|  |  |  |
| --- | --- | --- |
| C:\_Hochschule Heilbronn\6_AG MCI\4_Organisation\0_Corporate Design\1_Logo\UniTyLab_Logo.jpg |  | D:\Desktop\HHN Original (Vektor).emf |

Bachelor

Thesis

**Konzeption und Evaluation von Benutzerkonditionierung**

**in Virtual Reality**

|  |  |
| --- | --- |
| **Autor** | Robert Zlomke |
| **Studiengang** | Angewandte Informatik  Hochschule Heilbronn |
| **Matrikel-Nr.** | 193045 |
| **Abgabe** | 15.04.2021 |
| **Referent** | Prof. Dr.-Ing. Gerrit Meixner |
| **Korreferent:** | Philip Schäfer |

* 1. Danksagung
  2. Abstract

Virtuelle Umgebungen zu Fuß zu erkunden ist die realistischste und natürlichste Schnittstelle. Zeitgleich ist dies auch technisch und logistisch am anspruchsvollsten.

Allgemein wurden in bisherigen Arbeiten sich auf die Fortbewegung in übereinstimmenden oder leeren Umgebungen konzentriert. Weniger wurde darauf geachtet, wie sich eine Änderung der Nichtübereinstimmung zwischen der physischen Umgebung und ihrer virtuellen Darstellung auf die Benutzer auswirkt.

**Schlagwörter**: Wissenschaftliche Arbeit, Dokumentvorlage, Hochschule

## Inhaltsverzeichnis

[I. Danksagung 2](#_Toc66885967)

[II. Abstract 3](#_Toc66885968)

[1 Inhaltsverzeichnis 4](#_Toc66885969)

[2 Einleitung 6](#_Toc66885970)

[2.1 Unterpunkt 1 6](#_Toc66885971)

[3 Stand der Technik 7](#_Toc66885972)

[3.1 Konditionierung 7](#_Toc66885973)

[3.2 Virtual Reality 9](#_Toc66885974)

[3.2.1 Natural Walking 9](#_Toc66885975)

[3.2.2 Redirected Walking 10](#_Toc66885976)

[*3.2.2.1* Subtile kontinuierlicheRepositionierung 12](#_Toc66885977)

[3.2.2.2 Subtilediskrete Repositionierung 13](#_Toc66885978)

[3.2.2.3 Offene kontinuierliche Repositionierung 13](#_Toc66885979)

[3.2.2.4 Offene diskrete Repositionierung 13](#_Toc66885980)

[3.2.2.5 Offene kontinuierliche Neuausrichtung 14](#_Toc66885981)

[3.2.2.6 Subtile kontinuierliche Neuausrichtung 16](#_Toc66885982)

[3.2.2.7 Offene diskrete Neuausrichtung 17](#_Toc66885983)

[3.2.2.8 Subtile diskrete Neuausrichtung 18](#_Toc66885984)

[3.2.3 Walking-In-Place 19](#_Toc66885985)

[3.2.4 Abstahierte Schnittstelle des Gehens 19](#_Toc66885986)

[3.2.5 Laufbänder 19](#_Toc66885987)

[3.2.6 (Bewegungsverhalten) 20](#_Toc66885988)

[4 Problemstellung, Zielsetzung und Vorgehensweise 21](#_Toc66885989)

[4.1 Problemstellung 21](#_Toc66885990)

[4.2 Zielsetzung 21](#_Toc66885991)

[4.3 Vorgehensweise 21](#_Toc66885992)

[4.4 Technologien 21](#_Toc66885993)

[4.4.1 Unity 3D 21](#_Toc66885994)

[4.4.2 Oculus Quest 22](#_Toc66885995)

[4.4.3 Visual Studio 22](#_Toc66885996)

[4.4.4 Programmiersprache C# 23](#_Toc66885997)

[4.4.5 Git Versionskontrolle 23](#_Toc66885998)

[5 Umsetzung 24](#_Toc66885999)

[5.1 Einbinden der Oculus Quest in Unity 3D 24](#_Toc66886000)

[5.2 Entwicklung des Malus 25](#_Toc66886001)

[5.3 Entwicklung der Szenarien 25](#_Toc66886002)

[5.4 Datenerfassung 27](#_Toc66886003)

[5.5 Evaluation 30](#_Toc66886004)

[6 Zusammenfassung, Bewertung und Ausblick 31](#_Toc66886005)

[7 Anhang 32](#_Toc66886006)

[7.1 Abkürzungsverzeichnis 32](#_Toc66886007)

[7.2 Abbildungsverzeichnis 33](#_Toc66886008)

[7.3 Literaturverzeichnis 34](#_Toc66886009)

[7.4 Eidesstattliche Erklärung 37](#_Toc66886010)

## Einleitung

### Unterpunkt 1

Die Erkundung virtueller Umgebungen (VE) durch Gehen ist die realistischste und natürlichste

Schnittstelle für VR-Systeme (Virtual Reality), in denen Benutzer anthropomorphe Charaktere

verkörpern [1]. Es ist auch die technisch anspruchsvollste Schnittstelle aufgrund von

Hardwareproblemen (Tracking-Systeme, zu tragende Instrumente) und Umweltproblemen

(Ungleichheit zwischen den Ausmaßen des VE und dem verfügbaren physischen Raum,

Vorhandensein von physikalischen Elementen, die in der Simulation nicht vorhanden sind).

Obwohl verschiedene Lösungen für diese Probleme vorgestellt wurden, sind diese

nicht immer an häusliche Umgebungen wie Wohnungen, Büros und andere Orte anpassbar, die

nicht explizit als VR-Labor konzipiert sind. [2]

Wir sind der Meinung, dass die inländische VR ein grundlegendes Sprungbrett für ihre breite Akzeptanz darstellt. Diese Umgebungen bieten jedoch unterschiedliche andere Herausforderungen im Vergleich zu VR-Labors: in Häusern, Möbeln schränkt den navigierbaren Raum ein. Stattdessen sind VR-Labore normalerweise groß und leere Räume. Das Einbeziehen von Objekten aus der unmittelbaren physischen Umgebung des Benutzers in die Simulation durch Ersetzen durch nicht übereinstimmende virtuelle Gegenstücke bietet das Potenzial, ein überzeugendes VR-Erlebnis zu bieten. Wenn sich frühere Arbeiten jedoch auf die Bewegung in übereinstimmenden Umgebungen [3] oder leeren Umgebungen konzentrierten [4], wurde weniger darauf geachtet, ob sich eine Änderung der Nichtübereinstimmung zwischen der physischen Umgebung und ihrer virtuellen Darstellung auf das Bewegungsverhalten der Benutzer auswirkt. Wenn ja, gibt es bestimmte Änderungen, die wirksamer sind als andere?

## Stand der Technik

### Konditionierung

Die Pawlowsche Konditionierung ist ein wertvolles Labor Modell, um den Erwerb, die Ausprägung, die Generalisierung und die Hemmung von bedrohungsbezogenem Verhalten bei verschiedenen Spezies zu untersuchen4-6. Zwei pawlowsche Konditionierungsparadigmen haben als Arbeitsgrundlage gedient, um die Mechanismen zu verstehen, die der erlernten Bedrohung und der Rolle der kontextuellen Prozesse zugrunde liegen: die operante Konditionierung und die Kontextkonditionierung. Bei der aversiven Pawlowschen Konditionierung lernt eine Versuchsperson, dass ein neutraler Stimulus in der Umgebung (konditionierter Stimulus, CS, z.B. ein Ton oder ein Bild) das Auftreten eines aversiven Ergebnisses (unkonditionierter Stimulus, US, z.B. ein elektrischer Schock) vorhersagt. Sobald die Versuchsperson die CS-US-Assoziation erlernt hat, löst die Präsentation des CS allein bedrohungsbezogene Abwehrreaktionen aus, wie z. B. Erstarren und erhöhte sympathische Erregung. Die Methode der operanten Konditionierung ist leicht von Versuchstieren auf den Menschen übertragbar, wo einfache sensorische Reize wie Töne und Lichter oder komplexere Reize wie Gesichter und Objektkategorien als CSs dienen können. Die pawlowsche Bedrohungskonditionierung dient nach wie vor als wertvolles und nachvollziehbares Modell für Psychopathologien, die durch akute oder phasische Bedrohungsreaktionen gekennzeichnet sind, die durch diskrete Reize ausgelöst werden, wie z.B. Zwangsstörungen, Phobien und Symptomcluster der PTBS, die Abwehrreaktionen auf Trauma-Erinnerungen beinhalten [siehe Ref. 4 für eine Übersicht].

Bei der pawlowschen Kontextkonditionierung wird der US ausgelöst, während sich die Versuchsperson in einer bestimmten Umgebung (d.h. in der Konditionierungskammer) befindet, aber nicht durch einen diskreten CS signalisiert wird. Bemerkenswert ist, dass der Begriff "Kontext" in der Forschung zum assoziativen Lernen sehr weit gefasst werden kann und von den sensorischen Details der Umgebung (Anblicke, Gerüche, Geräusche usw.) über innere Zustände bis hin zu Zeitwahrnehmungen reicht7. Die Kontextkonditionierung bietet ein geeigneteres Modell für Psychopathologien, die durch anhaltende oder "freischwebende" Angst gekennzeichnet sind, wenn es keinen eindeutigen bedrohungsauslösenden Stimulus in der Umgebung gibt8, wie z.B. bei der Generalisierten Angststörung oder Symptomclustern der PTBS, die hyperarousal in Abwesenheit von bedrohungsauslösenden Hinweisen beinhalten.

Der Kontext interagiert auch mit der Cue-Konditionierung, um die Ausprägung des assoziativen Speichers CS-US zu modulieren. Zum Beispiel könnte ein CS, der den US in einem Kontext vorhersagt, dann ohne den US in einem anderen Kontext präsentiert werden, d.h. Extinktion. Das Testen des CS im ersten (d.h. Akquisitions-), zweiten (d.h. Extinktionskontext) oder einem neuen Kontext kann zeigen, welche CS-Assoziation die Testperson abruft. Die Forschung der neurologischen Verhaltensweisen hat gezeigt, wie der Kontext die Ausprägung von assoziativen CS-US-Erinnerungen beeinflusst7, 9; z.B. können Cue-ausgelöste Reaktionen bei einem späteren Test im Extinktionskontext reduziert werden oder im Extinktionskontext (spontane Erholung) oder im Akquisitions- oder neuen Kontext wieder auftauchen. Insbesondere konzentriert sich der vorliegende Artikel weitgehend auf Paradigmen der Kontextkonditionierung, bei denen nur die räumlich-zeitliche Umgebung den US in Abwesenheit eines diskreten bedrohungsauslösenden Hinweises signalisiert. In der Literatur über den Menschen wurde der Begriff Kontextkonditionierung oft fälschlicherweise verwendet, um Paradigmen zu beschreiben, bei denen der Kontext die Konditionierung des Cues moduliert.

Trotz der zunehmenden Anzahl von Studien zu kontextuellen Prozessen bei Versuchstieren gibt es bis heute nur wenige systematische Untersuchungen zur Kontextkonditionierung beim Menschen (Tabelle 1). Dies ist größtenteils auf die Herausforderung zurückzuführen, einen räumlich-zeitlichen Kontext zu schaffen, der mit einem US gekoppelt werden kann. Bei Versuchstieren wird der Kontext manipuliert, indem der physische Ort des Tieres (z. B. eine andere Konditionierungsbox) oder der Raum (z. B. Veränderung der Wandfarben, der Bodenbeschaffenheit und/oder des Geruchs) verändert wird. Das offensichtlichste Analogon zur Konditionierungskammer für menschliche Probanden ist der Laborversuchsraum. Obwohl der Raum der vorherrschende Kontext ist, ist er oft nicht das Ziel der Kontextkonditionierung. Mit anderen Worten, anders als in der Labor Tierforschung beabsichtigt der Experimentator in Studien zur Konditionierung von Menschen im Allgemeinen nicht, dass die Versuchsperson eine Assoziation zwischen der physischen Umgebung des Testraums (bestehend aus bestimmter Beleuchtung, Möbeln, Computern, Forschungsgeräten usw.) und den USA herstellt.

Die Technologie der virtuellen Realität kann genutzt werden, um kontextuelle Stimuli zu optimieren und die Chance zu erhöhen, dass die Kontextkonditionierung kontextuelle Lernsysteme im Gehirn aktiviert. In erster Linie kann VR ein starkes Gefühl der "Präsenz" hervorrufen, bei dem Menschen denken, sich verhalten, fühlen und das Gefühl haben, sich im virtuellen Raum und nicht in der realen Welt zu befinden34. Ein entscheidendes Merkmal von VR ist das Headset (oder Head Mounted Display), das dazu beiträgt, den sensorischen Input von der äußeren Umgebung zu reduzieren und zusätzlich ermöglicht, dass Kopfbewegungen in visuelle Rotationen übersetzt werden, wodurch eine Bewegungsparallaxe für den Teilnehmer entsteht, wodurch das Gefühl verstärkt wird, in eine virtuelle Umgebung vertieft zu sein und vom gegenwärtigen physischen Ort entfernt zu sein - dies wird als "Immersion" bezeichnet. Bis heute haben nur eine Handvoll Studien zur Kontextkonditionierung VR-Headsets eingesetzt. VR war bisher nicht weit verbreitet, da frühe Systeme kostspielig waren und in einigen Fällen viel Platz benötigten, z. B. [ref. 35]. Mit der kommerziellen Freigabe von VR-Headsets ist die Möglichkeit, die Forschung zur Kontextkonditionierung zu erweitern, nun in Reichweite von fast jedem Forschungslabor, das bereits menschliche Konditionierung mit psychophysiologischer Ausrüstung untersucht. Zugängliche und validierte Protokolle für kommerziell erhältliche VR-Systeme werden sich zunehmend als nützlich für die Untersuchung von Kontextkonditionierung beim Menschen erweisen. [5]

### Virtual Reality

Die virtuelle Realität ermöglicht es dem Benutzer, sich aus der realen Umgebung herauszuziehen und in eine virtuelle Umgebung einzutauchen, in der er beobachten, interagieren, einen Dialog führen und sogar neue Fähigkeiten erlernen kann. Diese Flexibilität hat in verschiedenen Bereichen wie Robotik, Stadtplanung, Industrie, Kunst und Bildung usw. neue Horizonte eröffnet. In den letzten zehn Jahren hat der technologische Fortschritt die Benutzererfahrung in der virtuellen Welt deutlich verbessert. Trotz der Tatsache, dass die Fortbewegung nicht das primäre Ziel von Virtual-Reality-Anwendungen ist, wird sie als eine der wichtigsten Komponenten der Interaktion in Virtual-Reality-Erfahrungen angesehen, da sie für die Erkundung der virtuellen Umgebung unverzichtbar und entscheidend ist (Lee et al., 2018). Das Aufkommen neuer Virtual-Reality-Geräte wie die kommerziellen Veröffentlichungen der HTC Vive und der Oculus Rift/Quest, markierte den Beginn dessen, was einige Forscher als "eine neue Ära in der Geschichte der virtuellen Realität" bezeichnen. In dieser neuen Ära haben sich gängige Fortbewegungsmittel wie Standard-Game-Controller und Joysticks als nicht angepasst erwiesen und verursachen Desorientierung in immersiven virtuellen Umgebungen (Lathrop & Kaiser, 2002; Ruddle & Lessels, 2006). Es ist also nicht klar, wie man effektiv und kostengünstig eine virtuelle Umgebung erkunden kann. Auf diese Weise wurde die Suche nach einer effizienten Methode zur Bewegung des Spielers in der virtuellen Umgebung zu einer der wichtigsten Herausforderungen und löste ein erneutes Interesse an den Lokomotionstechniken in virtuellen Umgebungen aus. Die Lokomotion in der virtuellen Realität ermöglicht es dem Benutzer, sich in einer unendlich großen virtuellen Welt zu bewegen, sie kann als selbstangetriebene Bewegung in der virtuellen Umgebung definiert werden, während er in einer raumgroßen realen Umgebung eingeschlossen bleibt (Hale & Stanney, 2014). Locomotion ist eine wesentliche Form der Mensch-Computer-Interaktion. In der virtuellen Realität hat sie einen direkten Einfluss auf viele Aspekte der Benutzererfahrung, wie z. B. Spaß, Frustration, Müdigkeit, Bewegungskrankheit und Präsenz. Wenn die virtuelle Umgebung die gleiche Größe wie die physische Umgebung des Benutzers hat und das System ihm erlaubt, sich frei zu bewegen, kann der Benutzer natürlich gehen, um die Fortbewegung in der virtuellen Welt zu steuern. Wenn die virtuelle Umgebung jedoch größer ist als die physische Umgebung, muss eine alternative Lokomotionsmethode verwendet werden. Infolgedessen sind neue Lokomotionstechniken entstanden. Die Wahl der Lokomotionstechnik ist in Bezug auf das Verständnis, die Entwicklung, den Einsatz und das Experimentieren noch nicht ausgereift. So stellt die Suche nach einer effizienten Lokomotionstechnik, die den Präsenz nicht verändert, keine Reisekrankheit verursacht und Ermüdung vermeidet, eine echte Herausforderung dar, die heute Gegenstand zahlreicher Forschungsstudien ist (Bowman et al., 2004; Kitson et al., 2017a). [6]

In den nachfolgenden Unterkapiteln werden die unterschiedlichen Schnittstellen und dessen Stand der Technik genauer beschrieben.

#### Natural Walking

Als High-Fidelity-Techniken werden solche bezeichnet, die den Akt des Gehens am genauesten reproduzieren. Besonders in den Szenarien, in denen der Benutzer durch einen virtuellen Charakter verkörpert wird, ist das echte Gehen in einem physischen Raum die natürlichste Wahl. Natürliches Gehen bezieht sich in der Regel auf Szenarien, in denen sich die virtuelle Umgebung vollständig innerhalb des physischen Raums befindet, in dem das VR-System eingesetzt wird. Unter diesen Umständen entspricht ein Schritt in der realen Welt einem Schritt von gleicher Länge in der virtuellen Umgebung [2]. Es ist jedoch mit verschiedenen Techniken ebenso möglich, wenn der physische Raum eine kleinere Größe hat als die virtuelle Umgebung. Auf diese Thematik wird genauer im nachfolgenden Kapitel 3.2.2 Redirected Walking eingegangen.

Diese Fortbewegungsschnittstelle erzeugt das größte Gefühl der Präsenz und löst am wenigsten okulomotorische (Augenbewegung) Unbehagen aus. Die Fortbewegung wird als natürlich, leicht und unkompliziert bewertet [7]. Der Vergleich findet dabei zwischen den weniger realitätsnahen Fortbewegungstechniken wie Walking-in-Place [1] oder virtuellem Fliegen [8] statt. Es hat sich auch gezeigt, dass natürliches Gehen durch die höhere und feinmotorischere Beweglichkeit der Personen zu weniger Kollisionen mit Hindernissen führt [9]. Ebenso erfordert eine natürliche Schnittstelle weniger Trainingszeit, und dies unabhängig von der Schwierigkeit der virtuellen Umgebung [10]. Die Bahnkurven [11] und das Verhalten in der Navigation [12] stimmen beim natürlichen Gehen gleichermaßen überein mit der Fortbewegung in der realen Welt zum Großteil überein. [2]

#### Redirected Walking

In Szenarien, in denen Benutzer virtuelle Umgebungen erkunden müssen, die größer als der verfügbare physische Raum sind, stößt die natürliche Schnittstelle des Gehens schnell an seine Grenzen und ist somit im geringeren Maße geeignet. Beim sogenannten "Redirected Walking" handelt es sich um verschiedene Techniken, die es den Benutzern ermöglichen, virtuelle Räume, die größer sind als ihre physische Umgebung, physisch zu erkunden. Sie verändern, wie der physische Pfad der realen Welt in der virtuellen Welt wahrgenommen wird. Dies geschieht durch die gezielte und im besten Falle vom Benutzer unbemerkten Manipulation der Route. So wird es dem Benutzer ermöglicht, sich durch viel größere virtuelle Umgebungen natürlich bewegen zu können [13]. [2]

Suma et al. Stellten eine Taxonomie von Umleitungstechniken vor [14], die auf eine frühere Arbeit von Steinicke et al. basieren und diese erweiterten (siehe Abb. 1: Eine Taxonomie für Umleitungstechniken) [15] [2]. Ebenso wurden von Steinicke et al. Analysen der menschlichen Sensibilität gegenüber diesen Manipulationen vorgestellt [16] [17] [18]. Die Klassifizierung der Techniken basiert darauf, ob die Beeinflussung auf die Translation, Rotation der Krümmung stattfindet. Ebenso entscheidend ist das Ausmaß der Veränderung (diskret oder kontinuierlich) und ob sie subtil oder offenkundig ist. Mit der neuen Taxonomie wird ermöglicht, eine oder mehrere Umleitungsmetaphern basierend auf einer Vielzahl von Kriterien auszuwählen und anzuwenden, die für das Design der virtuellen Umgebung relevant sind [14] [2]. Auf diese Punkte wird in den nachfolgenden Unterkapiteln genauer eingegangen und genauer erläutert.



Abb. 1: Eine Taxonomie für Umleitungstechniken

Andere Arbeiten konzentrieren sich auf die Kombination von passiver Haptik [7] mit umgeleitetem Gehen, so dass Benutzer zu Proxy-Requisiten in der realen Welt umgeleitet werden, die sie berühren können, wenn sie sich virtuellen Objekten nähern [13, 25]. Das Hauptproblem bei diesen Ansätzen ist, dass ein großer VR-Laboraufbau von mindestens 50m×50m Bodenfläche erforderlich ist, um nicht wahrnehmbare Manipulationen zu ermöglichen [21, 24]. Peck et al. haben vorgeschlagen, visuelle Distraktoren [20] zu verwenden, die den Benutzer ablenken und somit stärkere Manipulationen der virtuellen Welt ermöglichen, ohne dass er sie bemerkt. Bei diesem Ansatz werden jedoch kleine bewegliche Objekte in die virtuelle Welt injiziert, die den Benutzer absichtlich ablenken. Aus diesem Grund sind virtuelle Distraktoren für die professionelle Architekturexploration nicht realisierbar. [18]

##### Subtile kontinuierliche Repositionierung

Eine kontinuierliche Repositionierung kann auf subtile Weise angewendet werden, indem Translationsverstärkungen auf die physische Fortbewegung des Benutzers angewendet werden. Dadurch wird die Gehbewegungen effektiv skaliert, um größere Entfernungen in der virtuellen Umgebung zurückzulegen [19]. Diese Methode kann erweitert werden, indem die beabsichtigte Bewegungsrichtung des Benutzers geschätzt und die Translationen nur in diese Richtung skaliert werden, wodurch die Übertreibung des oszillierenden Kopfwippens und des Schwankens von Gehbewegungen reduziert, wird [20].

Diese Technik bleibt subtil, solange die angewandten Verstärkungen klein genug sind, um eine Erkennung zu vermeiden. Steinicke et al. führten eine psychophysikalische Studie zu Erkennungsschwellen durch. Diese führte zur Erkenntnis das gelaufene Distanzen in der realen Welt um 14 Prozent herunterskaliert und um 26 Prozent hochskaliert werden können, wenn sie auf virtuelle Bewegungen abgebildet werden. Diese Ungleichgewichtung deckt sich mit früheren Erkenntnissen, dass Benutzer dazu neigen, Distanzen zu unterschätzen [15]. [21] [14]

##### Subtile diskrete Repositionierung

Angesichts der abrupten Verschiebung, die für eine diskrete Neupositionierung erforderlich ist, scheint es schwierig, diese Technik auf subtile Weise anzuwenden. Forschungen haben jedoch herausgefunden, dass illusorische Bewegung am Rande des Blickfelds, der Peripherie, des Benutzers verwendet werden können, um abrupte Translationen in der Umgebung zu tarnen [22]. Diese kleinen diskreten Aktualisierungen können periodisch wiederholt werden, um den Bereich der unbemerkten skalierten Gehgewinne zu vergrößern. [14] Die Distanzwahrnehmung wird also durch Selbstbewegungsillusionen kompensiert, ohne vom Benutzer selbst erkannt zu werden [2].

##### Offene kontinuierliche Repositionierung

Eine Repositionierung kann erreicht werden, indem die virtuelle Umgebung kontinuierlich um die Position des Benutzers verschoben wird. Dadurch kann der Benutzer Bereiche in der virtuellen Umgebung aufsuchen, die zuvor innerhalb der Grenzen des physischen Arbeitsbereichs nicht zugänglich waren. Dies kann desorientierend wirken, wenn die virtuelle Welt unerwartet übersetzt wird, und kann die virtuelle Umgebung instabil erscheinen lassen. Diese Störung kann gemildert werden, indem die Übersetzung mit bekannten Metaphern als Navigationshilfen gekoppelt wird, die mit der Bewegung assoziiert werden, wie z. B. Aufzüge, Rolltreppen, oder Fahrzeuge. [14]

##### Offene diskrete Repositionierung

Diskrete Techniken der Repositionierung können durch sofortige Verschiebung erreicht werden, wodurch der Benutzer effektiv an eine neue Position im virtuellen Raum teleportiert wird. Diese Technik ist potenziell desorientierend, wenn der Benutzer nicht erwartet, dass die virtuelle Position manipuliert wird. Um dieses Problem zu entschärfen, haben Forscher sich Forscher für das Konzept der Portale entschieden. [14]

Portale sind ein gängiges Konzept in der Science-Fiction und Fantasy. In der Fiktion ist ein Portal ein magischer oder technologischer Zugang, der entfernte Orte miteinander verbindet, sei es durch Zeit oder vor allem durch Raum. Sie können in zwei Formen auftreten: Entweder muss eine Person durch den Rahmen eines Objekts (ein Spiegel, ein Schrank, ein Tor usw.) treten, das als Portal dient, oder, wenn das Portal allein ist, erscheint es in einer "magischen" Form, z. B. als Wirbel aus Energie. [23]

Darüber hinaus bietet das Konzept der virtuellen Portale die Möglichkeit, große Entfernungen innerhalb von Architekturmodellen intuitiv zurückzulegen [18]. Benutzer berichteten über ein höheres Maß an Präsenz und eine gesteigerte Fähigkeit, Distanzen zu schätzen, wenn Portale verwendet wurden, um von einer virtuellen Nachbildung der realen Umgebung in eine unbekannte Umgebung zu teleportieren, verglichen mit dem unmittelbaren Betreten der ungewohnten Umgebung. Dies wurde dadurch von den Benutzern bevorzugt, da die Metapher des Wurmlochs die Vorstellung unterstützt, in eine andere Welt versetzt zu werden [23]. [14]

##### Offene kontinuierliche Neuausrichtung

Sobald die Probanden die Grenzen des Tracking Systems erreichen, wird eine Technik des Zurücksetzens der Probanden eingesetzt. Beim Zurücksetzen wird der optische Fluss so manipuliert, dass der Benutzer sich von einem physischen Hindernis, wie z. B. einer Wand, wegbewegen kann, während er ein kontinuierliches Gefühl für seine Position im virtuellen Raum erhält. Nach dem Zurücksetzen bewegt sich der Benutzer auf demselben virtuellen Weg, auf dem er sich zuvor befunden hat. Bei der Freeze-Backup-Methode erhält der Benutzer mehr Raum für die virtuelle Erkundung, indem er Schritte rückwärts macht, während er an einer festen Position in der virtuellen Umgebung eingefroren wird. Unter den anderen beiden Methoden, Freeze-Turn und 2:1-Turn, überwindet der Benutzer physische Hindernisse, indem sie sich physisch um 180 Grad drehen und ihre Position im virtuellen Raum vor und nach der Drehung beibehalten. In der 2:1-Drehbedingung wird die Verstärkung der Drehung verdoppelt, so dass eine 180er-Drehung im physischen Raum einer 360 Drehung im virtuellen Raum. [24] Auf die Methode Freeze-Turn-Reset wird im Kapitel 3.2.2.7 genauer eingegangen.

Als Beispiel dient die nachfolgende Darstellung (siehe Abb. 2: Techniken des Zurücksetzens). Der Weg, den eine Person in der virtuellen Umgebung wahrnimmt, ist in (a) dargestellt, während der Weg im physischen Raum, den die Person unter den verschiedenen Rücksetzmethoden nimmt, in (b) und (c) gezeigt wird. In diesem Beispiel betrachtet eine Person an der Position (0,0) im physischen Raum die virtuelle Umgebung an der Position (0,0). In (a) geht die Person in der virtuellen Umgebung vorwärts, wo sie durch ein Signal bei (4,0) gewarnt wird, das anzeigt, dass sie sich den Verfolgungsgrenzen nähert und ihre Position im physischen Raum zurücksetzen muss. Die Person läuft dann in der virtuellen Umgebung weiter bis (12,0). Die entsprechenden Wege in der physischen Umgebung für die drei Rücksetzmethoden sind in (b) und (c) dargestellt. Rote Pfeile zeigen die physische Bewegung während einer Rückstellung an. [24]



Abb. 2: Techniken des Zurücksetzens

Da die Erteilung expliziter Anweisungen an den Benutzer die Anwesenheit unterbrechen kann, wurde vorgeschlagen, visuelle Ablenkungen zu verwenden, um Kopfdrehungen während der Intervention hervorzurufen und so die Möglichkeit zu bieten, Rotationsgewinne anzuwenden.

Die optimalen Distraktoren sind VE-abhängig und sollten so gestaltet werden, dass sie für die VE so natürlich wie möglich sind. Mögliche Beispielimplementierungen von Distraktoren sind ein virtuelles Haus zu erkunden und einen Hund vorbeilaufen zu lassen oder durch ein virtuelles Kunstmuseum zu gehen und sich von einem Dozenten in eine neue Richtung weisen zu lassen. [25]

##### Subtile kontinuierliche Neuausrichtung

Die Technik „Redirected Walking“ funktioniert, indem die virtuelle Szene interaktiv um den Benutzer gedreht wird, so dass der Benutzer dazu gebracht wird, kontinuierlich auf die am weitesten entfernte "Wand" des Tracker-Bereichs zuzugehen. Der Benutzer bemerkt diese Drehung nicht, da der Algorithmus die Grenzen der menschlichen Wahrnehmungsmechanismen für die Erfassung von Position, Orientierung und Bewegung ausnutzt. Bei einem Tracker mit begrenzter Größe gibt es einen Kompromiss: Je mehr Rotationsverzerrung (was dazu führt, dass der Benutzer in engeren Bögen läuft), desto größer ist die virtuelle Umgebung, die wir darstellen können. Je mehr Rotationsverzerrung, desto wahrscheinlicher ist es jedoch, dass der Benutzer die Rotation erkennt. Um Redirected Walking für Tracker mit begrenzter Fläche nutzbar zu machen zu können, wird der Benutzer gezwungen, sich an strategisch platzierten Wegpunkten in der VE umzusehen. Während der Benutzer sich dreht, um sich umzusehen, kann mit einer größeren Rotationsverzerrung gearbeitet werden. Die VE wird so gedreht, dass eine Richtung, die zuvor außerhalb des Tracker-Bereichs lag, nun sicher innerhalb des verfolgten Bereichs liegt. Obwohl die Notwendigkeit von Wegpunkten eine große Einschränkung für die VE darstellt, gibt es einige Aufgabenstellungen, die sich trotzdem auf natürlichen Weg für Wegpunkte eignen.

(Hintergrund Theorie: Der Mensch verlässt sich in erster Linie auf vestibuläre, visuelle und auditive Hinweise für Gleichgewicht und Orientierung3. Der Mensch nutzt diese Sinne auch, um festzustellen, ob er sich selbst bewegt (Eigenbewegung) oder ob sich die Objekte um ihn herumbewegen (Fremdbewegung). Frühere Forschungsergebnisse deuten darauf hin, dass die Konsistenz mehrerer Hinweise die Wahrscheinlichkeit erhöht, dass der Benutzer die Drehung als Eigenbewegung und nicht als Fremdbewegung wahrnimmt4. Wenn Redirected Walking die Konsistenz zwischen visuellen, auditiven und vestibulären Hinweisen aufrechterhält, sollte der Benutzer nicht das Gefühl haben, dass sich die Welt um ihn herum willkürlich bewegt. Das Ziel ist es, die Wahrscheinlichkeit zu maximieren, dass alle vom Benutzer wahrgenommenen Bewegungen Eigenbewegungen sind.)

Die nachfolgende Abbildung (siehe Abb. 3: Redirected Walking) zeigt die Draufsicht auf den Weg des Benutzers in der virtuellen Umgebung (oben in blau) und im Labor (unten in rot). Hier ist zu erkennen, wie der Benutzer in einem Zickzack-Muster durch die VE geht, während er im physischen Raum hin und her läuft. [13]



Abb. 3: Redirected Walking

Die Erkennbarkeit von Rotationsverstärkungen wurde sowohl im Kontext von Kopfdrehungen als auch von Ganzkörperdrehungen untersucht. Eine durchgeführte umfassende Studie ergab, dass Benutzer physisch um ca. 49 % mehr bzw. 20 % weniger als die wahrgenommene virtuelle Rotation gedreht werden können, ohne den Unterschied zu bemerken, wenn sich die Szene mit und nicht gegen die Kopfdrehung bewegt. Experimentelle Ergebnisse haben jedoch gezeigt, dass die unmerkliche Umlenkung eines Benutzers entlang eines Kreisbogens durch Krümmungsverstärkung einen sehr großen Arbeitsbereich mit einem Radius von mindestens 22 Metern erfordert. [21] [14]

##### Offene diskrete Neuausrichtung

Neben den offenen Methoden der kontinuierlichen Neuorientierung von Williams et al [24] existiert auch eine diskrete Technik „Freeze-and-Turn“ zum zurücksetzen des Benutzers [14].

Wenn das Tracking-Gerät bei dieser Methode feststellt, dass die Testperson eine Grenze erreicht hat, zeigt der Computer dem Teilnehmer an, dass er zurücksetzen muss, indem er sich umdreht. Das Display des HMD wird eingefroren, wodurch die Position und der Drehrichtungswinkel des Teilnehmers im virtuellen Raum eingefroren werden, und der Teilnehmer dreht sich um 180 Grad. Das Display wird wieder aufgetaut, das Tracking wird aktualisiert, und der Proband kann seine Route fortsetzen. [24] Als Beispiel zeigt Abbildung 2c) (siehe Abb. 2: Techniken des Zurücksetzens) das Zurücksetzen mit der „Freeze-and-Turn“ Methode.

Während die Diskontinuität, die durch das Einfrieren und Aufheben der Bewegungsverfolgung eingeführt wird, für den Benutzer offensichtlich ist, bleibt das Zurücksetzen als eine Notfall-Technik nützlich, um zu verhindern, dass der Benutzer den Arbeitsbereich verlässt. [14]

##### Subtile diskrete Neuausrichtung

Für eine subtile und diskrete Neuorientierung wurde die Technik "Change Blindness Redirection" von Suma et al [26] entwickelt und basiert dem Phänomen der Veränderungsblindheit. Diese kann beobachtet werden kann, wenn Benutzer Änderungen an einer visuellen Szene nicht bemerken, die außerhalb ihres Gesichtsfeldes stattfinden [14].

Veränderungsblindheit ist damit zu erklären, dass sich Menschen auf den ersten Eindruck einer Umgebung verlassen, die ursprüngliche Repräsentation überschrieben wird und widersprüchliche Merkmale kombiniert werden. Menschen können zwei Überzeugungen festhalten, ohne zu bemerken, dass sie grundsätzlich widersprüchlich sind. Die geänderte Szene überschreibt bzw. maskiert also die ursprüngliche Szene, was in der Literatur als am häufigsten angenommenen Grunde für die Veränderungsblindheit ist. [27]

Im Gegensatz zu den kontinuierlichen Neuorientierungen, wird die Position oder Ausrichtung von architektonischen Merkmalen, insbesondere Türen, in einer virtuellen Szene zur Laufzeit sofort geändert. In nachfolgender Abbildung (siehe Abb. 4: Diskreter Szenenwechsel) wird ein Beispiel eines Szenenwechsels veranschaulicht. Die Tür und der angrenzende Korridor im Winkel von 90 Grad gedreht werden, sobald der Benutzer wegschaut. Benutzer, die den Raum verlassen, gehen den Korridor in einer anderen Richtung als beim Betreten hinunter. [26]

Im Gegensatz zu früheren Umleitungstechniken führt dieser Ansatz, der auf einem dynamischen Umgebungsmodell basiert, weder zu visuellen Konflikten bei der Manipulation der Zuordnung zwischen physischen und virtuellen Bewegungen noch erfordert er eine Unterbrechung der Präsenz, um den Benutzer anzuhalten und explizit neu auszurichten. Die Ergebnisse von Suma et al zeigten, dass die Teilnehmer in der Lage waren, ihre räumliche Ausrichtung innerhalb der virtuellen Welt beizubehalten. Nur einer von 77 Teilnehmern in beiden Studien bemerkte definitiv, dass ein Szenenwechsel stattgefunden hatte, was darauf hindeutet, dass die Umleitung der Änderungsblindheit eine bemerkenswert überzeugende Illusion darstellt.

Jedoch sind diese Art von Techniken weitgehend auf Innenumgebungen mit manipulierbaren Merkmalen wie z.B. Türen in engen Räumen oder Gängen beschränkt und wären in spärlichen, offenen Umgebungen wie Außenszenen oft geometrisch nicht anwendbar. [14] [26]



Abb. 4: Diskreter Szenenwechsel

#### Walking-In-Place

Walking-in-Place (WIP) ist eine Kategorie von Techniken, bei denen, wie der Name schon sagt, der stationäre Gang des Benutzers auf die Bewegung in der VE abgebildet wird [33, 42]. Diese Methode imitiert das Gehen. Sie ermutigt die Benutzer, ihren ganzen Körper so weit wie möglich in eine realistische Gehbewegung zu versetzen, ohne sich jedoch tatsächlich vorwärts zu bewegen [6].

Eine umgeleitete Version von WIP verhindert, dass der Benutzer in einem dreiseitigen CAVE-Setup auf die fehlende Rückwand schaut [23]. WIP wird typischerweise über Sensoren oder Infrarotmarker implementiert, aber es wurde auch auf einem Wii Balance Board [47] und durch die Erkennung von Fußschatten auf dem Boden demonstriert [51]. Es wurde berichtet, dass WIP weniger realistisch ist als natürliches Gehen und zu Problemen im Zusammenhang mit Latenz und mangelnder Geschmeidigkeit während der Bewegung führt [6]. [2]

##### User-Body-zentrierte Techniken

Bei neigungsbasierten Fortbewegungstechniken wird der ganze Körper oder Oberkörper zur Bewegung eingesetzt. Das Gehen wird durch das Navigieren in Richtung des Körperschwerpunkts, was durch das Neigen des ganzen Körpers oder nur von Teilen davon in die gewünschte Richtung erreicht wird. Die Neigung selbst kann durch Standard-Tracking-Technologie oder mit speziellen Geräten wie Kraftmessplatten, z. B. dem Wii Balance Board [11, 40] oder speziellen, wie dem von Marchal et al. Konstruierten Prototypen [27], erfasst werden. [28]

Eine weitere gängige lehnungsbasierte Fortbewegungsmethode, die die Kopfbewegungserfassung nutzen kann, ist das Walkingin- Place. Diese Methode imitiert das Gehen. Sie ermutigt die Benutzer, ihren ganzen Körper so weit wie möglich in eine realistische Gehbewegung zu versetzen, ohne sich jedoch tatsächlich vorwärts zu bewegen. Viele Studien wurden durchgeführt, um diese Methode mit anderen Lokomotionsmethoden zu vergleichen (Bozgeyikli et al., 2016b, 2016a; McCullough et al., 2015) oder sogar neue Varianten dieser Methode unter Verwendung verschiedener Gesteneingaben vorzuschlagen (Berger & Wolf, 2018; Nilsson et al., 2013). Die Ergebnisse der Studien, die diese Fortbewegungsmethode mit anderen Methoden wie Armschwingen, Zeigen und Teleportieren, lehnungsbasierte Methoden in sitzender Position, Joystick usw. verglichen, zeigten, dass Walking-in-Place die meisten dieser Techniken übertrifft. Walking-in-Place kann den Platzbedarf für die Bewegung durch die virtuelle Umgebung reduzieren, mit weniger Simulatorkrankheit. Allerdings weist es einige Schwachstellen und Grenzen auf. Die erste Einschränkung bezieht sich auf die Tatsache, dass in vielen Fällen viele Sensoren am Körper des Benutzers angebracht werden müssen. Einige Studien versuchten, diese Probleme zu lösen, indem sie andere Parameter zur Erkennung des Gehens verwendeten, wie z. B. die Position und Ausrichtung des Head Mounted Displays (siehe Abbildung 4). Außerdem ist der Immersionsgrad des Benutzers noch gering (Lee et al., 2018; Nilsson et al., 2013). Darüber hinaus ist die stehende Position nicht immer geeignet, um die virtuelle Umgebung zu erkunden. Die zweite Einschränkung ist, dass diese Technik möglicherweise nicht geeignet ist, um eine signifikant große virtuelle Umgebung zu erkunden, da dies erhebliche Zeit und körperliche Energie erfordern würde. So könnte Walking-in-Place zur Erkundung eines Ortes oder einer Stadt anstrengend sein. Der Schwachpunkt ist, dass der Immersionsgrad der Benutzer durch die Tatsache, dass die Benutzer das Gehen imitieren, ohne tatsächlich zu gehen, und durch die Schwierigkeit der erforderlichen Gesten reduziert wird. Zielasko und Kollegen (2016) nutzten die im HMD eingebauten Inertialsensoren, um die Position und die Drehung des sitzenden Benutzers zu bestimmen. Der Teilnehmer musste nach oben und nach unten schauen, um vorwärts und rückwärts zu gehen. Die Ergebnisse zeigten, dass der Joystick die kopfbewegungsbasierte Fortbewegungstechnik übertraf und besser abschnitt. Es ist erwähnenswert, dass die Verwendung der Kopfbewegung zum Navigieren und Drehen in der virtuellen Umgebung die Verwendung des Kopfes für andere Interaktionen opfert, wie z. B. das Umschauen in 360°, ohne sich tatsächlich zu drehen. McMahan und Kollegen (2012) schlugen verschiedene Ansätze der gleichen Fortbewegungstechnik in einer Höhlendisplay-Konfiguration vor. Sie entwarfen und implementierten eine neue High-Fidelity-Lokomotion-Technik, die auf dem Tracking der Kopfposition basiert, die sie Human Joystick nannten, und verglichen sie mit der Joystick-Navigation. Sie erfassten den horizontalen 2D-Vektor von der Mitte des CAVE zur verfolgten Kopfposition des Benutzers und verwendeten ihn als 2D-Vektor eines Joysticks für die Fortbewegung, unabhängig von der Richtung, in die der Benutzer schaut (siehe Abbildung 5). Sie fanden heraus, dass die Teilnehmer mit der Human Joystick-Methode in Verbindung mit einem High-Fidelity-Display ein größeres Gefühl der Präsenz hatten. Bei einer Low-Fidelity-Anzeige übertrafen Tastatur und Maus jedoch die Human Joystick-Methode.

#### Abstahierte Schnittstelle des Gehens

"Virtuelles Fliegen" gehört zu den häufigsten und an den weitesten verbreiteten Techniken, bei denen die Manipulation des Blickpunkts über ein Eingabegerät erfolgt, z. B. einen Joystick oder eine Maus- und Tastaturschnittstelle [2]. Eine andere Variante dieses Konzepts, der "Human Joystick", ist eine von McMahan et al. für den Einsatz in CAVE-Systemen entwickelte Technik, bei der die Position des Benutzers in Bezug auf das Zentrum als Bewegungsvektor verwendet wird [18]. [2]

#### Laufbänder

Omnidirektionale Laufbänder unterstützen das Gehen in jede Richtung [35]. Sie erweitern zwar den virtuell begehbaren Raum und simulieren das Gehen besser, aber ihre Größe würde eine radikale Umgestaltung der häuslichen Umgebung erfordern, in der sie zum Einsatz kommen sollen. Laufbänder simulieren auch kein unebenes Terrain oder Hindernisse. [2]

#### (Bewegungsverhalten)

Im Unterschied zu den zuvor zitierten Arbeiten untersuchen wir in dieser Arbeit Faktoren, die das Verhalten der Benutzer beim Gehen in einer VE beeinflussen. Andere Forscher haben das Fortbewegungsverhalten durch die Analyse trajektographischer Maße und kinematischer Daten untersucht. Cirio et al. [4] haben ein Rahmenwerk von trajektographischen Kriterien vorgestellt, das den Realismus von virtuellen mit realen Trajektorien vergleicht. Das Rahmenwerk umfasst neun Metriken, die sich auf die Form, die Leistung und die kinematischen Merkmale beziehen. Bei uneingeschränkten zielgerichteten Aufgaben stimmen die Trajektorien in VE mit denen in einer realen Version der Aufgabe überein [4]. Fink et al. [7] fanden heraus, dass sich Trajektorien zur Hindernisvermeidung nur geringfügig zwischen einer VE und einer angepassten realen Umgebung unterscheiden. Warren und Fajen haben ein Modell der Lenkdynamik vorgeschlagen, das versucht darzustellen, wie Individuen eine Umgebung in Richtung eines Ziels navigieren würden, während sie stationäre oder bewegliche Hindernisse vermeiden [45]. Obwohl dieses Modell unter kontrollierten Bedingungen genau ist, reicht es nicht aus, um das Verhalten von Menschenmengen darzustellen [1]. Ruddle et al. [24] fanden heraus, dass natürliches Gehen die Benutzer dazu veranlasste, um Hindernisse herumzugehen, wohingegen Interfaces mit geringerer Originaltreue eine größere Inzidenz von Hindernisdurchdringungen aufwiesen. Sanz et al. [26] haben Trajektorien zur Hindernisvermeidung in einer 10 m langen immersiven Projektionsumgebung verglichen, in der die Teilnehmer realen physischen Hindernissen und Personen ausweichen mussten, sowie passenden virtuellen (nicht existierenden) Gegenstücken. Ihre Ergebnisse deuten darauf hin, dass die Teilnehmer dazu neigen, einen größeren Abstand (ca. 5 cm mehr) zu verwenden, wenn sie mit virtuellen Hindernissen umgehen als mit physischen. Fajen hat die Rolle der Affordanzwahrnehmung in VEs untersucht und gezeigt, dass Benutzer ihre Körpergröße und Handlungsmöglichkeiten berücksichtigen, wenn sie entscheiden, welche Route sie nehmen [5]. Zum Beispiel bei der Entscheidung, wie man sich bewegt, um einem Hindernis auszuweichen. Passive Haptik kann auch die Trajektorien der Teilnehmer beeinflussen. Im "Pit Room" [13] gab eine erhöhte Kante den Teilnehmern die Illusion, neben einer Grube zu stehen. Folglich folgten ihre Trajektorien der Kante. Das Hauptaugenmerk dieser Arbeit lag auf der Untersuchung, ob die Illusion überzeugend war. In unserer Arbeit sind wir daran interessiert, das Verhalten der Benutzer zu verstehen, wenn sie gezwungen sind, eine Navigationsentscheidung zu treffen, die auf dem Konflikt zwischen den visuellen Reizen der VE und ihrem Wissen über den realen Raum, in dem sie sich befinden, basiert. [2]

## Problemstellung, Zielsetzung und Vorgehensweise

### Problemstellung

Virtuelle Umgebungen zu Fuß zu erkunden ist die realistischste und natürlichste Schnittstelle. Zeitgleich ist dies auch technisch und logistisch am anspruchsvollsten.

Allgemein wurden in bisherigen Arbeiten sich auf die Fortbewegung in übereinstimmenden oder leeren Umgebungen konzentriert. Weniger wurde darauf geachtet, wie sich eine Änderung der Nichtübereinstimmung zwischen der physischen Umgebung und ihrer virtuellen Darstellung auf die Benutzer auswirkt

### Zielsetzung

* Konzipieren eines Szenarios, in dem das Bewegungsverhalten der Nutzer über festgelegte Reize in der VR beeinflussen zu können
* Evaluation, um die benötigten Daten zu sammeln und auszuwerten

### Vorgehensweise

* Szenario wird nach der Definition der Konditionierungsreize entwickelt
* Definition der Reize zur Konditionierung der Benutzer in VR (Malus)
* Schnittstelle zur Oculus Quest implementieren
* Auswahl der Erhebungsmethoden und nötigen Daten
* Evaluierung

### Technologien

Für die bestmögliche Umsetzung des Projekts werden verschiedene Technologien benutzt, die im Nachfolgenden erläutert werden.

#### Unity 3D

Unity 3D ist eine Laufzeit- und für Spiele und andere 3D und anderer interaktive 3D-Grafik-Anwendungen. Ebenso das meistgenutzte Tool im Bereich VR-Entwicklung. In Szenen werden die 3D Umgebungen erstellt. Diese besteht zum größten Teil aus Objekten, Materialien bzw. Texturen, Animationen, Audio und Lichtquellen. So entsteht eine virtuelle Welt. Der Android Support ist sehr umfangreich, was der Entwicklung mit der Oculus Quest zugutekommt. [29]

#### Oculus Quest

Bei der Oculus Quest handelt es sich um eine mobile VR Brille. Sie kann also genutzt werden, ohne zusätzlich einen leistungsstarken Computer oder Konsole zu benötigen. Ebenso werden keine externen Sensoren oder Kameras für das Tracking gebraucht.

Die Auflösung des Displays ist 1.600x1.440 Pixel pro Auge mit einer Wiederholrate von 72Hz. Das Tracking passiert über die Brille selbst. Auf dieser befinden sich an der Vorderseite 4 Kameras, die mit Weitwinkel arbeiten. Mit 6DoF, in Englisch Six Degrees of Freedom, ist die Nutzung aller sechs Freiheitsgrade möglich. Somit stehen viele Steuermöglichkeiten zur Verfügung und eine Bewegung im virtuellen Raum gegeben. Dafür elementar ist die Bedienung mit den zwei dazugehörigen Controllern (siehe Abb. 1: Oculus Quest mit Controller). [30]



Abb. 5: Oculus Quest mit Controller[[1]](#footnote-1)

#### Visual Studio

Visual Studio 2019 ist eine Entwicklungsumgebung mit vielen Tools und Dienste für Entwickler. Die eingesetzte Programmiersprache ist hierbei C#. Die Software eigenen Refactorings vereinfachen die Organisation des geschrieben Codes durch vorgeschlagene Aktionen. IntelliCode von Visual Studio gibt dem Entwickler Unterstützung mit dem Einsatz von künstlicher Intelligenz. Beinhaltend sind unter anderem die Bereitstellung von Codevervollständigung und die Suche nach Codeproblemen. Mit der integrierten Codebereinigung ist es möglich mit einem Klick Warnungen und Vorschläge zu ermitteln und zu beheben.

Mit Debugging kann präzise nach Laufzeitfehlern gesucht werden. [31]

#### Programmiersprache C#

C# ist eine objektorientierte und moderne Programmiersprache. Hier werden sogenannte Objekte zur Kapselung von Funktionalität und Daten genutzt. Dies wird erweitert durch die Möglichkeit der Vererbung und Polymorphie. [32] [33]

#### Git Versionskontrolle

Git ist eine kostenlose Open Source Software. Sie dient bei der Softwareentwicklung zur verteilten Versionskontrolle. Die Nutzung wird heutzutage in Unternehmen und von privaten Entwicklern genutzt. Durch die Plattformunabhängigkeit lässt sich Git in fast jeder Entwicklungsumgebung integrieren und nutzen. Die erste Version des Tools wurde im Jahre 2005 veröffentlicht.

Der Nutzen einer Versionskontroller besteht darin, Änderungen verteilt dem Softwareprojekt hinzufügen zu können. Diese werden zusätzlich protokolliert und sind somit nachvollziehbar. Es kann zu einem späteren Zeitpunkt auch auf frühere Versionen zugegriffen werden.

Die Verwaltung geschieht über dezentrale Git-Repositories, im Gegensatz zu anderer Software zur Versionskontrolle, welche mit einer zentralen Datenbank arbeiten. Jeder Entwickler, der an einem Projekt mit Git arbeitet, besitzt eine lokale Kopie des Repositories. Diese wird durch Programmieren weiterentwickelt und die Änderungen mit den anderen Entwicklern des Projekts geteilt.

Ein weiterer großer Bereich ist das Aufteilen und Zusammenführen von verschiedenen Entwicklungsständen. Hierbei kann man ein Softwareprojekt in Unterverzweigungen, den Branchen aufteilen. Dies wird vor allem genutzt, um parallel zum Hauptzweig an verschiedenen Features zu arbeiten. Zum Schluss werden die Branches wieder mit dem Hauptzweig zusammengeführt.

Es gibt viele Webanwendungen, welche Git nutzen und weitere Funktionen hinzufügen. Die Verwaltung von Softwareprojekten wird damit noch übersichtlicher, zugänglicher, transparenter und für andere Entwickler online verfügbar. Der Entwicklungsfortschritt kann grafisch und in Statistiken dargestellt werden. Auch ein eigenes Wiki über das Projekt kann für die Community erstellt und öffentlich zur Verfügung gestellt werden. Die Webseite GitHub ist ein Beispiel für solch eine Plattform [34].

## Umsetzung

### Einbinden der Oculus Quest in Unity 3D

Um selbst entwickelte grafische Anwendung bzw. Spiele in Unity 3D ausführen oder testen zu können, muss eine aktive Verbindung von der Oculus Quest zum Computer und der Entwicklungsumgebung hergestellt werden. (Unity Version erzänzen)

Um die Verbindung zum PC erfolgreich herstellen zu können, wird die Software Oculus Link benötigt. Diese wird vom Unternehmen selbst online angeboten. Erforderlich ist ebenso das Einrichten der Oculus Quest Brille mit einem Facebook Account.

In Unity 3D ist es nötig, den Oculus Support einzurichten. Dazu gibt es aktuell noch zwei Möglichkeiten, welche ihre Vor- und Nachteile bieten. Das veraltete XR-Plugin Tool ist schon standardmäßig integriert und bietet dazu die Möglichkeit, OpenVR zu nutzen. Dies wird benötigt, wenn weitere VR-Brillen in der Entwicklungsumgebung genutzt werden sollen. Ein Beispiel ist die HTCVive. Da die Umsetzung einzig mit der Oculus Quest durchgeführt wird, fiel die Entscheidung auf das neue XR-Management. Dies befindet sich in ständiger Weiterentwicklung, somit wird in naher Zukunft ebenso OpenVR unterstützt werden. Dies könnte bedeutsam sein, falls das Projekt weiterentwickelt werden soll.

Ebenso benötigt wird das Asset Oculus Integration, welches sich über den integrierten Asset Store finden und importieren lässt. Darin befinden sich einige Prefabs und Skripte, die benötigt werden, um grundlegende Interaktionen wie das Umschauen und das Gehen in der Virtuellen Welt ermöglichen. Dazu muss das Prefab OVRPlayerController vom Assets Order per Drag and Drop in die Hierarchie der Szene gezogen werden. Zusätzlich wird das Skript CharacterCameraContraint benötigt, welches noch hinzugefügt werden muss.

Ist die Oculus Quest erfolgreich mit Oculus Link verbunden, kann die VR-Brille direkt in der Unity Entwicklungsumgebung über den Play Mode ausgeführt werden. Dies wird vor allem in der Entwicklung benötigt. Für die spätere Evaluation läuft die Anwendung jedoch Standalone, da ohne Kabel und angeschlossenen Computer ein freieres Bewegen der Probanden möglich ist. Dazu werden in den Grafikeinstellungen einige Änderungen vorgenommen und die Plattform wird zu Android gewechselt. Ebenso muss der Account als Developer freigeschalten werden und die Oculus Quest im Entwicklermodus laufen.

### Entwicklung des Malus

Dieser Teil wurde zunächst separat zum Hauptprojekt entwickelt und nach der Fertigstellung integriert. Ziel ist es, einen visuellen und akustischen Malus auszulösen, sobald der Proband vom Weg abkommt. Visuell sichtbar sein soll ein rotes, pulsierendes Blinken im kompletten Sichtfeld. Ebenso soll neben dem visuellen Reiz ein akustisches Piepen ertönen. Die Frequenz beide Stimuli ist zeitgleich und wird bei steigender Dauer schneller.

Für die Umsetzung in Unity 3D wurde das UI Objekt Canvas implementiert. Dies ist dazu da um Inhalte wie z.B. Bilder, Buttons für ein Menü oder Text anzeigen zu lassen. Eine Besonderheit in der Entwicklung einer VR-Anwendung im Vergleich zu einer normalen 3D Umgebung ist, dass das Canvas mit dem Kamera Objekt des OVRPlayerController verknüpft werden muss. Sonst besteht das Problem der fehlenden Anzeige in der VR-Brille.

Unter dem Canvas wurde ein Image Objekt erstellt und die Größe angepasst. Die Farbe wurde Rot wurde als Hintergrundfarbe eingestellt und der Alpha-Wert zunächst auf 0, sodass beim Start der Anwendung die Farbe erstmal nicht sichtbar ist. Für das Erzeugen des Impulses wurde eine Animation erstellt. Diese nimmt den Alpha-Wert des Images und verändert diesen, um so ein Impuls zu erzeugen. Ebenso wurde eine AudioSource Komponente dem Image Objekt hinzugefügt, die dazu benutzt wird, um die Audiodatei einbinden zu können. Diese wurde zuvor per Drag and Drop in die Assets des Projekts gelegt und dient neben dem visuellen zusätzlich noch einen akustischen Reiz. Um das Audio zeitgleich mit dem visuellen Impuls abspielen zu lassen, wurde ein Event PlayAudioSource in der Animation erstellt. Dazu wurde zusätzlich ein Skript dem Image Objekt hinzugefügt, welches das Event implementiert, die AudioSource Komponente aufruft und das Audio abspielt.

Um die Animation ausführen zu können, wird ein Animator benötigt. Dieser bindet die Erstellen Animationen ein und ist später die Schnittstelle, um die Animationen per Skript ausrufen und starten zu können.

Nach erfolgreichem Testen wurde der Canvas als Prefab, der Animator mit den entsprechenden Animationen und der AudioSource in das Hauptprojekt integriert. Der Aufruf des Malus wurde zum späteren Zeitpunkt bei der Entwicklung der Szenarien implementiert.

### Entwicklung der Szenarien

Der zentralste Bestandteil des Projekts ist das Entwickeln der Szenarien. In diesen soll der Proband in der virtuellen Umgebung einen Weg vom Start bis zum Ziel durchlaufen. Die Schwierigkeit dabei ist das sich verschiedene Hindernisse auf dem Weg befinden. Diese Umgebung ist in allen drei Szenarien gleich.

Zu Beginn des Weges bilden ein Karton und eine Kiste eine Gasse, durch die der Proband durchlaufen muss. Nach diesem Hindernis steht ein Tisch halb im Weg hinein, an dem man vorbeikommen muss (siehe Abb. 2: Hindernisse Kisten und Tisch).



Abb. 6: Hindernisse Kisten und Tisch

Daraufhin bilden zwei hohe Zäune eine Gasse und ein liegender Pfosten, quer liegend über dem Weg, die Hindernisse (siehe Abb. 3: Hindernisse Zaun und Pfahl). Bei diesen Gegenständen besteht entweder die Möglichkeit diese zu Umgehen oder durch Hindurchlaufen bzw. drüber steigen zu bewältigen.



Abb. 7: Hindernisse Zaun und Pfahl

Zum Erkennen, ob sich der Proband auf dem Weg befindet, wurden Box Collider implementiert. Zusätzlich dazu wurde ein Player Skript erstellt und als Komponente an den OVRPlayerController Objekt gehängt. Die darin in C# implementierte Methoden erkennt, ob der Proband den Weg betritt oder verlässt. Befindet sich der Proband auf dem Weg und verlässt diesen, wird der akustische und visuelle Malus über den Zugriff auf den Animator aktiviert. Ebenso wird hier auf das DataRecorder Skript zugegriffen. Beim erstmaligen Betreten der Collider wird der Timer des Spiels gestartet. Sobald ein Malus aktiviert wird, wird darüber ebenso der Timer des Malus gestartet und beim wieder betreten des Weges gestoppt.

Ein weiterer Box Collider befindet sich am Ende des Weges. Sobald der Proband diesen betritt, wird die Zeit des Szenarios gestoppt und er bekommt mit einem grünen optischen Impuls und einem Audio signalisiert, dass das Ziel erreicht wurde.

Den Szenaren unterscheiden sich grundsätzlich im zu laufenden Weg. In einem dieser Szenarien stimmt der visuelle mit dem physischen Weg überein. Im visuellen Szenario unterscheidet sich der visuelle, richtige Weg vom physischen Weg. Das erste Szenario ich gleichzusetzen mit dem Physischen, unterscheidet sich jedoch im fehlenden negativen Malus.

Zum Start des Szenarios wurde ein Menü erstellt, über welches entweder ein explizites Szenario oder über den Menüpunkt Start das Spiel mit einer zufälligen Auswahl gestartet werden kann (siehe). Zur Interaktion mit dem Menü kann der linke oder rechte Controller genutzt werden und die Auswahl wird mit der Trigger Taste bestätigt. Für die Implementierung wurde ein Canvas verwendet. Unter diesem wurden die Menüpunkte als Buttons hinzugefügt. Die Funktionalität der Buttons wurde mit dem selbstentwickelten Skript MenuHandler implementiert, welches alle benötigten Methoden enthält um die verschiedenen Szenarien starten zu können. Die Zuweisung der richtigen Methode für jeden Button wurde im OnClick() Bereich des jeweiligen Inspectors zugewiesen. Für die Benutzung der Controller wurde das Prefab UIHelpers aus dem Oculus Integration Asset hinzugefügt. Dies beinhaltet zum einen Objekt LaserPointer, welches dafür zuständig ist den Laser Pointer zum Anvisieren der Buttons zu visualisieren. An selber Stelle befindet sich zum anderen das Objekt EventSystem, was die Funktionalität des Laserpointers und Controller implementiert. Hier lässt sich unter anderem der Controller bzw. Laserpointer auswählen und welche Taste des Controllers zur Eingabe genutzt werden soll.

### Datenerfassung

Für die Evaluation wurden zwei verschiedene Arten von Daten vorgesehen. Quantitative Daten, die zur Laufzeit der Anwendung entstehen, und qualitative Daten durch die spätere Befragung der Probanden mit Hilfe eines Online Fragebogens.

Die Daten, die automatisch von Unity während der Laufzeit der Szenarien getracked werden sind:

* welches Szenario aktiv war
* Zeit, die der Proband braucht, um den kompletten Weg bis zum Ziel zu laufen
* die Anzahl der Malus, die aktiviert wurde
* die Dauer pro Malus, als wie lange der Proband gebraucht hat, um wieder auf den richtigen Weg zurück zu kommen

Dazu wurde das Skript DataRecorder erstellt und an das OVRPlayerController Objekt hinzugefügt. In diesem Skript wurden mit C# Methoden erstellt, um die erforderlichen Daten, wie zuvor erläutert, aufzuzeichnen und zu speichern. Die Methoden werden durch das Player Skript aufgerufen.

Das Skript CSVManager ist für den Export der aufgezeichneten Daten in eine CSV Datei zuständig. Dies ist für das Auslesen und weitere Auswertung der Daten entscheidend. Die Datei Report.csv wird lokal im Assets Ordner Report erstellt bzw. abgelegt. Für den Fall, das Daten erhoben und abgespeichert werden sollen, die CSV Datei aber noch nicht existiert, wird diese und der dazugehörige Ordner automatisch erstellt. Dazu werden standardmäßig folgende Header in gleicher Reihenfolge erstellt:

* Szenario
* Gebrauchte Zeit
* Malus aktiviert

Sobald ein Malus öfter als einmal aktiviert wurde, werden die dazugehörigen Header für die Dauer der einzelnen Malus dynamisch hinzugefügt. Ein Beispiel der CSV Datei mit gefüllten Daten zeigt sich in Abb. 2: CSV Datei mit Beispieldaten.



Abb. 8: CSV Datei mit Beispieldaten

Es wurde ein Online Fragebogen erstellt, um zusätzlich qualitative Daten zu erheben. Für die Umsetzung wurde EvaSys verwendet. Hierbei handelt es sich um eine webbasierte Feedback Management Lösung. Online Umfragen, klassische Umfragen in Papierform oder selbst Hybridumfragen sind möglich. [35] In unserem Fall setzen wir auf eine reine Online-Umfrage, da eine Papier- oder Hybridumfrage keine Vorteile bietet und das Auswerten mit mehr Aufwand zusammenhängt.

Die Fragen der Online Umfrage wurde in verschiedene Fragegruppen zusammengefasst, auf die im folgenden Abschnitt im Einzelnen genauer eingegangen wird. Als erstes geht es um die persönlichen Daten des Probanden (siehe Abb. 3: Fragebogen - Persönliche Daten). Hier wird das Geschlecht und das Alter als Angaben gemacht. Diese wurden als Single Choice Fragen umgesetzt und das Alter in Gruppen unterteilt, was das empirische Auswerten unkomplizierter und übersichtlicher gestaltet.



Abb. 9: Fragebogen - Persönliche Daten

In der zweiten Fragengruppe geht es um allgemeine Angaben der Probanden (siehe Abb. 4: Fragebogen - Allgemeine Angaben). Die Erfahrungen in Bereich Gaming und Virtual Reality werden jeweils in einer Skala von 1 bis 5 abgefragt. Wobei 1 keine Erfahrung und 5 viel Erfahrung bedeutet. Die Anzahl der Ankreuzfelder wurden im kompletten Fragebogen konsistent gehalten. In der dritten Frage geht es um die Art der VR-Technologien, mit welche der Proband schon in Erfahrung getreten ist. Hierbei handelt es sich um eine Multiple Choice Frage, da eine oder mehrere Antworten möglich sind.



Abb. 10: Fragebogen - Allgemeine Angaben

Die nächste Gruppenfrage bezieht sich explizit auf die Durchführung der Szenarien (siehe Abb. 5: Fragebogen - Beurteilung Szenario). Für jedes der drei Szenarien werden dieselbe Art Fragen benutzt, um die Auswirkungen auf den Probanden zu ermitteln. Die Skala im Wert von 1 bis 5 ist hier ebenso, wie zuvor erläutert, die Art der Antwortmöglichkeiten. Der Wert 1 bedeutet Dazu gehört die Natürlichkeit des Laufens, die Projektion des Denkens und Handelns in der virtuellen Umgebung. Das Bewusstsein der physischen Umgebung ist ebenso ein Teil der Fragestellung. Die letzte Frage geht explizit auf den Einfluss des negativen Malus auf den Probanden ein.



Abb. 11: Fragebogen - Beurteilung Szenario

Die letzte Fragegruppe ergibt ein offenes und persönliches Feedback (siehe Abb. 6: Fragebogen - Persönliches Feedback). Der Frage Typ hierbei sind offene Fragen mit unbegrenzter Anzahl an Zeichen.

Hier geht es unter anderem darum, was die Probanden besonders positiv und negativ an der Durchführung der Szenarien fanden. Ein weiterer Punkt ist, was genau die Probanden am negativen Malus beeinflusst hat. Zum Abschluss können konkrete Verbesserungsvorschläge gebracht werden, was besonders Relevant für die Weiterentwicklung des Projektes ist.



Abb. 12: Fragebogen - Persönliches Feedback

Nach der Datenerfassung mit Hilfe des Fragenbogens können die Daten direkt in EvaSys dargestellt und ausgewertet werden.

### Evaluation

Bei der Ankunft in der Studie (siehe Abb. 1 Die Teilnehmer erhielten einen kurzen Überblick über die Studie, der Informationen zum Oculus Rift-Headset und eine allgemeine Beschreibung der Konditionierungsaufgabe enthielt. Nach Einverständniserklärung wurde den Teilnehmern die Verwendung

des Videospiel-Controllers erklärt. (Aufgabenbeschreibung)

## Zusammenfassung, Bewertung und Ausblick

## Anhang

### Abkürzungsverzeichnis

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

### Abbildungsverzeichnis

[Abb. 1: Oculus Quest mit Controller 8](#_Toc65407026)

[Abb. 2: CSV Datei mit Beispieldaten 12](#_Toc65407027)

[Abb. 3: Fragebogen - Persönliche Daten 13](#_Toc65407028)

[Abb. 4: Fragebogen - Allgemeine Angaben 13](#_Toc65407029)

[Abb. 5: Fragebogen - Beurteilung Szenario 13](#_Toc65407030)

[Abb. 6: Fragebogen - Persönliches Feedback 14](#_Toc65407031)

### Literaturverzeichnis

|  |  |
| --- | --- |
| [1] | J. N. Templeman, D. P. S. und L. E. Sibert, „Virtual Locomotion: Walking in Place through Virtual Env ironments,“ Dezember 1999. |
| [2] | A. L. Simeone, I. Mavridou und W. Powell, „Altering User Movement Behaviour in Virtual Environments,“ April 2017. |
| [3] | P. Fink, P. Foo und W. H. Warren, „Obstacle avoidance during walking in real and virtual environments,“ Januar 2007. |
| [4] | S. Razzaque, D. Swapp, M. Slater, M. C. Whitton und A. Steed, „Redirected Walking in Place,“ 2002. |
| [5] | M. C. W. Kroes, J. E. Dunsmoor, W. E. Mackey, M. McClay und E. A, „Context conditioning in humans using commercially available immersive Virtual Reality,“ 2017. |
| [6] | H. Cherni, N. Métayer und N. Souliman, „Literature review of locomotion techniques in virtual reality,“ 2020. |
| [7] | M. Usoh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater und F. P. Brooks, „Walking > Walking-in-Place > Flying, in Virtual Environments,“ 1999. |
| [8] | D. A. Bowman, D. Koller und L. F. Hodges, „Travel in Immersive Virtual Environments: An Evaluation of Viewpoint,“ 1997. |
| [9] | R. A. Ruddle und S. Lessels, „The Benefits of Using a Walking Interface,“ April 2009. |
| [10] | R. A. Ruddle, E. Volkova und H. H. Bülthoff, „Learning to Walk in Virtual Reality,“ Mai 2013. |
| [11] | M. C. Whitton, J. V. Cohn, J. Feasel, P. Zimmons, S. Razzaque, S. .. Poulton, B. McLeod und F. P. Brooks, Jr., „Comparing VE Locomotion Interfaces,“ März 2005. |
| [12] | E. A. Suma, S. L. Finkelstein, M. Reid, S. V. Babu, A. C. Ulinski und L. F. Hodges, „Evaluation of the Cognitive Effects of Travel Technique in Complex Real and Virtual Environments,“ 2010. |
| [13] | S. Razzaque, Z. Kohn und M. C. Whitton, „Redirected Walking,“ 2001. |
| [14] | E. A. Suma, G. Bruder, F. Steinicke, D. M. Krum und M. Bolas, „A Taxonomy for Deploying Redirection Techniques in Immersive Virtual Environments,“ 2012. |
| [15] | F. Steinicke, G. Bruder, L. Kohli, J. Jerald und K. Hinrichs, „Taxonomy and Implementation of Redirection Techniques for Ubiquitous Passive Haptic Feedback,“ 2008. |
| [16] | F. Steinicke, G. Bruder, J. Jerald, H. Frenz und M. Lappe, „Analyses of Human Sensitivity to Redirected Walking,“ 2008. |
| [17] | F. Steinicke, G. Bruder, T. Ropinski und K. Hinrichs, „Moving Towards Generally Applicable Redirected Walking,“ 2008. |
| [18] | G. Bruder, F. Steinicke und K. H. Hinrichs, „Arch-Explore: A natural user interface for immersive architectural,“ 2009. |
| [19] | B. Williams, G. Narasimham, T. P. McNamara, T. H. Carr, J. J. Rieser und B. Bodenheimer, „Updating orientation in large virtual environments using scaled translational gain,“ 2006. |
| [20] | V. Interrante, B. Ries und L. Anderson, „Seven League Boots: A New Metaphor for Augmented Locomotion through Moderately Large Scale Immersive Virtual Environments,“ 2007. |
| [21] | F. Steinicke, G. Bruder, J. Jerald, H. Frenz und M. Lappe, „Estimation of Detection Thresholds for Redirected Walking Techniques,“ 2009. |
| [22] | G. Bruder, F. Steinicke und P. Wieland, „Self-Motion Illusions in Immersive Virtual Reality Environments,“ 2011. |
| [23] | F. Steinicke, G. Bruder, K. Hinrichs und A. Steed, „Gradual transitions and their effects on presence and distance estimation,“ 2009. |
| [24] | B. Williams, G. Narasimham, B. Rump und T. McNamara, „Exploring Large Virtual Environments with an HMD when Physical Space is Limited,“ 2007. |
| [25] | T. Peck, H. Fuchs und M. Whitton, „Evaluation of Reorientation Techniques and Distractors for Walking in Large Virtual Environments,“ 2008. |
| [26] | E. Suma, S. Clark, S. Finkelstein und Z. Wartell, „Leveraging Change Blindness for Redirection in Virtual Environments,“ 2011. |
| [27] | D. Simons, „Current Approaches to Change Blindness,“ 2000. |
| [28] | D. Zielasko, S. Horn, S. Freitag, B. Weyers und T. Kuhlen, „Evaluation of Hands-Free HMD-Based Navigation Techniques for Immersive Data Analysis,“ 2016. |
| [29] | [Online]. Available: https://unity.com/de. [Zugriff am 23 02 2021]. |
| [30] | D. Ziesecke, „Alle Infos zur Oculus Quest – technische Daten, Preis und Laufzeit,“ 03 03 2019. [Online]. Available: https://vr-legion.de/news/alle-infos-zur-oculus-quest-technische-daten-preis-und-laufzeit/. [Zugriff am 23 02 2021]. |
| [31] | TerryGLee, „Neues in Visual Studio 2019,“ 10 11 2020. [Online]. Available: https://docs.microsoft.com/de-de/visualstudio/ide/whats-new-visual-studio-2019?view=vs-2019. [Zugriff am 22 02 2021]. |
| [32] | B. Wagner, „Überblick über C#,“ 28 01 2021. [Online]. Available: https://docs.microsoft.com/de-de/dotnet/csharp/tour-of-csharp/. [Zugriff am 23 02 2021]. |
| [33] | O. B. Dr. Veikko Krypczyk, „Einführung in die Programmierung: Objekte, nichts als Objekte,“ 19 04 2016. [Online]. Available: https://entwickler.de/online/development/einfuehrung-programmierung-objektorientierte-programmentwicklung-197372.html. [Zugriff am 24 02 2021]. |
| [34] | S. Augsten, „Definition „Git SCM“ - Was ist Git?,“ 27 08 2019. [Online]. Available: https://www.dev-insider.de/was-ist-git-a-850847/. [Zugriff am 24 03 2021]. |
| [35] | [Online]. Available: https://evasys.de/evasys/. [Zugriff am 28 02 2021]. |
| [36] | C. Spinger, „Oculus Quest im Test: Virtual Reality für die Masse,“ 30 04 2019. [Online]. Available: https://www.vrnerds.de/oculus-quest-im-test-virtual-reality-fuer-die-masse/. [Zugriff am 27 02 2021]. |
| [37] | A. L. Simeone, E. Velloso und H. Gellersen, „Substitutional Reality: Using the physical environment to design virtual reality experiences,“ Januar 2015. |
| [38] | J. N. Templeman, P. S. Denbrook und L. E. Sibert, „Virtual locomotion: Walking in place through virtual environments,“ Dezember 1999. |
| [39] | L. F. Hodges, R. Kooper, T. C. Meyer, B. O. Rothbaum, D. Opdyke, J. J. Graaff, J. S. Williford und M. M. North, „Virtual Environments for Treating the Fear of Heights,“ 1995. |

### Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Zeichnungen oder Abbildungen in dieser Arbeit sind von mir selbst erstellt worden oder mit einem entsprechenden Quellennachweis versehen.

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

Heilbronn, 19. September 2016 Vorname Name

1. https://www.vrnerds.de/wp-content/uploads/2019/04/Oculus-Quest-2-1.jpg [36] [↑](#footnote-ref-1)