit wurde im Lastenheft festgelegt, was für den Umfang dieser Arbeit erwartet wird. Für diese Arbeit war es wichtig, einen Prototyp zu erstellen, der während einer Studie evaluiert werden kann. Eine komplette Portierung aller Funktionalitäten ist dazu nicht nötig, da für den Umfang der Arbeit, die Konzeption eines Assistenzsystems für die Bildauswertung in VR im Vordergrund stand.

Zu den Schwerpunkten der Realisierung gehörten der Umgang mit hochauflösten Aufnahmen, die Implementierung der Interaktionskonzepte und die Erstellung einer immersiven Umgebung.

Eine Reihe von Lösungen für die Herausforderung des Umgangs mit hochauflösten Aufnahmen wurden recherchiert und ausprobiert. In Kapitel 6.3 wird beschrieben, wie das Problem für VIEW gelöst wurde.

In Kapitel 6.4 wird dargelegt, wie die virtuelle Umgebung von VIEW gestaltet wurde und aus welchen Komponenten er besteht.

Die einzelnen Interaktionskonzepte wurden während mehreren Treffen diskutiert und mit Hilfe von Mock-Ups am Prototypen ausgetestet. In Kapitel 6.5 wird beschrieben, wie die Interaktionen umgesetzt wurden.

Die einzelnen Teile von VIEW wurden weitestgehend modular erstellt, damit sie später ausgetauscht werden konnten, falls eine bessere Herangehensweise entdeckt wurde. Beispielsweise kann der genutzte MapServer beliebig gewechselt werden. Die Interaktionen wurden mit Hilfe einer Abstraktionsschicht von den ausgelösten Aktionen getrennt.

6.2 Genutzte Software und Schnittstellen

In diesem Kapitel befindet sich eine Zusammenfassung zu der genutzten Software und sonstigen Komponenten von VIEW.

Software

Unity3D Game Engine

Quelle: <https://unity3d.com/>

Lizenz: Proprietär

Blender Erstellung von 3D-Modellen

Quelle: <https://www.blender.org>

Lizenz: Frei unter GPL

OBS Studio Bildschirm Aufnahme

Quelle: <https://obsproject.com/download>

Lizenz: Frei unter GPL

GeoServer MapServer

Quelle: <http://geoserver.org/>

Lizenz: Frei unter GPL

QGIS Georeferenzieren von Bilddateien

Quelle: <http://www.qgis.org/en/site/forusers/download.html>

Lizenz: Frei unter GPL

Paint.NET Bearbeitung von Bildern

Quelle: <https://www.getpaint.net/>

Lizenz: »Freeware« - Details in Fußnote ¹

Frameworks

SteamVR Framework zur Unterstützung von HTC Vive

Quelle: <http://store.steampowered.com/steamvr>

Lizenz: Nutzung frei - Steamworks VR API Code Lizenz²

¹ Geschichte der Paint.NET Lizenz: https://en.wikipedia.org/wiki/Paint.NET#Development_details

² SteamWorks VR API™ Code License <https://github.com/ValveSoftware/steamworks-vr-api/blob/master/LICENSE>

VRTK Erweiterte Unterstützung für SteamVR

Quelle: <https://github.com/thestonefox/VRTK>

Lizenz: Frei unter der MIT Lizenz

SharpConfig Helfer für die Unterstützung von Konfigurationsdateien.

Quelle: <https://github.com/cemdervis/SharpConfig>

Lizenz: Frei unter MIT Lizenz

Log4net Helfer für das Logging

Quelle: <https://logging.apache.org/log4net/>

Lizenz: Apache License

UnitySlippyMap Schnittstelle zum Laden von SlippyMaps in Unity

Quelle: <https://github.com/jderrough/UnitySlippyMap>

Lizenz: Frei unter LGPL

SQLiter Für UnitySlippyMap benötigte SQL Implementierung

Quelle: <https://github.com/wilsonfreitas/sqliter>

Lizenz: Frei - keine Spezifikation angegeben

Assets

Future Beeps Lite Umgebungsgeräusche

Quelle: <https://www.assetstore.unity3d.com/en/#!/content/63357>

Lizenz: Frei unter Asset Store Provider Agreement¹

SkyBox Volume 2 Skybox

Quelle: <https://www.assetstore.unity3d.com/en/#!/content/3392>

Lizenz: Frei unter Asset Store Provider Agreement

ISS Ambient Noise Rauschen der Raumstation

Quelle: <https://youtu.be/IrdfEmYtB0>

Lizenz: Standard YouTube License

Lost Earth Brücke Umgebung für den Prototypen

Quelle: Fraunhofer IOSB

Lizenz: Interne Nutzung

Aswan Temple 3D-Modell des Mandulis Tempels

Quelle: VassilisP. (3DWarehouse.SketchUp.com²)

Lizenz: Abhängig vom Autor - siehe Fußnote³

Montserrat-ExtraBold Schriftart

Quelle: <https://fonts.google.com/specimen/Montserrat>

Lizenz: Frei unter SIL Open Font License

¹ Siehe Punkt 6.5 in Unity Asset Store Provider Agreement: https://unity3d.com/legal/as_provider

² <https://3dwarehouse.sketchup.com/model/a3f3eb13c08efcadbf35f2a4d5209a3/Kalabsha-Temple-Aswan>

³ 3D Warehouse: Terms of Use: <https://3dwarehouse.sketchup.com/tos.html>

Anmerkung Komponenten die ausgetestet und anschließend verworfen wurden, befinden sich nicht in dieser Übersicht. Zudem befindet sich Software, die indirekt benötigt wurde, nicht in dieser Liste. Hierzu gehören beispielsweise Microsoft Windows, Excel, Visio und Word, Notepad++ und TortoiseGIT. Außerdem befindet sich Software, die für andere Teile der Arbeit genutzt wurden, nicht in dieser Übersicht. Hierzu gehört beispielsweise die Unreal Engine für die Bildqualitätsbewertung und PSPP für die Evaluation.

6.3 Umgang mit hochauflösten Luftbildaufnahmen

Bilder werden in Unity mit Hilfe von Texturen dargestellt. Bei den aktuellen Engines und neusten Grafikkarten werden Texturen mit einer Auflösung von bis zu 8192 x 8192 Pixeln unterstützt – in einigen Fällen sogar 16k x 16k. Hierbei gibt die GPU Memory die Grenzen an. Das Problem hierbei ist, dass Luftbildaufnahmen oft eine deutlich höher Auflösung haben. Die NASA Earth Texture ist beispielsweise 65536 x 32768 Pixel groß.

Es könnten spezielle Shader genutzt werden, die mehrere Texturen zu einer zusammenschneiden, um dieses Grenzen zu umgehen¹. Das Problem ist jedoch, dass jede einzelne Luftbildaufnahme manuell angepasst werden müsse und das eigentliche Problem der limitierten GPU Memory nicht gelöst wird.

6.3.1 Mega-Textures, Giga-Texture und Virtual Textures

Die ersten Versuche des Ladens von hochauflösten Texturen in Unity wurden mit Megatextures versucht. Hierbei handelt es sich um Verfahren, die es erlauben, die Grenzen von 8k x 8k Texturen zu umgehen. Das Verfahren nutzt eine Kombination aus Mipmapping und dem Ausschneiden von derzeit benötigten Texturen Bereichen, um somit nur aktuell sichtbare Texturalelemente anzuzeigen.

*Graphine*² unterstützt beispielsweise bis zu 256k x 256k.

*Amplify*³ unterstützt mit einem ähnlichen Verfahren Texturen im Terabyte Bereich.

Das Problem bei dieser Herangehensweise ist, dass diese Middlewares proprietär sind und die Umsetzung sich als sehr schwierig erwiesen hat. Des Weiteren ist die Darstellung von hochauflösten Aufnahmen nicht das eigentliche Anwendungsgebiet solcher Software. Zudem wäre die weitere Nutzung sehr unflexibel.

6.3.2 Slippy Maps

Ein weiterer Lösungsansatz ist die Nutzung von *Slippy Maps*. Slippy Maps wird üblicherweise als Bezeichnung für Karten, die mit einem Browser geladen werden können, genutzt. Im Kapitel 3 wurde bereits kurz angesprochen, dass hierbei das Protokoll WMS für Rastergrafiken und das OpenStreetMap-Verfahren für Vektordaten genutzt wird.

Zur Umsetzung von Slippy Maps für VIEW gab es eine Reihe von Möglichkeiten. Es wird für die Implementierung einerseits ein MapServer benötigt und andererseits eine Client-Implementierung, die auf diesen Server zugreift.

1 Unity3D: Using textures larger than 4096x4096: <http://www.41post.com/4416/programming/unity3d-using-textures-larger-than-4096x4096>

2 Graphine: <http://graphinesoftware.com/>

3 Amplify: <http://amplify.pt/unity/amplify-texture-2/>

Tiling Die Aufnahmen müssen zur Nutzung für den MapServer in einzelne Tiles unterschiedlicher LOD zerschnitten werden. Nach mehreren (fehlgeschlagenen) Versuchen dieses Tiling mit Hilfe von unterschiedlicher Software (ImageMagick, python-magic, GenerateSlippyMapTiles, GDAL2Tiles, Mapslicer, ...) selbst durchzuführen, wurde die Information entdeckt, dass eine Reihe von MapServern dieses Tiling nativ durchführen können.

Tile Server Der gewählte Mapserver kann beliebig gewählt werden. Zu den einzigen Muss-Kriterien gehört, dass er das WMS-Protokoll unterstützt. Zu der engeren Auswahl gehörten die Server *GeoServer*, *MapServer*, *degree* und *MapGuide*.

Der Mapserver GeoServer wurde gewählt, da es sich um eine Java-basierte Implementierung handelt und somit auf allen Java-unterstützenden Betriebssystemen laufen kann. Zudem unterstützt GeoServer nativ das Importieren von georeferenzierten TIFF Dateien (GeoTiff).

Der gewählte MapServer kann entweder direkt auf der lokalen Maschine laufen oder extern auf einer, über das lokale Netzwerk oder über das Internet erreichbare Maschine, laufen. Zudem kann er nach Belieben ausgetauscht werden.

Tile Client Zum Laden, der vom MapServer geliefert den Tiles gab es mehrere Kandidaten:

*UtyMap*¹ ist eine sehr gut unterstützte Implementierung von SlippyMaps für Unity. Sie wird genutzt, um 3D-Welten für Spiele anhand von gegebenen Karten zu erstellen. Das Problem ist, dass die Nutzung sehr schwierig ist und dass die Bibliothek wegen der langen Liste an Features besonders groß ist.

*UnitySlippyMap*² scheint nicht mehr weiter unterstützt zu werden. Diese Implementierung ist jedoch deutlich verständlicher aufgebaut und es gibt keine Liste an (für VIEW) unnötigen Features. Das Problem jedoch ist, dass es unter Unity 5 nicht funktioniert. Nach langem anpassen der einzelnen Klassen, gelang es mir jedoch es zum Laufen zu bringen.

*World Maps*³ ist eine sehr simple Implementierung, die scheinbar nur das niedrigste LOD lädt. Daher kann sie für VIEW nicht genutzt werden.

Neben den erwähnten Kandidaten gab es noch eine Reihe von proprietären Implementierungen wie *map-ity* und *Online Maps*. *UnitySlippyMaps* wurde am Ende gewählt, da es alle benötigten Protokoll unterstützt und dies ohne unnötig komplex zu sein.

Georeferenzieren von Bilddateien Zur Georeferenzierung der nicht-georeferenzierten Bilddateien wurde QGIS Desktop genutzt. Mit Hilfe des eingebauten *Georeferencer*-Tools lassen sich Punkte auf einer Referenzdatei und der zu georeferenzierenden Aufnahme markieren. Sind mindestens drei Punkte markiert, so kann die Referenzierung ausgerechnet werden. Nach diesem Prozess kann die Datei als GeoTIFF abgespeichert werden und nativ auf den GeoServer geladen werden.

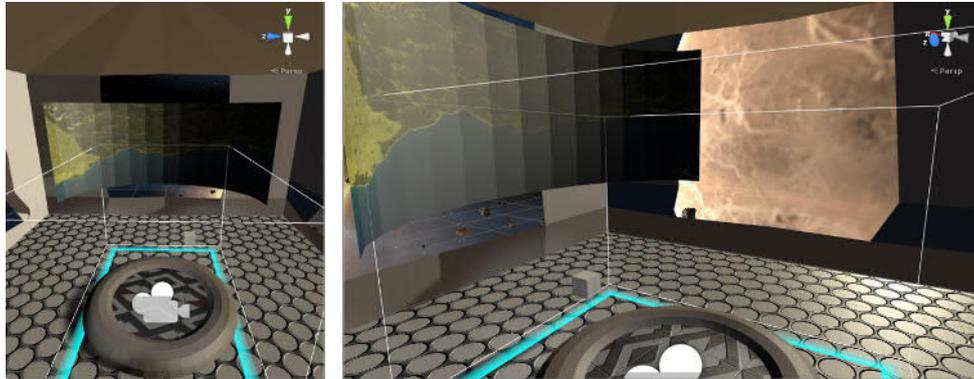
1 UtyMap: <https://github.com/reinterpretcat/utymap>

2 UnitySlippyMap: <https://github.com/jderrough/UnitySlippyMap>

3 World Maps: <https://www.assetstore.unity3d.com/en/#!/content/54085>

6.4 Gestaltung vom virtuellen Raum

In dieser Sektion wird beschrieben, wie die einzelnen Komponenten in Unity und Blender realisiert wurden. In der ersten Iteration des Prototyps bestand der virtuelle Raum lediglich aus einer Kiste und einem eckig-gewölbten Display (siehe Abb. 6.2). Als Leitfaden hierzu wurden die Erkenntnisse zur Immersion, welche in den vorherigen Kapiteln dargelegt wurden, genutzt.



(a) Blick auf das *Map-Display*

(b) Blick aus dem Fenster

Abbildung 6.2: Ausschnitte der ersten Iteration

6.4.1 Darstellung der Karte

Zur Darstellung der vom GeoServer gelieferten Tiles sind mehrere Prozesse nötig. Einerseits werden die Tiles von *UnitySlippyMaps* auf einer Fläche auf den Koordinaten $(0, 0, 0)$ angezeigt. Diese werden dann mit Hilfe von einer *RenderTexture* auf die Leinwand der Brücke projiziert. Diese Leinwand hat während der Entwicklung den Namen *MapDisplay* erhalten und wird im folgenden auch so genannt.

Anzeigen der Tiles auf dem MapDisplay *UnitySlippyMap*¹ ist eine Slippy Map Implementierung in C# für Unity. Sie erlaubt es Daten von mehreren Arten von Tile Anbietern herunterzuladen und anzuzeigen. Nativ werden zumindest *OSM tiles*, *WMS tiles*, *VirtualEarth/Bing maps tiles* und *MBTiles databases* offiziell unterstützt. Weitere können selbst implementiert werden.

Die Tiles werden auf einer Fläche Richtung x- und z-Koordinaten angezeigt und werden durch ein fixiertes Kamera-Objekt betrachtet (siehe Abb. 6.3). Die Tiles bewegen sich unter dieser Kamera. Die Kamera selbst bewegt sich lediglich beim Zoomen nach oben und unten. Tiles, die vom Blickfeld der Kamera nicht mehr erfasst werden, werden entladen. Neue Bereiche, welche ins Blickfeld der Kamera geraten, werden dynamisch geladen. Beim Zoomen werden schrittweise die verschiedenen LOD der Tiles geladen.

¹ *UnitySlippyMap*: <https://github.com/jderrough/UnitySlippyMap>

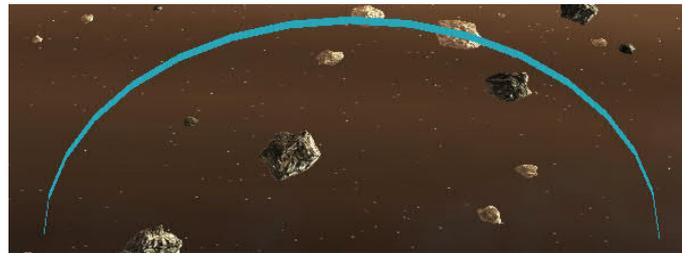


Abbildung 6.3: UnitySlippyMap und RenderTexture-Kamera

Dieser Aufbau bietet die Option, eine Unity *RenderTexture*¹ zu nutzen. Hierbei wird der Blickpunkt, also alles was die Kamera sieht, als eine Textur auf ein Objekt projiziert. Üblicherweise werden *RenderTextures* bei der Implementierung von Sicherheitskamera in Spielen genutzt. Sie bieten den Vorteil, dass die Form des Zielobjekts beliebig geändert werden kann, ohne Änderungen an der Darstellungsweise der Karte vornehmen zu müssen. In diesem Fall wird die Slippy Map auf die Kartenleinwand – das *MapDisplay* – projiziert (siehe Abb. 6.4).



(a) Blick vom Auswertetisch auf das *MapDisplay*



(b) Querschnitt des *MapDisplays* von oben

Abbildung 6.4: Überblick zum *MapDisplay*

Die Brücke wurde auf die Koordinaten (-1000, -1000, -1000) verschoben, damit sie nicht mit der Slippy Map interferiert. Dies bedeutet in anderen Worten, dass die Slippy Map außerhalb des für den Spieler sichtbaren Bereichs liegt.

UnitySlippyMap benötigt für das Herunterladen der Tiles mehrere Angaben, wie URL, Name der Schicht, Koordinaten und Server Typ. Trotz der Nutzung eines lokalen GeoServers ist es nicht wünschenswert, dass diese Daten hart-kodiert werden. Aus diesem Grund wurde in der Implementierung eine Konfigurationsdatei, welche mit Hilfe von *SharpConfig*² geparkt wird, genutzt.

Neben der *.exe*-Datei des fertigen Builds befindet sich ein Ordner mit dem Namen *cfg*. In diesem Ordner liegt die Datei *mapInfo.cfg*, deren Inhalt wie folgt aussieht:

1 Unity *RenderTexture*: <https://docs.unity3d.com/Manual/class-RenderTexture.html>

2 *SharpConfig*: <https://github.com/cemdervis/SharpConfig>

```

1 [General]
2 center = {32.88,23.97}
3
4 [Assuan]
5 Name = Assuan radar image
6 MapType = RADAR
7 ServerType = WMS
8 URL = localhost:8080/geoserver/wms?&TILED=true
9 layer = ViLandVR:AssuanDam_Radar_v2
10
11 (...weitere Layer...)

```

Hierbei sind für UnitySlippyMap die wichtigsten Informationen die URL und der Name der Schicht. Diese beiden Informationen in Kombination mit dem Typ des Servers erlauben es dem Plugin, die einzelnen Tiles herunterzuladen. Die einzelnen Schichten und Tile-Provider werden anschließend als MapInfo-Objekt zusammengefasst und zwischengespeichert. Sie können dann einzeln geladen, deaktiviert und überlagert werden.

Da der GeoServer lokal läuft, kann der Cache und die maximale Anzahl an gleichzeitigen Downloads erhöht werden:

```

1 TileDownloaderBehaviour.Instance.MaxCacheSize = 500000000; //500 MB
2 TileDownloaderBehaviour.Instance.MaxSimultaneousDownloads = 60;

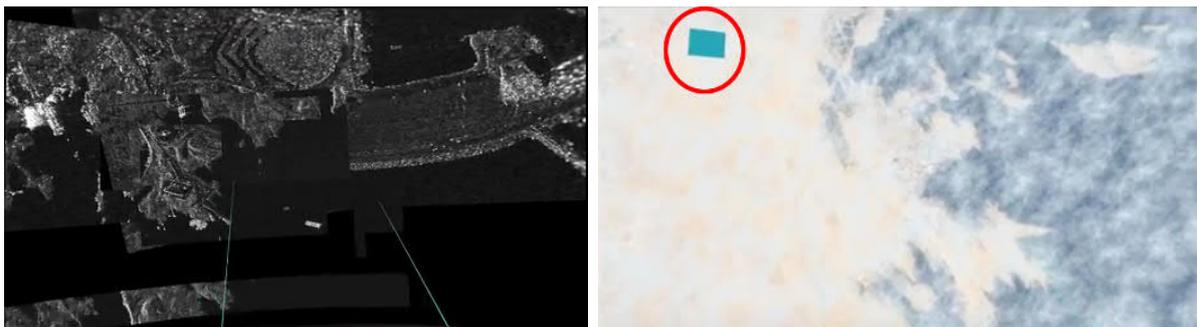
```

In der für die Studie genutzten Version existieren noch zwei Fehler, welche nicht behoben werden konnten, da die Ursache nicht lokalisiert werden konnte.

Der erste Fehler kommt in manchen Fällen vor, bei denen der Benutzer die Karte zu schnell weit hinaus und dann wieder hinein zoomt. Dies löst manchmal aus, dass UnitySlippyMap die Tiles in der falschen Position oder mit dem falschen LOD anzeigt. Zu sehen ist der Effekt in Abb. 6.5a).

Der zweite Fehler tritt zeitweise beim Laden von neuen Tiles auf. Der Effekt ist, dass eine oder eine Reihe von Tiles nicht geladen werden (siehe Abb. 6.5b).

Die beiden Fehler lassen sich korrigieren, in dem das Gebiet aus dem sichtbaren Bereich hinausgezogen wird und wieder hinein gezogen wird. Dies verursacht, dass der korrupte Bereich neu geladen wird. Während der Studie ist es in einigen Fällen zum Vorkommen dieser Fehler gekommen. Die Benutzer wurden in diesen Fällen auf die Lösung hingewiesen.



(a) Tile-Bug #1 – Falsch-platzierte Tiles

(b) Tile-Bug #2 – Fehlendes Tile rot markiert

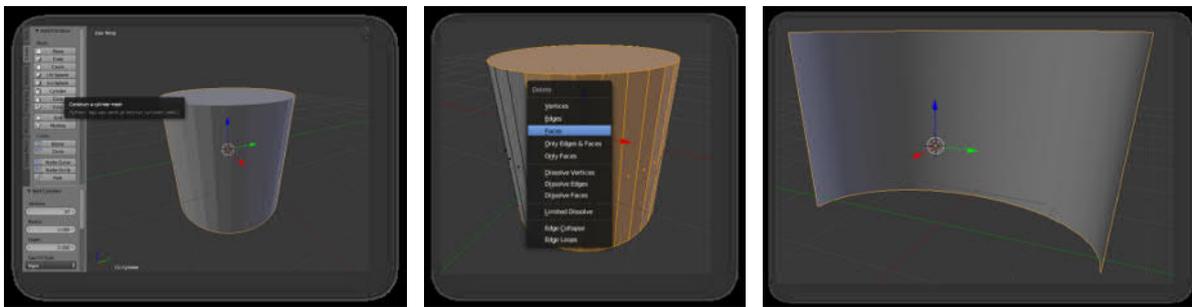
Abbildung 6.5: Überblick zum *MapDisplay*

Erstellung der Leinwand in Blender Das *MapDisplay* wurde in Blender auf der Basis eines Zylinders erstellt. Hierbei wurde die Blender Grundform *Cylinder* genutzt (siehe Abb. 6.6a). Anschließend wurde die Hälfte davon abgetrennt und die obere und untere Fläche entfernt (siehe Abb. 6.6b). Im folgenden Schritt wurden die Normalen des Objekts invertiert. Vor dem Durchführen dieses Schritts ist nur das äußere des ehemaligen Zylinders zu sehen. Durch das Invertieren der Normalen wird nur das Innere des Zylinders sichtbar gemacht.

Des Weiteren, muss sichergestellt werden, dass die Texturen korrekt auf dem Objekt angezeigt werden. Bei einfachen Formen wie dieser, reicht es die Blender-Funktion *Smart UV Projekt* zu nutzen. Bei komplexen Formen muss das UV Mapping per Hand durchgeführt werden.

Zu diesem Zeitpunkt wirkt das Objekt noch *eckig*, dies kann im *Object Mode* mit Hilfe von dem Werkzeug *Shading:Smooth* korrigiert werden (siehe Abb. 6.6c).

Anschließend kann das Objekt in Unity importiert werden. Die Scale, die in Blender angewendet wurde, ist identisch zu der Scale, die in Unity benutzt wird und somit kann sie auch in Unity noch angepasst werden.



(a) Zylinder Grundform in Blender

(b) Entfernung mehrerer Flächen des Zylinders

(c) Fertiges Objekt in Blender

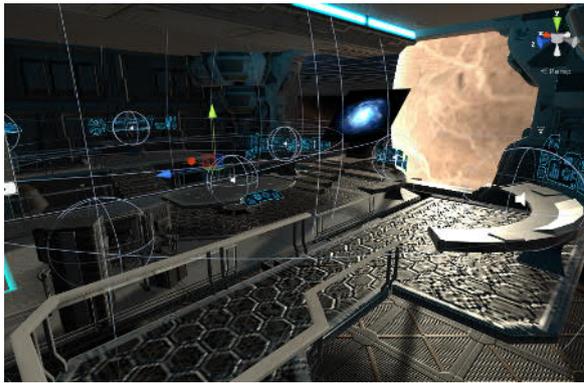
Abbildung 6.6: Erstellung des *MapDisplays* in Blender

In Unity ist es wichtig für das *MapDisplay* einen *Unlit*-artigen Shader zu nutzen. Dieser hat den Effekt, dass das Objekt sich selbst beleuchtet. Bei anderen Shadern würden sich Schatten bilden, was für die Bildauswertung unerwünscht ist.

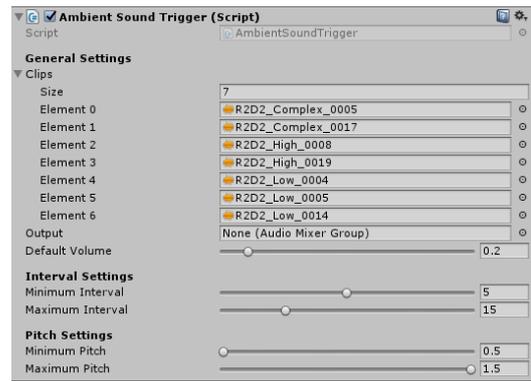
6.4.2 Importieren von Google Earth Modellen nach Unity

Der bei der Studie anzuzeigende Tempel wurde von SketchUp.com als *.kmz*-Datei heruntergeladen. Diese Datei ist im eigentlichen ein Archiv und kann nach der Umbenennung zu *.zip* mit als ZIP-Archiv geöffnet werden. In diesem Archiv befindet sich eine *.dae*-Datei und ein dazu gehöriger Ordner mit Texturen. Diese müssen zusammen extrahiert werden und können dann als Asset direkt in Unity importiert werden.

6.4.3 Abspielen von Geräuschen



(a) Die Geräusche werden auf mehreren Orten abgespielt



(b) Die einzelnen Werte können im Editor angepasst werden

Abbildung 6.7: Umgebungsgeräusche im virtuellen Raum

In VIEW gibt es zwei Arten von Tonquellen. Einerseits wird ein konstantes ambientes Rauschen abgespielt, welches das Rauschen einer Raumstation simuliert. Hierzu wurde das Geräusch der ISS genutzt. Andererseits werden in definierbaren Abständen Computer-Geräusche simuliert. Die Computer, die Geräusch machen, wurden in Unity definiert (siehe Abb. 6.7a). Hat der Benutzer beide Kopfhörer auf, so wirken diese Geräusche, als ob sie tatsächlich von diesen Orten stammen würde. Entfernt der Benutzer sich von ihnen, so werden sie leiser.

Die Geräusche stammen vom *Future Beeps Lite* Asset Paket. Zur Steuerung der Einstellungen wie Häufigkeit, Lautstärke, Pitch, ... wurde das *AmbientSoundTrigger*-Script erstellt. Diese spielt in pseudo-zufälligen Intervallen bei einem zufälligen Computer ein zufälliges Geräusch ab. Die minimale und maximale Dauer dieses Intervalls kann mit Hilfe des *Unity Inspector* angepasst werden (siehe Abb. 6.7b). Eine *Coroutine* startet nach einer gewissen Zeit die Methode *playRandomSoundAtRandomLocation()* (siehe Code).

```

1 void playRandomSoundAtRandomLocation()
2 {
3     //Choose random audio file from list
4     int randomSoundIndex = Random.Range(0, clips.Length);
5     //Choose random audio source from all sources
6     int randomAudioSourceIndex = Random.Range(0, audioSources.Length);
7     //Choose random pitch from min to max
8     float randomPitch = Random.Range(minimumPitch, maximumPitch);
9
10    //Set the values
11    AudioSource source = audioSources[randomAudioSourceIndex];
12    source.clip = clips[randomSoundIndex];
13    source.pitch = randomPitch;
14
15    //Play the sound
16    source.Play();
17 }

```

6.4.4 Die Lost Earth Brücke

Die Entwurfs-Szene von der ersten Iteration wurde verworfen, nachdem ich das Blender-Modell, welches für *Lost Earth* genutzt wurde erhalten habe. Beim Importieren dieses Modells in Unity gab es eine Reihe von Herausforderungen.

Das Blender-Modell enthielt etliche versteckte Layer und Objekt, die in Unity einzeln per Hand deaktiviert und/oder gelöscht werden mussten. Zudem wurden die Texturen nicht korrekt importiert und mussten einzeln wieder mit den respektiven Objekten verbunden werden. Dieses sowohl für die Albedo-Schicht (optisch), als auch für die Normal Map. In Abbildung 6.8b ist die gesamte Brücke als Querschnitt zu erkennen. Der begehbare Bereich wurde auf das hier grün-markierte Gebiet eingeschränkt, da im Vorfeld einige Probanden Wege gefunden hatten, sich aus dem Raumschiff hinaus zu teleportieren. Rund um das Modell wurden Licht-blockierende Wände eingefügt, da das Modell viele Löcher hat, welche Licht in das Raumschiff hineinlassen und somit eine unnatürliche Beleuchtung erzeugen (siehe Abb. 6.8g).

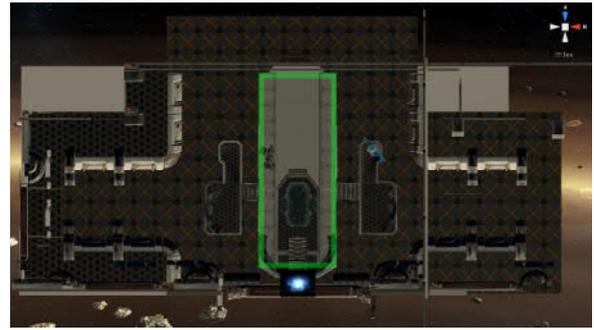
Als Außenumgebung wurde die Skybox *DSBWP* vom Skybox Volume 2 Asset Paket genutzt. Diese bildet einen Planeten aus nächster Nähe und ein breites Feld an Sternen ab. Des Weiteren wurden die Einstellungen von Unity so angepasst, dass metallene Objekte die festgelegten Lichtquellen reflektieren (siehe Abb. 6.8c).

Mit Hilfe von einer weiteren Emissionstextur-Schicht wurden die Leuchten an der Decke so konfiguriert, dass sie blaues Licht absondern (siehe Abb. 6.8e). Der *Unlit Transparent Cutout*-Shader wurde für die schwebenden Monitore genutzt, um sie halb-transparent und farb-emittierend zu gestalten (siehe Abb. 6.8f).

Innerhalb der versteckten Blender-Layer, habe wurde ein Computer Tisch mit Monitoren gefunden. Dieser wurde extrahiert und als Debug-Konsole genutzt. Das Objekt wurde so konfiguriert, dass VRTK die Konsolen Ausgabe auf dem Monitor anzeigt, damit während der Entwicklung nicht jedes Mal das HMD ausgezogen werden musste, um die Debugging Konsole auf dem Desktopmonitor zu lesen (siehe Abb. 6.8d).



(a) Blick aus dem Fenster



(b) Querschnitt der Brücke von oben mit begehbarem Bereich (grün)



(c) Blick Richtung Auswertetisch



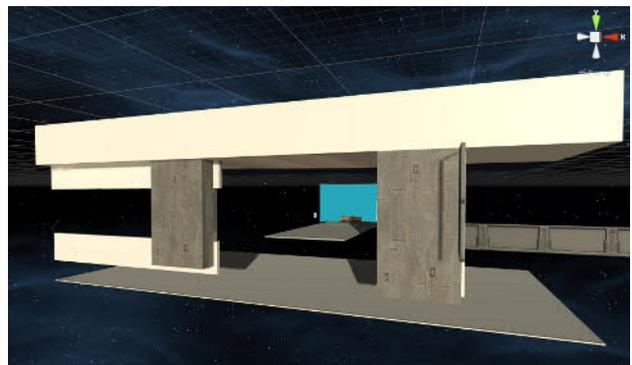
(d) Blick Richtung Debug Konsole



(e) Texturfarben basierte Emission von Licht



(f) Monitor sind halbtransparent und emittieren farbiges Licht



(g) Licht-blockierende Wände

Abbildung 6.8: Die Lost Earth Brücke

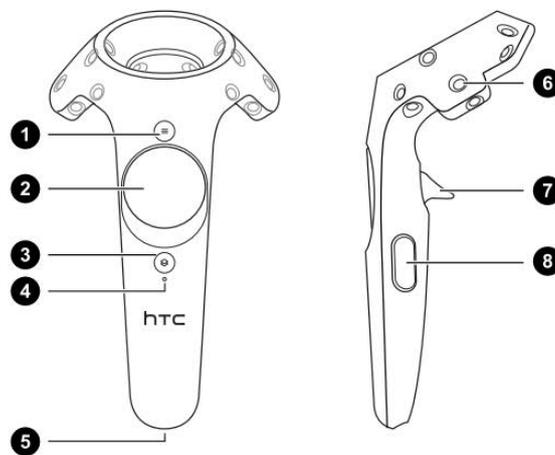
6.5 Umsetzung der Interaktionskonzepte

Für die erste Iteration wurde als Framework für die Interaktionen NewtonVR¹ genutzt. Es zeigte sich jedoch schnell, dass der Schwerpunkt dieses Frameworks auf der Berechnung von physikalischen Ereignissen liegt. Daher wurde zum VRTK Framework gewechselt. Während dem Lauf der Arbeit wurde, das VRTK mehrmals aktualisiert und bietet mittlerweile viele nützliche Funktionen². VRTK erlaubt es beispielsweise unabhängig vom der genutzten Hardware, die einzelnen Interaktionen zu implementieren. Des Weiteren bietet es die Möglichkeit ein VR-System zu simulieren, damit die Entwicklung von VR-Hardware auch ohne VR-Headset umsetzbar ist.

Die *Trigger* werden genutzt für die Interaktion mit Objekten auf Distanz. Der *Grip*-Knopf wird genutzt zum Greifen von Objekten. Mit Hilfe des *Menü*-Knopfs kann der Benutzer seine Position im Raum zurücksetzen. Der *System*-Knopf aktiviert des Steam-Overlay und bietet Zugriff auf die Headset-Kamera. Das *Touchpad/Trackpad* wird genutzt zur Teleportation.

In Abbildung 6.10 werden die Bezeichnungen für die einzelnen Eingabe-Möglichkeiten des HTC Vive Controllers zusammengefasst.

In VIEW werden dem Benutzer auf dem rechten Controller alle Knöpfe erklärt (siehe Abb. 6.10).



1	Menu button
2	Trackpad
3	System button
4	Status light
5	USB charging adapter
6	Tracking sensor
7	Trigger
8	Grip button

Abbildung 6.9: HTC Vive
Controller Knöpfe
Quelle: Vive User Guide[HTC16]

¹ NewtonVR: <https://github.com/TomorrowTodayLabs/NewtonVR>

² VRTK Features: <https://vrtoolkit.readme.io/>

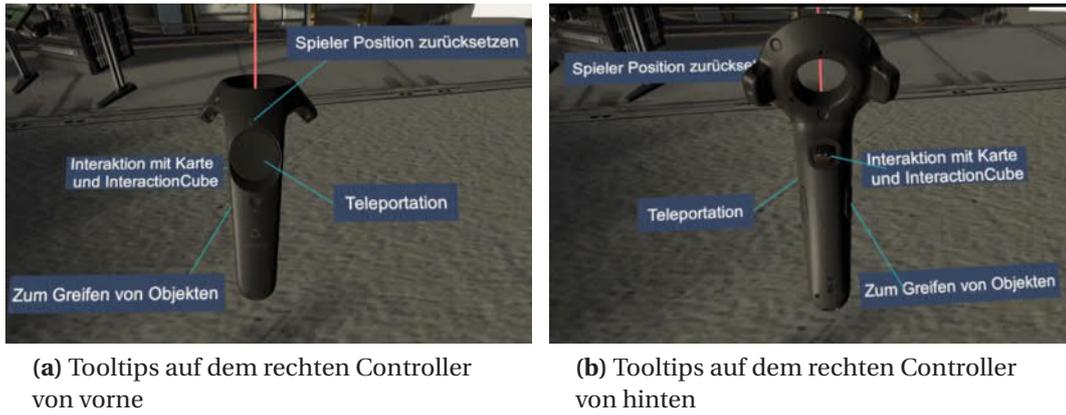


Abbildung 6.10: Tooltips auf dem rechten Controller

6.5.1 Lokomotion

Der Benutzer bewegt sich über größere Distanzen mit Hilfe der *Arc Teleport*-Funktion. Durch Drücken auf das Touchpad erscheint ein Bogen, welcher grün wird, wenn der Benutzer auf eine valide Oberfläche zeigt (siehe Abb. 6.11a). Lässt der Benutzer das Touchpad los, wird das Bild für 0,6 ms schwarz und der Benutzer wird zu dem angezielten Bereich teleportiert. Ist der gewählte Bereich nicht zulässig, so wird der Bogen rot und beim Loslassen erfolgt keine Teleportation (siehe Abb. 6.11b). Kleinere Distanzen kann der Benutzer auch selbst durch *room-scale* im realen Raum zu Fuß erkunden.

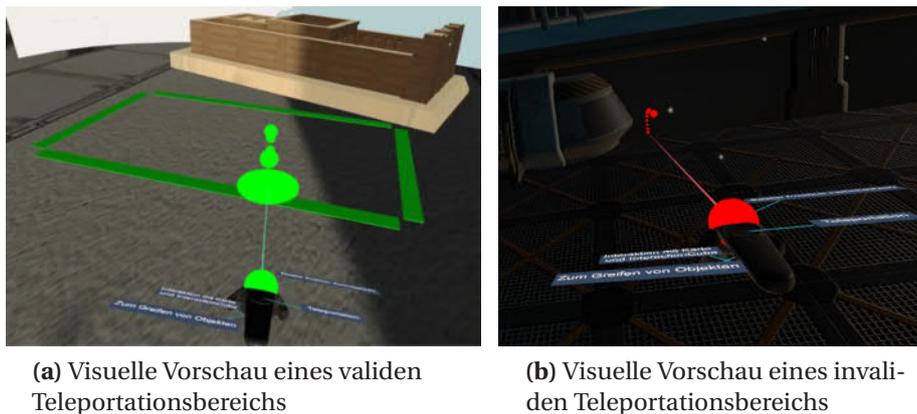
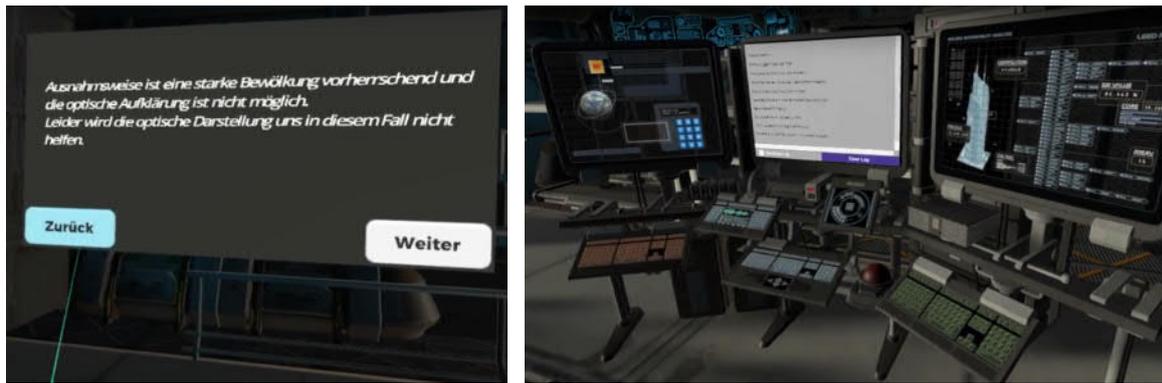


Abbildung 6.11: Teleportations Funktionalität im Überblick

6.5.2 Anweisungen und Debugging Konsole

Rechts vom Mapdisplay wird zwecks der Studie eine Tafel mit Anweisungen angezeigt (siehe Abb. 6.12a). Der Benutzer kann mit Hilfe der *Zurück*- und *Weiter*-Knöpfe zwischen den verschiedenen Seiten vor und zurück Blättern. Wird ein Knopf mit Hilfe des Strahls anvisiert, so färbt dieser sich blau, um dem Benutzer ein visuelles Feedback zu geben.

Die Debugging Konsole diene während der Entwicklung zur vereinfachten Inspektion der Fehlermeldungen (siehe Abb. 6.12b). Im Textfeld wird die gesamte Konsolenausgabe angezeigt, diese kann dann in VR gelesen werden, ohne das HMD abzunehmen. Dieses Objekt befand sich in einem versteckten Layer der Blender-Datei und wurde wegen des passenden Aussehens auch in der Version, die für die Studie genutzt wurde beibehalten.



(a) Tafel mit Instruktionen. Benutzer zielt mit Strahl auf *Zurück*

(b) Debugging Konsole

Abbildung 6.12: Weitere Objekte im virtuellen Raum

6.5.3 InteractionCube

Auf dem linken Controller befindet sich ein Würfel-förmiges Objekt mit Knöpfen und Beschriftung. Während der Entwicklung hat dieser Würfel die Beschriftung *InteractionCube* erhalten und wird auch hier so bezeichnet.

Das Konzept des InteractionCubes ist inspiriert von der Werkzeug- und Farbpalette der *Google Tilt Brush*-Anwendung [Goo16]. In dieser Anwendung werden die verfügbaren Werkzeuge in der linken Hand angezeigt. Mit der rechten Hand kann der Benutzer die gewünschte Option wählen.

In VIEW werden dem Benutzer ähnliche Menüs präsentiert. Auf dem vorderen Panel (siehe Abb. 6.13a) hat der Benutzer Zugriff auf die Optionen *Karte zurücksetzen* und *3D-Modus*. Auf dem rechten Panel (siehe Abb. 6.13c) kann er zwischen den unterschiedlichen Schichten der Darstellung wechseln. Das linke, hintere und obere Panel wurden für diesen Prototyp nicht genutzt, können aber für zukünftige Versionen mit weiteren Funktionen versehen werden. Hierzu gehören zum Beispiel die zuvor erklärten Konzepte der Helligkeit- und Kontrastanpassung.

Zum Anwählen einer Option zielt der Benutzer mit dem Strahl des rechten Controllers auf das Feld. Trifft der Strahl auf eine valide Option, so färbt er sich grün und die Option wird farblich hervorgehoben (siehe Abb. 6.13d). Wird keine valide Option getroffen, so bleibt der Strahl rot. Zum Anwählen der gewünschten Option, betätigt der Benutzer den *Trigger*.



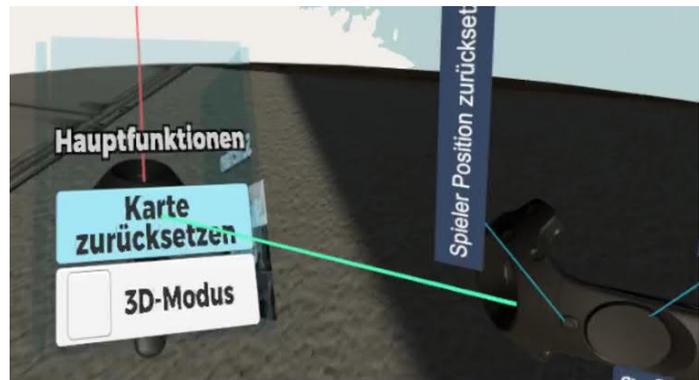
(a) Hauptfunktionen-Panel des Interaction-Cubes



(b) Proband bei der Interaktion mit dem Interaction-Cube



(c) Schichten-Panel des InteractionCubes



(d) Anwählen einer InteractionCube Option

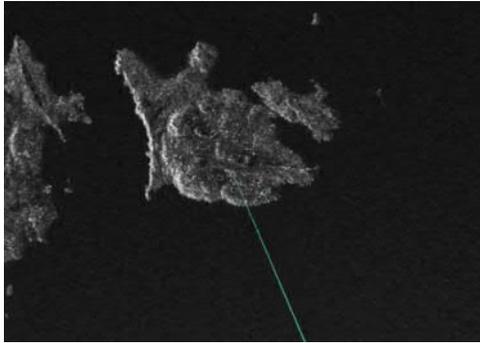
Abbildung 6.13: Überblick zum Interaktioncube

Karte zurücksetzen Diese Funktion dient dazu, dem Benutzer eine Möglichkeit zu bieten, schnell zur ursprünglichen Position auf der Karte zurückzuspringen. Des Weiteren, wird der Zoom-Level auf den initialen Wert zurückgesetzt. Das Ganze passiert nicht ruckartig, sondern in einer flüssigen Bewegung, damit der Nutzer nicht die Orientierung verliert.

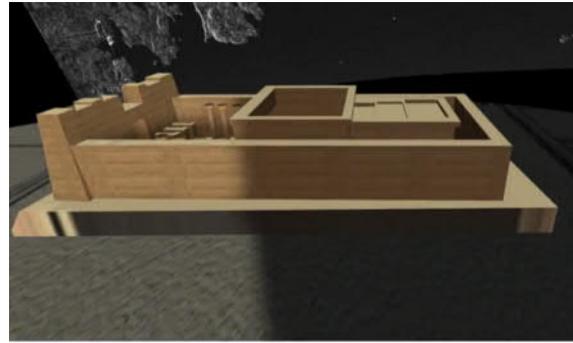
3D-Modus Ähnlich wie *Google Earth* können in VIEW beliebige Punkte auf der Karte mit zusätzlichen Funktionen und Informationen besetzt werden. In dieser Version des Prototyps wurde der *Mandulis-Tempel* in Ägypten mit einem 3D-Modell versehen. Bei Aktiviertem 3D-Modus kann der Benutzer den Tempel auf der Karte mit Hilfe des Strahls anzielen (siehe Abb. 6.14a). Betätigt er anschließend den *Trigger*, so wird vor ihm das 3D-Modell im virtuellen Raum geladen (siehe Abb. 6.14b). Das Modell befindet sich auf einem unsichtbaren, drehbaren Podest. Dies erlaubt es dem Benutzer durch Greifen des Modells, es hin und her zu drehen. Zum Greifen, bewegt der Benutzer den Controller zum Objekt hin. Ist der Controller nahe genug, färbt sich das Objekt blau und der Benutzer kann es mit Hilfe des *Grip*-Knopfs greifen. Sobald das Objekt gegriffen ist, verschwindet die blaue Verfärbung wieder und der Benutzer kann es durch Bewegen des Controllers in beliebige horizontale Richtungen rotieren.

Beim Aufrufen des Tempels wächst er bis zu einer vordefinierten Größe. Wird der 3D-Modus deaktiviert, schrumpft das Objekt wieder, bis es nicht mehr sichtbar ist und wird dann deaktiviert.

Während der Evaluation (siehe Kapitel 7) sollen die Probanden die Position des Tempels auf der Karte selbst entdecken. Da eine visuelle Angabe der Verfügbarkeit eines 3D-Modells in diesem Bereich die Position des Tempels verraten würde, wird kein Indikator auf der Karte angezeigt. Derartige Indikatoren sollten jedoch für zukünftige Versionen implementiert werden.



(a) Position des Mandulis Tempels auf der Karte



(b) 3D-Modell des Mandulis Tempels

Abbildung 6.14: 3D-Modus am Beispiel *Mandulis Tempel*

Schichten Auf der rechten Seite des InteractionCubes hat der Benutzer Zugriff auf die unterschiedlichen Schichten der Karte. Diese können im Code beliebig gesetzt werden, beziehungsweise in der *mapInfo.cfg*-Datei konfiguriert werden. In der für die Studie genutzte Version wurden die Schichten *Vektor*, *Optisch* und *Radar* konfiguriert.

Für die Vektor-Darstellung wird der OpenStreetMap TileServer vom Fraunhofer IOSB genutzt¹. Für die optische Schicht wurde die von *NASA Visual Earth* bereitgestellten Aufnahmen des Assuan-Gebiets² genutzt. Diese wurden mit Hilfe von *Paint.NET* mit Wolken überdeckt. Der Zweck davon ist, die Probanden während der Studie zu motivieren, die Radar-Aufnahmen zu nutzen. Für die Radar-Aufnahmen wurden die Bilder des TerraSar-X Radar Satelliten³ genutzt.

Jede Option ist beschriftet und hat als Hintergrund eine Vorschau der jeweiligen Darstellung (siehe hierzu Abb. 6.13c).

Beim Anwählen einer Schicht, wird die aktuelle Schicht deaktiviert und anschließend die gewünschte Schicht aktiviert. Die zuvor genutzten Schichten bleiben noch im Hintergrund geladen und können somit schnell wieder aufgerufen werden, ohne dass sie erneut heruntergeladen werden müssen. Zudem ist es möglich mehrere Schichten zu überlagern.

¹ Fraunhofer IOSB Tile Server: <https://tile.iosb.fraunhofer.de/>

² NASA Visual Earth - Asawn High Dam: <https://visibleearth.nasa.gov/view.php?id=85992>

³ TerraSAR-X Radar Satellite Image Aswan Dam: <http://www.satimagingcorp.com/gallery/terrasar-x/terrasar-x-radar-satellite-image-aswan-dam/>

6.6 Anwendungsbeispiel

Zum Abschluss dieses Kapitels wird in dieser Sektion ein allgemeines Anwendungsbeispiel beschrieben. Hierzu gehört sowohl die Vorbereitung als auch die Nutzung von VIEW. In Kapitel 7 folgt ein konkretes Anwendungsbeispiel, im Kontext der durchgeführten Studie. In Abbildung 6.15 ist das Anwendungsbeispiel illustriert.

Aufbau Zur Vorbereitung der Nutzung muss *Rolle#1* den GeoServer vorbereiten. Hierzu gehört das Starten des GeoServers, das Laden der benötigten Schichten auf diesen Server und das Anpassen der Konfigurationsdatei. *Rolle#2* startet VIEW und setzt das VR-HMDs auf.

Rollen *Rolle#1* ist zuständig für die Vorbereitung des GeoServers. *Rolle#2* ist zuständig für die Bildauswertung. Die Aufgabe von *Rolle#1* kann von *Rolle#2* übernommen werden.

Szenario Sobald der Aufbau vorbereitet ist, kann *Rolle#2* mit der Bildauswertung beginnen. Hierzu gehört das Bewegen der Karte, das Heran- und Hinauszoomen der Bereiche und das Suchen nach Auffälligkeiten. Abhängig von den Begebenheiten wechselt er mit Hilfe des InteractionCubes zwischen den unterschiedlichen Schichten.

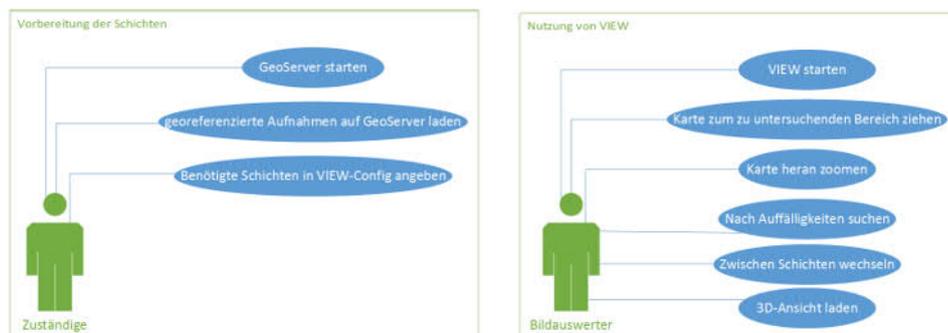


Abbildung 6.15: Use Cases zur Vorbereitung und zur Nutzung von VIEW

Evaluierung von VIEW

Zur Evaluierung von dem während dieser Masterarbeit erstellten Prototypen VIEW wurde eine Studie mit 16 Probanden durchgeführt. Schwerpunkt dieser Studie war die Untersuchung der Presence, Interaktionen und Gebrauchstauglichkeit des Systems.

In diesem Kapitel wird der Aufbau, die Durchführung und die Resultate dieser Studie präsentiert.

7.1 Versuchsaufbau

Zur Evaluierung von VIEW sind eine Reihe von Faktoren zu beachten. Dazu kommt, dass diese Faktoren voneinander abhängig sein können. Beispielsweise könnte ein nicht-intuitives Interaktionskonzept die Immersion und die Usability vermindern.

Presence und Immersion In Kapitel 2 und Kapitel 5 wurden die Faktoren Presence und Immersion untersucht. Laut Schubert et al. handelt es sich bei der Immersion um eine Variable der Technologie, welche objektiv beschrieben werden kann [Igr]. Bei der *Presence* handelt es sich hingegen um eine Variable, die subjektiv vom Benutzer gefühlt wird. Daher werden zur Messung von Presence, Fragebögen genutzt, bei welchen der Benutzer selbst sein Empfinden angibt.

Interaktion und Arbeitsbelastung Die Interaktionskonzepte und deren Umsetzung wurden in Kapitel 5 und Kapitel 6 diskutiert. Um die kognitive Immersion zu gewährleisten, ist es nötig, dem Nutzer intuitive und nicht-obstruktive Werkzeuge zu bieten, welche es ihm ermöglichen, die Aufgabe effektiv zu lösen.

Gebrauchstauglichkeit Neben den bereits behandelten Themen, ist es wichtig, zu untersuchen, ob das erstellte Assistenzsystem den Benutzer bei seiner Aufgabe als Bildauswerter unterstützt. Hierzu bietet sich an, die Gebrauchstauglichkeit (*Usability*) zu untersuchen.

Die Usability eines Systems wird im ISO Standard *ISO 9241 Teil 11* [Iso98] definiert. Es wird angegeben, dass die Usability nur gemessen werden kann, wenn der Kontext des Systems beachtet wird. Dies heißt genauer: *wer nutzt das System, wofür wird es genutzt und in welchem Umfeld wird es genutzt*. Dazu kommt, dass die Messung der Usability mehrere Aspekte hat:

Effektivität Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen.

Effizienz Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen.

Zufriedenheit Freiheit von Beeinträchtigung und positive Einstellung gegenüber der Nutzung des Produkts

7.1.1 Hypothese

Das Postulat, welches für die Studie gestellt wurde, ist:

Postulat: *VIEW bietet dem Bildauswerter eine effektive und immersive Arbeitsunterstützung. Dies wird ermöglicht durch die Nutzung von intuitiven Interaktionskonzepten.*

Beim Postulat werden mehrere Aspekte hervorgebracht, welche einzeln geprüft werden müssen. Die Frage, ob die Arbeitsunterstützung *immersive* ist, hängt von der vom Benutzer empfundenen Wahrnehmung ab – der *Presence*. Damit die Arbeitsunterstützung *effektiv* ist, muss sie gebrauchstauglich (Kontext Usability) sein. Ferner müssen Fragen zu der *Intuitivität* und der *Affordance* gestellt werden.

Das Postulat wird in drei Hypothesen aufgeteilt, um diese Trennung klarer zu machen:

Presence *Die Probanden fühlen sich in die virtuelle Welt vertieft.*

Usability *Den Probanden gelingt es die Aufgabe zu erledigen.*

Intuitivität *Die Interaktionskonzepte werden als intuitiv empfunden.*

Die Aussagen der drei Hypothesen werden mit Hilfe von mehreren Subhypothesen ausgebreitet:

Presence Im Kontext der Immersion werden (wie in Kapitel 5 bereits angedeutet) nicht alle Aspekte der Immersion untersucht. Die höchste Priorität hat die Sicherstellung, dass der Benutzer sich in die virtuelle Welt vertieft fühlt [Igr]. Um dieses Ziel zu erreichen bedarf es mehrerer Faktoren. Es bedeutet im direkten Sinne, dass die *Presence* sichergestellt werden muss [Igr15]. Des Weiteren, darf die Interaktion nicht überfordernd wirken [Lac16] und sie darf nicht Symptome der VR-Krankheit auslösen [Goo15]. Letztlich müssen die technischen Faktoren wie die Bildschirmwiederholrate, das FOV und die erzielten Bilder pro Sekunde beachtet werden.

Daher werden im Kontext der Immersion folgende Subhypothesen gestellt:

P1 Die Probanden haben das Gefühl, sie würden sich physikalisch auf der Brücke des Raumschiffs befinden [Sch99] [Sch01]

P2 Die Probanden fokussieren ihre Aufmerksamkeit auf die Aufgabe [Sch99] [Sch01]

P3 Die gewählten Lokomotionskonzepte verursachen keine Vorkommen von VR-Cybersickness (mehrere Best Practice Guides z. B. [Goo15])

P4 Die Bildwiederholrate und andere technischen Aspekte genügen zur Sicherstellung eines angenehmen Erlebnisses (mehrere Best Practice Guides z. B. [Goo15])

P5 Die Probanden verspüren eine niedrige subjektiv wahrgenommene Arbeitsbelastung [Lac16]

Usability Im Kontext der Gebrauchstauglichkeit wird vor allem auf die Aspekte Effektivität und Zufriedenheit geachtet. Dies schließt sich aus der zu Beginn dieses Kapitels erklärten ISO-Richtlinie [Iso98]. In diesem Zusammenhang werden folgende Subhypothesen festgelegt:

U1 Die Probanden empfinden den Prototyp als gebrauchstauglich

U2 Der Prototyp erlaubt das effektive Bewältigen der Aufgabe

U3 Die Nutzung des Prototyps ist zufriedenstellend und nicht frustrierend

Intuitivität In Kontext der Intuitivität, wird vor allem in Frage gestellt, ob die Probanden die Interaktionen auf eine natürliche Art nutzen können, um die Aufgabe zu lösen. Sie sollen nicht Schwierigkeiten damit haben, die Konzepte kennenzulernen [Nor99].

I1 Die Probanden können die Interaktionskonzepte nachvollziehen

I2 Die Probanden verstehen die Interaktionskonzepte ohne weitere Anweisungen

7.1.2 Fragebögen und Bewertungsart

Für die Untersuchung zu den einzelnen Subhypothesen werden standardisierte Fragebögen genutzt. Im folgenden wird dargelegt, welche Fragebögen für welche Untersuchungen genutzt werden.

Presence

Zur Untersuchung der Presence (Subhypothesen **P1-P5**) werden zwei Fragebögen benötigt. Einerseits muss die subjektiv empfundene Arbeitsbelastung gemessen werden (**P5**), andererseits müssen die spezifischen Faktoren der Presence untersucht werden (**P1 und P2**). Die Subhypothesen **P3** und **P4** werden anhand eines freien Interviews und mit Hilfe einer technischen Bewertung untersucht.

Schlüsse über die Arbeitsbelastung (**P5**) lassen sich über den *NASA Task Load Index* (kurz NASA-TLX) schließen[Har88]. Hierbei handelt es sich um einen Fragebogen, welcher die subjektiv wahrgenommene Arbeitsbelastung (Workload) eines Probanden beim Bewältigen einer Aufgabe einschätzt. Der NASA-TLX wurde von der *Human Performance Group* in NASA's (National Aeronautics and Space Administration) Ames Forschungszentrum (ARC) entwickelt.

Offiziell wird der originale Fragebogen nur in der englischen Variante angeboten. Für diese Studie wurde die deutsche Übersetzung von Christian Lehsing und Katharina Seifert genutzt (entnommen von [Gru]).

Ursprünglich bestand der NASA-TLX aus zwei Teilen. In einem ersten Teil wird anhand von sechs Unterskalen der gesamte Arbeitsaufwand eingeschätzt. Jeder dieser sechs *Tasks* kann 100 Punkte (in fünf-Punkt Schritten) erreichen. Diese Punkte werden summiert und bilden somit den sogenannten *task load index*.

Im zweiten Teil wird versucht, diesen Unterskalen ein individuelles Gewicht zuzuweisen. Dabei werden die Probanden gebeten, jeden *Tasks* paarweise zu vergleichen und den *Task* auszuwählen, welcher ihnen als wichtiger erscheint. Für jede Unterskala wird gezählt, wie oft sie gewählt wurde. Diese Anzahl wird dann mit dem dazugehörigen Wert des ersten Teils multipliziert und anschließend durch 15 geteilt. Das Resultat ist der sogenannte *work load score*, der von 0 bis 100 reicht.

Dieser zweite Teil wird in einer anderen Variante des Fragebogens, dem RawTLX [Bye89], nicht durchgeführt. Laut Hart [Har06] reicht es, den ersten Teil durchzuführen, um ein gleichwertiges Resultat zu erreichen. Dies bedeutet, dass der Fragebogen, den die Probanden beantworten sollen, aus folgenden Abfragen besteht:

Geistige Anforderung

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

Körperliche Anforderung

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

Zeitliche Anforderung

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

Leistung

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

Anstrengung

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

Frustration

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

Die Probanden sollen retrospektiv für jede Abfrage ihre empfundene Beanspruchung auf einer zwanzig-stufigen Likert-Skala ankreuzen.

Obwohl es viele digitale Verfahren gibt, um diese Werte abzufragen, wurde der RawTLX-Fragebogen auf Papier ausgeteilt, da das Ausfüllen auf dem Computer als anstrengender empfunden wird und somit zu einem erhöhten *Workload* führt [Noy07]. Der vollständige Fragebogen befindet sich im Anhang (2).

Zur Messung der Presence bieten sich mehrere Fragebögen an:

- Witmer & Singer: Presence questionnaire [Wit98]
- Slate- Usoh-Steed Presence questionnaire [Uso00]
- Igroup Presence Questionnaire (IPQ) [Igr15]

Zudem ist Presence messbar mit Hilfe von Verfahren wie EKG. Es ist möglich Herzrate und ähnliche biologisch-orientierte Messungen durchzuführen, um Schlüsse über die verspürte Presence zu schließen [Mal15]. Dies ist jedoch eher passend, bei VR-Anwendungen, wie Shooter und Achterbahn-Simulatoren, bei denen der Proband viel Aufregung verspürt.

Der Fragebogen von Witmer und Singer ist relativ zutreffend, aber für die geplante Dauer der Studie zu lang. Der Fragebogen von Slate-Usoh-Steed behandelt nicht genügend die in den Subhypothesen festgelegten Faktoren.

Für die Untersuchung der Presence wurde der *Igroup Presence Questionnaire* gewählt. Der Grund hierfür ist, dass beide Subhypothesen (**P1** und **P2**) sich direkt aus dem Resultat des Fragebogens schließen lassen. Der IPQ Fragebogen besteht aus 14 Aussagen, welche auf einer Skala von -3 bis +3 bewertet werden sollen. Das Resultat der Durchführung dieses Fragebogens, sind drei Faktoren, welche Aussagen über die unterschiedlichen Facetten der Presence geben. Zusätzlich ergibt sich der **G**-Wert, welcher eine allgemeine Aussage darüber gibt, ob der Benutzer das Gefühl hatte, sich in der virtuellen Umgebung zu befinden [Sla94].

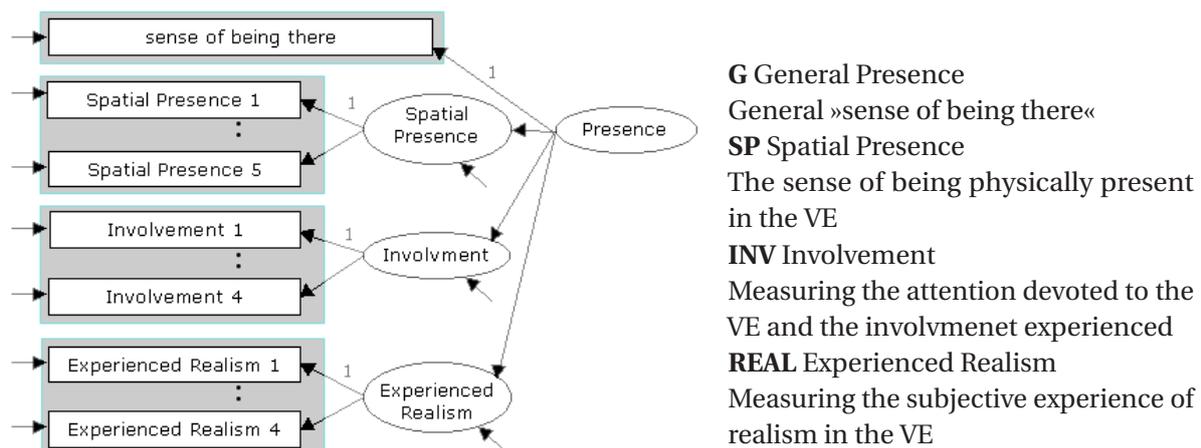


Abbildung 7.1: Aufbau der konfirmatorischen Faktorenanalyse [Igr]

Diese Werte und die Gewichtung der einzelnen Fragen wurden aus einer konfirmatorischen Faktorenanalyse geschlossen [Sch99][Sch01]. Der Aufbau befindet sich zusammengefasst in Abb. 7.1. Ein zusätzlicher Vorteil des IPQ-Fragebogens ist, dass die offizielle deutsche Übersetzung von den Autoren gegeben ist. Der komplette Fragebogen befindet sich im Anhang 3.

In Bezug auf die Subhypothesen, kann **P1** aus **SP** und **P2** aus **INV** geschlossen werden. Zudem bietet **G** eine allgemeinere Angabe zum Gefühl *sich im virtuellen Raum* zu befinden. **REAL** - der Wert der angibt, wie real die virtuelle Welt wirkt - ist für diese Studie nicht von Interesse.

Weiterhin müssen zur Verbesserung der Presence die technischen Aspekte sichergestellt werden (**P4**). In Kapitel 4 wurden die Aspekte der Bildqualität untersucht und in Kapitel 5.1.2 wurden die Anforderungen an Hardware und Design geschildert.

Außerdem werden die Probanden in einem freien Interview darüber befragt, ob sie Anzeichen der VR-Krankheit, also Schwindelgefühle oder Übelkeit, bei dem Bewegen in *room-scale VR* oder beim Teleportieren, verspürt haben (**P3**).

Usability

Für die Untersuchung der Usability bietet sich die SUS (System Usability Scale) [Bro96]. Hierbei handelt es sich um zehn Abfragen, die mit Hilfe einer Likert-Skala bewertet werden sollen. Die Likert-Skala ist fünf-stufig und reicht von »Stimme überhaupt nicht zu« bis zu »Stimme voll zu«. Die Durchführung dieser Befragung gibt einen globalen Überblick zu der subjektiven Einschätzung der Gebrauchstauglichkeit eines Systems. SUS kann auch mit einer geringen Anzahl an Probanden eine zuverlässige Einschätzung zu der Usability des Systems bieten [Sau13]. Offiziell wird der originale Fragebogen nur in der englischen Variante angeboten. Für diese Studie wurde die deutsche Übersetzung von CoreMedia [Loh13] genutzt. Der komplette Fragebogen befindet sich im Anhang 4. Folgende Abfragen werden an den Probanden gestellt:

- Ich denke, dass ich dieses System gerne regelmäßig nutzen würde.
- Ich fand das System unnötig komplex.
- Ich denke, das System war leicht zu benutzen.
- Ich denke, ich würde die Unterstützung einer fachkundigen Person benötigen, um das System benutzen zu können.

- Ich fand, die verschiedenen Funktionen des Systems waren gut integriert.
- Ich halte das System für zu inkonsistent.
- Ich glaube, dass die meisten Menschen sehr schnell lernen würden, mit dem System umzugehen.
- Ich fand das System sehr umständlich zu benutzen.
- Ich fühlte mich bei der Nutzung des Systems sehr sicher.
- Ich musste viele Dinge lernen, bevor ich mit dem System arbeiten konnte.

Zur Berechnung des SUS, müssen zuerst alle Werte summiert werden. Die Werte der Fragen 1, 3, 5, 7 und 9 sind die jeweilige Bewertung auf der Likert-Skala minus 1. Die Werte der Fragen 2, 4, 6, 8 und 10 sind 5 minus die jeweilige Bewertung auf der Likert-Skala. Durch diese Berechnung erhält jede Frage einen Wert zwischen 0 und 4. Im Anschluss wird die berechnete Summe mit 2.5 multipliziert und ergibt somit den SUS-Score. Dieser liegt zwischen 0 und 100. Dieser Wert erlaubt die Unterscheidung zwischen einem nutzbarem und einem unbrauchbarem System.

Intuitivität

Während der Studie werden Notizen zu den einzelnen durchgeführten Interaktionen aufgenommen. Untersucht werden die Interaktionen, die dazu benötigt werden, um die Aufgabe erfolgreich abzuschließen. Die Probanden erhalten neben den in VIEW gegebenen Anweisungen, keine Erklärung zu den unterschiedlichen Interaktionen. Optionale Interaktionen, die für das Bewältigen der Aufgabe nicht benötigt werden, wie das Zurücksetzen der Karte oder der Spielerposition, werden nicht beachtet. Folgende Interaktion haben Priorität:

- Startknopf betätigen
- Teleportieren
- Greifen
- Weiter/Zurück betätigen
- Karte ziehen
- Karte zoomen
- Layer wechseln
- 3D-Tool aktivieren
- 3D-Tool nutzen

Die Interaktionen werden mit drei möglichen Werten bewertet:

-1 Der Proband konnte die Aktion nicht durchführen und benötigte Anweisungen

0 Der Proband konnte die Aktion selbstständig durchführen, hatte jedoch Schwierigkeiten (z. B. Öfters falschen Knopf betätigt)

+1 Der Proband konnte die Aktion selbstständig und problemlos durchführen

Diese Werte werden anschließend in einer Tabelle jeweils *pro Proband* und *pro Interaktion* gemittelt. Zusätzlich werden die Probanden im freien Interview befragt, ob sie die Interaktionen als intuitiv empfunden haben.

Sonstige Abfragen

Neben den bereits geschilderten Fragebögen, wurde noch ein weiterer Fragebogen angefertigt, welcher Auskunft über persönliche Aspekte der Probanden geben soll. Der komplette Fragebogen befindet sich im Anhang (1). Er beinhaltet folgende Abfragen:

- Alter
- Geschlecht
- Bildungsgrad
- Erfahrung mit Computerspielen
- Erfahrung mit Virtual Reality Anwendungen

In einem freien Interview wurden die Probanden darüber befragt, wie gut sie die Bildqualität gefunden haben, ob sie die Interaktionskonzepte als intuitiv empfunden haben, ob sie irgendeine Art von Schwindelgefühlen empfunden haben und ob ihnen Effekte wie auf Metall spiegelnde Sonneneinstrahlung bemerkt haben. Zudem erhielten die Probanden hier die Möglichkeit, frei über ihre Meinung zu VIEW zu berichten.

Zusammenfassung

Im Anhang (siehe Anhang 5) befindet sich eine Grafik, welche den Ablauf vom Postulat zu den einzelnen Bewertungsarten zusammenfasst. In Abbildung 7.2 befindet sich eine vereinfachte Version.

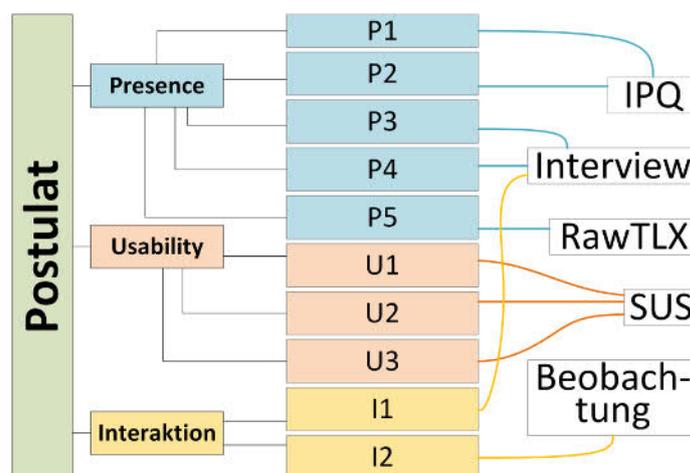


Abbildung 7.2: Vereinfachte Version der Übersicht zu den Hypothesen

7.1.3 Szenario

In diesem Unterkapitel werden der Ablauf, die Story und die genutzten Inhalte der Studie erläutert. Der Ablauf enthält eine kurze Zusammenfassung des Vorhabens und der Aufgaben, die die Probanden erhalten. Anschließend wird die Story, die den Probanden erzählt wird genauer dargestellt. Zuletzt werden die Hilfsmittel, die für die Studie genutzt wurden aufgelistet.

Ablauf

Das Einführungslevel besteht aus einer einfachen Szene, die dazu dient, dem Benutzer Zeit zu geben, sich mit VR vertraut zu machen. Es wird ihm die Möglichkeit geboten, die einzelnen Interaktionskonzepte, wie das Teleportieren und das Greifen von Objekten auszuprobieren. Ein Ausschnitt des Einführungslevels ist in Abb. 7.3c zu sehen. Zusätzlich wird sichergestellt, dass er das VR-Headset richtig aufgesetzt und eingestellt hat. Genauer bedeutet dies, dass die Halterung des Headsets angenehm auf dem Kopf befestigt ist und die IPD-Einstellung korrekt ist. Vor der Studie wurde jeder Proband aufgefordert, mit Hilfe des online IPD-Tools¹ den eigenen Augenabstand herauszufinden.

Das IPD-Tool erlaubt es, eine ungefähre Schätzung des Pupillen-Abstands zu erhalten. Dazu wird lediglich eine Webcam und eine Karte (beispielsweise eine beliebige Mitgliedskarte) benötigt. Der Proband soll die Karte vor den Mund, unterhalb der Nase halten (siehe Abb. 7.4a). Sobald der Proband in Position ist, wird ein Bild genommen. Anschließend können die Position der Pupillen und die Ecken der Karte angepasst werden (siehe Abb. 7.4b). Sobald dies erfolgt ist, ist der Prozess abgeschlossen und das Tool gibt den ungefähren Pupillenabstand in Millimeter an (siehe Abb. 7.4c).

Ähnlich wie in der VIEW-Szene werden Tooltips über den einzelnen Knöpfen des Controllers in VR angezeigt. Zudem wird dem Probanden erklärt, dass er mit dem System-Knopf (kleiner Knopf unter dem Touchpad) die Headset-Kamera aktivieren kann, um beispielsweise leichter das Kabel von den Beinen zu lösen, falls sich dieses verdreht hat. Sobald der Proband sich bereit fühlt, kann er durch betätigen des großen roten Knopfes das Hauptlevel laden. Hier wird der Proband gebeten, den Controller mit dem *Interaction Cube* in die linke Hand zu nehmen. VIEW übernimmt dies im Normalfall automatisch, durch häufiges Wechseln der Controller, könnte dieser Mechanismus jedoch überlistet werden.

Nach dem Betätigen des Knopfes kommt es zum Hauptteil der Aufgabe. Dem Probanden wird eine Story (in folgendem Unterkapitel *Story* 7.1.3 genauer beschrieben) erzählt, welche ihn in die

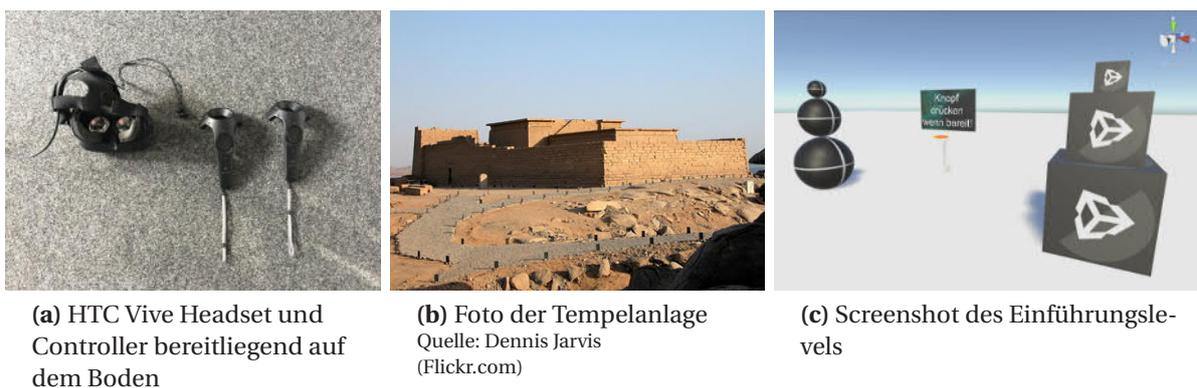


Abbildung 7.3: Überblick zu unterschiedlichen genutzten Materialien

¹ IPD-Tool: <http://www.trylive.com/demos/trylive-eyewear/pupil-distance>



Abbildung 7.4: IPD-Tool Übersicht

Lage eines Bildauswerters versetzen soll. Während dessen soll er eine Reihe von Aufgaben erledigen. Diese beinhalten:

- Wechseln von Kartenansichten (Vektor, optisch, Radar)
- Navigation über die Karte mit Hilfe von Bewegungen und Zoomen
- Identifizierung einer Insel, dessen Position nur ungefähr angegeben wird
- Aktivierung des 3D-Tools und hervorrufen des 3D-Modells vom Mandulis-Tempel
- Betrachtung dieses 3D-Modells aus mehreren Winkeln

Die erste Hälfte der Anweisungen werden ihm rechts vom Hauptdisplay angezeigt. Mit den Knöpfen *Weiter* und *Zurück* kann er zwischen den unterschiedlichen Seiten wechseln. Diese Art der Übermittlung der Story erlaubt es dem Benutzer, die Kopfhörer anzubehalten. Anfangs war es geplant, dem Probanden die Story vorzulesen. Dies würde jedoch verhindern, dass die 3D-Audio Effekte gehört werden können.

Die zweite Hälfte der Anweisung werden dem Probanden mündlich übermittelt. Er wird gebeten, die Kopfhörer abzulegen. Dies ist nötig, da die weiteren Anweisungen abhängig von den Aktionen des Probanden sind und angepasst werden müssen.

Im Abschluss der Aufgabe wird der Proband gebeten das Headset auszuziehen und eine Reihe von Fragebögen (SUS, Raw TLX und Presence) auszufüllen und auf die Fragen des freien Interviews zu antworten.

Story

Nach der zuvor bereits beschriebenen Einführung beginnt der Hauptteil der Versuchsdurchführung. Den Probanden soll eine glaubwürdige Aufgabe zugeteilt werden, damit sie sich in die Lage eines Bildauswerters versetzen können. Zur Erreichung dieses Ziels haben wir eine Story geschrieben, welche den Probanden während ihrer Aufgabe nach und nach mitgeteilt wird. Hierbei ist die Hauptidee, dass ein Tempel im Assuan-Damm Gebiet gesucht wird und anschließend inspiziert werden soll.

Im folgenden Teil werden Story-Inhalte, welche dem Probanden erzählt werden in

kursiver Schriftart, innerhalb eines Kastens

dargestellt. Dies macht die Trennung zwischen Story und der dazugehörigen Erklärung übersichtlicher und lesbarer.



Abbildung 7.5: Unterschiedliche Darstellungen des Assuan-Gebiets

Der Proband befinden sich nach dem Aufsetzen der HTC Vive auf der virtuellen Brücke von Lost Earth. Hier wird bereits die Vektor-Darstellung des Assuan-Damm Gebiets (siehe Abb. 7.5a) auf dem großen Karten Display angezeigt. Rechts davon befinden sich seine Missionsziele.

Missionsziele werden in diesem Fenster angezeigt.

Drücken Sie auf Weiter, sobald Sie bereit für die Mission sind!

Sobald der Proband seine Orientierung gefunden hat, kann er mit *Weiter* zur nächsten Seite wechseln.

Aufständische haben den Assuan-Staudamm unter ihre Kontrolle gebracht und drohen ihn zu sabotieren, um die Stromversorgung lahm zu legen. Wegen der wichtigen Rolle des Wasserkraftwerks für ganz Ägypten (10% der Stromerzeugung Ägyptens), wären die Folgen dramatisch.

Dem Proband wird erzählt, was passiert ist und welche Folgen dies haben könnte.

Auf dem großen Display haben wir bereits die Vektordarstellung als Übersichtskarte für das Gebiet des Dammes geladen. Augenzeugen haben berichtet, dass die Aufständischen in Richtung Süd-Westen über den Nassersee geflüchtet sind.

Anschließend wird ungefähr angedeutet, wo unser Ziel sich befinden könnte.

Die örtlichen Gegebenheiten müssen für eine Luftüberwachung ausgekundschaftet werden.

Wir müssen herausfinden, wo sie sich befinden könnten!

Auf dem Kasten in Ihrer linken Hand können Sie nun zur optischen Karte wechseln.

Nachdem der Proband den Kontext des Szenarios erhalten hat, wird er gebeten, zu dem optischen Layer zu wechseln.

Ausnahmsweise ist eine starke Bewölkung vorherrschend und die optische Aufklärung ist nicht möglich.

Leider wird die optische Darstellung uns in diesem Fall nicht helfen.

Er wird bemerken, dass durch die Wolken nicht genug zu erkennen ist (siehe Abb. 7.5b).

Der Radarsatellit TerraSAR-X hat ein hochauflöstes Radarbild der Szene erstellt. Radar-Aufnahmen erlauben es, 'durch' Wolken zu sehen. Versuchen wir es also mit den Radar-Aufnahmen!

Da die Auswertung mit Hilfe von der optischen Darstellung nicht möglich ist, wird die Nutzung der Radar-Aufnahme motiviert. Der Proband wird gebeten, zum Radar-Layer zu wechseln (siehe Abb. 7.5c).

Fragestellung #1:

Wir wissen, dass die Aufständischen in Richtung Süd-Westen geflohen sind. Sie können mit ihrem Boot noch nicht so weit gekommen sein!

Wo könnten Sie sich befinden?

Dies ist die letzte Anweisung, die der Proband von dem Missionsdisplay erhält. Ab hier werden weitere Instruktionen vom Studienleiter gegeben.

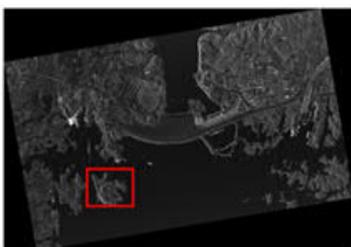
Für weitere Instruktionen, reden Sie bitte mit dem Einsatzleiter (Julien Hoffmann)!

Nehmen Sie dafür bitte die Kopfhörer ab.

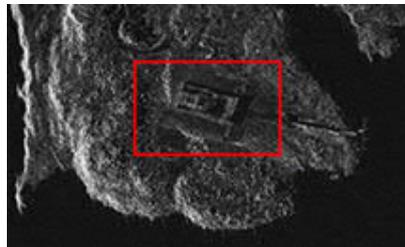
Der Proband soll nach einem möglichen Ort suchen, wo die Aufständischen sich befinden könnten. Hierbei muss er beide Controller nutzen, um sich durch das Gebiet zu navigieren. Gibt er dabei an, dass er eine Insel entdeckt hat, kann zum nächsten Punkt übergegangen werden. Falls er die Insel nicht findet, werden weitere Instruktionen erteilt, um dem Probanden zu helfen. Die gesuchte Insel befindet sich Süd-westlich vom Damm (siehe Abb. 7.6a).

Sehr gut! Es ist gut möglich, dass Sie sich auf dieser Insel befinden. Versuchen Sie die Insel genau zu betrachten. Erkennen Sie irgendeine Form von Gebäuden, Zelten, Fahrzeugen oder ähnlichen Strukturen. Oder sieht es wie eine verlassene Insel aus?

Der Proband soll die Insel genau betrachten und Rückmeldung geben, falls er etwas erkennt. Die Umrise des Tempels sind auf dem Radarbild deutlich zu erkennen.



(a) Bereich der gesuchten Insel



(b) Bereich der gesuchten Tempelanlage



(c) 3D-Modell des Mandulis Tempel
Quelle: VassilisP. (SketchUp.com)

Abbildung 7.6: Radaraufnahme und 3D-Modell des gesuchten Mandulis-Tempels

*Sehr interessant. Möglicherweise haben wir eine 3D-Übersicht dieses Gebäudes!
Wo exakt befindet sich diese Struktur?
Wählen Sie mit Hilfe des 3D-Tool das Gebäude an!*

Durch das Drücken auf den Bereich des Gebäudes (siehe Abb. 7.6b), wird ein 3D-Modell des Tempels geladen (siehe Abb. 7.3b).

Das ist aufregend! Wir haben tatsächlich ein 3D Modell dieses Bereichs.

Dieses 3D-Modell lässt sich durch Kopf- und Körperbewegungen aus verschiedenen Winkeln betrachten. Des Weiteren, kann der Benutzer mit Hilfe der Controller nutzen, um das Objekt greifen und drehen. Um sicher zustellen, dass er diese Möglichkeiten der Interaktion ausnutzt, werden ihm folgende Frage gestellt:

Fragestellung #2

Schauen Sie sich das Modell genau an.

- *Was für eine Art Gebäude könnte dies sein?*
- *Wie ist der aktuelle Aufbau der Infrastruktur?*
- *Wo könnten die Aufständischen sich versteckt haben?*
- *Gibt es alternative Routen, über die wir vordringen könnten?*

Wenn er nicht errät, dass es sich bei dem Gebäude um einen Tempel handelt, wird ihm erklärt, dass es der Mandulis-Tempel ist. Der Proband soll den Tempel aus verschiedenen Winkeln betrachten und versuchen schlüssige Folgerungen zu den Fragen zu finden. Bei diesen Fragen ist es für die Evaluierung nicht zwingend wichtig, was der Proband antwortet. Von Interesse ist es, herauszufinden, ob es dem Probanden leicht fällt, das 3D Modell aus verschiedenen Winkeln zu betrachten.

Tolle Arbeit! Wir wären dann fertig mit der Bildauswertung. Sie können das Headset nun ausziehen.

Sobald der Proband sich genügend mit den Fragen beschäftigt hat, wird er darauf hingewiesen, dass er die Aufgabe erfolgreich abgeschlossen hat und das Headset nun ablegen kann.

Inhalte

Für diese Studie werden die drei, in Abbildung 7.5 gezeigten, hochaufgelösten Luftaufnahmen des Assuan-Gebiets und das 3D-Modell des Tempels benötigt:

Vektor OpenStreetMap TileServer des Fraunhofer IOSBs ¹

Optisch Optische Aufnahme von NASA Visual Earth²

Radar TerraSar-X Radar Satelliten Aufnahmen ³

Tempel 3D-Modell des Mandulis Tempel von VassilisP ⁴.

¹ Fraunhofer IOSB Tile Server: <https://tile.iosb.fraunhofer.de/>

² NASA Visual Earth - Asawn High Dam: <https://visibleearth.nasa.gov/view.php?id=85992>

³ TerraSAR-X Radar Satellite Image Aswan Dam:

<http://www.satimagingcorp.com/gallery/terrasar-x/terrasar-x-radar-satellite-image-aswan-dam/>

⁴ 3D-Modell des Temples:

<https://3dwarehouse.sketchup.com/model/a3f3eb13c08efcadbf35f2a4d5209a3/Kalabsha-Temple-Aswan>

Test Objekt

Virtual Reality Image Exploitation Workspace (VIEW) ist der entwickelte Prototyp zur Durchführung von Bildauswertung in der virtuellen Realität (VR). VIEW würde mit Hilfe der Game Engine Unity3D¹ entwickelt und basiert auf C#.

Beim Starten der Anwendung wird zuerst das Einführungslevel geladen. In diesem befinden sich drei Kugeln und drei Kasten, welche gegriffen und geworfen werden können. Außerdem befindet sich ein Knopf, einige Meter von der Startposition, entfernt. Der Nutzer muss sich zum Knopf teleportieren und mit dem Controller den Knopf hinunter drücken, um zum Hauptlevel – der *Lost Earth Brücke* zu wechseln. Das 3D-Modell der Lost Earth Brücke ist das gleiche, das im Serious Game *Lost Earth 2307* genutzt wurde.

Mit Hilfe der beiden Controller, kann der Benutzer die Luftbildaufnahme, welche auf der Leinwand (*MapDisplay*) angezeigt werden, verschieben und vergrößern. Auf dem linken Controller befindet sich das *InteractionCube*. Das Konzept des InteractionCubes wurde von *Tilt Brush* [Goo16] inspiriert. Mit Hilfe des rechten Controller kann der Benutzer die unterschiedlichen Aktionen anwählen. Hierzu gehört das Wechseln zwischen den unterschiedlichen Luftbildaufnahmen des Gebiets.

Die Aufnahmen werden von einem lokalen GeoServer² mit Hilfe von UnitySlippyMap³ heruntergeladen. UnitySlippyMap zeigt diese dann – außerhalb des dem Nutzer sichtbaren Bereichs – auf einer Fläche an. Diese Fläche wird von einer zusätzlichen Kamera betrachtet. Das Blickfeld der Kamera wird auf das *MapDisplay* mit Hilfe von einer *RenderTarget*⁴ projiziert.

Wählt der Benutzer bei aktiviertem 3D-Modus ein Gebäude an, für das es ein 3D-Modell gibt, so wird dieses im Raum geladen. Neben dem MapDisplay befindet sich ein weiteres Display mit Anweisungen. Diese können mit Hilfe der *Weiter* und *Zurück* Knöpfe durchblättert werden.

Voraussetzungen

In diesem letzten Abschnitt des Versuchsaufbaus wird zusammengefasst, welche Komponenten für die Durchführung der Studie benötigt werden.

Folgende Hardware wird für den Versuchsaufbau benötigt:

- HTC Vive (HMD, Controller, 2 Lighthouses, ...)
- Computer, welcher die Mindestanforderungen erreicht

Folgende Software wird für den Versuchsaufbau genutzt:

- VIEW
- Die im Abschnitt *Inhalte* beschriebenen Luftbildaufnahmen
- Lokaler, konfigurierter GeoServer
- OBS Studio zur Bildschirmaufnahme
- Online IPD-Tool zur Einschätzung des Pupillen-Abstands

Sonstiges:

- Fragebögen und leere Blätter für Notizen (Person, SUS, IPQ, RawTLX)
- Beliebige Karte (für IPD-Tool)

1 Unity3D: <https://unity3d.com/>

2 GeoServer: <http://geoserver.org/>

3 UnitySlippyMap: <https://github.com/jderrough/UnitySlippyMap>

4 Unity RenderTexture: <https://docs.unity3d.com/Manual/class-RenderTarget.html>

7.2 Versuchsdurchführung

7.2.1 Probanden

Die Studie hat über einen Zeitraum von zwei Wochen stattgefunden. Es haben 16 Probanden teilgenommen (14 männlich, 2 weiblich). Der jüngste Teilnehmer hatte 21 Jahre, der älteste 59. Das Durchschnittsalter betrug 35,4 Jahre. 13 Probanden haben als ihr Fachgebiet Informatik angegeben. Die anderen drei haben jeweils Physik, Medienproduktion und Mathematik angegeben.

Die Probanden sollten ihre Erfahrung mit Computer- und Konsolenspielen auf einer Skala von 0 (keine Erfahrung) bis 5 (sehr erfahren) einschätzen. Hierbei war der Mittelwert 3,75. Auf einer weiteren Skala sollten sie ihre Erfahrung mit Virtual Reality angeben – der Mittelwert dabei war 2,56.

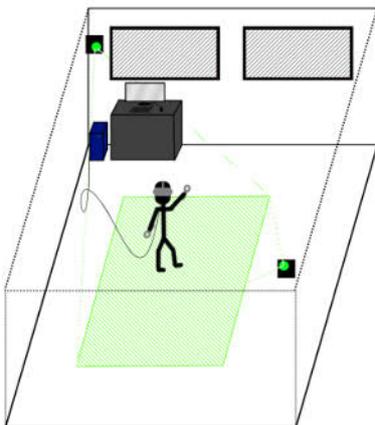
Die meisten der Angaben, sie hätten Erfahrung mit VR, haben HTC Vive als das genutzte Gerät angegeben (12 von 14). Weitere Angaben waren einmal CAVE, dreimal Oculus Rift und zweimal GearVR/Google Cardboard.

7.2.2 Studienumgebung

Die Studie wurde im Innolab-Raum am Fraunhofer IOSB durchgeführt. Eine Skizze des Raums ist in Abbildung 7.7a zu sehen. Grün markiert ist der in VR nutzbare Bereich. Dieser ist einerseits durch das Sichtfeld der Lighthouses und andererseits durch Bürotische und Stühle im Raum eingeschränkt.

Bevor der Proband in den Raum gebeten wird, werden die Fragebögen bereit gelegt. Auf dem Computer wird die Aufnahme-Software vorbereitet und der GeoServer wird gestartet. Anschließend wird der Proband in den Raum gebeten und begrüßt. Die Versuchsperson erhält dann eine kurze Zusammenfassung zum Ablauf der Studie und wird gebeten den Fragebogen zur Person (siehe Anhang 1) auszufüllen. Während dieser ausgefüllt wird, wird VIEW geladen. Das HTC Vive HMD und die Controller liegen zu diesem Zeitpunkt bereit auf dem Boden. Das IPD-Tool wird benutzt, um den Pupillen-Abstand des Probanden zu messen und das HMD wird entsprechend eingestellt.

Sobald der Proband bereit ist, wird er gebeten die HTC Vive aufzusetzen, die Controller in die Hände zu nehmen und die Kopfhörer in seine Ohren zu stecken. Während dessen wird die Bildschirm-Aufnahme gestartet. Es folgt das zuvor beschriebene Szenario. Im Anschluss wird der Proband gebeten, die restlichen Fragebögen auszufüllen (IPQ, RawTLX und SUS). Zum Schluss wird noch ein freies Interview durchgeführt.



(a) Skizze des Innolab-Raums. Der spielbare Bereich ist grün markiert



(b) Foto der Studienumgebung. *Lighthouse* rot markiert. Auf dem großen Monitor ist die Aufnahme-Software zu sehen

Abbildung 7.7: Übersicht zur Studienumgebung

7.2.3 Berechnung

Zur Berechnung der Statistiken und Generierung der Diagramme wurde *Microsoft Excel*¹ und *PSPP*² genutzt. Am Ende der Studie wurden alle Werte in PSPP eingefügt. Microsoft Excel wurde zum Zusammensammeln der Werte genutzt und später zum Umwandeln der Tabellen in das LaTeX-Format mit Hilfe von *Excel2Latex*³.

Bei PSPP handelt es sich um ein um ein Programm zur Durchführung von statistischen Analysen. Es wurde erstellt als eine freie Alternative zu SPSS⁴. Zur Setzung der benötigten Hilfewerte in PSPP wurden folgende Variablen gesetzt:

RawTLX

$$\text{RawTLX} = (5 * TLXq1 + 5 * TLXq2 + 5 * TLXq3 + 5 * TLXq4 + 5 * TLXq5 + 5 * TLXq6) / 6$$

SUS

$$\text{SUS} = ((SUSq1 - 1) + (5 - SUSq2) + (SUSq3 - 1) + (5 - SUSq4) + (SUSq5 - 1) + (5 - SUSq6) + (SUSq7 - 1) + (5 - SUSq8) + (SUSq9 - 1) + (5 - SUSq10)) * 2.5$$

IPQ

$$\text{real1u} = (-1) * REAL1 + 6$$

$$\text{sp2u} = (-1) * SP2 + 6$$

$$\text{inv3u} = (-1) * INV3 + 6$$

$$G = G1 + 4$$

$$SP = \text{MEAN}(SP1, \text{sp2u}, SP3, SP4, SP5)$$

$$INV = \text{MEAN}(INV1, INV2, \text{inv3u}, INV4)$$

$$REAL = \text{MEAN}(\text{real1u}, REAL2, REAL3, REAL4)$$

Interaktion

$$\text{INT} = \text{MEAN}(\text{INT_Start}, \text{INT_Teleport}, \text{INT_Grab}, \text{INT_Button}, \text{INT_Pan}, \text{INT_Zoom}, \text{INT_Layer}, \text{INT_Activ_3D}, \text{INT_Use_3D})$$

Im Anschluss hierzu können, mit Hilfe von PSPP, eine Reihe von komplexen statistischen Operationen automatisch durchgeführt werden.

1 Microsoft Excel: <https://products.office.com/en/excel>

2 GNU PSPP: <https://www.gnu.org/software/pspp/>

3 Excel2Latex: <https://www.ctan.org/tex-archive/support/excel2latex/?lang=en>

4 IBM SPSS: <https://www.ibm.com/analytics/us/en/technology/spss/>

7.3 Bewertung

In diesem Teil wird eine Zusammenfassung der gesammelten Werte dargelegt. Hierzu gehört die Berechnung der Mittelwerte, Varianz, Skewness, Kurtosis, ... Zudem wird für jeden Datensatz mit Hilfe vom D'Agostino-Pearson omnibus Test untersucht, ob er nahezu normal verteilt ist.

7.3.1 Bewertung der Normalverteilung

Zur Untersuchung der Normalverteilung wird der D'Agostino-Pearson omnibus Test durchgeführt. In diesem Kontext werden folgende Hypothesen aufgestellt:

H0 Die Daten sind nicht normal verteilt

H1 Die Daten sind normal verteilt

Bei der Durchführung dieses Tests wird der Signifikanzwert p gesucht. Falls dieser höher als der Signifikanzniveau ($\alpha = 0.05$) ist, kann die Null-Hypothesen verworfen werden und die Hypothese der Normalverteilung akzeptiert werden. Folgende Berechnung wurde hierzu in Excel durchgeführt:

$P_{Value} = ChiDist(DP, 2)$, wobei:

$$DP = z_{g1}^2 + z_{g2}^2$$

$$z_{g1} = Skewness / Error_{Skewness}$$

$$z_{g2} = Kurtosis / Error_{Kurtosis}$$

Die Resultate zum D'Agostino-Pearson omnibus Test sind in Tabelle 7.1 zusammengefasst. Skewness, Kurtosis, Signifikanzwert und Signifikanzniveau wurden jeweils mit γ_1 , κ , p und α abgekürzt, um die Breite der Tabelle zu reduzieren.

$n = 16$	γ_1	γ_1 Err	κ	κ Err	z_{g1}	z_{g2}	DP	p	$p > \alpha$ (0.05)
SUS	1.037	0.564	0.178	1.058	1.837	0.168	3.404	0.1824	TRUE
RawTLX	0.200	0.564	-1.283	1.058	0.355	-1.213	1.596	0.4501	TRUE
G	-1.654	0.564	2.539	1.058	-2.931	2.399	14.347	0.0008	FALSE
SP	-1.165	0.564	1.048	1.058	-2.065	0.991	5.247	0.0725	TRUE
INV	0.736	0.564	0.908	1.058	1.304	0.858	2.438	0.2956	TRUE
REAL	0.337	0.564	-1.397	1.058	0.597	-1.320	2.099	0.3500	TRUE
Int_Mean	0.088	0.564	-1.278	1.058	0.156	-1.208	1.483	0.4765	TRUE

Tabelle 7.1: D'Agostino-Pearson omnibus test auf Datensätzen

Für den Faktor G gilt $p < \alpha$, daher kann dessen Null-Hypothese nicht verworfen werden. Für die Faktoren SUS, RawTLX, SP, INV, REAL und Int_Mean kann die Null-Hypothese verworfen werden und die Hypothese der Normalverteilung *H1* akzeptiert werden.

7.3.2 Bewertung der Arbeitsbelastung

In Tabelle 7.2 werden die von PSPP berechneten statistischen Werte zusammengefasst. In Abbildung 7.9 werden sie visuell dargestellt. In Tabelle 7.3 befinden sich die berechneten Kategorien des RawTLX und die jeweiligen statistischen Berechnungen. Aus Gründen der Übersichtlichkeit wurden die Mittelwerte mit Hilfe von der konditionellen Formatierung in Excel gefärbt. Rot steht hierbei für die höchst-mögliche 100 und grün für die niedrigst-mögliche Beanspruchung 0. In Abbildung 7.8 wird der Box-Plot dargestellt.

RawTLXScore		
N	<i>Valid</i>	16
	<i>Missing</i>	0
Mean		32.14
S.E. Mean		3.36
Std Dev		13.44
Kurtosis		-1.28
Skewness		.20
Range		42.50
Minimum		13.33
Maximum		55.83
Percentiles	50 (Median)	30.42

Tabelle 7.2: Übersicht zu den RawTLX Werten

	Gesamtbeanspruchung	Geistige Anforderung	Körperliche Anforderung	Zeitliche Anforderung	Leistung	Anstrengung	Frustration
p1	15.0	20	15	15	15	15	10
p2	55.8	70	70	20	30	80	65
p3	45.8	75	65	5	15	60	55
p4	30.0	80	10	10	55	15	10
p5	33.3	35	5	65	25	35	35
p6	45.0	65	75	25	60	30	15
p7	23.3	30	30	15	25	35	5
p8	17.5	25	10	40	5	20	5
p9	19.2	25	15	10	15	35	15
p10	13.3	30	10	5	15	15	5
p11	28.3	25	45	5	25	50	20
p12	30.8	35	40	5	40	30	35
p13	20.8	25	35	15	10	20	20
p15	41.7	65	10	5	75	70	25
p16	46.7	80	20	20	75	65	20
p17	47.5	70	20	25	35	60	75

Tabelle 7.3: Zusammenfassung zu den RawTLX Werten der einzelnen Probanden

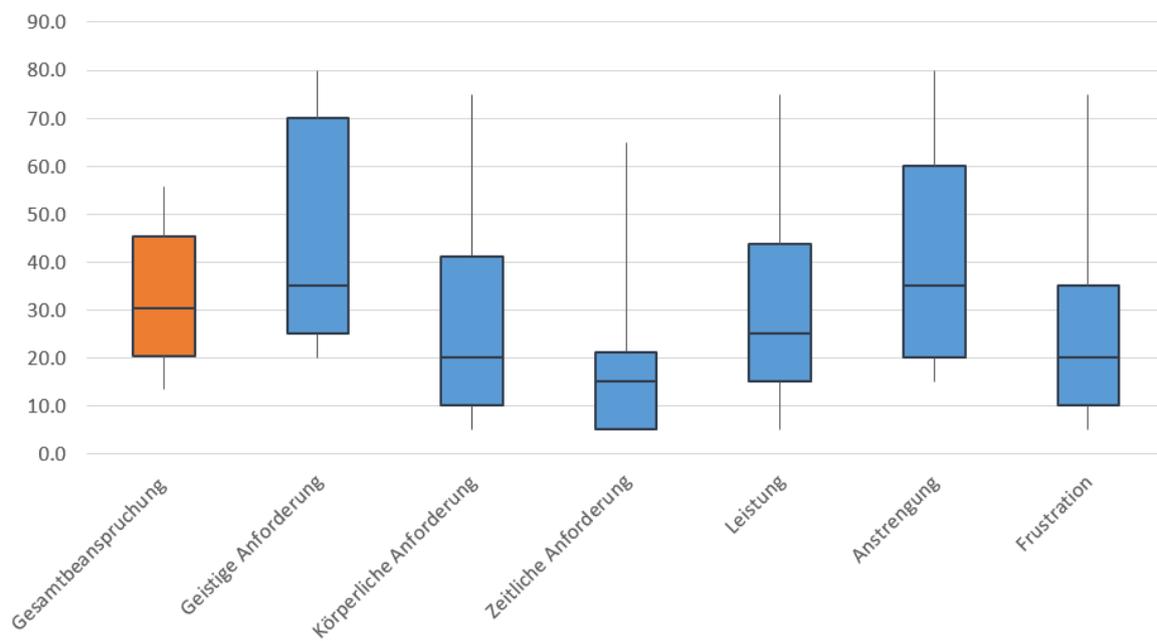


Abbildung 7.8: Box Plot zu den RawTLX Werten

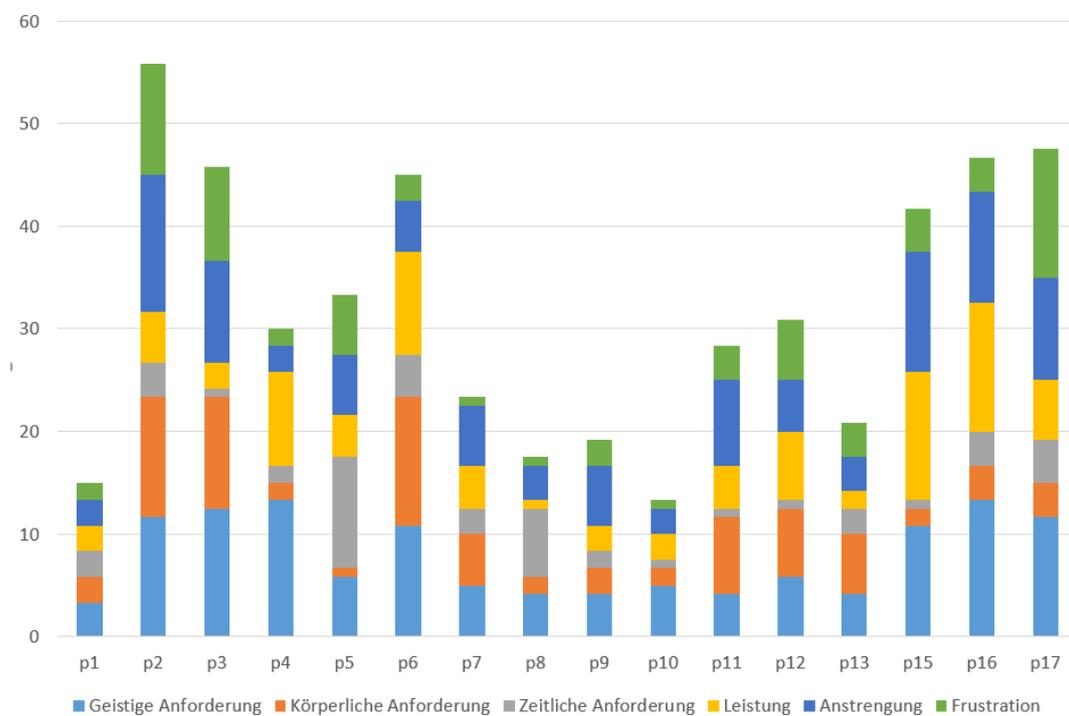


Abbildung 7.9: Visualisierung der Zusammensetzung der RawTLX Werte

7.3.3 Bewertung der Presence

Eine Zusammenfassung zu den berechneten Presence Faktoren befindet sich in Tabelle 7.9. Diese Werte werden mit den von *I*Group gelieferten Beispielen [Igr15] verglichen. Die grafische Darstellung – das sogenannte *Presence-Profil* ist in Abb 7.3.3 zu sehen. Die genauen Werte befinden sich in Tabelle 7.4. In den weiteren Tabellen werden die von PSPP berechneten statistischen Werte zu **G** 7.5, **SP** 7.6, **INV** 7.7 und **REAL** 7.8 zusammengefasst.

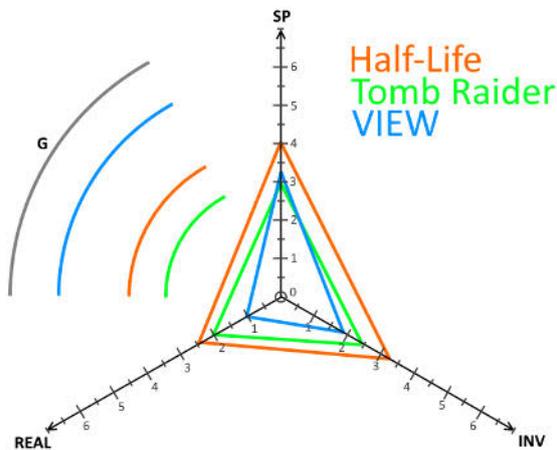


Abbildung 7.10: Presence Profil

IPQ Werte	Spiel	Mittelwert
G	Tomb Raider	3
	Half Life	3.93
	VIEW	5.75
SP	Tomb Raider	3.06
	Half Life	3.99
	VIEW	3.20
INV	Tomb Raider	2.4
	Half Life	3.27
	VIEW	1.81
REAL	Tomb Raider	1.92
	Half Life	2.34
	VIEW	1.06

Tabelle 7.4: Mittelwerte der verglichenen Spiele

G	
Mean	5.75
S.E. Mean	.36
Std Dev	1.44
Kurtosis	2.56
Skewness	-1.65
Range	5
Minimum	2.00
Maximum	7.00
Percentiles 50 (Median)	6.00

Tabelle 7.5: Übersicht zum IPQ G-Wert

SP	
Mean	3.20
S.E. Mean	.24
Std Dev	.96
Kurtosis	1.05
Skewness	-1.17
Range	3.40
Minimum	.80
Maximum	4.20
Percentiles 50 (Median)	3.40

Tabelle 7.6: Übersicht zum IPQ SP-Wert

INV	
Mean	1.81
S.E. Mean	.27
Std Dev	1.09
Kurtosis	.91
Skewness	.74
Range	4.25
Minimum	.25
Maximum	4.50
Percentiles 50 (Median)	1.75

Tabelle 7.7: Übersicht zum IPQ INV-Wert

REAL	
Mean	1.06
S.E. Mean	.28
Std Dev	1.13
Kurtosis	-1.40
Skewness	.34
Range	3.50
Minimum	-.50
Maximum	3.00
Percentiles 50 (Median)	.50

Tabelle 7.8: Übersicht zum IPQ REAL-Wert

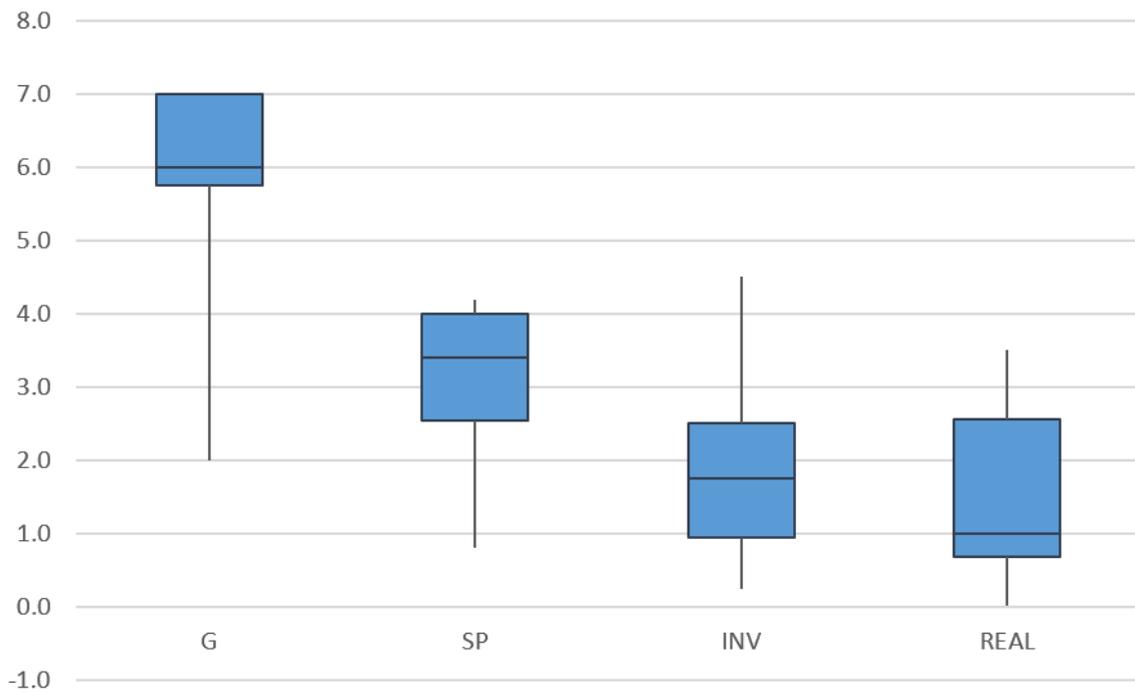


Abbildung 7.11: Box Plot zu den IPQ Werten

	G	SP	INV	REAL
p1	2	0.8	1	0.25
p2	6	3.4	2.5	0.5
p3	6	2.2	0.5	0
p4	7	4.2	2.75	2.5
p5	5	3.2	2.5	-0.25
p6	6	2.6	1.25	0
p7	7	3.4	2	0.5
p8	3	2	0.25	-0.5
p9	5	3.8	0.75	2.5
p10	6	3.4	1.5	1.75
p11	7	4	1.5	3
p12	7	3.6	2.25	0.5
p13	6	4	4.5	1.5
p15	6	2.4	0.75	0.5
p16	7	4.2	2.75	2.25
p17	6	4	2.25	2

Tabelle 7.9: Zusammenfassung zu den IPQ Werten der einzelnen Probanden

7.3.4 Bewertung der Usability

In Tabelle 7.10 werden die von PSPP berechneten statistischen Werte zusammengefasst. In Abbildung 7.11 ist die Box Plot Darstellung zu sehen.

In Tabelle 7.12 befinden sich die Rohdaten und jeweiligen statistischen Berechnungen. Aus Gründen der Übersichtlichkeit wurden die Mittelwerte mit Hilfe von der konditionellen Formatierung in Excel gefärbt. Rot steht hierbei für niedrigst-mögliche 0 und grün für die höchstmögliche Gebrauchstauglichkeit 100.

SUS		
N	Valid	16
	Missing	0
Mean		79.38
S.E. Mean		2.09
Std Dev		8.34
Kurtosis		.18
Skewness		1.04
Range		27.50
Minimum		70.00
Maximum		97.50
Percentiles	50 (Median)	76.25



Tabelle 7.10: Übersicht zum SUS-Wert

Tabelle 7.11: Box Plot zu den SUS Werten

	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	SUS Score
p1	4	2	4	3	4	1	4	1	5	1	82.5
p2	3	1	4	4	5	1	4	2	5	1	80.0
p3	4	2	2	2	4	1	4	3	4	2	70.0
p4	5	1	3	1	3	1	5	2	3	4	75.0
p5	4	2	4	2	4	2	4	2	4	4	70.0
p6	5	2	4	2	5	2	4	1	5	4	80.0
p7	5	1	4	1	4	1	5	2	5	1	92.5
p8	1	2	5	1	4	1	4	2	3	1	75.0
p9	4	1	2	3	4	1	4	1	4	2	75.0
p10	4	1	5	2	5	2	5	1	5	1	92.5
p11	3	1	5	2	5	1	5	1	2	2	82.5
p12	4	2	3	2	4	2	4	2	4	2	72.5
p13	5	1	5	1	5	1	5	1	4	1	97.5
p15	4	1	4	4	5	2	4	2	3	1	75.0
p16	5	2	4	1	4	1	4	4	3	3	72.5
p17	4	1	4	3	4	1	4	1	3	2	77.5

Tabelle 7.12: Zusammenfassung zu den SUS Werten der einzelnen Probanden

7.3.5 Bewertung der Interaktion

Im Versuchsaufbau wurde die Bewertung der Interaktion bereits geschildert. Zur besseren Übersichtlichkeit wird hier noch einmal die Zusammenfassung geschildert:

-1 Der Proband konnte die Aktion nicht durchführen und benötigte Anweisungen

0 Der Proband konnte die Aktion selbstständig durchführen, hatte jedoch Schwierigkeiten (z. B. Öfters falschen Knopf betätigt)

+1 Der Proband konnte die Aktion selbstständig und problemlos durchführen

In Tabelle 7.13 befinden sich die Rohdaten und in Tabelle 7.14 jeweiligen statistischen Berechnungen. Aus Gründen der Übersichtlichkeit wurden die Werte mit Hilfe von der konditionellen Formatierung in Excel gefärbt. Rot steht hierbei für das Minimum -1 und grün für das Maximum +1. In Tabelle werden die statistischen Werte zusammen gefasst und in Abbildung 7.12 werden die Werte visuell zusammengefasst.

	Startknopf betätigen	Teleportieren	Greifen	Weiter/Zurück betätigen	Karte ziehen	Karte zoomen	Layer wechseln	3D-Tool aktivieren	3D-Tool nutzen	Mean
p1	0	1	-1	1	1	1	1	1	1	0.67
p2	1	1	1	1	1	1	1	1	1	1.00
p3	1	1	-1	1	1	1	1	1	1	0.78
p4	1	0	-1	1	1	1	0	1	1	0.56
p5	1	1	-1	0	1	0	1	1	1	0.56
p6	1	1	-1	1	1	0	1	1	1	0.67
p7	1	1	0	1	1	1	0	1	1	0.78
p8	0	1	0	1	1	1	1	1	1	0.78
p9	1	1	0	1	1	1	1	1	1	0.89
p10	1	1	1	1	1	1	1	1	1	1.00
p11	1	1	1	1	1	1	1	1	1	1.00
p12	1	1	-1	0	1	-1	1	1	1	0.44
p13	1	1	1	1	1	1	1	1	1	1.00
p15	1	0	-1	1	1	-1	1	1	1	0.44
p16	0	1	0	1	1	0	1	1	1	0.67
p17	1	1	0	1	1	0	-1	1	1	0.56
MEAN	0.81	0.88	-0.19	0.88	1.00	0.50	0.75	1.00	1.00	0.74

Tabelle 7.13: Rohdaten und Mittelwerte zu den beobachteten Interaktionen der einzelnen Probanden

Variable	N	Mean	Std Dev	Variance
INT_Start	16	.81	.40	.16
INT_Teleport	16	.88	.34	.12
INT_Grab	16	-.19	.83	.70
INT_Button	16	.88	.34	.12
INT_Pan	16	1.00	.00	.00
INT_Zoom	16	.50	.73	.53
INT_Layer	16	.75	.58	.33
INT_Activ_3D	16	1.00	.00	.00
INT_Use_3D	16	1.00	.00	.00

Tabelle 7.14: Zusammenfassung zu den Beobachteten Interaktionen

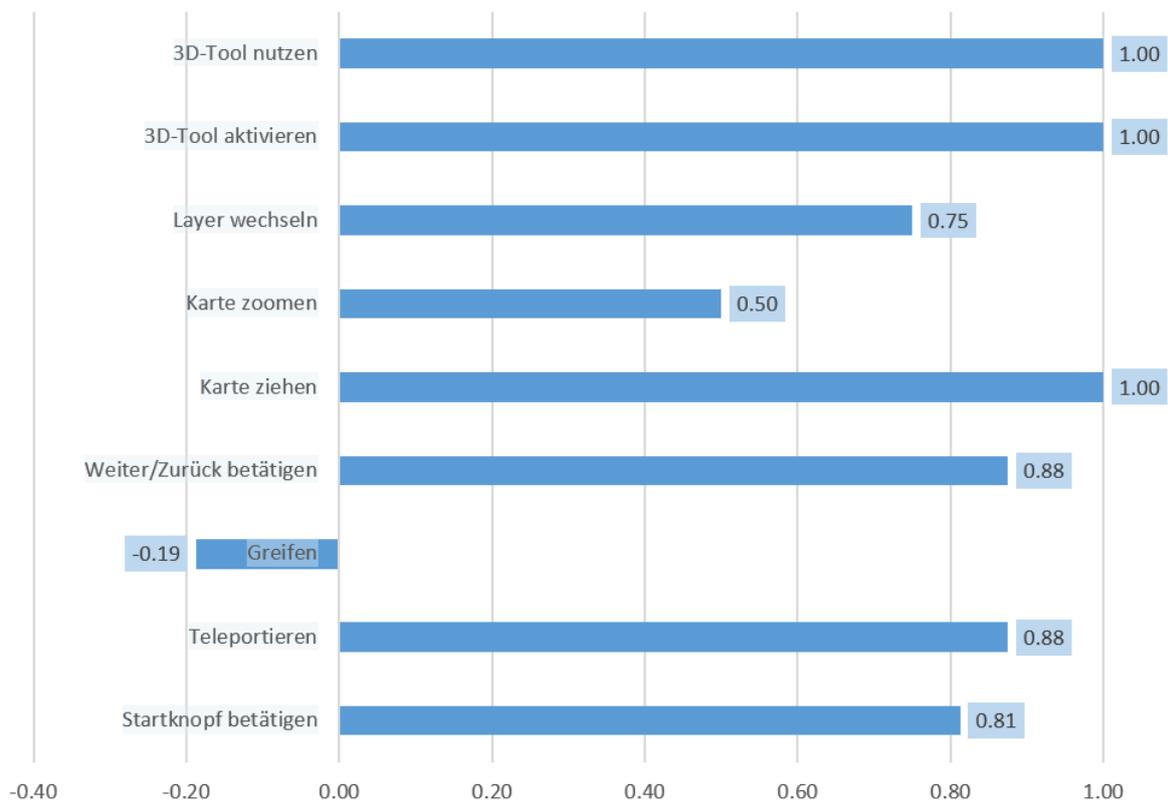


Abbildung 7.12: Visuelle Übersicht zu den Interaktionswerten

7.3.6 Bewertung der Korrelationen

Zur Untersuchung der Korrelationen wurde eine *Bivariate Correlations Analysis* mit PSPP durchgeführt. Da die resultierende Tabelle äußerst groß ist (21 x 44 Zellen), habe ich eine Zusammenfassung der Tabelle erstellt, welche nur Faktoren beinhaltet, die signifikante Korrelationen mit anderen Faktoren aufzeigen (absolut Wert des Pearson Correlation Wertes > 0.50). Die signifikanten Werte wurden in Tabelle 7.15 farblich markiert.

	N= 16	SUS	SP	REAL	Alter	Spiele Erf.	VR Erf.	INT Tele.	INT Grab	INT Zoom
SUS	Pearson Correl. Sig. (2-tailed)	1								
SP	Pearson Correl. Sig. (2-tailed)	0.12 0.655	1							
REAL	Pearson Correl. Sig. (2-tailed)	0.18 0.513	0.73 0.001	1						
Alter	Pearson Correl. Sig. (2-tailed)	-0.26 0.339	0.56 0.025	0.71 0.002	1					
Spiele Erfahr.	Pearson Correl. Sig. (2-tailed)	0.19 0.491	-0.08 0.781	-0.4 0.125	-0.5 0.051	1				
VR Erfahr.	Pearson Correl. Sig. (2-tailed)	0.56 0.024	0.28 0.298	0.18 0.515	-0.25 0.355	0.28 0.294	1			
INT Tele.	Pearson Correl. Sig. (2-tailed)	0.2 0.447	-0.04 0.881	-0.15 0.577	-0.17 0.537	0.68 0.004	0.19 0.48	1		
INT Grab	Pearson Correl. Sig. (2-tailed)	0.63 0.009	0.45 0.08	0.45 0.077	0.18 0.513	0.14 0.607	0.67 0.004	0.38 0.146	1	
INT Zoom	Pearson Correl. Sig. (2-tailed)	0.44 0.09	-0.04 0.889	0.2 0.454	0.03 0.919	0.28 0.288	0.36 0.176	0.27 0.317	0.49 0.053	1
TLXq4	Pearson Correl. Sig. (2-tailed)	-0.36 0.177	0.22 0.407	0.15 0.578	0.26 0.338	-0.54 0.031	-0.46 0.076	-0.56 0.023	-0.38 0.145	-0.54 0.031

Tabelle 7.15: Zusammenfassung der Pearson Correlation Werte (signifikante Werte farblich markiert)

Weiterhin habe ich mit Hilfe von PSPP für jedes der markierten Faktoren-Paare, eine lineare Regressionsanalyse durchgeführt, um den entsprechenden Sigma-Wert zu berechnen. Die komplette Ausgabe befindet sich im Anhang (siehe Anhang 1). Die Zusammenfassung der Student-T und Sigma-Werte befinden sich in Tabelle 7.16. Für die Bestimmung der t-Werte wird als ν in diesem Fall $n - 1 = 15$ gewählt. Hierbei werden jeweils folgende Hypothesen gestellt:

H_0 Es existiert keine signifikante lineare Korrelationen zwischen den Werten

H_1 Es existiert eine signifikante lineare Korrelationen zwischen den Werten

Da das Sigma für die jeweiligen Paare statistisch signifikant niedrig ist, können die Null-Hypothese verworfen werden und die Hypothese der signifikanten Korrelation akzeptiert werden.

Unabhängige Variable	Abhängige Variable	t	Sig.	Adjusted R square
Alter	SP	2.52	0.025	0.26
Alter	REAL	3.75	0.002	0.47
VR Erfahrung	INT Grab	3.39	0.004	0.41
VR Erfahrung	SUS	2.52	0.024	0.26
Spielerfahrung	INT Teleport	3.47	0.004	0.42
INT Grab	SUS	3.03	0.009	0.35
REAL	SP	3.95	0.001	0.49
TLXq4	Spielerfahrung	-2.40	0.031	0.24
TLXq4	INT Teleport	-2.55	0.023	0.27
TLXq4	INT Zoom	-3.37	0.05	0.41

Tabelle 7.16: Zusammenfassung der Sigma Werte zu den potenziell signifikanten Korrelationen

7.3.7 Bewertung der technischen Aspekte

Neben dieser Studie wurden weitere Untersuchungen zu den technischen Aspekten bereits in Kapitel 4.3 und Kapitel 5.1.2 durchgeführt. Die gegebenen Hardwareanforderungen wurden bereits in den jeweiligen Kapiteln behandelt.

Die Anforderungen an VIEW waren wie folgt:

- Dem Benutzer muss die Möglichkeit geboten werden, sich auf Elemente zu fokussieren und Text zu lesen.
- Die App muss das Head-Tracking beibehalten, es darf nicht unterbrochen werden. Zu dem dürfen keine virtuellen Objekte an den Kopf des Benutzers angebracht werden, da dies die Immersion stört
- Die Horizont-Linie muss stabil bleiben. Der Boden darf nicht geneigt werden und die Kamera darf nicht geschüttelt werden
- Der Benutzer muss stets die Controller über die Kamerabewegungen behalten.
- Der Controller muss wie ein Laser Pointer bei der Interaktion mit UI Elementen genutzt werden. Zudem sollte der Controller in VR sichtbar sein und das Anzeigen eines Laserstrahls ist ratsam.
- Der Cursor muss die gleiche Tiefe annehmen wie die Objekte, auf die gezielt werden

Diese Anforderungen waren bereits im Vorfeld erfüllt, da sie bei der Entwicklung des Prototyps beachtet wurden.

Eine weitere Anforderung resultiert aus den Anforderungen an die Hardware. Sowohl die Bildschirmwiederholrate (Hz) der Displays als auch die Bildwiederholrate der Anwendung (FPS) sollen nicht unter 90 fallen.

Dies ist im Falle von VIEW unproblematisch. Die sichtbare Szene besteht lediglich aus einem Raum und die genutzte Hardware liegt weit über den Hardware-Empfehlungen für die Nutzung des HTC Vives. Aus diesem Grund werden oft weit über 100 FPS erzielt.

Zu Bemerkungen ist jedoch, dass das sehr schnelle Hinaus- und Hineinzoomen der Karte ein Fehler hervorrufen kann, der zum Ruckeln des Bildes führen kann. Dieser Fehler ist bei zwei Probanden aufgetreten. Die HTC Vive zeigt dem Benutzer an, wenn es zu unerwarteten Framedrops während der normalen Nutzung kommt. Dies ist bei keinem der Probanden aufgetreten.

7.4 Diskussion

In diesem Abschnitt werden die Resultate der Bewertung und die dazugehörigen qualitativen Aussagen der Probanden näher diskutiert.

7.4.1 Interaktion

Bei der Bewertung der Interaktion ist besonders auffällig, wie niedrig der Wert für das Greifen von Objekten ausgefallen ist. Dies deckt sich auch mit den Aussagen, die die Probanden während dem freien Interview gegeben haben. Mehrere Probanden haben angegeben, dass der Grip-Knopf schwierig zu erreichen ist:

- »Schwierig ist das Greifen, weil der Knopf auf dem Controller merkwürdig platziert ist« - P03
- »Ich finde den Seitenknopf komisch« - P06
- »Die Bedienung war am Anfang ungewohnt, ich musste oft hinschauen, um zu sehen, was ich drücke, aber danach ging es gut. Ich stelle mir das vor wie mit einer neuen Fernbedienung, da sieht man am Anfang immer hin, wenn man auf die 5 drücken will – danach klappt das auch, ohne hinzuschauen « - P09

Des Weiteren empfanden einige Probanden das Greifen an sich nicht ideal umgesetzt:

- »Das Greifen ist nicht so gelungen, man greift die Koordinaten der Objektmitte, ich hätte lieber, man könnte auch die Kanten greifen« - P10
- »Interaktionen gingen ganz gut - Frickelig war das Greifen« - P12
- »Greifen war nicht ganz intuitiv« - P17

Zudem haben die Probanden versucht die Objekte auf Distanz zu greifen und den Startknopf auf Distanz zu betätigen, was jedoch nicht möglich ist.

In der Bewertung hat sich gezeigt, dass es eine Korrelation zwischen den Werten *Greifen* und *VR Erfahrung* gibt. Es liegt nahe, dass Benutzer, die schon mehrmals VR ausprobiert haben, besser mit dieser Art der Interaktion klarkommen.

Die restlichen Interaktionen wurden als intuitiv und angenehm beschrieben. Besonders viel Lob erhielt das Konzept des InteractionCubes. Beliebt war auch die Beschriftung auf den Controllern. Vier Probanden haben versucht den Startknopf spielerisch mit Hilfe von Objekten zu aktivieren. Sie haben hierbei Freude gezeigt, als mit Hilfe dieser Aktion das Hauptlevel geladen wurde.

- »InteractionCube ist eine schöne Idee, viel besser wie Management von Untermenüs anhand von Tabs« - P08
- »InteractionCube ist eine sehr gute Idee, sehr schlau« - P12
- »[VIEW] ist einfach zu bedienen, das war mir alles vertraut« - P10
- »Teleportation hat mir sehr gut gefallen« - P11
- »Cool fand ich die Beschriftungen auf dem Controller – besonders, dass sie auch auf der Rückseite nicht spiegelverkehrt waren« - P13
- »Interaktionen mit Karte super! Skalierung auch intuitiv wie erwartet. Es war super, dass die Karte genau am Strahl hängen bleibt« - P13

Sechs Probanden kamen nicht sofort mit dem Zoomen der Karte klar. Sie haben unterschiedliche Arten von Interaktionen ausprobiert, bevor sie die richtige gefunden hatten. Sie haben dabei versucht, die Trigger zu betätigen und dann den Controller Richtung Körper zu sich zu ziehen. Zwei dieser sechs Probanden haben die erwartete Geste nicht selbstständig herausgefunden und ihnen musste das Konzept des Zoomens erklärt werden. Dies ist bedingt durch den Versuchsaufbau. Den Benutzern wurden die unterschiedlichen Gesten nicht im Voraus erklärt, sie sollten diese selbst entdecken.

Mit der Teleportation hatten zwei Probanden Schwierigkeiten. Sie haben das Konzept jedoch ohne zusätzliche Erklärung selbst herausgefunden. Bei der Bewertung hat sich herausgestellt, dass es eine Korrelation zwischen den Werten *Teleportieren* und *Spiel Erfahrung gibt*. Da jedoch nur zwei Probanden Schwierigkeiten mit dem Teleportieren hatten, lässt sich keine naheliegende Erklärung hierfür aufzeigen. Gäbe es mehr Probleme mit dieser Interaktion, könnte die Hypothese aufgestellt werden, dass Benutzer, die mehr Erfahrung mit Computer- bzw. Konsolenspielen haben, besser mit dem Teleportieren klarkommen.

Die Aktivierung und Nutzung des 3D-Modus verlief ohne Probleme. Schwierigkeiten hatten die Probanden jedoch erneut mit dem Greifen des 3D-Modells. Grund hierfür war einerseits, die zuvor beschriebenen Probleme des Greifens; andererseits, weil die Probanden nicht erwartet hatten, dass der Tempel sich nur horizontal drehen lässt. Es würde sich anbieten in zukünftigen Versionen, mit Hilfe einer Anzeige der Freiheitsgrade anzugeben, um welche Achsen sich das Objekt drehen lässt.

7.4.2 SUS

Der Prototyp wurde von den Probanden als nützlich beschrieben. Dies spiegelt sich auch bei den SUS Werten. Viele gaben an, dass sie das System als sinnvoll empfinden.

- »Sinnvolle Umsetzung von VR. Es gibt zurzeit viel Mumpitz, das hier ist gut« - P3
- »Spielerische Elemente haben Spaß gemacht und motiviert weiter mit der Aufgabe zu fahren« - P13
- »Funktionen alle nützlich, es hat mir nichts gefehlt« - P15

Einige Probanden haben angegeben, die Schrift der Instruktionen auf dem Monitor wäre schwer lesbar. Sie haben sich jedoch wie erwartet im Anschluss dem Monitor genähert.

- »Schrift war schlecht zu lesen« - P15

Bei der Untersuchung der Korrelationen ist aufgefallen, dass es eine signifikante Korrelation zwischen dem System Usability Score und den Werten Greifen und VR Erfahrung gibt. Naheliegend ist im ersten Fall, dass Probanden die besser mit dem Greifen klarkamen, das System als gebrauchstauglicher empfanden. Des Weiteren ist es möglich, dass Probanden, die mehr VR Erfahrung haben, sich auch in VR wohler fühlen und daher leichter bzw. natürlicher mit den Interaktionskonzepten des Prototyps umgehen.

7.4.3 Presence

Die virtuelle Umgebung – das Lost Earth Raumschiff – ist besonders gut bei den Probanden angekommen. Viele haben während der Nutzung oder im späteren freien Interview ihre Begeisterung dazu geäußert:

- »Die Audio-Clips waren sehr gut. Staubpartikel und Effekte waren cool. Die Texturen waren hübsch, hat zur Atmosphäre beigetragen« - P10

- »Super fand ich die neue Ausgestaltung der Brücke« - P11
- »Ich fühlte mich wohl auf der Brücke. Ich wollte am liebsten die ganze Zeit dort herumlaufen« - P12

Mehrere Probanden haben angegeben, dass sie auch ihre Hände in VR sehen wollen. Einerseits, um die Immersion zu erhöhen und andererseits, um die Knöpfe des Controllers besser zu treffen.

- »Ich würde gerne meine eigenen Finger sehen« - P12
- »Meine Hände sind nicht zu sehen, das wäre noch cool – ich weiß nicht ob das geht« - P15

Die Effekte wie die Sonneneinstrahlung, die auf dem Metall spiegelt und der Staub, der dem Raum mehr Fülle geben soll, wurden von den meisten Probanden nicht bewusst bemerkt.

- »Ich habe nicht auf Staubeffekte und so geachtet – sie sind nicht negativ aufgefallen und es hat nicht gewirkt, als ob etwas fehlt« - P07
- »Effekte sind mir nicht bewusst aufgefallen, unbewusst haben sie aber wahrscheinlich Vorteile« - P09
- »Die Effekte habe ich höchsten unbewusst wahrgenommen« - P12
- »Die Effekte habe ich nicht beachtet« - P15

Die Ton-Kulisse wurde größtenteils gelobt. Zwei Probanden gaben an, sie würden lieber ohne Ton arbeiten.

- »Immersion definitiv gesteigert durch Audio. Danach nimmt man die ambienten Geräusche nicht mehr wahr, was gut ist – sie stören nicht« - P09
- »Der Sound erzeugt eine höhere Immersion, aber bei der Arbeit habe ich lieber Ruhe« - P11

Bei der Untersuchung der Korrelation ist aufgefallen, dass es eine signifikante Korrelation zwischen Alter und jeweils SP und REAL gab. Es wäre möglich, dass dies daraus gründet, dass jüngere Probanden vertrauter mit neuen Technologien sind und durch diese *Sättigung* schwieriger zu beeindrucken sind - was zur Folge haben könnte, dass das tatsächliche Empfinden einer physikalischen Immersion schwieriger zu erreichen ist.

Bei der Bewertung ist der G-Wert im Vergleich zu den Spielen *Half-Life* und *Tomb Raider* besonders hoch aufgefallen. Im Vergleich sind jedoch die Werte SP und INV niedrig. Möglicherweise könnte ein höheres Involvement erreicht werden, falls die Probanden mehr Zeit mit VIEW verbracht hätten. Spieler benötigen ungefähr 12 Stunden¹, um *Half-Life* abzuschließen. Den Spielern wird also deutlich mehr Zeit und Story geboten, um in die Spielwelt einzutauchen.

Zwei Probanden haben angegeben, sie wären stark anfällig für Übelkeit bei der Nutzung von Spielen. Der erste Proband gab im Interview an, er hätte keine Übelkeit verspürt – kam jedoch später zurück und gab an, er glaube, dass ihm doch etwas übel sei. Der zweite Proband gab bereits im Interview etwas Schwindelgefühle an. Ein weiterer Proband hat angegeben, er sei nicht sicher, ob ihm schwindelig ist. Alle weiteren Probanden gaben an, sie hätten keine Übelkeit oder sonstiges Unwohlsein verspürt.

¹ Half-Life - HowLongToBeat.com: <https://howlongtobeat.com/game.php?id=4247>

7.4.4 Technische Aspekte

Die empfundene Bildqualität wurde als ausreichend bis sehr gut beschrieben. Einige Probanden gaben jedoch an, dass die Bildqualität an den Rändern nicht ausreichend sei. Des Weiteren haben sich zwei Probanden über die Auflösung der Treppen-Textur beschwert.

- »Bildqualität am Rand war ungenügend. Ich wusste nicht, dass ich nur den Kopf bewegen soll – ich hätte gerne auch meine Augen bewegt« - P08

Wie bereits zuvor beschrieben, kam es in zwei Fällen zu einem Fehlverhalten von VIEW gegen Ende der jeweiligen Nutzungen, welches eine inkonsistente Bildwiederholrate verursacht hat. In beiden Fälle wurden die Probanden gebeten kurz zu warten, während die Anwendung neu gestartet wird. In allen anderen Fällen kam es weder zu Framedrops, noch zu einem Fallen der Bildwiederholrate unter 90 FPS.

7.5 Fazit der Evaluation

An dieser Studie haben 16 Probanden teilgenommen. Anhand von mehreren Fragebögen und einem freien Interview wurden die Faktoren *Presence*, *Usability* und *Intuitivität* untersucht. Im Folgenden werden die Erkenntnisse zu den einzelnen Subhypothesen zusammengefasst und ein allgemeines Fazit zu den jeweiligen Hypothesen gegeben.

7.5.1 Presence

Trotz den niedrigen SP und INV Werten ist der allgemeine Wert G im Vergleich zu den beiden Spielen hoch. Mehrere Probanden haben angegeben, dass sie sich komplett in die Aufgabe und die virtuelle Welt vertieft gefühlt haben (P1). Mehrere Probanden haben angegeben, dass das Kabel stört. Die Presence könnte demnach möglicherweise durch die Nutzung des TPCAST kabellos Adapter für die HTC Vive noch weiter erhöht werden.

Der niedrige Involvement-Wert lässt sich zum Teil dadurch erklären, dass die Probanden die Anwendung nicht über längere Zeit genutzt haben. Im Gegensatz zu den Beispielspielen *Half-Life* und *Tomb Raider*, wurde der Prototyp nicht während mehreren Stunden genutzt. Weiterhin könnte durch Anwendung von Serious Games Aspekten das Involvement weiter erhöht werden (P2).

Bei drei von 16 Probanden hat die Nutzung von VIEW Schwindelgefühle ausgelöst. Von diesen drei haben zwei Probanden bereits im Vorfeld angegeben, dass sie bei jeglicher Art von Spielen sehr schnell Übelkeit empfinden. Bei den 13 anderen Probanden ist keine Art von Übelkeit oder Schwindelgefühlen aufgetaucht (P3).

Der Prototyp hatte bei den Versuchen eine konstante Bildwiederholrate von über 90 FPS. In zwei Fällen kam es zu einem Fehlverhalten, welches ein Stottern der Bilder ausgelöst hat. Dieses sollte in zukünftigen Versionen verbessert werden. Alle weiteren Aspekte wie FOV, niedrige Latenz und präzises Tracking wurden bereits im Vorfeld in Kapitel 5.1.2 bestätigt. Fazit für die technischen Anforderungen ist, dass die HTC Vive noch kein perfektes VR-Erlebnis bieten, es jedoch ausreichend für der Erstellung von Immersion ist. VIEW erfüllt im Grunde die Anforderungen an die Software, hat jedoch noch einige Fehler, die für die Gewährleistung einer idealen Anwendung behoben werden sollten (P4).

Die niedrige subjektiv empfundene Arbeitsbelastung konnte mit dem RawTLX-Fragebogen bestätigt werden. Dessen Durchschnittswert ist 32/100 (niedriger ist besser) (P5).

Aus den Resultaten der fünf Subhypothesen kann geschlossen werden, dass VIEW ein weitestgehend immersives Erlebnis bietet, bei welchem die Nutzer sich in die virtuelle Welt vertiefen können.

7.5.2 Usability

Die Probanden haben angegeben, dass es sich bei VIEW, um ein nützliches Werkzeug handelt. Der durchschnittlich erzielte System Usability Score von 79/100 bestätigt diese Aussagen (höher ist besser) (U1). Keiner der Probanden hatte Schwierigkeiten damit, die gestellte Aufgabe zu erledigen (U2). Des Weiteren war die allgemeine Rückmeldung zur Zufriedenheit sehr positiv (U3).

Die allgemeine positive Rückmeldung der Probanden und der hohe erzielte System Usability Score zeigt, dass der Prototyp gebrauchstauglich ist. Einige Probanden haben jedoch angegeben, sie würden sich noch weitere Features wie ein Maßstab und die Möglichkeit der Helligkeits- und Kontrast-Anpassung wünschen. Dies sollte bei zukünftigen Versionen berücksichtigt werden.

7.5.3 Intuitivität

Die Probanden gaben im Interview an, dass sie alle Interaktionen, bis auf das Greifen als intuitiv und sinnvoll empfanden (I1). Der Großteil der Interaktionen konnte von den Probanden problemlos ausgeführt werden, ohne dass sie Anweisungen zu der Funktionsweise von VIEW erhalten haben. Deutliche Probleme gab es jedoch bei dem Konzept des Greifens. Viele Probanden hatten sichtlich Schwierigkeiten beim Umgang mit diesem Konzept. Dies zeigt sich auch bei den qualitativen Aussagen der Probanden (I2).

Es konnte ein hoher Grad an Intuitivität nachgewiesen werden. Das Konzept des InteractionCubes kam besonders gut bei den Probanden an. Die Interaktion mit dem MapDisplay hat zudem auch viel Lob erhalten.

Es lässt sich demnach schließend, dass der Großteil der in VIEW umgesetzten Interaktionen als intuitiv empfunden wird. Es gibt jedoch noch Faktoren, welche in zukünftigen Version verbessert werden sollten. Besonders das Konzept des Greifens muss überarbeitet werden. Es sollte entweder dem Benutzer erlaubt werden, auf Distanz zu greifen oder ihm ganz genau erklärt werden, wie dieses Konzept funktioniert. Des Weiteren bietet es sich an, bei fixierten Objekten wie dem 3D-Modell des Tempels, die Achsen, um die sich das Objekt drehen kann, zu visualisieren.

Es sollte weiterhin versucht werden, die Nutzung des Grip-Buttons zu vermeiden. Eine Reihe von Probanden, hatten Schwierigkeiten ihn zu finden und eine angenehme Fingerpositionierung einzunehmen, um ihn zu betätigen. Es würde sich anbieten zu versuchen, alle Interaktionsmöglichkeiten auf den Trigger zu binden und die Aktionen kontextabhängig auszulösen.

Obwohl das Teleportieren keine größeren Schwierigkeiten ausgelöst hat, würde es sich anbieten, feste Teleportationspunkte auf dem begehbaren Bereich zu markieren, um somit den Umgang potentiell noch einfacher zu gestalten.

Weiterhin sollte dem Benutzer mehr Feedback zu fehlgeschlagenen Aktionen gegeben werden. Wird beispielsweise die maximale Zoomstufe erreicht, so könnte dies durch Vibrationen der Controller signalisiert werden.

Da einige Probanden Schwierigkeiten mit dem Konzept des Zoomens der Karte hatten, sollte die Funktion kurz erklärt, oder alternative Konzepte getestet werden. Anzumerken ist, dass die Konzepte wegen des Aufbaus der Studie nicht im Vorfeld erklärt wurden. Ein kurzes Training oder eine Demonstration könnten die erzielten Werte weiter erhöhen.

Obwohl es noch kleinere Verbesserungsmöglichkeiten gibt, lässt sich schließen, dass das Postulat mit einigen Einschränkungen bestätigt wurde. VIEW bietet dem Bildauswerter eine effektive und *weitestgehend* immersive Arbeitsunterstützung. Dies wird ermöglicht durch die Nutzung von *größtenteils* intuitiven Interaktionskonzepten.

Fazit und Ausblick

In dieser Arbeit wurde ein Assistenzsystem für die Bildauswertung in VR konzipiert, entwickelt und evaluiert. Ein Schwerpunkt dieser Arbeit war die Entwicklung von intuitiven Interaktionskonzepten, welche zusätzlich keine Symptome der VR-Krankheit hervorrufen. Die Herausforderung hierbei war, dass aktuell keine festgelegten, universell genutzten Interaktionskonzepte für VR existieren – es jedoch eine Reihe von Paradigmen zu befolgen gibt [Mal].

Der entwickelte Prototyp mit dem Namen *Virtual Reality Image Exploitation Workspace (VIEW)* wurde in einer Studie mit 16 Teilnehmern evaluiert. Die Studie hat gezeigt, dass der Großteil der entwickelten Konzepte intuitiv ist und eine angenehme Nutzung erlaubt. Probleme gab es jedoch bei der Umsetzung des Konzepts zum Greifen von Objekten. Die Probanden hatten Schwierigkeiten den dazugehörigen Knopf auf dem Controller zu finden und diesen zu benutzen.

Eine weitere Herausforderung war der Umgang mit hochaufgelösten Luftbildaufnahmen. Deren Auflösung ist oft deutlich höher als die Auflösung, die von aktuellen Game Engines und Grafikkarten unterstützt wird. Die Radar-Aufnahme des Assuan-Damms hat beispielsweise eine Auflösung von 46k x 30k Pixel¹ – Unity3D hingegen unterstützt maximal 8k x 8k Pixel².

Zur Lösung dieses Problems wurden sogenannte *Slippymaps* genutzt. Ein lokal-laufender Map-server liefert die aktuell betrachteten Bildausschnitte in der entsprechenden Auflösung an die Anwendung. Diese projiziert die Kartenfläche auf eine gekrümmte Leinwand. Da die gesendeten Bildausschnitte deutlich kleiner sind, ist es mit Hilfe dieser Umsetzung möglich Luftbildaufnahmen mit beliebig-hoher Auflösung zu betrachten.

Weiterhin wurden die Aspekte der Immersion untersucht. In diesem Kontext wurde vor allem die Presence (die vom Nutzer empfundene Immersion) untersucht. Zur Einbindung des Prototyps in die Vision des »Bildauswertepplatz in VR« und mit dem Ziel eine hohe Presence zu ermöglichen, wurde als Umgebung das 3D-Modell des Raumschiffs aus dem Serious Game *Lost Earth 2307* importiert.

In der Studie hat sich gezeigt, dass der Prototyp eine allgemein hohe Empfindung von Presence erlaubt, die Benutzer jedoch nicht ein großes Gefühl von tatsächlicher physikalischer Präsenz in VR empfanden. Des Weiteren ist der Wert des Involvements – der Beteiligung – niedrig ausgefallen.

In der Bildauswertung wird ein sehr hoher Anspruch auf die Bildqualität gesetzt. Aus diesem Grund wurden zur Untersuchung der subjektiv wahrgenommenen Bildqualität in VR, systematische Experimente mit dem *HTC Vive*-Headset unternommen. Diese zeigten, dass die aktuelle Hardware noch nicht auf dem Stand von herkömmlichen Desktopmonitoren ist; sie jedoch ausreichend gut für den Konzeptentwurf und die Entwicklung eines Bildauswerteprogramms ist. Große Mängel haben sich an den Rändern der Linsen gezeigt. Die Bauform der Linsen verursacht eine Verzerrung des

1 TerraSAR-X Radar Satellite Image Aswan Dam:

<http://www.satimagingcorp.com/gallery/terrasar-x/terrasar-x-radar-satellite-image-aswan-dam/>

2 Unity3D Textures: <https://docs.unity3d.com/Manual/class-TextureImporter.html>

Bildes in den äußeren Bereichen. Des Weiteren fallen die Abstände zwischen einzelnen Pixel sowie Sub-Pixel beim Lesen von Texten auf.

Zusammenfassend lässt sich sagen, dass das Vorhaben der Konzeption einer gebrauchstauglichen VR-Anwendung für die Bildauswertung gelungen ist, es jedoch noch Raum für Verbesserungen gibt. VIEW erlaubt die intuitive Interaktion mit hochauflösten Luftbildaufnahmen in einer weitestgehend immersiven Umgebung.

Ausblick Für weitere Arbeiten sollten die Aspekte des adaptiven Lernens und das Konzept des Flows [Che07] auf VIEW angewendet werden. Hierbei sollten Untersuchungen zu deren Auswirkung auf die empfundene Presence – insbesondere den Aspekt der Beteiligung – durchgeführt werden. Die Hypothese hierbei ist, dass E-Learning in Kombination mit länger dauernden Aufgaben das Empfinden von physikalischer Präsenz und von Beteiligung erhöht. Li Z. et al. geben an, dass bei der Kombination von VR mit E-Learning sowohl die Immersion als auch die Lernfähigkeit und Lernbereitschaft der Benutzer sich steigern [Li09].

Des Weiteren sollten die zwar entwickelten aber aus Zeitgründen nicht umgesetzten Interaktionskonzepte, zusätzlich implementiert und evaluiert werden. Insbesondere das Konzept der *MapCards*, kleine Karten-förmige Ausschnitte von ausgewählten Bereichen, sollte umgesetzt werden. Dieses Konzept bietet unter anderem die Möglichkeit, Gebäude und die Erdoberfläche dreidimensional darzustellen. Dies könnte die Fähigkeiten der Darstellung dreidimensionaler Welten in VR noch tiefergründiger ausnutzen. Besonders bei der Darstellung von Radar-Aufnahmen würden sich hier viele interessante Möglichkeiten bieten. So könnte beispielsweise der von Bonnaffe et al. entwickelte Prozess zur Darstellung von Radar-Aufnahmen in VR genutzt werden, um diese auf den Mapcards anzuzeigen [Bon07].

Letztlich sollten Untersuchungen zu der Bildqualität von zukünftigen VR-HMD durchgeführt werden. Obwohl das *True VR* – das VR-Erlebnis, welches nicht von der realen Welt unterscheidbar ist – noch weit entfernt ist [Mar17], so werden bereits jetzt Headsets mit deutlich besseren Spezifikationen angekündigt. *PiMAX* hat beispielsweise Anfang des Jahres ein 8k-HMD angekündigt¹. HMDs mit höherer Auflösung, deutlich höherer Pixeldichte und größerem, weniger verzerrtem FOV könnten die subjektiv wahrgenommene Bildqualität deutlich verbessern. Zudem würde die Unterstützung von Eye-Tracking einerseits das Foveated Rendering erlauben² und andererseits könnte es weitere Chancen zu tiefgreifenderen Gaze-basierten Interaktionskonzepten erlauben [Jac95].

1 PiMAX's 8k Headset: <http://www.roadtovr.com/hands-pimaxs-8k-headset-proves-high-fov-vr-coming/>

2 NVIDIA Partners with SMI on Innovative Rendering Technique That Improves VR:
<https://blogs.nvidia.com/blog/2016/07/21/rendering-foveated-vr/>

Literaturverzeichnis

- [Abr14] ABRASH, Michael: What VR Could, Should, And Almost Certainly Will Be Within Two Years (2014), URL <http://media.steampowered.com/apps/abrashblog/AbrashDevDays2014.pdf>
- [Ada04] ADAMS, Ernest: Postmodernism and the three types of immersion. *Gamasutra: The Art And Business of Making Games* (2004), Bd. 9
- [Adr16] ADRIAN HEINRICH HOPPE: *Interaktionskonzepte fuer WIMP-Applikationen in Virtual Reality*, Master thesis, KIT (2016)
- [Aeh13] AEHNELT, Mario; BADER, Sebastian; RUSCHER, Gernot; KRÜGER, Frank; URBAN, Bodo und KIRSTE, Thomas: Situation aware interaction with multi-modal business applications in smart environments (2013), in: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Bd. 8018 LNCS, Springer, Berlin, Heidelberg, S. 413–422, URL http://link.springer.com/10.1007/978-3-642-39226-9_45
- [Bar95] BARFIELD, Woodrow. und FURNESS, Thomas A.: *Virtual Environments and Advanced Interface Design*, Bd. 55, Oxford University Press (1995), URL <http://linkinghub.elsevier.com/retrieve/pii/0160932796884249>
- [BEN16] BENJ EDWARDS: The Wacky World of VR in the 80s and 90s (2016), URL <http://www.pcmag.com/feature/343351/the-wacky-world-of-vr-in-the-80s-and-90s>
- [Bil08] BILLEN, Magali I.; KREYLOS, Oliver; HAMANN, Bernd; JADAMEC, Margarete A.; KELLOGG, Louise H.; STAADT, Oliver und SUMNER, Dawn Y.: A geoscience perspective on immersive 3D gridded data visualization. *Computers and Geosciences* (2008), Bd. 34(9): S. 1056–1072, URL <http://linkinghub.elsevier.com/retrieve/pii/S009830040800071X>
- [Bjo04] BJORK, S und HOLOPAINEN, Jussi.: *Patterns in game design (game development series)*, Charles River Media, 1 Aufl. (2004), URL <http://www.citeulike.org/group/1820/article/423551>
- [Bon07] BONNAFFE, Florence; JENNETTE, Dave und ANDREWS, John: A method for acquiring and processing ground-based lidar data in difficult-to-access outcrops for use in three-dimensional, virtual-reality models. *Geosphere* (2007), Bd. 3(6): S. 501–510, URL <http://geosphere.gsapubs.org/cgi/doi/10.1130/GES00104.1>
- [Boz16] BOZGEYIKLI, Evren; RAIJ, Andrew; KATKOORI, Srinivas und DUBEY, Rajiv: Point And Teleport Locomotion Technique for Virtual Reality (2016), in: *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play - CHI PLAY '16*, ACM Press, New York, New York, USA, S. 205–216, URL <http://dl.acm.org/citation.cfm?doid=2967934.2968105>

- [Bri16] BRIC, Justin D.; LUMBARD, Derek C.; FRELICH, Matthew J. und GOULD, Jon C.: Current state of virtual reality simulation in robotic surgery training: a review. *Surgical Endoscopy* (2016), Bd. 30(6): S. 2169–2178, URL <http://www.ncbi.nlm.nih.gov/pubmed/26304107><http://link.springer.com/10.1007/s00464-015-4517-y>
- [Bro96] BROOKE, John: SUS - A quick and dirty usability scale. *Usability evaluation in industry* (1996), Bd. 189(194): S. 4–7, URL <http://hell.meiert.org/core/pdf/sus.pdf>
- [Bur96] BURDEA, G.C.: Force and Touch Feedback for Virtual Reality (1996), URL <https://www.mysciencework.com/publication/show/61a6677ed9f79826a5eba79a7413a62f>
- [Bye89] BYERS, J. C.; BITTNER, A. C. und HILL, S. G.: Traditional and raw task load index (TLX) correlations: Are paired comparisons necessary? (1989), in: *Advances in industrial ergonomics and safety*, S. 481–485
- [Cha13] CHA, Yeonjoo und MYUNG, Rohae: Extended Fitts' law for 3D pointing tasks using 3D target arrangements. *International Journal of Industrial Ergonomics* (2013), Bd. 43(4): S. 350–355
- [Che07] CHEN, Jenova: Flow in games (and everything else). *Communications of the ACM* (2007), Bd. 50(4): S. 31, URL <http://portal.acm.org/citation.cfm?doid=1232743.1232769>
- [Cro15] CRONIN, Beau: The hierarchy of needs in virtual reality development (2015), URL <https://medium.com/@beaucronin/the-hierarchy-of-needs-in-virtual-reality-development-4333a4833acc#.eu69u9w2i>
- [Dav86] DAVIS, Fred D: *A Technology Acceptance Model for Empirically Testing New End-User Information Systems*, Dissertation, Massachusetts Institute of Technology (1986)
- [Dav16] DAVID MATTHEW, Whittinghill; BRADLEY, Ziegler; CASE, Tristan und BRENNAN, Moore: System and method for reducing simulator sickness (2016), URL <https://www.google.com/patents/US20160300391>
- [Deu15] DEUSE, Jochen; WEISNER, Kirsten; HENGSTEBECK, André und BUSCH, Felix: Gestaltung von Produktionssystemen im Kontext von Industrie 4.0, in: *Zukunft der Arbeit in Industrie 4.0*, Springer Berlin Heidelberg, Berlin, Heidelberg (2015), S. 99–109, URL http://link.springer.com/10.1007/978-3-662-45915-7_11
- [Duh04] DUH, HBL; PARKER, DE und FURNESS, TA: An independent visual background reduced simulator sickness in a driving simulator. *Presence: Teleoperators And Virtual ...* (2004), Bd. 13(5): S. 578–588, URL <http://www.mitpressjournals.org/doi/abs/10.1162/1054746042545283>
- [Epi] EPIC GAMES: Virtual Reality Best Practices | Unreal Engine, URL <https://docs.unrealengine.com/latest/INT/Platforms/VR/ContentSetup/>
- [Fer16] FERNANDES, Ajoy S. und FEINER, Steven K.: Combating VR sickness through subtle dynamic field-of-view modification (2016), in: *2016 IEEE Symposium on 3D User Interfaces, 3DUI 2016 - Proceedings*, S. 201–210
- [Fit54] FITTS, Paul M.: The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* (1954), Bd. 47(6): S. 381–391
- [For97] FORSBERG, A.S.; LAVIOLA, J.J.; MARKOSIAN, L. und ZELEZNIK, R.C.: Seamless interaction in virtual reality. *IEEE Computer Graphics and Applications* (1997), Bd. 17(6): S. 6–9, URL <http://ieeexplore.ieee.org/document/626956><http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=626956>

- [Fur04] FURUMURA, T und CHEN, L: Large scale parallel simulation and visualization of 3D seismic wavelield using the earth simulator. *Cmes-Computer Modeling In Engineering And Sciences* (2004), Bd. 6(2): S. 153–168, URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.596.1781&rep=rep1&type=pdf>
- [Goo15] GOOGLE: A new dimension - Designing for Google Cardboard - VR design guidelines (2015), URL <http://www.google.com/design/spec-vr/designing-for-google-cardboard/a-new-dimension.html#>
- [Goo16] GOOGLE: Tilt Brush (2016), URL <https://www.tiltbrush.com/>
- [Gre14] GREG KUMPARAK: Oculus Founder Palmer Luckey On Why Now Is The Time For VR, And The Biggest Hurdles Ahead (2014), URL <https://techcrunch.com/2014/03/08/oculus-ceo-palmer-luckey-on-why-now-is-the-time-for-vr-and-the-biggest-hurdles-ahead/>
- [Gru] GRUBERT, Jens: NASA Task Load Index - German Raw, URL <http://jensgrubert.bplaced.net/nasa-tlx-short/TLX-German-short.html>
- [Gui87] GUIARD, Y: Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. *Journal of motor behavior* (1987), Bd. 19(4): S. 486–517, URL <http://www.tandfonline.com/doi/abs/10.1080/00222895.1987.10735426>
- [Har88] HART, Sandra G. und STAVELAND, Lowell E.: Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology* (1988), Bd. 52(C): S. 139–183
- [Har06] HART, SANDRA, G.: NASA-task load index (NASA-TLX); 20 years later. *Human Factors and Ergonomics Society Annual Meeting* (2006): S. 904–908
- [Hec16] HECHT, Jeff: Optical Dreams, Virtual Reality. *Optics and Photonics News* (2016), Bd. 27(6): S. 24–31
- [HTC] HTC: VIVE™ United Kingdom | Discover Virtual Reality Beyond Imagination, URL <https://www.vive.com/uk/>
- [HTC16] HTC: Vive PRE User Guide (2016), URL http://www.htc.com/managed-assets/shared/desktop/vive/Vive_PRE_User_Guide.pdf
- [Hub05] HUBONA, Geoffrey S. und SHIRAH, Gregory W.: Spatial Cues in 3D Visualization, in: *Ambient Intelligence for Scientific Discovery*, 2, Springer Berlin Heidelberg (2005), S. 104–128, URL http://link.springer.com/10.1007/978-3-540-32263-4_6
- [Igr] IGROUP PROJECT CONSORTIUM: Immersion vs. Presence, URL <http://www.igroup.org/pq/ipq/construction.php>
- [Igr15] IGROUP PROJECT CONSORTIUM: Igroup Presence Questionnaire (2015), URL <http://www.igroup.org/pq/ipq/download.php#English>
- [Iso98] ISO: ISO 9241-11:1998 - Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability (1998), URL <https://www.iso.org/standard/16883.html>
- [Jac95] JACOB, Robert J K: Eye Tracking in Advanced Interface Design. *Virtual Environments and Advanced Interface Design* (1995): S. 258–290

- [JOJ16] JOJO PUTHUPARAMPIL: Virtual test drives next big trend for cars (2016), URL <http://www.hypergridbusiness.com/2016/06/virtual-test-drives-the-next-big-trend-for-cars/>
- [Jul17] JULIA BECK: Try before you buy with Expedia (2017), URL <http://www.virtual-reality-in-tourism.com/try-before-you-buy-with-expedia/>
- [Kau15] KAULICH, Christian: *Immersion und Interaktion in Virtual Reality Anwendungen*, Master thesis, Universität Leipzig (2015), URL https://www.leipzigschoolofmedia.de/fileadmin/content/Dokumente/Masterarbeiten/MM/Masterarbeit_Kaulich-Christian_2015.pdf
- [Ken00] KENNEDY, Robert S.; STANNEY, Kay M. und DUNLAP, William P.: Duration and Exposure to Virtual Environments: Sickness Curves During and Across Sessions. *Presence: Teleoperators and Virtual Environments* (2000), Bd. 9(5): S. 463–472, URL <http://www.mitpressjournals.org/doi/abs/10.1162/105474600566952>
- [Kre06] KREYLOS, Oliver; BAWDEN, Gerald W.; BERNARDIN, Tony; BILLEN, Magali I.; COWGILL, Eric S.; GOLD, Ryan D.; HAMANN, Bernd; JADAMEC, Margarete A.; KELLOGG, Louise H.; STAADT, Oliver G. und SUMNER, Dawn Y.: Enabling Scientific Workflows in Virtual Reality (2006), in: *Proceedings of the 2006 ACM International Conference on Virtual Reality Continuum and Its Applications*, ACM Press, New York, New York, USA, S. 155–162, URL <http://portal.acm.org/citation.cfm?doid=1128923.1128948>
- [Lac16] LACKEY, S. J.; SALCEDO, J. N.; SZALMA, J.L. und HANCOCK, P.A.: The stress and workload of virtual reality training: the effects of presence, immersion and flow. *Ergonomics* (2016), Bd. 0139(April): S. 1–13, URL <http://www.ncbi.nlm.nih.gov/pubmed/26977540>
- [Li09] LI, Zhenbo; YUE, Jun und JAUREGUI, David Antonio Gomez: A new virtual reality environment used for e-learning (2009), in: *ITME2009 - Proceedings 2009 IEEE International Symposium on IT in Medicine and Education*, S. 445–449, URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.367.6353&rep=rep1&type=pdf>
- [Lin98] LIN, Ching-Rong und LOFTIN, R. Bowen: Application of Virtual Reality in the Interpretation of Geoscience data (1998), in: *VRST '98 Proceedings of the ACM symposium on Virtual reality software and technology*, ACM Press, New York, New York, USA, S. 187–194, URL <http://portal.acm.org/citation.cfm?doid=293701.293736>
- [Lof00] LOFTIN, R.B. und NELSON, H.R.: Interaction with geoscience data in an immersive environment (2000), in: *Proceedings IEEE Virtual Reality 2000 (Cat. No.00CB37048)*, IEEE Comput. Soc, S. 55–62, URL <http://ieeexplore.ieee.org/document/840363/>
- [Loh13] LOHMANN, Kris und SCHÄFFER, Jörg: System Usability Scale (SUS)–An Improved German Translation of the Questionnaire (2013), URL <https://minds.coremedia.com/2013/09/18/sus-scale-an-improved-german-translation-questionnaire/>
- [Mal] MALAIKA, Yasser: GDC Vault - Interaction Design in VR: The Rules Have Changed (Again), URL <http://www.gdcvault.com/play/1022810/Interaction-Design-in-VR-The>
- [Mal15] MALIŃSKA, Marzena; ZUŻEWICZ, Krystyna; BUGAJSKA, Joanna und GRABOWSKI, Andrzej: Heart rate variability (HRV) during virtual reality immersion. *International journal of occupational safety and ergonomics : JOSE* (2015), Bd. 21(1): S. 47–54, URL <http://www.ncbi.nlm.nih.gov/pubmed/26327262>

- [Mar16] MARTINDALE JON: Oculus Rift vs. HTC Vive: How Should We Walk in VR? | Digital Trends (2016), URL <http://www.digitaltrends.com/virtual-reality/vr-locomotion-movement-omni-hover-junkers/>
- [Mar17] MARK METRY: When will VR Become Indistinguishable from Reality? | VU Dream (2017), URL <http://www.vudream.com/when-will-vr-become-indistinguishable-from-reality/>
- [Nak14] NAKAMURA, Jeanne und CSIKSZENTMIHALYI, Mihaly: The concept of flow, in: *Flow and the Foundations of Positive Psychology: The Collected Works of Mihaly Csikszentmihalyi*, Springer (2014), S. 239–263
- [Nic16] NICK PETTIT: What is Room-Scale VR? (2016), URL <http://blog.teamtreehouse.com/what-is-room-scale-vr>
- [Nom01] NOMURA, Junji und SAWADA, Kazuya: Virtual reality technology and its industrial applications. *Annual Reviews in Control* (2001), Bd. 25: S. 99–109
- [Nor99] NORMAN, Donald A.: Affordance, conventions, and design. *Interactions* (1999), Bd. 6(3): S. 38–42, URL <http://portal.acm.org/citation.cfm?doid=301153.301168>
- [Noy07] NOYES, Jan M und BRUNEAU, Daniel P J: A self-analysis of the NASA-TLX workload measure. *Ergonomics* (2007), Bd. 50(4): S. 514–519
- [Ohn07] OHNO, Nobuaki und KAGEYAMA, Akira: Scientific visualization of geophysical simulation data by the CAVE VR system with volume rendering. *Physics of the Earth and Planetary Interiors* (2007), Bd. 163(1-4): S. 305–311
- [Oi] OLED-INFO: PenTile: technology explained, URL <https://www.oled-info.com/pentile>
- [Pal17] PALMISANO, Stephen; MURSIC, Rebecca und KIM, Juno: Vection and cybersickness generated by head-and-display motion in the Oculus Rift. *Displays* (2017), Bd. 46: S. 1–8, URL <http://linkinghub.elsevier.com/retrieve/pii/S0141938216300713>
- [Par08] PARSONS, Thomas D und RIZZO, Albert A: Affective outcomes of virtual reality exposure therapy for anxiety and specific phobias: A meta-analysis. *Journal of Behavior Therapy and Experimental Psychiatry* (2008), Bd. 39: S. 250–261, URL <https://pdfs.semanticscholar.org/bfb1/012c4a26ace569a4a43fb17aa2d656ce4a16.pdf>
- [Pau15] PAUL JAMES: This HTC Vive PAX Reactions Video is Pure, Honest Hype (2015), URL <http://www.roadtovr.com/this-htc-vive-pax-reactions-video-is-pure-honest-hype/>
- [Rav16] RAVASZ, Jonathan: Design Practices in Virtual Reality (2016), URL <https://uxdesign.cc/design-practices-in-virtual-reality-f900f5935826#.ol3kzkoy3>
- [RIK14] RIK HENDERSON: Just how good is Oculus Rift Development Kit 2 in comparison to DK1? - Pocket-lint (2014), URL <http://www.pocket-lint.com/news/129752-just-how-good-is-oculus-rift-development-kit-2-in-comparison-to-dk1>
- [Rol13] ROLLER, Wolfgang; BERGER, Anton und SZENTES, Daniel: Technology based training for radar image interpreters (2013), in: *2013 6th International Conference on Recent Advances in Space Technologies (RAST)*, 6, S. 1173–1177, URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6581181>

- [Sau13] SAURO, Jeff: 10 Things to Know About the System Usability Scale (SUS) (2013), URL [http://www.measuringusability.com/blog/10-things-SUS.php?utm_source=feedly&utm_medium=feed&utm_campaign=Feed:+MeasuringUsability+\(Measuring+Usability:+Quantitative+Usability+and+Statistics\)](http://www.measuringusability.com/blog/10-things-SUS.php?utm_source=feedly&utm_medium=feed&utm_campaign=Feed:+MeasuringUsability+(Measuring+Usability:+Quantitative+Usability+and+Statistics))
- [Sch99] SCHUBERT, Thomas; FRIEDMANN, Frank und REGENBRECHT, Holger: Embodied presence in virtual environments. *Visual representations and interpretations* (1999): S. 268–278, URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.40.3047&rep=rep1&type=pdf>
- [Sch01] SCHUBERT, Thomas; FRIEDMANN, Frank und REGENBRECHT, Holger: The Experience of Presence: Factor Analytic Insights. *Presence: Teleoperators and Virtual Environments* (2001), Bd. 10(3): S. 266–281, URL http://www.hci.otago.ac.nz/pubs/2001_SchubertFriedmannRegenbrecht_PRESENCEJune2001_p266.pdf
- [Sha15] SHANMUGAM, Prasanth: UX And Virtual Reality - Designing for interfaces without screens (2015), URL <http://www.uxness.in/2015/08/ux-virtual-reality.html>
- [Sla94] SLATER, Mel; USOH, Martin und STEED, Anthony: Depth of Presence in Virtual Environments. *Presence: Teleoperators and Virtual Environments* (1994), Bd. 3(2): S. 130–144, URL <http://www.mitpressjournals.org/doi/10.1162/pres.1994.3.2.130>
- [St.15] ST. PIERRE, Matthew E.; BANERJEE, Salil; HOOVER, Adam W. und MUTH, Eric R.: The effects of 0.2 Hz varying latency with 20-100 ms varying amplitude on simulator sickness in a helmet mounted display. *Displays* (2015), Bd. 36: S. 1–8
- [Ste92] STEUER, Jonathan: Defining Virtual Reality: Dimensions Determining Telepresence. *Journal of Communication* (1992), Bd. 42(4): S. 73–93, URL <http://doi.wiley.com/10.1111/j.1460-2466.1992.tb00812.x>
- [Str99] STREITZ, Norbert A.; GEISLER, Jörg; HOLMER, Torsten; KONOMI, Shin'ichi; MÜLLER-TOMFELDE, Christian; REISCHL, Wolfgang; REXROTH, Petra; SEITZ, Peter; STEINMETZ, Ralf; GEIBLER, Jiirg und MILLER-TOMFELDE, Christian: i-Land: An Interactive Landscape for Creativity and Innovation (1999), in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'99)*, May, ACM Press, New York, New York, USA, S. 120–127, URL <http://portal.acm.org/citation.cfm?doid=302979.303010>
- [Str01] STREITZ, Norbert A; TANDLER, Peter; MÜLLER-TOMFELDE, Christian und KONOMI, Shin: Roomware - Towards the Next Generation of Human-Computer Interaction Based on an Integrated Design of Real and Virtual Worlds. *Human-Computer Interaction in the New Millennium* (2001): S. 553–578
- [Str15] STREICHER, Alexander und ROLLER, Wolfgang: Towards an Interoperable Adaptive Tutoring Agent for Simulations and Serious Games. *International Conference on Theory and Practice in Modern Computing, MCCSIS 2015* (2015): S. 194–197
- [Str16] STREICHER, Alexander und LEHMANN, Eva: Das Technologieakzeptanzmodell für kartenbasierte Lernspiele in der Bildauswertung. *DeLFI 2016* (2016)
- [Uni] UNITY FOUNDATION: Unity - Optimisation for VR in Unity, URL <https://unity3d.com/learn/tutorials/topics/virtual-reality/optimisation-vr-unity>
- [Uso00] USOH, Martin; CATENA, Ernest; ARMAN, Sima und SLATER, Mel: Using presence questionnaires in reality. *Presence: Teleoperators and Virtual Environments* (2000): S. 1–16, URL <http://discovery.ucl.ac.uk/106368/>

- [vdM09] VAN DER MEIJDEN, O. A. J. und SCHIJVEN, M. P.: The value of haptic feedback in conventional and robot-assisted minimal invasive surgery and virtual reality training: a current review. *Surgical Endoscopy* (2009), Bd. 23(6): S. 1180–1190, URL <http://link.springer.com/10.1007/s00464-008-0298-x>
- [Voi] VOIT, Michael: SmartControlRoom, URL <https://www.iosb.fraunhofer.de/servlet/is/6620/>
- [VRSa] VRS.ORG: Who Coined the Term “Virtual Reality”?, URL <https://www.vrs.org.uk/virtual-reality/who-coined-the-term.html>
- [VRSb] VRS.ORG.UK: Virtual Reality in the Military, URL <https://www.vrs.org.uk/virtual-reality-military/>
- [Wit98] WITMER, Bob G und SINGER, Michael J: Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoper. Virtual Environ.* (1998), Bd. 7(3): S. 225–240, URL <http://www.mitpressjournals.org/doi/abs/10.1162/105474698565686>
- [Wol11] WOLF, Katrin; NAUMANN, Anja; ROHS, Michael und MÜLLER, Jörg: A taxonomy of microinteractions: Defining microgestures based on ergonomic and scenario-Dependent requirements (2011), in: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Bd. 6946 LNCS, Springer, Berlin, Heidelberg, S. 559–575, URL http://link.springer.com/10.1007/978-3-642-23774-4_45
- [Yao14] YAO, Richard; HEATH, Tom; DAVIES, Aaron; FORSYTH, Tom; MITCHELL, Nate und HOBERMAN, Perry: Oculus VR best practices guide. *Oculus VR* (2014)
- [Yi07] YI, Ji Soo; AH KANG, Youn; STASKO, John T. und JACKO, Julie A.: Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics* (2007), Bd. 13(6): S. 1224–1231, URL <http://ieeexplore.ieee.org/document/4376144/>

Abbildungsverzeichnis

1.1	Der VR-Prozess	1
1.2	Kapitel Aufbau	4
2.1	Visionen von Arbeitsumgebungen der Zukunft	6
2.2	Überblick zum SmartControlRoom	6
2.3	Interaktionskonzepte	8
2.4	Lokomotion in VR	10
2.5	Möglichkeiten der Reduzierung von VR-Krankheit	12
2.6	Prozess von der Immersion zur Presence	14
2.7	The hierarchy of needs in virtual reality	14
3.1	Oculus Rift HMDs	16
3.2	Überblick zum HTC Vive System	16
3.3	Benötigte Komponenten für die Nutzung von PS-VR	17
3.4	VR Headsets welche Smartphones als Display nutzen	17
3.5	Relevante Virtual Reality Anwendungen	18
3.6	VR und die virtuelle Welt	19
3.7	Beispiele unterschiedlicher VR Eingabegeräte	20
3.8	Das Uncanny Valley in VR	20
3.9	Unterschiedliche Arten von Luftbildaufnahmen	22
3.10	Ausschnitt des ViLand Interfaces im Überblick	23
3.11	Ausschnitte aus dem Bildauswertungs-Programm ViLand	23
3.12	Qualitätsmerkmale von Bildausgabegeräten	23
3.13	Das Blickfeld in der Computergrafik	26
3.14	Computergrafik Verfahren	26
4.1	Vereinfachter Ablauf vom Sensor zum Gehirn	29
4.2	Fliegengitter Effekt	30
4.3	Ring Effekt	30
4.4	Light Ray Effekt	31
4.5	Vergleich der Subpixel-Muster	33
4.6	Überblick zu den beiden erstellten Szenen	34
4.7	Beispiele der Texturen, die bei diesem Experiment genutzt wurden	35
4.8	Unterschiedliche LOD eines Bildausschnittes	36
5.1	Skizzen für Lokomotionskonzepte	45
5.2	Skizzen für Zoomkonzepte	46
5.3	Übersicht zu den <i>MapCards</i>	47
5.4	<i>Touchpad Shortcuts</i> Konzeptzeichnungen	47

5.5	Tilt Brush Interaktion	48
5.6	InteractionCube Konzeptzeichnungen	48
5.7	Anpassung von Bildwerten wie Kontrast und Helligkeit	49
6.1	Architektur von VIEW	51
6.2	Ausschnitte der ersten Iteration	56
6.3	UnitySlippyMap und RenderTexture-Kamera	57
6.4	Überblick zum <i>MapDisplay</i>	57
6.5	Überblick zum <i>MapDisplay</i>	58
6.6	Erstellung des <i>MapDisplays</i> in Blender	59
6.7	Umgebungsgeräusche im virtuellen Raum	60
6.8	Die Lost Earth Brücke	62
6.9	HTC Vive Controller Beschreibung	63
6.10	Tooltips auf dem rechten Controller	64
6.11	Teleportations Funktionalität im Überblick	64
6.12	Weitere Objekte im virtuellen Raum	65
6.13	Überblick zum Interaktioncube	66
6.14	3D-Modus am Beispiel <i>Mandulis Tempel</i>	67
6.15	Use Cases	68
7.1	IPQ Faktoren	73
7.2	Hypothesen (vereinfacht)	75
7.3	Überblick zu unterschiedlichen genutzten Materialien	76
7.4	IPD-Tool Übersicht	77
7.5	Unterschiedliche Darstellungen des Assuan-Gebiets	78
7.6	Radaraufnahme und 3D-Modell des gesuchten Mandulis-Tempels	79
7.7	Übersicht zur Studien Umgebung	82
7.8	Box Plot zu den RawTLX Werten	86
7.9	Visualisierung der RawTLX Werte	86
7.10	Presence Profil	87
7.11	Box Plot zu den IPQ Werten	88
7.12	Visuelle Übersicht zu den Interaktionswerten	91
1	Fragebogen zur Person	113
2	Deutsche Version des NASA Raw TLX Fragebogens	114
3	IPQ Presence Fragebogen - Seite 1/2	115
4	IPQ Presence Fragebogen - Seite 2/2	116
5	System Usability Scale Fragebogen	117
6	Hypothesen Aufbau	118

Tabellenverzeichnis

2.1	Taxonomy of VR input	10
4.1	Zusammenfassung der Spezifikationen aktueller HMDs	32
4.2	Zusammenfassung der empfohlenen Hardware Anforderungen	32
5.1	Arten von Immersion	40
5.2	Hardware Anforderungen für VR Geräte	41
5.3	Zusammenfassung gängiger Interaktionskonzepte für VR-Anwendungen	44
7.1	D'Agostino-Pearson omnibus test auf Datensätzen	84
7.2	Übersicht zu den RawTLX Werten	85
7.3	Zusammenfassung zu den RawTLX Werten der einzelnen Probanden	85
7.4	Mittelwerte der verglichenen Spiele	87
7.5	Übersicht zum IPQ G-Wert	87
7.6	Übersicht zum IPQ SP-Wert	87
7.7	Übersicht zum IPQ INV-Wert	87
7.8	Übersicht zum IPQ REAL-Wert	87
7.9	Zusammenfassung zu den IPQ Werten der einzelnen Probanden	88
7.10	Übersicht zum SUS-Wert	89
7.11	Box Plot zu den SUS Werten	89
7.12	Zusammenfassung zu den SUS Werten der einzelnen Probanden	89
7.13	Interaktion Rohdaten	90
7.14	Zusammenfassung zu den Beobachten Interaktionen	91
7.15	Pearson Correlation	92
7.16	Sigma der Korrelationen	93

Appendix A: Fragebögen der Evaluation

1 Fragebogen zur Person

Proband#: _____ Datum: __/__/17 Uhrzeit: __: __

FRAGEBOGEN ZUR PERSON

Allgemein

Alter _____
Geschlecht weiblich männlich

LEHRE UND BERUF

Höchster erreichter Bildungsabschluss (Nur eine Option ankreuzen):

- Grundschule
- Weiterführende Schule (Haupt- oder Realschulabschluss, Abitur)
- Aufstiegsfortbildung (z.B. Meister, Techniker, ...)
- Hochschulabschluss (2-4 Jahre, z.B. Bachelor)
- Hochschulabschluss (4+ Jahre, z.B. Master)
- Promotion oder vergleichbarer akademischer Grad

Studiengang, Fachgebiet: _____

ERFAHRUNG

	Keine Erfahrung			Sehr erfahren		
Erfahrung mit Computer-/Konsolenspielen	<input type="radio"/>					
Erfahrung mit Virtual Reality	<input type="radio"/>					

Falls Sie Erfahrung mit VR haben, welches Produkt haben Sie benutzt?

EINVERSTÄNDNIS-ERKLÄRUNG

Ich stimme zu, dass die aufgenommenen Daten Zwecks der Studie und des Projektes genutzt werden dürfen. Das aufgenommene Bild/Video Material dient zum Überprüfen der gesammelten Daten.

Hinweis: Da Sie ein Headset tragen werden, wird Ihr Gesicht kaum erkennbar sein.

Falls Sie einverstanden sind, dass die Bilder zusätzlich in Publikationen (Papier, Präsentation, ...) genutzt werden dürfen, kreuzen Sie bitte zusätzlich die folgende Markierung entsprechend an:

- Mein Gesicht darf in den Videos zu sehen sein
 Ja Nein
- Das Bild/Video Material darf zusätzlich in Publikationen genutzt werden
 Ja Nein

Unterschrift: _____ Datum: _____

Abbildung 1: Fragebogen zur Person

3 Presence

Proband#: ____

Datum: __/__/17

Uhrzeit: __: __

IGROUP PRESENCE QUESTIONNAIRE

Kreuzen Sie bitte in jeder Zeile nur eine der sieben Optionen an.

Beispiel:	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>überhaupt nicht</i>			<i>sehr stark</i>			
In der computererzeugten Welt hatte ich den Eindruck, dort gewesen zu sein...	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>trifft gar nicht zu</i>			<i>trifft völlig zu</i>			
Ich hatte das Gefühl, dass die virtuelle Umgebung hinter mir weitergeht.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>trifft gar nicht zu</i>			<i>trifft völlig zu</i>			
Ich hatte das Gefühl, nur Bilder zu sehen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>hatte nicht das Gefühl</i>			<i>hatte das Gefühl</i>			
Ich hatte nicht das Gefühl, in dem virtuellen Raum zu sein	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>trifft gar nicht zu</i>			<i>trifft völlig zu</i>			
Ich hatte das Gefühl, in dem virtuellen Raum zu handeln statt etwas von außen zu bedienen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>trifft gar nicht zu</i>			<i>trifft völlig zu</i>			
Ich fühlte mich im virtuellen Raum anwesend.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>extrem bewusst</i>	<i>mittelmäßig bewusst</i>			<i>unbewusst</i>		
Wie bewusst war Ihnen die reale Welt, während Sie sich durch die virtuelle Welt bewegten (z.B. Geräusche, Raumtemperatur, andere Personen etc.)?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	<i>trifft gar nicht zu</i>			<i>trifft völlig zu</i>			
Meine reale Umgebung war mir nicht mehr bewusst.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Abbildung 3: IPQ Presence Fragebogen - Seite 1/2

	trifft gar nicht zu						trifft völlig zu
Ich achtete noch auf die reale Umgebung.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	trifft gar nicht zu						trifft völlig zu
Meine Aufmerksamkeit war von der virtuellen Welt völlig in Bann gezogen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	vollkommen real		weder noch				gar nicht real
Wie real erschien Ihnen die virtuelle Umgebung?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	überhaupt nicht		etwas				vollständig
Wie sehr glich Ihr Erleben der virtuellen Umgebung dem Erleben einer realen Umgebung?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	wie eine vorgestellte Welt						nicht zu unterscheiden von der realen Welt
Wie real erschien Ihnen die virtuelle Welt?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	trifft gar nicht zu						trifft völlig zu
Die virtuelle Welt erschien mir wirklicher als die reale Welt.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Abbildung 4: IPQ Presence Fragebogen - Seite 2/2

4 SUS

Proband#: ____

Datum: __/__/17

Uhrzeit: __: __

SYSTEM USABILITY SCALE QUESTIONNAIRE

Kreuzen Sie bitte in jeder Zeile nur eine der fünf Optionen an.

Beispiel:	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Stimme überhaupt nicht zu			Stimme voll zu	
Ich denke, dass ich dieses System gerne regelmäßig nutzen würde.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich fand das System unnötig komplex.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich denke, das System war leicht zu benutzen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich denke, ich würde die Unterstützung einer fachkundigen Person benötigen, um das System benutzen zu können.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich fand, die verschiedenen Funktionen des Systems waren gut integriert.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich halte das System für zu inkonsistent.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich glaube, dass die meisten Menschen sehr schnell lernen würden, mit dem System umzugehen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich fand das System sehr umständlich zu benutzen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich fühlte mich bei der Nutzung des Systems sehr sicher.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ich musste viele Dinge lernen, bevor ich mit dem System arbeiten konnte.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Abbildung 5: System Usability Scale Fragebogen

5 Zusammenfassung der Hypothesen

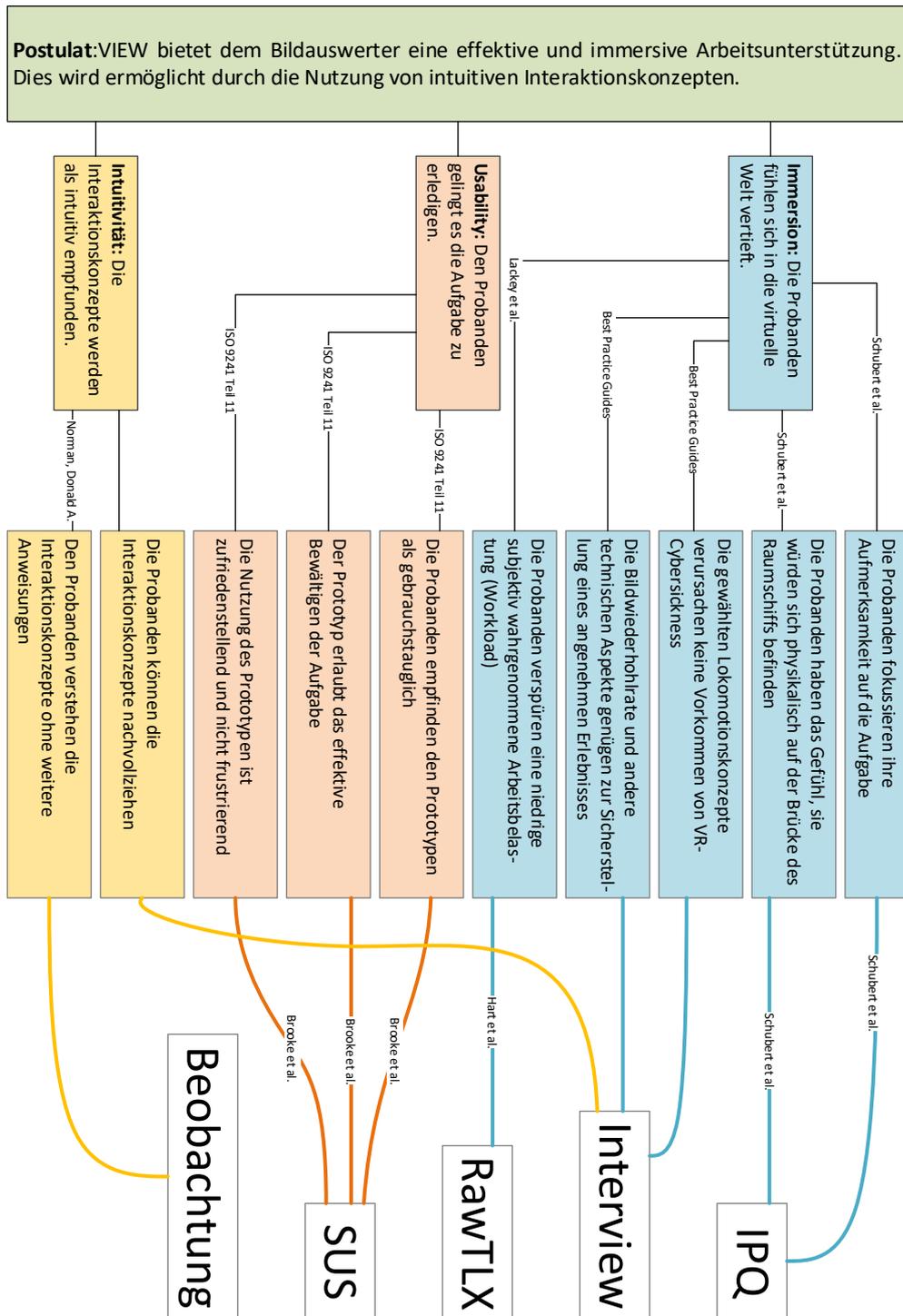


Abbildung 6: Zusammenfassung zum Aufbau der Verbindung zwischen Postulat und Fragebögen

Appendix B: Studien Ergebnisse

1 Korrelationen und Regressionsanalyse

CORRELATIONS

CORRELATION
 /VARIABLES = SUS SP REAL Alter SpieleErfahrung VRErfahrung INT_Teleport INT_Grab
 /PRINT = TWOTAIL NOSIG.

Correlations

		SUS	SP	REAL	Alter	SpieleErfahrung	VRErfahrung	INT_Teleport	INT_Grab
SUS	Pearson Correlation	1.00	.12	.18	-.26		.19	.56	.20
	Sig. (2-tailed)		.655	.513	.339		.491	.024	.447
	N	16	16	16	16	16	16	16	16
SP	Pearson Correlation	.12	1.00	.73	.56		-.08	.28	-.04
	Sig. (2-tailed)	.655		.001	.025		.781	.298	.881
	N	16	16	16	16	16	16	16	16
REAL	Pearson Correlation	.18	.73	1.00	.71		-.40	.18	-.15
	Sig. (2-tailed)	.513	.001		.002		.125	.515	.577
	N	16	16	16	16	16	16	16	16
Alter	Pearson Correlation	-.26	.56	.71	1.00		-.50	-.25	-.17
	Sig. (2-tailed)	.339	.025	.002			.051	.355	.537
	N	16	16	16	16	16	16	16	16
SpieleErfahrung	Pearson Correlation	.19	-.08	-.40	-.50	1.00		.28	.68
	Sig. (2-tailed)	.491	.781	.125	.051			.294	.004
	N	16	16	16	16	16	16	16	16
VRErfahrung	Pearson Correlation	.56	.28	.18	-.25	.28	1.00		.19
	Sig. (2-tailed)	.024	.298	.515	.355	.294			.480
	N	16	16	16	16	16	16	16	16
INT_Teleport	Pearson Correlation	.20	-.04	-.15	-.17	.68	.19	1.00	
	Sig. (2-tailed)	.447	.881	.577	.537	.004	.480		
	N	16	16	16	16	16	16	16	16
INT_Grab	Pearson Correlation	.63	.45	.45	.18	.14	.67	.38	1.00
	Sig. (2-tailed)	.009	.080	.077	.513	.607	.004	.146	
	N	16	16	16	16	16	16	16	16

REGRESSION

REGRESSION
 /VARIABLES= Alter
 /DEPENDENT= SP
 /METHOD=ENTER
 /STATISTICS=COEFF R ANOVA.

Model Summary (SP)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.56	.31	.26	.82

ANOVA (SP)

	Sum of Squares	df	Mean Square	F	Sig.
--	----------------	----	-------------	---	------

Regression	4.29	1	4.29	6.33	.025
Residual	9.47	14	.68		
Total	13.76	15			

Coefficients (SP)

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	1.76	.61	.00	2.91	.011
Alter	.04	.02	.56	2.52	.025

REGRESSION

```
REGRESSION
/VARIABLES= Alter
/DEPENDENT= REAL
/METHOD=ENTER
/STATISTICS=COEFF R ANOVA.
```

Model Summary (REAL)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.71	.50	.47	.83

ANOVA (REAL)

	Sum of Squares	df	Mean Square	F	Sig.
Regression	9.61	1	9.61	14.05	.002
Residual	9.58	14	.68		
Total	19.19	15			

Coefficients (REAL)

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	-1.09	.61	.00	-1.78	.095
Alter	.06	.02	.71	3.75	.002

REGRESSION

```
REGRESSION
/VARIABLES= VRErfahrung
/DEPENDENT= SUS
/METHOD=ENTER
/STATISTICS=COEFF R ANOVA.
```

Model Summary (SUS)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.56	.31	.26	7.16

ANOVA (SUS)

	Sum of Squares	df	Mean Square	F	Sig.
Regression	326.04	1	326.04	6.36	.024
Residual	717.71	14	51.27		
Total	1043.75	15			

Coefficients (SUS)

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	69.01	4.48	.00	15.40	.000
VRErfahrung	4.04	1.60	.56	2.52	.024

REGRESSION

```
REGRESSION
/VARIABLES= INT_Grab
/DEPENDENT= SUS
```

```

/METHOD=ENTER
/STATISTICS=COEFF R ANOVA.
    
```

Model Summary (SUS)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.63	.40	.35	6.71

ANOVA (SUS)

	Sum of Squares	df	Mean Square	F	Sig.
Regression	412.61	1	412.61	9.15	.009
Residual	631.14	14	45.08		
Total	1043.75	15			

Coefficients (SUS)

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	80.55	1.72	.00	46.75	.000
INT_Grab	6.29	2.08	.63	3.03	.009

REGRESSION

```

REGRESSION
/VARIABLES= VRErfahrung
/DEPENDENT= INT_Grab
/METHOD=ENTER
/STATISTICS=COEFF R ANOVA.
    
```

Model Summary (INT Grab)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.67	.45	.41	.64

ANOVA (INT Grab)

	Sum of Squares	df	Mean Square	F	Sig.
Regression	4.71	1	4.71	11.50	.004
Residual	5.73	14	.41		
Total	10.44	15			

Coefficients (INT Grab)

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	-1.43	.40	.00	-3.58	.003
VRErfahrung	.49	.14	.67	3.39	.004

REGRESSION

```

REGRESSION
/VARIABLES= INT_Teleport
/DEPENDENT= SpieleErfahrung
/METHOD=ENTER
/STATISTICS=COEFF R ANOVA.
    
```

Model Summary (SpieleErfahrung)

R	R Square	Adjusted R Square	Std. Error of the Estimate
.68	.46	.42	.98

ANOVA (SpieleErfahrung)

	Sum of Squares	df	Mean Square	F	Sig.
Regression	11.57	1	11.57	12.06	.004
Residual	13.43	14	.96		
Total	25.00	15			

Coefficients (SpieleErfahrung)

	Unstandardized Coefficients	Standardized Coefficients		

	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>	<i>t</i>	<i>Sig.</i>
<i>(Constant)</i>	1.50	.69		.002	.17
<i>INT_Teleport</i>	2.57	.74		.683	.47

REGRESSION

```

REGRESSION
  /VARIABLES= REAL
  /DEPENDENT= SP
  /METHOD=ENTER
  /STATISTICS=COEFF R ANOVA.

```

Model Summary (SP)

<i>R</i>	<i>R Square</i>	<i>Adjusted R Square</i>	<i>Std. Error of the Estimate</i>
.73	.53	.49	.68

ANOVA (SP)

	<i>Sum of Squares</i>	<i>df</i>	<i>Mean Square</i>	<i>F</i>	<i>Sig.</i>
<i>Regression</i>	7.26	1	7.26	15.62	.001
<i>Residual</i>	6.50	14	.46		
<i>Total</i>	13.76	15			

Coefficients (SP)

	<i>Unstandardized Coefficients</i>		<i>Standardized Coefficients</i>	<i>t</i>	<i>Sig.</i>
	<i>B</i>	<i>Std. Error</i>	<i>Beta</i>		
<i>(Constant)</i>	2.55	.24		.0010	.73
<i>REAL</i>	.61	.16		.73	3.95

Danksagung

Danken möchte ich all denjenigen, die mich über den Verlauf meines Studiengangs und vor allem in den letzten sechs Monaten unterstützt haben.

Ganz besonders möchte ich meinem Betreuer Alexander Streicher danken. Sie haben mich dazu motiviert, über meine Grenzen hinaus zu denken und gaben mir immer wieder konstruktive Kritik und wertvolle Hinweise. Vielen Dank für die Mühen und Zeit, die Sie in mich und meine Arbeit investiert haben.

Außerdem möchte ich mich bei meinem betreuenden Professor, Prof. Dr.-Ing. Jürgen Beyerer, für seine hilfreichen Anmerkungen zu meiner Arbeit bedanken.

Danken möchte ich auch den Mitarbeitern und HiWis des Fraunhofer IOSB. Sie konnten oft nützliche Hinweise und Antworten zu meinen Problemen bieten. Ein herzlichen Dank sende ich auch an die Probanden die sich die Zeit und Mühe zur Teilnahme an der Studie genommen haben.

Zudem möchte ich mich bei meiner Familie und meinen Freuden bedanken. Besonders gilt dieser Dank meinen Eltern, ohne die mein Studium schon im Vorhinein nicht möglich gewesen wäre. Letztlich möchte ich mich bei meiner Partnerin bedanken, die mir stets Motivation und moralische Unterstützung geboten hat.