Interaktionen im dreidimensionalen Raum

Im Kontext von kollaborativen Mixed-Reality-Anwendungen

Christian Blank

HAW Hamburg, Technik und Informatik,
Berliner Tor 7, Hamburg, Germany
christian.blank@hawhamburg.de
http://i2e.informatik.haw-hamburg.de

Zusammenfassung. In der vorliegenden Arbeit wird die These zur Kombination von interpretierten Gesten und physikbasierter Interaktion aufgestellt und es wird ein Lösungsansatz vorgestellt, mit dem diese These untersucht werden kann. Zudem wird das Konzept der Evaluierung erläutert, mit dem die Lösung und die These untermauert werden sollen. Im Anschluss werden Chancen und Risiken beleuchtet und es wird auf den aktuellen Stand der Entwicklung aufgezeigt.

Schlüsselwörter: Mixed-Reality;Gestenerkennung;Interaktion;3D

1 Einleitung

Vor einem Jahrzehnt war es nenneswert, wenn man im World Wide Web (WWW) gesurft ist. Heute ist man ständig und rund um die Uhr, mithilfe von Smartphones, Tablets, Smartwatches und Notebooks, online.

Die virtuelle Realität (VR), die man mit VR-Brillen, wie etwa der Oculus Rift, erleben kann, sind heute eher eine Ausnahme und für viele Personen gänzlich unbekannt. In wenigen Jahren wird sich aber auch hier ein ähnliches Muster wie zuvor mit dem WWW abzeichnen. Erste Studien, die die Auswirkung von VR über einen längeren Zeitraum beobachten, wurden bereits durchgeführt [SB14] und auch die Industrie drängt immer weiter in eine Vermischung aus virtueller und realer Welt vor. Projekte von Microsoft, Meta, Epson und Google zeigen klar auf, dass sehr viel Bewegung in diesem Feld existiert, das bereits vor mehr als 20 Jahren entstand.

Neben VR-Brillen werden immer häufiger auch See-Through-Brillen verwendet. Diese Art von Brillen ist dazu geeignet, virtuelle Objekte in eine reale Szene zu projezieren. Durch die vielfältigen Einsatzgebiete und die Ablösung von stationären Computern und traditionellen Eingabegeräten hinzu mobilen Lösungen ermöglicht die Verwendung von neuen Interaktionsform durch natürliche, intuitive Schnittstellen.

1.1 Motivation

Die Kommunikation mit einer Umgebung, egal ob real, virtuell oder einer Mischform, geschiet immer über die Aktorik und Sensorik des Nutzers. In VR wird in der Sensorik bisher sehr viel im audio-visuellen Bereich gearbeitet. Tast- und Geruchssinne werden kaum miteinbezogen. Im Bereich der Aktorik werden hingegen eine Vielzahl von Möglichkeiten angeboten und getestet. Unter anderem werden Sprachsteuerung, Mimikerkennung und Maus- und Tastatursteuerung für eine Kommunikation genutzt. Auch Gestenerkennung und allgemein die Analyse von Bewegungen werden verwendet, um Eingaben von einem Nutzer anzunehmen.

In dem konkreten Anwendungsszenario wird eine reale Szene durch eine Multimediabrille betrachtet und die Szene somit durch virtuelle Daten erweitert. Zu den virtuellen Daten gehören Bausteile, das Konstrukt und entfernte Personen. Neben den rein virtuellen Objekten, werden sich auch reale, physikalische Objekte in der Szene befinden. Die realen Objekte können mit den virtuellen Objekten interagieren. In der Szene können Personen Objekte aus Einzelteilen zusammensetzen, sie betrachten und über sie diskutieren.

Der allgemeine Aufbau ist in Abbildung 1.1 dargestellt. Ein Nutzer muss nur eine See-Through-Brille tragen und ist sonst nicht mit dem System verbunden. Die Brille ist ein **Mobile Viewer** und ermittelt ihre Position im Raum selbstständig. Die aktuelle Position wird zusammen mit dem Bildwinkel und der Auflösung an einen **Szenenrenderer** gesendet. Dieser kann aus den übermittelten Daten die Szene aus der Sicht des Devices rendern und überträgt die Daten anschließend zurück an die Brille. Die Würfel in der Hand des Nutzers agieren als Stellvertreter für virtuelle Objekte. Durch das **Objekttracking** können Bewegungen der Würfel ermittelt und auf die virtuellen Bausteine übertragen werden. In Abbildung 1.1, Mitte kann man die Murmelbahn erkennen, die von den beiden

Nutzern zusammen gebaut wird. Die grauen Avatare in den Bildern links und rechts sind die jeweiligen virtuellen Präsenzen der entfernten Nutzer. Diese Nutzer werden durch die **3D-Rekonstruktion** gescannt und ihr Mesh wird über das Netzwerk übertragen. Die Zusammensetzung der Konstrukte wird durch Constraints, die in der **Konstruktionslogik** verankert sind, gesteuert. Durch die Constraints wird sichergestellt, dass die virtuellen Konstrukte zu jedem Zeitpunkt valide sind. Das virtuelle Konstrukt kann durch **Gesten** manipuliert werden. Zusammen mit dem Objekttracking bildet die Gestenerkennung das **User Interaction Interface**. Der gesammte Aufbau ermöglicht es dem Nutzer, das virtuelle Konstrukt von allen Seiten zu betrachten. Durch die Nutzung von realen Objekten für Konstruktionsvorgänge in virtuellen Objekten wird die Umgebung als Mixed-Reality-Umgebung bezeichnet.

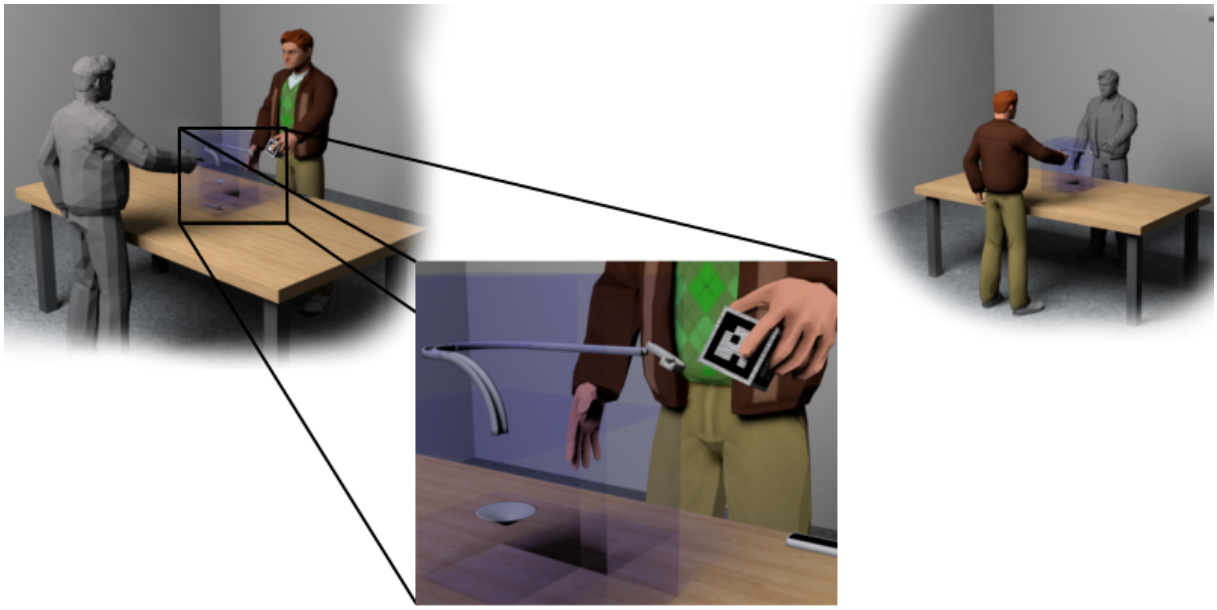

Abb. 1. Veranschaulichung des Konzeptes für die kollaborative Arbeit in Mixed Reality zur Erstellung von 3D-Modellen (Quelle: `http://i2e.informatik.haw-hamburg.de`)

In einer Mixed-Reality-Umgebung, die Möglichkeiten für ein Prototyping bereitstellt, sollten sich Objekte so verhalten, wie man es als Anwender erwarten würde. Da nicht nur rein virtuelle, sondern auch reale Objekte vorhanden sind, kann nicht erwartetes Verhalten noch schneller zu Unverständnis bei den Nutzern führen. Um dieses Problem zu lösen, wird versucht, eine Schnittstelle zu bieten, die es einem Anwender ermöglicht, mit realen und virtuellen Objekten in gleicher Weise zu interagieren.

1.2 Ziel

Das Ziel dieser Arbeit wurde bereits in dem vorherigen Abschnitt kurz angerissen. **Virtuelle Objekte sollen sich bei der Interaktion wie reale Objekte verhalten.** Das bedeutet, dass ein Nutzer durch die Bewegung seines Armes durch die Szene, virtuelle Objekte verschieben und drehen kann. Ebenso ist es möglich, dass ein Nutzer virtuelle Objekte in die Hand nimmt und sie fallen lassen kann. Neben der physikalischen Interaktion ist es auch gewünscht, dass es objekt- und kontextbezogene Interaktionsmöglichkeiten gibt, die durch den Computer interpretiert werden können. Es ist vorstellbar, dass sich ein virtuelles Objekt skalieren lässt oder das man ein zusammengesetztes virtuelles Objekt duplizieren kann.

Für diesen Zweck werden zwei Ansätze der Interaktion miteinander verbunden. Interpretierte Gesten werden für erweiterte Interaktionsformen und physikbasierte Interaktion für die Bewegung von virtuellen Objekten verwendet. These 1 ist die Grundlage der Arbeit und soll im weiteren Verlauf untersucht werden. Dabei wird zunächst eine Lösungsidee konzipiert und anschließend versucht, die These mithilfe der Lösung zu validieren.

These 1 *Durch die Kombination von interpretierten Gesten und physikbasierter Interaktion kann ein Benutzer intuitiver mit virtuellen und realen Objekten in einer Mixed-Reality-Umgebung arbeiten als es bei herkömmlichen Lösungen der Fall ist.*

1.3 Definitionen

Um im weiteren Verlauf dieser Arbeit ein einheitliches Verständnis für die einzelnen Begriffe zu haben, werden zunächst einige Definitionen gegeben. Die Begriffe werden in der Literatur teilweise unterschiedlich genutzt und deshalb an dieser Stelle definiert.

Mixed Reality Mixed Reality (MR) ist ein Teil der virtuellen Realität, in der reale und virtuelle Objekte in einer Szene gemeinsam dargestellt werden. Sie wird zum einen von der realen Umgebung und zum anderen von der virtuellen Umgebung begrenzt, wie in Abbildung 1.3 gezeigt. Augmented Reality (AR) ist die Erweiterung der Realität durch zusätzliche, virtuelle Informationen. Ein Beispiel wäre die Navigation im Straßenverkehr durch ein Head-up-Display, das die korrekte Fahrtrichtung für den Fahrer sichtbar auf die Frontscheibe projeziert und Verkehrsschilder markiert. Im Gegensatz dazu ist die Augmented Virtuality (AV) eine virtuelle Umgebung, in der reale Daten eingeblendet werden. Diese realen Daten können beispielsweise Webcams sein, deren Videostream in einer virtuellen Welt in einem Fenster gerendert wird.

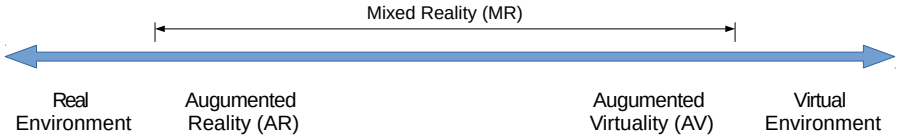

Abb. 2. Der Bereich zwischen vollständig realer und vollständig virtueller Welt wird als Mixed Reality bezeichnet (Quelle: [MK94])

Interpretierte Geste Eine interpretierte Geste kann als eine Abstraktion von Bewegungsmustern, denen eine Bedeutung zugewiesen wird, angesehen werden. Dabei werden Bewegungsabläufe des Nutzers aufgezeichnet und analysiert. Als Beispiele für interpretierte Gesten wären eine allgemeine Zeigegeste, eine Bestätigungsgeste oder eine Geste zum Vergrößern eines Ausschnittes zu nennen.

Physikbasierte Interaktion Die physikbasierte Interaktion dient als virtuelles Gegenstück zur Interaktion mit realen Objekten. Dabei hat die Bewegung eines Körpers direkten Einfluss auf ein oder mehrere virtuelle Objekte. Dabei können einfache Formen, wie Drehen und Schieben, oder kompliziertere Formen, wie etwa Greifen verwendet werden.

1.4 Aufbau

Die vorliegende Arbeit besteht aus vier Abschnitten. Nach der Einleitung folgt in Abschnitt 2 die Präsentation eines Lösungsansatzes. Im Abschnitt 3 wird die geplante Evaluierung der aufgestellten These mithilfe der Lösung vorgestellt. Zum Schluss wird in Abschnitt 4 ein Fazit gegeben, es wird auf Chancen und Risiken eingegangen und der aktuelle Stand der Entwicklung aufgezeigt.

2 Lösungsansatz

Die hier präsentierte Lösung vereinigt zwei Ansätze der räumlichen Interaktion: Physikbasierte Interaktion und Interpretation von räumlichen Gesten. Im vorherigen Abschnitt wurden bereits beide Begriffe definiert. Zu der physikbasierten Interaktion können unter anderem Lösungen gezählt werden, die durch die Verfolgung der Gelenkpunkte arbeiten [SYW08]. Ebenso kann Partikelverfolgung genutzt werden [HKI+12]. Ein Vertreter im Bereich der Interpretation von Gesten ist das Templatematching [KNQ12]. Neben den bereits genannten Ansätzen gibt es noch weitere Verfahren, die aber keinen Einfluss auf die präsentierte Lösung haben. Die gesuchte Lösung soll auf Skelettdaten arbeiten und weitestgehend unabhängig von den verwendeten Sensoren sein. Daraus resultiert der Wunsch nach einer Abstraktionsschicht. Neben der Berechnung der physikalischen Interaktion sollen die Bewegungen auch interpretiert werden. Die Ergebnisse sollten nach Möglichkeit in interaktiven Antwortzeiten (ca. 100-200 ms) zur Verfügung stehen. Andere Services können sich auf die Gestendaten einschreiben und werden informiert, sobald neue Daten verfügbar sind. Es ist möglich, die Bewegungen von verschiedenen Nutzern zur gleichen Zeit zu analysieren.

Durch die verschiedenen Skelettformate die durch die Sensoren erzeugt werden und die schnelle Entwicklung auf dem Markt, ist eine Abstraktionsschicht für die Sensordaten nötig. Die Skelette, die durch die Abstraktionsschicht erzeugt werden, beinhalten alle relevanten Informationen für die Gestenerkennung. Sie werden sowohl in der Analyse der interpretierte Gesten als auch in der Berechnung der physikbasierten Interaktion genutzt. Die Ergebnisse der beiden Analysen werden in einer gemeinsamen Schnittstelle anderen Services zur Verfügung gestellt.

Damit besteht die Verarbeitungspipeline aus drei größeren Abschnitten, zu sehen in Abbildung 2. Die Abstraktionsschicht Trame (gelb) erzeugt Skelettdaten und gibt diese an die weiter (rot). Die Hauptkomponente besteht aus der Gestenerkennung und der Modellerstellung. Über eine Schnittstelle (blau) werden die aufbereiteten Daten anschließend zur Verfügung gestellt.

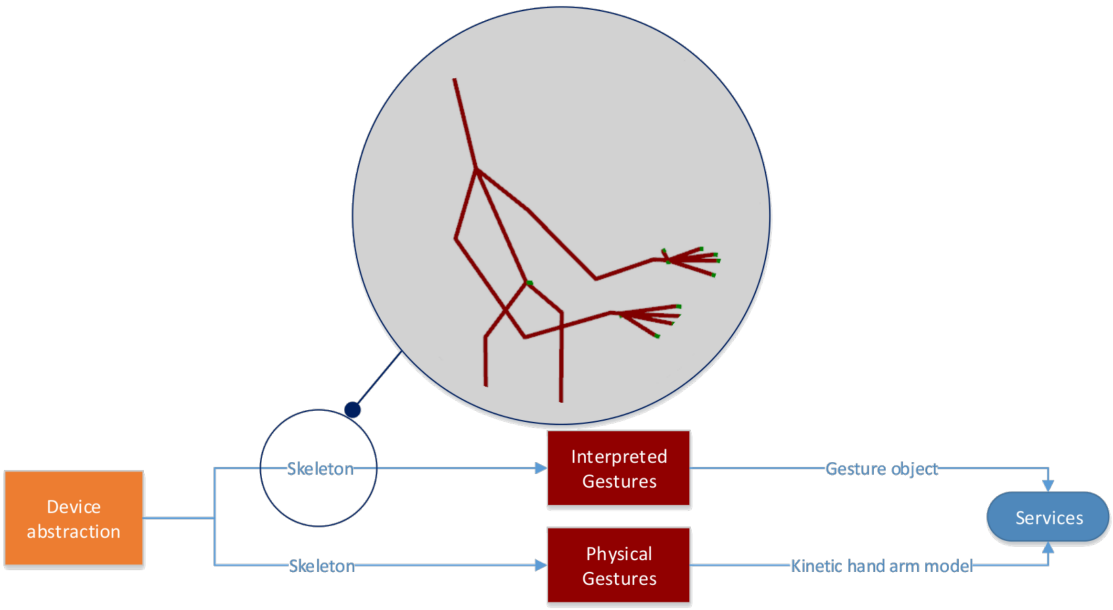

Abb. 3. Übersicht der Verarbeitungspipeline mit Geräteabstraktion, Gestenerkennung, Modellerstellung und Ausgabe (Quelle: `http://i2e.informatik.haw-hamburg.de`)

Im Nachfolgenden werden die drei Komponenten der Verarbeitungspipeline vorgestellt.

2.1 Abstraktionsschicht

Die verwendete Hardwareabstraktionsschicht wurde bereits in Anwendung 2[1] beschrieben und entstand in der Arbeit für Projekt 1. Durch die Verwendung von Trame werden einige große Vorteile gegenüber der direkten Verwendung der SDKs der Hersteller gewonnen. Durch das uniforme Skelettmodell, das auf einer Baumstruktur basiert, kann eine Unabhängigkeit von der verwendeten Hardware erzielt werden. Ein Wechsel zwischen Kameramodellen verschiedener Hersteller ist somit genauso möglich wie der Einsatz einer anderen Technologie, bspw. die Verwendung von Thalmic Myo.

Die Kombination verschiedener Sensoren ist ebenso möglich. So kann ein Körperskelett in Kombination mit dem Handmodell der Leap Motion verwendet werden. Das Körperskelett wird derzeit durch das Microsoft Kinect-SDK erzeugt. Durch die Verwendung von relativen Abständen zwischen den Gelenken in dem erzeugten Skelettmodell kann auf eine zeitliche Synchronisation der Frames von verschiedenen Sensoren verzichtet werden.

[1] `http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master2014-aw2/blank/bericht.pdf`

Erstellte Skelette können durch Trame serialisiert und deserialisiert werden. Somit ist es auch möglich, Datenströme, bestehend aus Skelettdaten, wiederzuverwenden und die Testbarkeit der Lösung zu verbessern.

2.2 Datenverarbeitung

Nachdem der Controller ein neues Skelett von der Abstraktionsschicht erhalten hat, wird das Skelett einer Pipeline zugeordnet. Eine Pipeline repräsentiert dabei immer einen Nutzer, sodass mehrere Nutzer parallel verarbeitet werden können. Somit ist die Lösung multiuserfähig.

Im ersten Schritt der Vorverarbeitung wird versucht, Messfehler und "Wackler" zu reduzieren, indem der Mittelwert der Skelettbewegung innerhalb eines Fensters von Frames berechnet wird. Das Ergebnis wird für alle weiteren Berechnungen verwendet. Durch diese Maßnahmen soll die Robustheit der Lösung gesteigert werden.

Um die Effizienz zu verbessern, werden mehrere Faktoren überprüft, bevor eine Auswertung durchgeführt wird. Die Ausrichtung des Körpers und des Kopfes eines Nutzers ist einer der wichtigsten Anhaltspunkte, um die Intention zu bestimmen. Es wird dabei geschaut, ob der Blick und der Körper zu dem zu manipulierenden Objekt gerichtet sind oder nicht. Neben der groben Intention wird noch eine adaptive Input-Zone festgelegt, in der der Nutzer agieren kann. Bewegungen außerhalb dieser Zone werden nicht als solche anerkannt und von der Weiterverarbeitung ausgeschlossen.

In dieser Lösung werden nur Gesten verarbeitet, die mit den Armen und Händen durchgeführt werden können. Zu diesem Zweck werden die Arm- und Handposition extrahiert und die restlichen Elemente des Körperskeletts verworfen. Durch diesen Schritt werden die Berechnungen und der Speicherverbrauch reduziert. Das Hand-Arm-Modell und das Templatematching wird parallel ausgeführt. Somit kann die Zeit, die für die Verarbeitung einer Eingabe benötigt wird, reduziert werden.

Kinetisches Hand-Arm-Modell Nachdem die Skelette gefiltert wurden, wird die Bewegungsrichtung und Geschwindigkeit der übrig gebliebenen Gelenke bestimmt. Diese Daten werden für die kontinuierliche Erstellung von einem kinetischem Hand-Arm-Modell benötigt. Hierfür wird ein Ansatz wie in [OKA11] gewählt. In vereinfachter Form zu sehen in Abbildung 2.2. Die Gliedmaßen zwischen den Gelenken werden dabei durch Ellipsoide approximiert. Neben der Hand wird in der angestrebten Lösung auch der Unterarm modelliert. Das kinetische Arm-Hand-Modell wird für die physikbasierte Interaktion benötigt. Durch die Verwendung dieser einfachen Approximation kann ein zu großer Rechenaufwand vermieden werden, wie es etwa bei der Partikelverfolgung der Fall gewesen wäre. Im Gegensatz zu [SYW08], bei dem nur eine Fingerspitze verfolgt wurde, erhält dieses Modell einen größeren Detailgrad, der ein natürlicheres Verhalten bei der Interaktion mit virtuellen Objekten bewirken soll.

Abb. 4. Darstellung eines approximierten Handmodells durch Ellipsoide (Quelle: [OKA11])

Templatematching Parallel zu der Berechnung des Hand-Arm-Modells wird ein Templatematching genutzt, um Gesten zu interpretieren. Bei dem **Templatematching** werden die Punkte der aktuellen Bewegung mit vorhandenen Templates für verschiedene Gesten verglichen. Ein Template bestimmt dabei aus einer Liste von Ortspunkten, die ein Gelenkpunkt durchlaufen muss. Für das Matching wird ein Algorithmus auf der Basis von [KNQ12] eingesetzt, der jedoch einige Änderungen enthält. So werden nicht nur zweidimensionale Gesten, sondern auch Gesten in

allen drei Dimensionen erkannt. Bei der Lösung von Kristensson wurden nur die Handflächen verfolgt. In der vorgestellten Lösung können alle enthaltenen Gelenkpunkte verfolgt und in einem Template einbezogen werden. Die Menge aller Templates, die in einem Vergleich miteinbezogen werden, wird als **Gestenset** bezeichnet.

Als Ergebnis erhält man die Wahrscheinlichkeit für jedes mögliche Template im Gestenset. Auf Basis dieser Wahrscheinlichkeiten wird im letzten Schritt eine Entscheidung getroffen.

2.3 Datenbereitstellung

Der Szenenrenderer schreibt sich auf mögliche Daten ein und erhält Updates, sobald neue Daten vorhanden sind. Die Bewegungsdaten der Arme aus dem kinetischen Arm-Hand-Modell werden in jeder Iteration berechnet und anderen Services zur Verfügung gestellt. Im Gegensatz dazu werden mögliche Gesten erst nach einem Entscheidungsprozess gepublished. Das bedeutet, dass in vielen Iterationen keine Geste erkannt wird. Die Entscheidung, ob eine Geste ausgeführt wurde oder nicht, wird durch die ins Verhältnis gesetzten Wahrscheinlichkeiten bestimmt. Dabei muss eine Geste nicht nur zu einer hohen Wahrscheinlichkeit ausgeführt worden sein, sondern auch einen größeren Abstand zu anderen Gesten besitzen. Die Festlegung der Grenzwerte erfolgt zunächst durch Schätzung und kann angepasst werden, wenn genügend Eingabedaten gesammelt wurden.

Zusätzlich zu der eigentlichen Geste erhalten die Services weitere Informationen über die Bewegung. Es ist geplant, die Skalierung, Orientierung und Geschwindigkeit, mit der die Geste ausgeführt wurde zu übergeben und somit Services zu ermöglichen auf diese Parameter gesondert einzugehen. Somit könnte es bei eine Skalierungsgeste interessant sein, wie schnell oder in welcher Position sie ausgeführt wird. Die Bewegungsdaten bestehen aus einem sogenannten Collider-Objekt, das Bewegungsrichtung und Beschleunigungswerte enthält für jedes Segment im Arm und der Hand enthält.

3 Evaluierung

Als technische Grundlage für die Evaluierung wird ein Teilsystem verwendet, dass eine große Nutzerinteraktion erlaubt, aber dennoch so überschaubar ist, dass Fehler lokalisiert werden können. Im Gegensatz zu einer vollständigen Konfiguration werden nur der Szenenrenderer, der Mobile Viewer und das User Interaction Interface verwendet, die in Abschnitt 1.1 eingeführt wurden. Services für die 3D-Rekonstruktion, die Verteilung oder komplexere Simulationslogiken werden nicht einbezogen.

3.1 Testumgebung

Nutzer tragen eine Multimediabrille, in diesem Fall die Epson Moverio BT-200, und stehen vor einem Tisch, der mit einer Microsoft Kinect und einer Leap Motion ausgestattet ist. Neben diesen 3D-Kameras wird eine 2D-Kamera für das Objekttracking verwendet. Der Aufbau ist in Abbildung 3.1 dargestellt. Neben den Kameras des Systems werden auch Überwachungskameras eingesetzt, die die Bewegung des Nutzers aufzeichnen. Somit können nach Beendigung der Untersuchungen weitere Beobachtungen gemacht werden. Wenn ein Proband das erste Mal das System nutzt, dann wird er verschiedene Dinge ausprobieren, auf die das System nicht reagiert. Häufen sich Versuche ohne Reaktion bei verschiedenen Probanden, dann können die entsprechenden Gesten in nachfolgenden Gestensets aufgenommen werden.

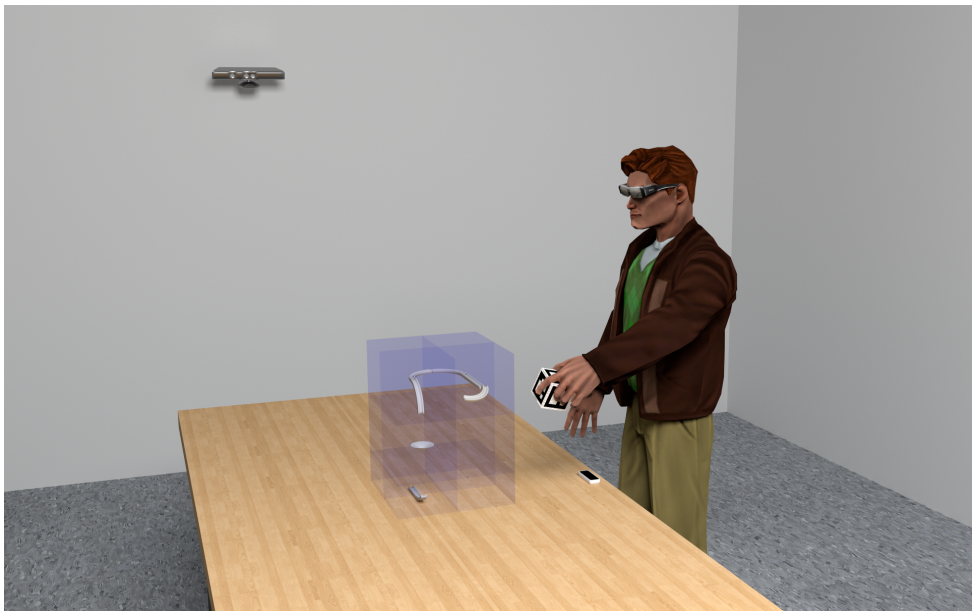

Abb. 5. Übersicht des Testaufbaus - Der Proband trägt eine See-Through-Brille und wird von verschiedenen Kameras aufgezeichnet (Quelle: `http://i2e.informatik.haw-hamburg.de`)

3.2 Funktionale Tests

Für funktionale Tests werden aufgezeichnete Bewegungen in die Gestenerkennung gespielt. Somit ist es möglich, automatisierte Tests durchzuführen. Es wird dabei die Dauer der Verarbeitung (Antwortzeit), die korrekte Entscheidung bei der Auswertung von Gesten und das Arm-Hand-Modell getestet.

3.3 Usability-Tests

Nachdem ein Proband den Testaufbau betreten hat, wird ihm etwas Zeit gegeben, einen ersten Eindruck von der MR-Umgebung zu erhalten und er kann sich in einer einem Sandkasten an das Userinterface gewöhnen. Nach dieser Eingewöhnungsphase erhält eine kurze Erklärung zu den einzelnen Gestenarten, die von dem System unterstützt werden. Dieser Schritt ist wichtig, da in der späteren Befragung genauer auf diese Punkte eingegangen wird. Dem Probanden werden verschiedene Aufgaben gestellt, die er nacheinander lösen muss. Neben den Aufgaben ändert sich auch die Eingabemöglichkeit, sodass teilweise nur interpretierte Gesten, nur physikbasierte Interaktion oder beide Interaktionsformen verwendet werden können.

Nachdem ein Proband alle Aufgaben absolviert hat, wird ein Interview geführt. In diesem Interview wird gemeinsam mit dem Probanden ein Fragebogen zur Ergonomie ausgefüllt. Dabei soll herausgefunden werden, welche Interaktionsform für den Proband in welcher Aufgabe am angenehmsten war und warum das möglicherweise so war.

3.4 Aufgaben

Die gestellten Aufgaben in den jeweiligen Szenen sind so aufgebaut, dass sie mit allen Interaktionsformen gelöst werden können, da sonst keine Vergleichbarkeit zwischen den Interaktionsformen gewährleistet wäre. Die Aufgaben sind so gestellt, dass sie einzelne Probleme bei der Konstruktion in der VR darstellen. Der Proband muss diese Aufgaben nacheinander lösen.

Aufgabe 1 Ein virtuelles Objekt steht auf dem Tisch in der Szene. Auf dem Tisch ist ein virtueller Bereich markiert, der sich nicht mit dem virtuellen Objekt überschneidet. Der Proband soll das Objekt vollständig in den markierten Bereich bewegen. Größe von Objekt und Gebiet sowie der Abstand zwischen Tisch und Gebiet variieren zwischen verschiedenen Durchgängen.

Aufgabe 2 Ein virtuelles Objekt steht vor dem Proband auf dem Tisch. Der Proband soll das Objekt in die ihm angezeigte Richtung drehen. Die Richtung, in die ein Objekt gedreht werden soll, und die Größe des Objektes variieren.

Aufgabe 3 Zwei virtuelle Objekte liegen auf dem Tisch. Ihre farblich markierten Seiten sollen miteinander verbunden werden. Dabei ist es nötigt, dass die einzelnen Objekte zuvor rotiert oder bewegt werden. Position, Lage und markierte Seite sind zufällig gewählt.

Aufgabe 4 Ein Konstrukt, das aus mehreren virtuellen Bausteinen besteht, liegt auf dem Tisch vor dem Probanden. Der Proband soll die Verbindung zwischen zwei Bausteinen in dem Konstrukt lösen. Die zu lösende Verbindung ist farblich markiert und wird zufällig gewählt.

Aufgabe 5 Mehrere virtuelle, nummerierte Objekte liegen verteilt auf dem Tisch. Der Proband soll die Objekte in der vorgegebenen Reihenfolge markieren. Nachdem alle Objekte markiert wurden, soll die Markierung von farblich gekennzeichneten Objekten wieder aufgehoben werden. Die Position, Nummerierung und die Anzahl der Objekte variiert ebenso wie die Kennzeichnungen.

3.5 Untersuchungsmethode

Eine Aufgabe wird einem Probanden mehrfach gestellt, sodass die Fallzahlen für Schlussfolgerungen, die sich auf statistischen Werten stützen sollen, statistisch relevant ist. Neben der Befragung der Probanden wird auch die Zeit gemessen, die benötigt wird, um die einzelnen Aufgaben zu lösen. Ebenso wird die Benutzerpräferenz im Hinblick auf Art und Dauer von Interaktionsformen gemessen. Eine Anzahl von Probanden die größer als 15 ist, sollte für eine erste Evaluierung ausreichend sein.

4 Fazit

In diesem Abschnitt wird der aktuelle Stand der Arbeit aufgezeigt und es werden vorhandene Chancen und Risiken erläutert. Der letzte Abschnitt gibt eine kurze Zusammenfassung der Arbeit wieder.

4.1 Aktueller Stand

In Projekt 1[2] wurde ein Konzept erarbeitet und es wurden große Teile der Sensorabstraktionsschicht Trame umgesetzt. Durch die Entwicklung von einem kleinen Szenario konnten erste Eindrücke gewonnen werden, wie sich die Arbeit in Mixed Reality anfühlt. In dem Szenario kann ein Nutzer eine virtuelle Murmelbahn mithilfe von realen Objekten aufbauen. Die realen Würfel (Abb. 4.1, links) repräsentieren dabei verschiedene Grundbausteine für eine Murmelbahn (Abb. 4.1, mittig und rechts). In einer Simulation kann man anschließend Murmeln durch diese Bahn rollen lassen und kann damit überprüfen, ob die eigene Konstrukion funktioniert.

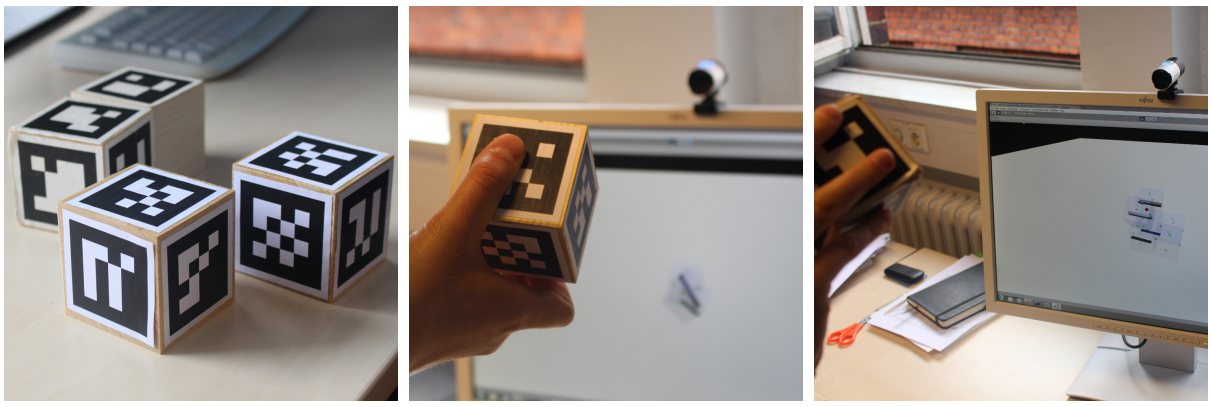

Abb. 6. Rotationsinvariantes markerbasiertes Tracking von Würfeln, the virtuelle Bausteine repräsentieren. (Quelle: `http://i2e.informatik.haw-hamburg.de`)

[2] Weitere Informationen zu Projekt 1 können dem Projektbericht entnommen werden: `http://i2e.informatik.haw-hamburg.de/assets/docs/p1/p1_blank_2014.pdf`.

Die Ergebnisse von Projekt 2 sind noch nicht vollständig. Es wurde weiter an der Abstraktionsschicht gearbeitet und das Templatematching wurde implementiert. Für die Abstraktionsschicht Trame wurde eine C#-Variante umgesetzt und es wurden verschiedene Erweiterungen entwickelt. Es ist nun möglich, Skelette zu serialisieren und deserialisieren. Trame besitzt eine Webservicekomponente und kann somit auch Daten über HTTP streamen. Die erste Version des Templatematchings hält sich strikt an den Algorithmus von [KNQ12].

Ein weiteres Ziel von Projekt 2 ist die Kommunikation der verschiedenen Teilsystem miteinander. Das Objekttracking, die Gestenerkennung, die Visualisierung, das Mobile Device und die Konstruktionslogik kommunizieren bereits miteinander. Die Kommunikation erfolgt durch eine eigene Middleware, die in mehreren Programmiersprachen implementiert ist.

Zudem sollen weitere Szenarien entworfen werden, die einzelne Arbeitsabläufe in einer Mixed-Reality-Umgebung simulieren. Für eine Machbarkeitsstudie wird ein "Sandkasten" implementiert, in dem ein Nutzer virtuelle Objekte durch physikalische Interaktion manipulieren kann. Der Nutzer trägt dabei das Mobile Device und kann die Szene durch dieses betrachten.

4.2 Chancen

Sollten die getroffenen Annahmen korrekt sein, dann würde eine neues Natural User Interface für die räumliche Interaktion zur Verfügung stehen, das für weitere Betrachtungen und auch im produktiven Einsatz verwendet werden kann.

Ebenso kann ein größeres Wissen und Verständnis für die Verwendung von Gesten bei der räumlichen Konstruktion in MR-Umgebungen erworben werden. Dazu gehören Vorlieben der Nutzer, Geschwindigkeit und Genauigkeit bei der Konstruktion, welche Probleme durch den gewählten Ansatz gelöst werden können und welche nicht gelöst werden bzw. neu entstehen.

Durch den Einsatz einer Abstraktionsschicht können verschiedene, kostengünstige Sensoren verwendet werden. Somit besteht die Möglichkeit, ein kostengünstiges Tool für die Erstellung von Prototypen in Konstruktionsprozessen zu realisieren, wie in Abschnitt 1.1 beschrieben.

4.3 Risiken

Ein wichtiger Aspekt bei der performanten Berechnung von physikbasierter Interaktion ist das kinetische Hand-Arm-Modell, das nur zu einem gewissen Grad realistisch sein kann. Eine zu große Abweichung zwischen der Form der realen Hand und der des Modells kann zu einem Verlust der Akzeptanz von Seiten der Nutzer führen.

Eine mögliche Alternative wäre der Einsatz von Partikelfiltern, wie in [HKI+12] beschrieben. Diese Art von Verfahren sind präziser, da sie wesentlich höher auflösen können und somit eine bessere Approxiamtion bieten, als das gewählte Verfahren. Im Gegensatz dazu sind sie aber auch rechenintensiver [HKI+12].

Das Templatematching, das zum Einsatz kommen soll, wurde für zweidimensionale Eingaben entwickelt. Dementsprechend muss es angepasst werden, damit auch dreidimensionale Eingaben analysiert und interpretiert werden können. Die theoretischen Aspekte wurden dabei durchdrungen, jedoch lässt sich bisher noch keine Aussage zu Ergebnissen aus dem praktischen Einsatz treffen, da es noch keine Untersuchungen in diesem Bereich gab. Es ist zu erwarten, dass die verwendete Methode für die Untersuchung der These 1 ausreicht. Es bleibt jedoch ein Restrisiko. Dieses Risiko sollte am Ende von Projekt 2 bereits geklärt sein und somit nicht die weitere Durchführung der Arbeit gefährden.

Ein wichtiger Aspekt bei dem Einsatz von Human-Computer-Interfaces ist die Reaktionszeit, die benötigt wird, um auf eine Eingabe des Nutzers zu reagieren. Eine zu lange Wartezeit kann zu einem Missverständnis führen, da ein Ergebnis nicht mehr direkt mit einer Aktion in Verbindung gebracht werden kann.

Die Antwortzeiten sollten so gering wie möglich gehalten werden. Führt der Nutzer eine längere Geste aus, dann sollten ihm Zwischenergebnisse aufgezeigt werden. In [KNQ12] werden die wahrscheinlichsten Eingaben, die ein Nutzer gerade tätigen will, angezeigt. Ebenso sollte darüber nachgedacht werden, inwieweit langandauernde Gesten überhaupt sinnvoll als Eingabe genutzt werden können und ob nicht besser Teilbewegungen als Gesten erkannt werden sollten.

4.4 Zusammenfassung

In dieser Arbeit wurde eine Motivation gegeben, welche herausstellt, warum die räumliche Interaktion in einer Mixed-Reality-Umgebung relevant ist. Es wurde ein Lösungsansatz vorgestellt und eine These aufgestellt. Ebenso wurde gezeigt, wie die aufgestellte These evaluiert werden kann. Das Konzept der Evaluierung ist der Hauptaspekt dieser Arbeit. Die Chancen und Risiken, auf die im letzten Abschnitt eingegangen wurden, bilden den Abschluss der vorliegenden Arbeit. Bezüglich der Risiken wurden ebenfalls Ideen vorgeschlagen, mit deren Hilfe die einzelnen Risiken minimiert werden können.

Literatur

[HKI+12] HILLIGES, Otmar ; KIM, David ; IZADI, Shahram ; WEISS, Malte ; WILSON, Andrew: HoloDesk: direct 3d interactions with a situated see-through display. In: *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems* ACM, 2012, S. 2421–2430

[KNQ12] KRISTENSSON, Per O. ; NICHOLSON, Thomas ; QUIGLEY, Aaron: Continuous Recognition of One-handed and Two-handed Gestures Using 3D Full-body Motion Tracking Sensors. In: *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*. New York, NY, USA : ACM, 2012 (IUI '12). – ISBN 978–1–4503–1048–2, S. 89–92

[MK94] MILGRAM, Paul ; KISHINO, Fumio: A taxonomy of mixed reality visual displays. In: *IEICE TRANSACTIONS on Information and Systems* 77 (1994), Nr. 12, S. 1321–1329

[OKA11] OIKONOMIDIS, Iasonas ; KYRIAZIS, Nikolaos ; ARGYROS, Antonis A.: Markerless and efficient 26-dof hand pose recovery. In: *Computer Vision–ACCV 2010*. Springer, 2011, S. 744–757

[SB14] STEINICKE, Frank ; BRUDER, Gerd: A Self-Experimentation Report about Long-Term Use of Fully-Immersive Technology. In: *Proceedings of the ACM Symposium on Spatial User Interaction (SUI)*, ACM Press, 2014, (accepted)

[SYW08] SONG, Peng ; YU, Hang ; WINKLER, Stefan: Vision-based 3D finger interactions for mixed reality games with physics simulation. In: *Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry* ACM, 2008, S. 7