



FAKULTÄT INFORMATIK
INSTITUT FÜR SOFTWARE- UND MULTIMEDIATECHNIK
PROFESSUR MEDIENGESTALTUNG

Bakkalaureatsarbeit zum Thema

Kombination gestenbasierter Interaktion in multimodalen Anwendungen

eingereicht am: 25.01.2011
eingereicht von: Dana Henkens
Matrikelnummer: 3249276

Verantwortlicher Hochschullehrer: Prof. Dr.-Ing. habil. Rainer Groh
Wissenschaftlicher Betreuer: Dipl.-Medieninf. Dietrich Kammer

EIDESSTATTLICHE ERKLÄRUNG

Hiermit versichere ich, die vorliegende Arbeit zum Thema

„Kombination gestenbasierter Interaktion in multimodalen Anwendungen“

selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Dresden, 25. Januar 2011

Dana Henkens

INHALTSVERZEICHNIS

1	Einleitung	3
1.1	Zielstellung	3
1.2	Aufbau der Arbeit	4
2	Grundlagen	5
2.1	Mensch-Computer-Interaktion	5
2.2	Benutzerschnittstellen	8
2.3	Eingabeaspekte	13
2.4	Gesten	20
3	Verwandte Arbeiten	23
3.1	Multimodale Anwendungen	23
3.2	Verwendung von Gesten	26
3.3	Gestenbeschreibung	28
3.4	Touch- und Point-Interaktion	32
4	Synthese	37
4.1	Kombination von Touch- und Point-Interaktion	37
4.2	Formalisierung räumlicher Handgesten	41
4.3	Formalisierung multimodaler Benutzereingaben	50
5	Multimodaler Showcase	53
5.1	Technologischer Aufbau	54
5.2	Implementierung	55
5.3	Herausforderungen	57
6	Zusammenfassung und Ausblick	59
	Literaturverzeichnis	61
	Abbildungsverzeichnis	66
	Tabellenverzeichnis	69

1 EINLEITUNG

Die Multitouch-Technologie ist auf dem heutigen IT-Markt nicht mehr weg zu denken. Ihr erfolgreicher Einsatz zeigt, wie sich neue Benutzerschnittstellen den Erfordernissen der Nutzer anpassen und zunehmend reale Handlungsmuster abbilden. Dennoch stellt sich der Einsatz berührungsempfindlicher Systeme nicht in jedem Anwendungsfall als praxistauglich und anwendbar heraus, da es Situationen gibt, in denen ein Nutzer Berührungen vermeiden möchte. Hierfür lassen sich Beispielsituationen aus dem häuslichen Alltag, in der Arbeitsumgebung und im öffentlichen Raum finden, in denen die Gefahr besteht, dass die Oberfläche des Gerätes verschmutzt oder beschädigt wird und die Bedienbarkeit damit gestört wird. Desweiteren bedingt das Konzept der Berührungseingaben einen geringen Abstand zum Display, was zum Nachteil wird, sobald die Aufwendungen des Agierenden steigen, um Teile der Benutzeroberfläche zu erreichen. In diesen Fällen sind alternative Interaktionstechniken gefordert, die vergleichsweise intuitiv und anwendbar sind. Das Fraunhofer Institut bezeichnet mit Multipointing-Interaktion, einer berührungslosen Steuerung von Systemen, die nächste Generation neuer User Interfaces. [P. Chojacki, 2008] Die Ablösung berührungsempfindlicher und greifbarer Schnittstellen durch ein berührungsloses Interaktionskonzept ist nicht abzusehen. Dennoch stellt das Point-Interface eine Option dar, den genannten Schwächen von Touch entgegenzuwirken. Um die Bedienung von Computer-Systemen gegenüber verschiedenen Alltagssituationen flexibler zu gestalten, liegt es nahe, sowohl berührungssensitive als auch berührungslose Eingaben in gestenbasierten Systemen zu unterstützen. Hierfür sind die Interaktionstechniken hinsichtlich ihres Zusammenwirkens in einem multimodalen Interface zu untersuchen.

1.1 ZIELSTELLUNG

Auf der Basis bekannter gestenbasierter Bedienkonzepte sowie Studien, die den Einsatz der berührungslosen und berührungsempfindlichen Interaktionstechniken evaluieren, ist es Ziel dieser Arbeit den Mehrwert einer Kombination der Bedienkonzepte aufzuzeigen und ihre Eignung hinsichtlich Schlüsselfaktoren wie Ausrichtung und Dimension des Displays herauszustellen. Unter dem Gesichtspunkt der Multimodalität werden dabei sowohl technische, als auch formale Aspekte näher untersucht. Im Mittelpunkt steht dabei die Erweiterung des Konzepts zweidimensionaler Touch-Gesten um eine Systematik zur Darstellung von Gesten im dreidimensionalen Raum. Dazu werden Potentiale der Gestenbeschreibungssprache für Mul-

titouch (GeForMT) aufgegriffen. Unabhängig von der Formalisierung wird ein multimodales Interaktionskonzept für die Navigation in einer Kartenanwendung prototypisch umgesetzt.

1.2 AUFBAU DER ARBEIT

Neben grundlegenden Begriffen der Interaktion mit Computersystemen, werden in Kapitel 2 zunächst Eigenschaften natürlicher, gestenbasierter und multimodaler Benutzerschnittstellen erläutert. Des Weiteren werden technologische Aspekte und Eigenschaften von Gesten thematisiert. In Kapitel 3 sind verwandte Arbeiten verarbeitet. Die Recherche zu multimodalen Anwendungen, die die Interaktion mit Gesten unterstützen, ist Ausgangspunkt für die Entwicklung neuer Bedienkonzepte. Es wird auf die Verwendung von berührungsempfindlichen als auch berührungsloser Gesten eingegangen. Darauf aufbauend werden Ansätze der formalen Beschreibung von Gesten vorgestellt und insbesondere das Konzept der Gestenbeschreibungssprache für Multitouch (GeForMT) erläutert. Es werden weiterhin zwei Studien herangezogen, die eine Kombination der verschiedenen Interaktionstechniken in Betracht ziehen. In Kapitel 4, der Synthese, werden auf dieser Grundlage Schlüsse gezogen, in wie weit sich die Interaktionstechniken eignen. Weiterhin werden Vorschläge erarbeitet, wie die Syntax der GeForMT adaptiert werden kann, um sie für die Beschreibung räumlicher Gesten heranzuziehen. Darauf baut die Untersuchung von Potentialen zur Verknüpfung verschiedenartiger Interaktionstechniken auf, mit der Beziehungen auf multimodaler Ebene beschrieben werden können. In Kapitel 5 wird das Konzept des Anwendungsprototypen vorgestellt und ein Einblick in die Umsetzung gegeben. Die Ergebnisse der Arbeit werden abschließend in Kapitel 6 resümiert und ein Ausblick für weiterführende Arbeiten gegeben.

2 GRUNDLAGEN

Das folgende Kapitel beschäftigt sich mit der gestenbasierten Interaktion zwischen Mensch und Computer. Es werden zunächst allgemeine Grundsätze aus dem Bereich der Mensch-Computer-Interaktion dargelegt und mit der Klärung von Begriffen das allgemeine Verständnis gestärkt. Nicht zuletzt dienen die grundlegenden Ausführungen der Eingrenzung des Themas. Die Definition und Abgrenzung der natürlichen und gestenbasierten Benutzerschnittstelle ist unter diesem Gesichtspunkt ein Hauptanliegen. Weiterhin spielt die Multimodalität im Kontext einer Kombination verschiedener Interaktionstechniken eine bedeutende Rolle. Darauf aufbauend werden technologische Aspekte für die Erkennung gestenbasierter Eingaben aufgegriffen. Der letzte Abschnitt widmet sich den Gesten und ihren Eigenschaften.

2.1 MENSCH-COMPUTER-INTERAKTION

Mensch-Computer-Interaktion (MCI) bezeichnet ein Teilgebiet der Informatik, welches den Fokus auf die benutzergerechte Gestaltung interaktiver Systeme legt. Neben der Software-Ergonomie, dem E-Learning, Kontextanalyse und dem Informationsdesign als wichtige Teilgebiete der MCI, ist das **Interaktionsdesign**¹ der zutreffende Rahmen, in dem sich diese Arbeit bewegt. Während es beim **Informationsdesign**² um die Darstellung und Wahrnehmung komplexer Informationen unter Betrachtung wahrnehmungspsychologischer Aspekte geht, befasst sich das Interaktionsdesign mit der Gestaltung des Verhaltens von Systemen, Produkten und Diensten in ihrer Wechselbeziehung zum Nutzer. (vgl. [Wikipedia, 2010]) Oft wird Interaktionsdesign mit dem Begriff **Interfacedesign** gleichgesetzt. Hierbei geht es aber um die konkrete Visualisierung der Schnittstelle. Eine klare Trennung kann an der Stelle nicht vorgenommen werden, da die Konzeption der Interaktion mit einem System auch die Gestaltung der Systemoberfläche nach sich zieht. (vgl. [Wikipedia, 2010]) Grundlegend kann zwischen Eingabe des Nutzers und Auswirkungen auf das System und die Ausgabe des Systems und Auswirkung auf den Nutzers unterschieden werden. Die vorliegende Arbeit beschäftigt sich insbesondere mit Eingabemethoden natürlicher Benutzerschnittstellen.

¹siehe [Herczeg, 2006, S. 7]

²von Robert E. Horn geprägter Begriff (vgl. [Kadera, 2007])

2.1.1 Begriffe

Um über **Interaktion** zwischen Mensch und Computer im Detail sprechen zu können, werden Teilaspekte der Interaktion nach [Boles, 1998, Kapitel 9.2] begrifflich zusammengefasst. Der Typ der Information, den ein Nutzer eingeben kann, wie z.B. die Positionierung oder Auswahl, wird als **Interaktionsaufgabe** bezeichnet. Nach [Foley et al., 1992] werden die fünf Basisinteraktionsaufgaben Positionierung, Texteingabe, Auswahl, Quantifizierung, Bestätigung und zusammengesetzte Interaktionsaufgaben unterschieden. Eine **Interaktionsform** fasst Paradigmen der Informationseingabe zusammen, von denen in [Boles, 1998, Kapitel 9.2.3] vier verbreitete Klassen von Eingabeformen genannt werden: Interaktionssprachen (Kommandosprachen), Masken, User-Interface-Komponenten und Direkte Manipulation. **Interaktionstechniken** beschreiben nach [Foley et al., 1992] die Möglichkeiten Eingabegeräte einzusetzen, um eine Interaktionsaufgabe zu erfüllen. Spezifiziert wird die Art und Weise wie Informationen eines Typs, unter Berücksichtigung des Eingabegerätes, an das System übermittelt werden. Beispielsweise wird für eine Positionierung üblicherweise eine Maus verwendet. Denkbar ist aber auch die Nutzung eines Fingers auf einem Touchscreen oder die Eingabe numerischer Koordinaten über eine Tastatur. Interaktionsformen stellen dabei Hilfsmittel dar, um Interaktionstechniken zu verdeutlichen. Die menschlichen Aspekte der Interaktion beziehen sich insbesondere auf das Wunschdenken und die Erwartungen des Nutzers. So wird unter dem **Interaktionsziel** das Ziel verstanden, dass der Nutzer in seiner Erwartungshaltung vor Augen hat. Jede Nutzereingabe ist somit mit einer mehr oder weniger konkreten Vorstellung verbunden, welche Auswirkungen seine Eingabe hat. Die tatsächliche Reaktion des Systems wird als **Interaktionsauswirkung** bezeichnet. (vgl. [Boles, 1998, Kapitel 9.2.5])

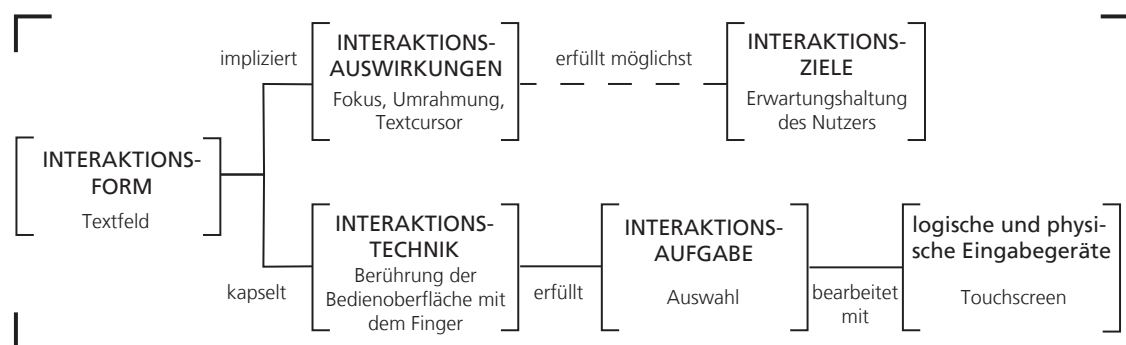


Abbildung 2.1: Terminologische Zusammenhänge am Beispiel "Auswahl eines Textfeldes" in Anlehnung an [Boles, 1998, Kapitel 9.2]

Abbildung 2.1 verdeutlicht die terminologischen Zusammenhänge anhand eines Beispiels: Der Nutzer hat die Aufgabe ein Textfeld zu selektieren, um eine Tastatureingabe in dem Feld einzuleiten. Als Eingabegerät steht ihm dabei ein Touchscreen zur Verfügung. Die Auswahl hat zur Folge, dass ein Fokusrahmen um das Eingabefeld erscheint und ein blinkender Textcursor gesetzt wird, der die Textposition markiert und zur Eingabe auffordert.

2.1.2 Interaktion und Kommunikation

Im Sinne des wechselseitigen Einwirkens von Akteuren oder Systemen ist der Begriff Interaktion eng mit dem der Kommunikation verknüpft und wird oft sogar synonym verwendet. Nach [Herczeg, 2006] verbergen sich hinter den Begrifflichkeiten grundlegend unterschiedliche Ansichten auf das Zusammenwirken von Mensch und Computer. Dabei stellt sich die grundlegende Frage, ob der Mensch mit dem Computer kommuniziert oder im weitesten Sinne handelt und agiert. Demnach werden zwei Modelle unterschieden:

Das **Kommunikationsmodell** vertritt den Ansatz, dass Mensch und Computer ähnlich wie Menschen untereinander kommunizieren. Man spricht auch von Dialogsystemen. (vgl. [Herczeg, 2006, S. 11]) Das Modell geht von einem Sender aus, der Informationen über einen Kanal an einen Empfänger übermittelt - einer gibt Informationen weiter, der andere nimmt sie auf. Die Rollen der beiden Kommunikationspartner Informant und Informierter zu sein, wechseln sich im Dialog ab. (vgl. [Herczeg, 2006, S. 11 u. 14])

Eine andere Ansicht auf die Kommunikation zwischen Mensch und Computer stellt das **Welt- oder Handlungsmodell**, oder auch **handlungsorientiertes Modell**, dar. Es lässt sich durch Einbezogenheit des Nutzers und dessen direkten Zugriff auf die Information charakterisieren, wodurch der Computer kaum noch wahrgenommen wird. HERZEG beschreibt das Modell wie folgt:

“Das Modell versucht den Computer als Handlungsraum zu gestalten, in dem Menschen, ähnlich wie in der physischen Welt, Objekte erzeugen, wahrnehmen oder verändern.” [Herczeg, 2006, S. 11]

Derartige Modelle stammen aus der späteren Entwicklungsphase der Informationstechnik, in der man zunächst durch grafische Aufbereitung und Metaphern eine Vereinfachung der Mensch-Computer-Beziehung anstrebte. Ein bekanntes Beispiel ist die Desktop-Metapher. Unterstützt wird der Nutzer in seinem Handeln durch Objekte und Werkzeuge. Objekte sind virtuelle Repräsentationen realer Gegenstände. Sie besitzen veränderliche Zustände und können durch Manipulation erzeugt, erkannt, verändert, kombiniert oder wieder beseitigt werden. Werkzeuge stellen in dem Sinne Hilfsmittel der Manipulation dar. Sie müssen in den jeweiligen Benutzerschnittstellen in geeigneter Weise kommuniziert und anwendbar gemacht werden. [Herczeg, 2006, S. 22 ff.]

2.1.3 Gestaltungsprinzipien

Nach [Norman und Nielsen, 2010] lassen sich sieben Grundprinzipien der Gestaltung von Interaktion formulieren, die beim Entwurf neuer Interfaces beachtet werden sollen. In [Schacht, 2010, S. 11] werden sie unter folgenden Begriffen zusammengefasst: *Sichtbarkeit, Rückmeldung, Konsistenz, umkehrbare Operation, Auffindbarkeit, Skalierbarkeit und Verlässlichkeit*.

Angelehnt an diese Prinzipien erklärt SAFFER die Gestaltungsgrundsätze in Bezug auf eine gestenbasierte Interaktion mit folgenden Eigenschaften einer Benutzerschnittstelle: *erfassbar, vertrauenswürdig, reaktionsfähig, geeignet, bedeutungsvoll, intelligent, verspielt, angenehm* und *zielgruppenorientiert*. (vgl. [Saffer, 2009, S. 19]) Die tragendsten Eigenschaften werden im folgenden erläutert. *Erfassbar* bedeutet, dass über die Eigenschaften manipulierbarer Objekte bereits die Möglichkeiten der Interaktion nahegelegt werden. Gesten müssen hinsichtlich der Situation, der Kultur und dem Kontext *geeignet* sein. Insbesondere kulturelle Unterschiede führen zu Verständigungsproblemen und Fehlinterpretationen. Rückmeldungen des Systems (*reaktionsfähig*) sind wichtig, um dem Nutzer das Gefühl der Kontrolle zu überlassen. *Angenehm* ist eine Eigenschaft, die sich in Ästhetik und Bedienbarkeit wieder spiegelt und damit das Vertrauen des Nutzer gewinnt. Ein Interface ist *zielgruppenorientiert*, wenn es den Bedürfnissen und Erfordernissen der Nutzer entgegenkommt. Der Anwender steht im Mittelpunkt des Designprozesses, für den das System optimiert wird. In dem Zusammenhang sind die verschiedenen Leistungsvoraussetzungen der Nutzer zu beachten.

2.2 BENUTZERSCHNITTSTELLEN

Eine computergestützte Benutzerschnittstelle (engl.: User Interface) umfasst nach DIN EN ISO 9241-110 alle Bestandteile eines interaktiven Systems, sowohl Hardware als auch Software, welche dem Nutzer Informationen und Steuerelemente zur Verfügung stellen, um eine Arbeitsaufgabe zu erledigen. Eine Benutzerschnittstelle eröffnet dem Nutzer verschiedene Möglichkeiten mit dem System zu interagieren. Der Handlungsraum formt sich nach dessen Gestalt und legt letztlich fest, wie der Nutzer mit dem System kommunizieren muss.

Abbildung 2.2 zeigt das Modell einer Benutzerschnittstelle in ihrer Wechselbeziehung zwischen Mensch und Computer nach [Groh, 2005]. Das Interface steht als Kommunikationsmittel zwischen den Systemelementen Rechner, Realität und Nutzer.

“Die Kommunikation zwischen dem Rechner und der Realität besitzt einen klaren (informations-)technischen Charakter. Die kommunikative Verbindung von Nutzer und Computer ist geprägt vom instrumentellen (werkzeughaften) Verhältnis, das der Nutzer zu seiner Technik entwickelt. Die Beziehung des Nutzers zur Realität ist dagegen vom Ziel- und Wunschdenken gesteuert.” [Groh, 2005, S. 31]

2.2.1 Natürliche Benutzerschnittstelle

Nach [Lee, 2010b] sind die Grundzüge der natürlichen Benutzerschnittstelle (engl.: Natural User Interface - NUI) eine intuitive Bedienung und effektive Umsetzung einer Absicht in eine Aktion des Systems mit wenig Aufwand. Mit den Worten *“[...] NUI generally refers to an interface that is highly intuitive and effectively becomes invisible to the user when performing a task.”* erklärt er, dass die eigentliche Aufgabe, welche mit dem Rechner gelöst werden soll, in den Vordergrund tritt, während die Schnittstelle unscheinbar wirkt.

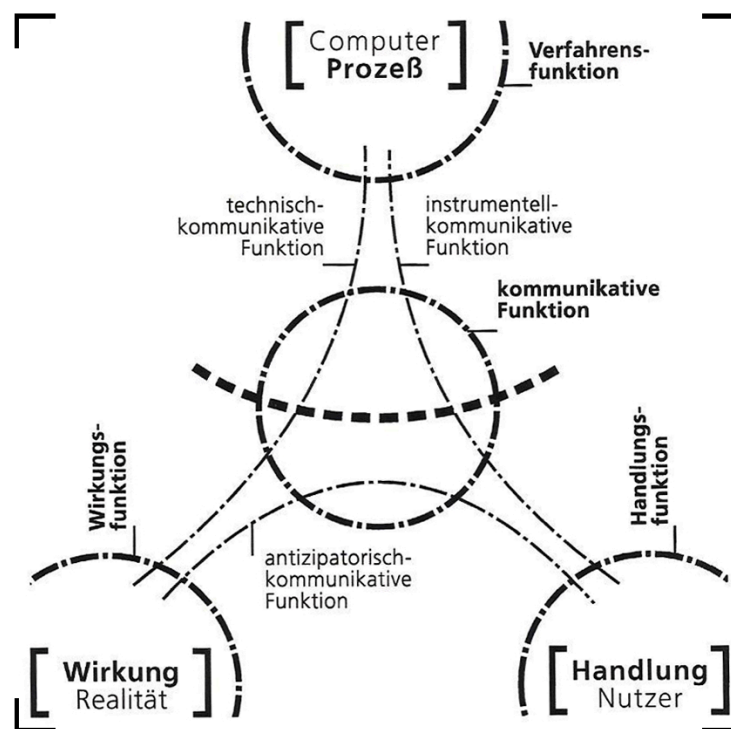


Abbildung 2.2: Modell einer Benutzerschnittstelle (vgl. [Groh, 2005, S. 31, Abb. 6])

"A natural user interface is a user interface designed to reuse existing skills for interacting directly with content." [Blake, 2010]

Der Definition nach BLAKE können weitere Merkmale natürlicher Benutzerschnittstellen entnommen werden. Zunächst einmal ist das Interface *"designed"* und folgt letztlich Gestaltungsprinzipien im Interesse einer einfachen Bedienung. Die Wiederverwendung existierender alltäglich gebrauchter Fertigkeiten des Nutzers sind von Bedeutung um die Bedienung natürlich und intuitiv zu gestalten. Der Mensch besitzt von Natur aus Fähigkeiten der verbalen und nonverbalen zwischenmenschlichen Kommunikation und der Interaktion mit seiner Umgebung, die dafür aufgegriffen werden. Die Gestalt der Benutzeroberfläche präsentiert sich durch Metaphern, welche aus der realen Welt entnommen sind und zu dem Verständnis des Nutzers im Umgang mit dem System beitragen. Weiterhin interagiert der Nutzer in direkter Weise mit dem Inhalt. Die primäre Interaktionsform ist die der direkten Manipulation. In Hinblick auf Grafische Benutzerschnittstellen (engl.: Graphical User Interface - GUI), haben Steuerelemente wie Schaltflächen oder dergleichen eine zweitrangige Bedeutung. Während sich Grafische Benutzerschnittstellen und Kommandozeilen-Interfaces (engl.: Command Line Interface - CLI) nach ihren Eingabegeräten richten und hinsichtlich dieser definiert sind, wird das NUI nach der Auffassung von BLAKE mit der Art der Interaktion in Beziehung gesetzt. Solange existierende und natürliche Fertigkeiten in der Interaktionstechnik Wiederverwendung finden, kann jede Interface-Technologie zum Einsatz kommen. In diesem Sinne beschreibt BLAKE das NUI als Nachfolger des CLI und GUI, welches ohne Instrument zur Abbildung und

Übersetzung von Nutzerhandlungen auskommt.

BOLLHOEFER beschreibt das NUI als *“Interface, welches unmittelbar durch einen oder mehrere Sinne des Nutzers bedient wird”* und unterscheidet aktive NUIs, in denen der Nutzer direkt aktiv mit dem System interagiert, und passive NUIs, in denen Nutzer eine passive Rolle einnehmen und eine Kommunikation ohne Handeln entsteht. Als Beispiele für aktive NUIs sei Multitouch genannt. Passiven NUIs kann ein Iris-Scan oder die Systemsteuerung durch Synapsenaktivitäten des Gehirns zugeordnet werden. [Bollhoefer et al., 2009, S. 6]

2.2.2 Gestenbasierte Benutzerschnittstelle

Gestenbasierte Benutzerschnittstellen stellen eine Untermenge der NUI dar, indem sie sich primär auf die Körpersprache und Bewegungsmuster beziehen. Gesten sind eine besonders intuitive Art der Kommunikation, die wir täglich gebrauchen. Die Bedienung von Computersystemen durch Gesten werden insbesondere von Interaktionsformen der direkten Manipulation unterstützt.

In [Saffer, 2009, S. 4] werden zwei Typen gestenbasierter Benutzerschnittstellen unterschieden: **Touch User Interfaces** und **Free-form Gestural Interfaces**. Es wird auch von berührungssensitiver oder Touch-Interaktion und analog von berührungsloser Interaktion gesprochen. Während sich die Touch-Interaktion auf Gesten beschränkt, welche unter Berührung einer Oberfläche erkannt werden, ist für die Erkennung berührungsloser Gesten im Raum nicht zwingend ein Eingabegerät notwendig. Der Körper allein agiert mit dem System und lässt eine Vielzahl möglicher Gesten zu. Kapitel 2.4 geht dazu gesondert auf das Thema Gesten ein.

Touch-Interface

Touchscreens sind berührungsempfindliche Systeme, dessen Eingabebereich sich objektiv und aus Nutzersicht mit dem Ausgabebereich decken. Das Display dient also neben der Darstellung gleichzeitig der Bedienung durch Berührungen mit den Fingern und Händen. *Touch* bezeichnet in diesem Sinne die Interaktionstechnik dieser Schnittstelle. Multitouch drückt die Eigenschaft des Erkennungssystems aus, mehrere Finger und ggf. mehrere Hände erkennen zu können. Mit dieser Charakteristik geht oftmals einher, dass die Bedienung durch mehrere Nutzer möglich ist.

Point-Interface

Point ist ein Begriff, der in Zusammenhang mit dem berührungslosen Interaktionskonzept des vom Fraunhofer Institut entwickelten iPoint Presenter (siehe Abschnitt 2.3.3) verwendet wird. Die Bezeichnung rührt daher, dass das System vorrangig Zeigehandlungen mit den Fingern unterstützt. Point wird im Folgenden als Synonym für gestenbasierte berührungslose Interaktion im dreidimensionalen Raum mit Orientierung an dem digitalen Ausgabebereich verwendet. Analog zu Touch-Interfaces bezeichnet Mulptpoint die Systemeigenschaft, mehrere finger- und handbasierte Eingaben simultan verarbeiten zu können.

2.2.3 Multimodale Benutzerschnittstelle

In Abschnitt 2.2.1 wurde erklärt, dass eine natürliche Benutzerschnittstelle durchaus ein multimodalen Systemkonzept folgen kann. Der Begriff Multimodalität wird in der Literatur nach verschiedenen Auffassungen interpretiert und demnach in unterschiedlichen Kontexten verwendet. Nach [Dix et al., 1993] zeichnet sich ein multimodales System dadurch aus, dass mehrere Sinneskanäle des Menschen, das heißt Sehen, Hören, Riechen, Schmecken, Tasten, bedient werden. Gemeint ist also die sensorische Modalität und damit die Informationsaufnahme durch den Menschen. [Blattner und Dannenberg, 1992] und [Lee, 2010b] hingegen beschreiben ein multimodales System als Anwendung, die mehrere Arten der Interaktion unterstützt und damit mehrere Möglichkeiten der Informationsweitergabe bietet. (vgl. [Boles, 1998, Kapitel 9.1.3])

In [Hedicke, 2000] werden diese beiden Ansichten auf Multimodale Benutzerschnittstellen (engl.: Multimodal User Interfaces - MMUI) zusammengefasst. Die Optionen der Informationsaufnahme werden als **Wahrnehmungsmodalitäten** und die Möglichkeiten der Informationsweitergabe als **Aktionsmodalitäten** bezeichnet. Neben traditionellen Aktions- und Wahrnehmungsmodalitäten umfasst die Definition nach [Hedicke, 2000] weitere kommunikative Mittel der Interaktion, wie zum Beispiel natürliche Sprache, einschließlich Gestik, sowie technische Eigenschaften einer Schnittstelle. (vgl. [Hedicke, 2000, S. 204]) Es ergeben sich aus technischer Sicht unter Vernachlässigung von Geschmacks- und Geruchssinn drei abgrenzbare Schnittstellenkomponenten, die sich an den menschlichen Sinnesmodalitäten orientieren. In Abbildung 2.3 werden sie als visuelles, auditives und haptisches Interface dargestellt. Jede Schnittstellenkomponente stellt auf der einen Seite die Eingabeoptionen und auf der anderen Seite ihre Ausgabeformate bereit, die mit der entsprechenden Sinnesmodalität des Interfaces in Verbindung stehen.

In engem Zusammenhang mit der Multimodalität eines interaktiven Systems, steht der Begriff **Wahrnehmungsgesteuerte Benutzerschnittstelle** (eng.: Perceptual User Interface - PUI). Das PUI konzentriert sich ebenso auf die Bedienung mehrerer Sinne des Nutzers und unterstützt mehrere Eingabemodalitäten, fokussiert darüber hinaus die computerseitige Unterstützung perzeptiver Fähigkeiten der Schnittstelle. Perzeptiv bedeutet, dass die Wahrnehmung und Informationsverarbeitung hauptsächlich unbewusst vonstatten geht und im Bewusstsein des Anwenders Bilder von Teilaspekten der Wirklichkeit entstehen. (vgl. [Rötting und Seifert, 2005, S. 285]) Das Interface versucht durch den Einsatz verschiedener Wahrnehmungs- und Aktionsmodalitäten der Realität nahezukommen. Eine Kombination visueller, sprachbasierter und gestenbezogener Interfacekomponenten soll dadurch die Kommunikation erleichtern und natürlicher gestalten. In [Geng et al., 2001, S. 1] wird auch von einem menschenähnlichen Verhalten des Computers gesprochen, daran angelehnt wie Menschen untereinander kommunizieren.

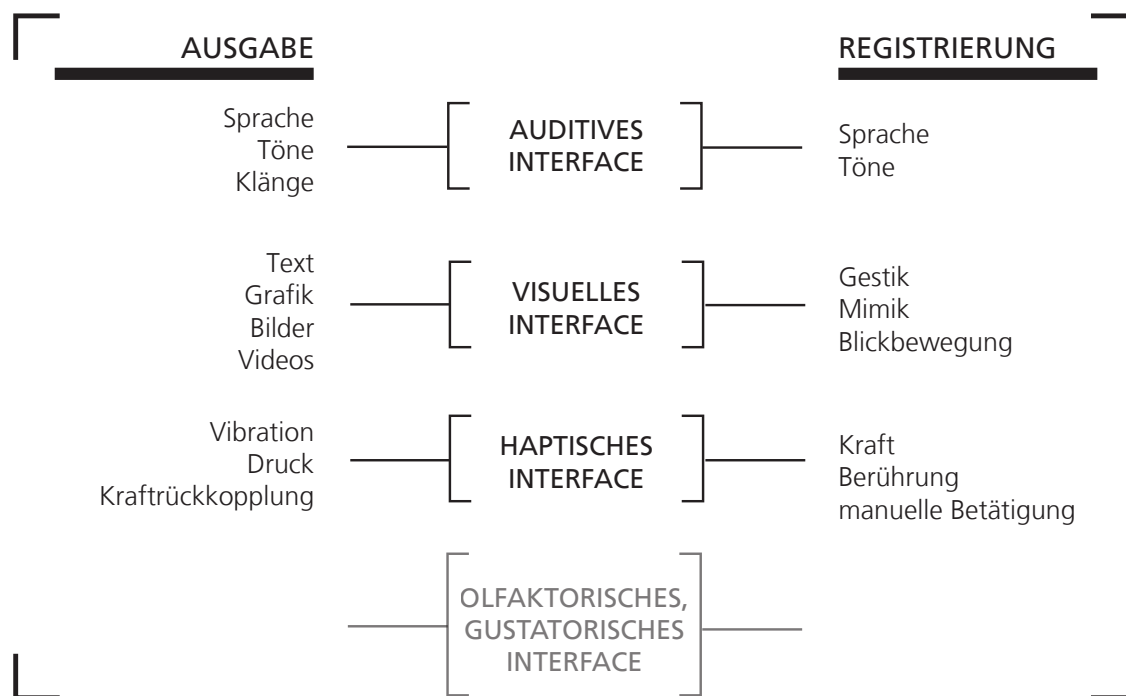


Abbildung 2.3: Technische Komponenten eines multimodalen Systems nach [Hedicke, 2000, S. 210]

Ziele der multimodalen Mensch-Maschine-Interaktion

Multimodale Systeme gestatten die Verlagerung der Informationsvermittlung auf mehrere Sinne. Dadurch wird die Beanspruchung eines Sinnes und die Belastung des Benutzers insgesamt reduziert. Die Ziele, die multimodale Anwendungen verfolgen, lassen sich nach [Rötting und Seifert, 2005, S. 286] unter den in Abbildung 2.4 aufgeführten Begriffen Natürlichkeit, Interaktionssicherheit, Flexibilität und Leistungsfähigkeit zusammenfassen. Sie werden im Folgenden erläutert.

Natürlichkeit bedeutet für den Nutzer, dass er seine Erfahrungen aus ähnlichen Situationen der Realität auf den Dialog mit dem Computer übertragen kann. Für die Interaktion ist kein spezielles computerbezogenes Wissen erforderlich.

Mit *Interaktionssicherheit* ist die Zuverlässigkeit und Robustheit des Systems gemeint, welche dadurch erreicht wird, dass Mehrdeutigkeiten von Nutzerhandlungen durch weitere Eingabeinformationen aufgelöst werden können. Eine redundante Informationsdarbietung sichert, dass die Information den Nutzer in geeigneter Weise erreicht.

Die *Flexibilität* der Interaktion bezieht sich auf örtliche, zeitliche und situative Parameter. Auch die Leistungsvoraussetzungen des Nutzers werden mit dem Angebot verschiedener Ein- und Ausgabeoptionen berücksichtigt. Die Bedienung des Systems kann somit auch für Personengruppen mit eingeschränkten Wahrnehmungsfähigkeiten gewährleistet werden.

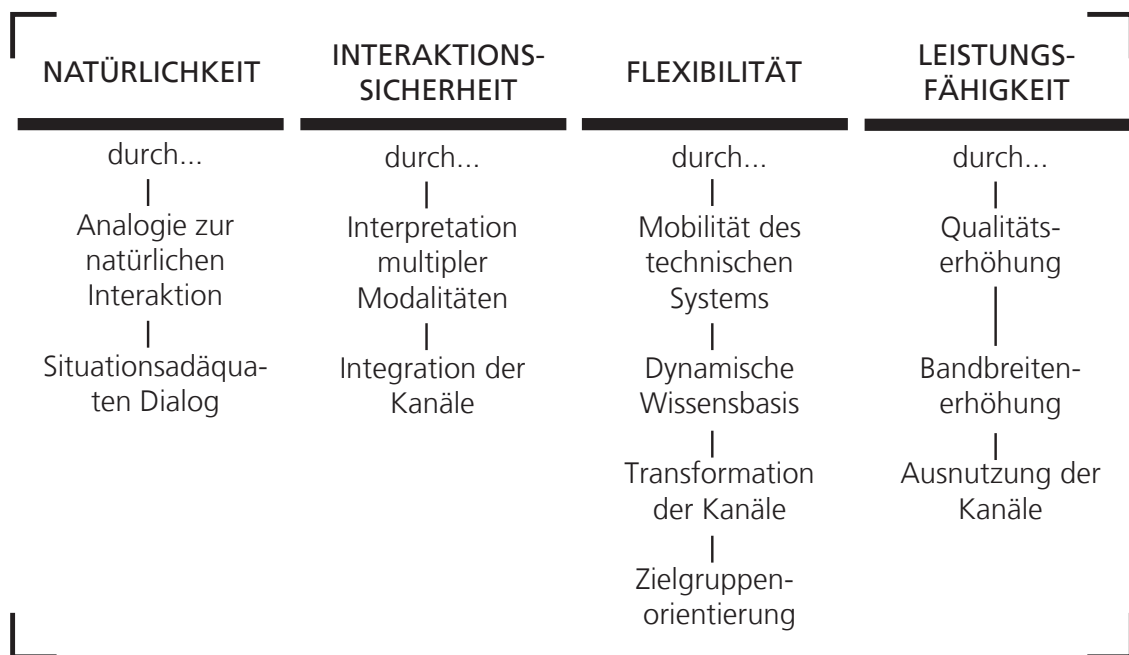


Abbildung 2.4: Ziele der multimodalen Mensch-Maschine-Interaktion nach [Rötting und Seifert, 2005, S. 286]

Die *Leistungsfähigkeit* eines multimodalen Systems nimmt in dem Sinne zu, dass die Qualität vermittelter Inhalte durch Ausschöpfung des Potentials der Informationskanäle erhöht wird. Beispielsweise kann ein Warnhinweis des Systems visuell, akustisch und zugleich über eine taktile Rückmeldung vermittelt werden, um ein Systemverhalten deutlich zu machen. Auf der anderen Seite stellt die Bandbreite ein Angebot an Möglichkeiten der Informationsweitergabe dar, in welchem der Nutzer je nach Erfordernis wählen kann.

Bei allen Vorteilen, die die Multimodalität mit sich bringt, bemerkt HEDICKE dennoch, dass Vorsicht in der Aussnutzung der Bandbreite, die sich über Wahrnehmungs- und Aktionsmodalitäten erstreckt, geboten ist. Es besteht die Gefahr, dass Nutzer mit der redundanten Informationsdarbietung überfordert werden. Bisher konzentrieren sich Anwendungen auf die visuelle Stimulation. Erfordert diese bereits die gesamte Aufmerksamkeit des Nutzers, hat dies schnell eine Reizüberflutung zur Folge, wenn weitere Sinne angesprochen werden. (vgl. [Hedicke, 2000, S. 207])

2.3 EINGABEASPEKTE

Interaktion entsteht durch den Informationsaustausch über Komponenten der Benutzerschnittstelle. Wie in Kapitel 2.2.3 gezeigt, sind drei Schnittstellenkomponenten relevant. Folgender Abschnitt befasst sich mit der Eingabeseite der haptischen und visuellen Schnittstellen. Nach dem Nutzerempfinden werden Systeme, welche Berührungseingaben erkennen, dem haptischen Interface zugeordnet. Die Erfassung von Posen und Bewegungen im Raum, erfordert ein optisches System oder zumindest Bewegungssensorik. Demnach wird die berührungslose Interaktion in das visuelle Interface eingeordnet.

2.3.1 Haptisches Interface

Die Eingabeoptionen in Touch-Interfaces sind auf Berührungseingaben beschränkt. Über den Tastsinn wird dem Nutzer ein haptisches Feedback gegeben. Unter dieser Einschränkung sind Eingaben nur auf einer Oberfläche möglich, die mit entsprechender Sensorik ausgestattet ist.

In [Meißner, 2010, S. 22] werden bereits Technologien, welche Berührungseingaben nach der Definition in Abschnitt 2.2.2 verarbeiten, ausführlich erklärt. Sie werden im Folgenden nochmals zusammengefasst und um induktive Touchscreens erweitert.

Resistive Touchscreens

Zwei elektrisch leitfähige Schichten sind stellenweise miteinander verbunden (4-8-Draht). Unter Druck auf dem Touchscreen kann der elektrische Widerstand zwischen den Schichten Aufschluss über die Position der Eingabe geben.

Beispiele: Elographics AccuTouch³, Nintendo DS⁴

Kapazitive Touchscreens

An den Rändern einer Glasschicht mit leitfähiger Beschichtung befinden sich Elektroden, die ein elektrisches Feld über die gesamte Oberfläche erzeugen. Es werden *kapazitive Oberflächen-Touchscreens* (engl.: Surface Capacitive Touchscreens) und *projizierte kapazitive Touchscreens* (engl.: Projected Capacitive Touchscreens) unterschieden.

Beispiele: Apples⁵ iPhone, iPod Touch oder iPad, MERL Diamond Touch⁶

Induktive Touchscreens

Über einen Induktionsstrom kann neben der Position der Eingabe auch der Druck auf die Oberfläche ermittelt werden. Insbesondere Grafiktablets nutzen die Druckempfindlichkeit der Technik für sich. Schwachstelle ist jedoch, dass Eingaben nur mit Hilfe eines speziellen Stiftes getätigt werden können.

Beispiele: Wacom Cintiq⁷

Optische Touchscreens

Optische Systeme zeichnen sich dadurch aus, dass Licht (zumeist im Infrarotbereich) von einer Kamera aufgenommen und Änderungen der Lichtverhältnisse ausgewertet werden. Folgende Ansätze werden unterschieden:

- *Frustrated Total Internal Reflection (FTIR)* - Ausnutzung der Totalreflexion [Han, 2010]

³Quelle: <http://www.wacom.com/cintiq/> [Aufruf: 03.01.2011]

⁴Quelle: http://www.nintendo.de/NOE/de_DE/nintendo_ds-systemuebersicht_11512.html [Aufruf: 03.01.2011]

⁵Quelle: <http://www.apple.com> [Aufruf: 03.01.2011]

⁶Quelle: <http://www.merl.com/projects/DiamondTouch/> [Aufruf: 03.01.2011]

⁷Quelle: <http://www.wacom.com/cintiq/> [Aufruf: 03.01.2011]

- *Diffused Illumination (DI) (Front- und Rear-DI)* - Ausnutzung von Kontrasten bei Berührung
- *Diffuse Surface Illumination (DSI)* - Zusätzliche Ausleuchtung der Oberfläche mit Infrarotlichtquellen an den Rändern
- *Laser Light Plane (LLP)* - Infrarot-Laser-Netz oberhalb der Glasschicht
- *LED Light Plane (LED-LP)* - erweitert LLP um Ausleuchtung durch einen IR-LED-Lichtrahmen

Beispiele: *MS Surface*⁸, *impressx*⁹ *xDesk*, *Perceptive Pixel Multi-Touch Workstation/ Wall*¹⁰

Akustische Touchscreens

Akustische Systeme arbeiten mit einem Glas, welches Schallwellen transportiert. Es sind zwei verschiedene Techniken bekannt: *Surface Acoustic Wave (SAW)* und *Acoustice Pulse Recognition (APR)*. SAW verwendet Signalgeber, welche Ultraschallwellen über dem Display erzeugen. Sensoren werten Veränderungen der Amplitude aus. In APR werten Sensoren das Schallsignal der Berührung selbst aus und errechnen daraus die Position der Eingabe.

Beispiele: *ELO Touchsystems APR-Touchscreen* und *intelliTouch-SAW-Touchscreen*¹¹

2.3.2 Visuelles Interface

Visuelle Systeme sind in der Lage Körperbewegungen zu erfassen. In Abbildung 2.5 werden Interpretationsmöglichkeiten in Relation zu den erkennbaren Bewegungsmustern aufgeführt.

Die Position des Agierenden gibt Aufschluss über dessen Präsenz und Position im Raum. Mit der Drehung des Oberkörpers kann er sich einem Bereich oder Objekt zuwenden. Er signalisiert damit seine Aufmerksamkeit. Unter Gesichtslinie wird die Ausrichtung des Kopfes verstanden. Kopfbewegungen werden vorallem zur Orientierung eingesetzt. Darüber hinaus sind aber auch Gesten möglich wie zum Beispiel das Kopfschütteln. Gesichtszügen sind mimetische Ausdrücke zu entnehmen und lassen auf einen bestimmten Gemütszustand schließen. Mit der Blickrichtung wird der Fokus der Betrachtung festgestellt. Gleichzeitig dienen Blickbewegungen der Orientierung und Erfassung der Umgebung.

Bewegungen von Arm, Hand und Fingern werden zusammengefasst betrachtet, da eine besondere Abhängigkeit besteht. An dieser Stelle wird deutlich gemacht, für welchen Einsatzzweck berührungslose Eingaben und Berührungseingaben sinnvoll sind. Während Bewegungen im Raum eher eine passive und unterstützende Wirkung haben, werden Berührungen als Auslöser oder Manipulation interpretiert. In einer Studie über Gesten in einem Hilfesystem

⁸Quelle: www.microsoft.com/surface/ [Aufruf: 03.01.2011]

⁹Quelle: <http://www.impressx.com> [Aufruf: 03.01.2011]

¹⁰Quelle: <http://www.perceptivepixel.com/solutions.php> [Aufruf: 03.01.2011]

¹¹Quelle: <http://www.elotouch.de/Produkte/Touchscreens/default.asp> [Aufruf: 03.01.2011]

KÖRPERMERKMAL	REGISTRIERUNG	INTERPRETATION
Körpervektor K	Position, Bewegung	Präsenz, Ortung
Frontvektor F	Ausrichtung, Neigung des Körpers	Aufmerksamkeit, Zuwendung
Gesichtslinie G	Kopfposition, Kopf-gestik	Orientierung
Gesichtszüge	Mimik	Emotionen
Blickvektor B	Blickrichtung, Blickbe-wegung, Gestik	Orientierung, Fokus, Aufmerksamkeit
Arm, Hand und Finger	Handposition, Hand-gestik	Zeigen, Imitieren, Handeln
	Berührung, Druck	Betätigen, Ertasten, Verändern, Zeichnen

Abbildung 2.5: Erkennung und Interpretation von Körperbewegungen

von WICKEY und ALEM wird dieser Sachverhalt deutlich: In einer Sammlung von Objekten sucht der Agierende zunächst alle Elemente ab, indem er mit der Hand über sie hinwegfährt. Wird ein passendes Element gefunden, zeigt er auf dieses mit aufgestreckten Zeigefinger. Erst mit einem "Tap", einer ausdrucksstarken Geste, die mit der Berührung der Oberfläche einhergeht, wird der Suchvorgang abgeschlossen und das Element endgültig ausgewählt.(vgl. [Wickey und Alem, 2007, S. 90])

Freiheitsgrade der Hand

Die menschliche Hand ist ein komplexes Gebilde aus einer Vielzahl von beweglichen Gelenken, Muskeln und Finger- und Handknochen. Für die Entwicklung von Trackingsystemen und Rekonstruktionen digitaler oder mechanischer Art, wird die menschliche Hand oft in einem Modell von Freiheitsgraden (engl.: Degrees of Freedom - DOF) betrachtet. Freiheitsgrade sind ausführbare Bewegungen um oder entlang einer Achse. Ein frei beweglicher Körper im Raum hat demzufolge 6 DOF (siehe Abbildung 2.6) - 3 Freiheitsgrade für die Position (Ortskoordinaten) und 3 Freiheitsgrade für die Orientierung (Drehwinkel) entlang der 3 Raumachsen. Die drei Raumebenen, die sich damit ergeben, werden Frontal-, Transversal- und Sagittalebene genannt. ¹² Als Greifraum wird der Bereich bezeichnet, der von einer Person mit den Händen erreicht werden kann, ohne dass dazu der Standort verlassen werden muss. Dies gilt für alle drei Raumebenen, womit sich eine sphärenartige Erreichbarkeitsgrenze um den Körper ergibt.

Die Hand besteht nach [Malerczyk, 2009] aus 15 für Handposen relevante Gelenke. Aus

¹²Begriffe nach den drei Körperebenen aus folgender Quelle: http://de.wikipedia.org/wiki/Lage-_und_Richtungsbezeichnungen [Aufruf: 10.01.2011]

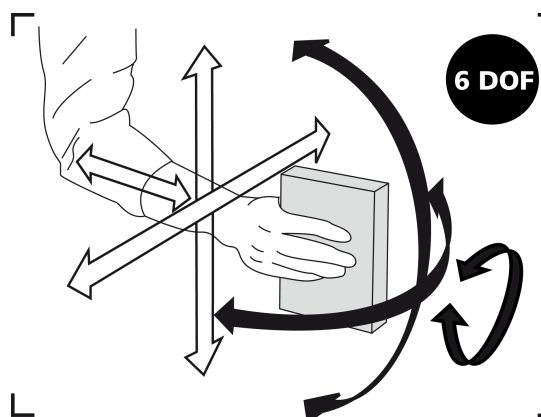


Abbildung 2.6: 6 DOF im Raum nach [Kantel, 2001]

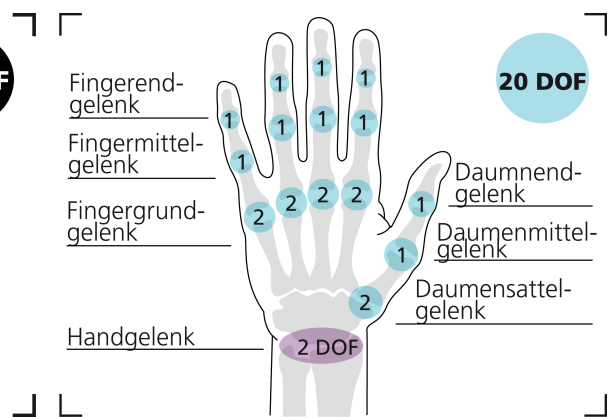


Abbildung 2.7: 20 Freiheitsgrade der Hand angelehnt an [Malerczyk, 2009, S. 29]

dieser Zahl ergeben sich 20 Freiheitsgrade der Hand (Handgelenk ausgenommen), welche in Abbildung 2.7 eingezeichnet sind. Handmodelle mit 20 oder 21 Freiheitsgraden sind gebräuchlich, um die volle Bewegungsfreiheit der Hand zu beschreiben. Diskutabel ist, ob das Daumenmittलगelenk mit einem oder zwei Freiheitsgraden bewertet wird. Dies macht den Unterschied von einem Freiheitsgrad aus. Je nach Relevanz und Verwendung von Handmodellen variiert die Anzahl der Freiheitsgrade. (vgl. [Malerczyk, 2009, S. 29]) Im Kontext dieser Arbeit stellt sich damit die Frage, wie die Bewegungen in den einzelnen Freiheitsgraden hinsichtlich verwertbarer Gesten zu bewerten sind und ob alle 20 DOF relevant für eine allgemeine Gestenbeschreibung sind.

2.3.3 Erkennung hand- und fingerbasierter Eingaben

Grundsätzlich sind zwei verschiedene Arten der Ausführung in Bezug auf die visuelle Ausgabeschnittstelle zu unterscheiden:

- Relative Bewegungs- und Positionserkennung der Hand
- Absolute Bewegungs- und Positionserkennung der Hand

Erste Variante erfasst Bewegungen relativ zu einer Ruheposition, die beim Nutzer liegt. Technische Hilfsinstrumente sind mit Sensoren ausgestattet, welche Veränderungen der Handbewegung auswerten. Nachteilig ist dabei, dass der Nutzer ein Eingabegerät mit sich führen muss. Im Folgenden werden Geräte zur relative Bewegungsmessung vorgestellt.

Datenhandschuh Durch faseroptische Kabel im Handrücken eines Handschuhs wird Licht geleitet. Die Lichtintensität wird an den Kabelenden gemessen und die Beugung und Bewegung der Finger aus diesen Daten errechnet. Der Vorteil eines Handschuhs, ist die Integration von Komponenten, die ein taktils Feedback und Kraftrückkopplung ermöglichen. Die Technik variiert: Die Bewegungserkennung kann, neben der Besagten Ausnutzung des Lichts, über mechanischen Weg oder unter Ausnutzung von Magnetismus erfolgen.

Beispiele: CyberGlove Systems CyberGlove II¹³, Essential Reality P5 Glove u.w.

Phantom Ein Phantom besteht im Wesentlichen aus einem drehbarem Arm mit drei Gelenken. Über einen Griff am Ende des Arms ist es dem Nutzer möglich Position und Drehwinkel der Hand über 6 Freiheitsgrade im Raum zu übertragen. Durch Kraftrückkopplung des Gerätes können darüber hinaus virtuelle Objekte ertastet werden.

Beispiele: SensAble Desktop Phantom¹⁴

Mobile Endgeräte Immer häufiger werden Bewegungssensoren in SmartPhones integriert, welche die Lage und damit die Orientierung des Gerätes erfassen und aus dieser Information beispielweise Rückschlüsse für die Orientierung der Bedienoberfläche ziehen. Unter Beachtung der Bedienung des Gerätes kann so in vielen Fällen auch auf die Handhaltung geschlossen werden. Unter Berücksichtigung der Beschleunigung können schnelle Bewegungen als Schütteln interpretiert werden.

Beispiele: Apple¹⁵ iPhone, Apple iPad u.w.

Controller In der Bedienung von Fernsehgeräten, Stereoanlagen und sonstigen Home-Entertainment-Geräten sind Fernbedienungen nicht wegzudenken. Controller heben das Konzept auf ein neues Niveau und ermöglichen eine kabellose Steuerung der Systeme mit Gesten.

Beispiele: HillcrestLabs Loop Pointer¹⁶, Kymera Magic Wand¹⁷, Nintendo Wiimote¹⁸

Im Gegensatz zu relativen Erkennungssystemen ist der Aufbau kamerabasierter Systeme fest, weshalb der Nutzer sich für die Eingabe nach der verwendeten Technik richten muss. Oftmals wird er sogar dazu aufgerufen, sich in eine exakte Position vor die Ausgabefläche zu stellen, so wie es beim iPoint Presenter der Fall ist. (siehe "Markerfreie Handerkennung") In optischen Systemen kann eine Unterscheidung zwischen einer markerfreien und markerbasierten Handerkennung vorgenommen werden.

Markerbasierte Handerkennung

Die Herausforderung optischer Systeme besteht vorallem darin, die menschliche Hand unter unbekannten Gegebenheiten des Umfeldes zu extrahieren. Fixpunkte können dem System dabei helfen, die Fehleranfälligkeit des Erkennungsprozesses zu verringern. Dazu werden Markierungen am Körper angebracht, die für den Erkenner eindeutig zuordenbar sind.

¹³Quelle: <http://www.cyberglovesystems.com/products/cyberglove-ii/overview> [Aufruf: 03.01.2011]

¹⁴Quelle: <http://www.sensable.com/haptic-phantom-desktop.htm> [Aufruf: 03.01.2011]

¹⁵Quelle: <http://www.apple.com> [Aufruf: 03.01.2011]

¹⁶Quelle: <http://hillcrestlabs.com/loop/index.php> [Aufruf: 03.01.2011]

¹⁷Quelle: <http://www.thewandcompany.com/> [Aufruf: 03.01.2011]

¹⁸Quelle: http://www.nintendo.de/NOE/de_DE/systems/zubehoer_1243.html [Aufruf: 03.01.2011]



Abbildung 2.8: Farbliche Markierungen an der Hand [Chua et al., 2002]



Abbildung 2.9: Handschuh mit farbigem Muster [Wang und Popovic, 2009]

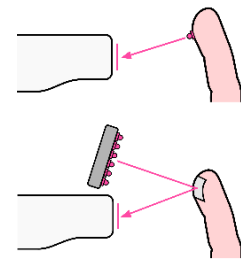


Abbildung 2.10: LED- und Reflektoren [Peitz und Podewski, 2009]

Kontrastreiche Muster, wie sie auch an Crash Dummies verwendet werden, stellen eine Möglichkeit dar. Sie können an markanten Punkten der Hand(ober)fläche aufgetragen sein. Eine weitere Variante ist eine kontrastreiche Farbgebung zu nutzen. Abbildung 2.8 zeigt eingefärbte Markierungen, die an Fingerspitzen, Handballen und Handgelenk geklebt wurden. In [Wang und Popovic, 2009] werden farbige Handschuhe mit auffälligem Muster (siehe Abbildung 2.9) für die dreidimensionale Rekonstruktion der Hand verwendet.

Des Weiteren existieren Tracking-Systeme, welche Licht im Infrarotbereich messen können. Die Wiimote, die bereits unter den Controllern für relative Bewegungsmessungen genannt wurde, besitzt über den internen Beschleunigungssensor hinaus eine integrierte Infrarotkamera. Infrarot-LEDs eignen sich als Lichtquelle und können als Tracking-Punkte verwendet werden. Die Spielekonsole Nintendo Wii sieht es vor, dass die Wiimote in der Hand gehalten wird und sich die Infrarotquelle in fester Anordnung an dem Ausgabegerät befindet. In [Lee, 2010a] werden Nutzungsvarianten aufgezeigt, indem der Controller fixiert und als feste Kamera benutzt wird. Daran angelehnt werden in [Peitz und Podewski, 2009] Methoden der Handerkennung untersucht. Abbildung 2.10 zeigt die technischen Alternativen auf: Entweder werden Infrarot-LEDs als Marker an den Fingern angebracht (obere Darstellung) oder reflektierende Folie sorgt dafür, dass Infrarotstrahlen zur Controller-Kamera geleitet werden (untere Darstellung). (vgl. [Lee, 2010a])

Markerfreie Handerkennung

Hilfsmittel wie Controller, die in der Hand gehalten werden sowie Markierungen oder Datenhandschuhe, die zunächst angelegt werden müssen, sind hinsichtlich der Natürlichkeit der Interaktion störende Faktoren. Die Realisierung einer markerfreien Gestenerkennung in Echtzeit ist allerdings mit erhöhtem Rechenaufwand und mehreren Bildverarbeitungsschritten verbunden. In einem Vorverarbeitungsschritt müssen relevante Regionen unter weitestgehend unbekannten Bedingungen der Umgebung, wie beispielsweise Lichtverhältnisse oder Bewegungen im Hintergrund, voneinander getrennt werden. Weitere Herausforderungen ergeben sich nach [Malerczyk, 2009, S. 20] mit der Dimension des Suchraums, der Selbstverdeckung einzelner Handsegmente, der Verarbeitung großer Datenmengen in Echtzeit und

der Verfolgung von Bewegungen mit höheren Geschwindigkeiten. Optische Systeme lassen sich hinsichtlich der Anzahl verwendeter Kameras und den hauptsächlich verwendeten Methoden und Algorithmen für die Segmentierung und Merkmalsextraktion, sowie Verfahren zur Distanzermittlung klassifizieren.



Abbildung 2.11: iPoint Presenter

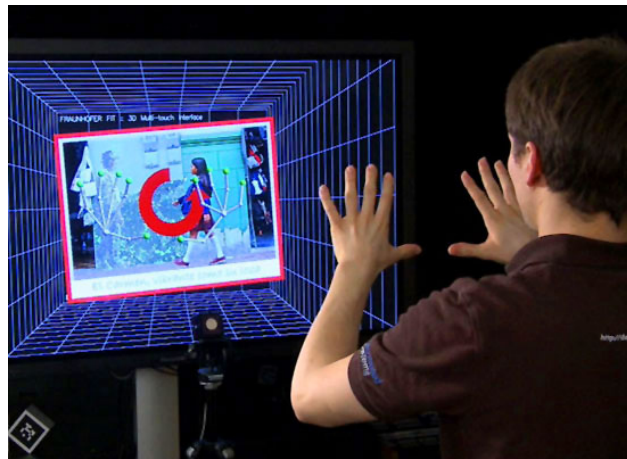


Abbildung 2.12: Forschungsprojekt des FIT

Die Abbildungen zeigen Beispielsysteme des Fraunhofer Instituts. Der iPoint Presenter¹⁹ (Abbildung 2.11) arbeitet mit zwei Infrarotkameras, die in einer Box untergebracht sind. Der Nutzer muss in dem Fall unter oder über der Box agieren, damit Gesten erkannt werden. In einem aktuelleren Forschungsprojekt²⁰ des Instituts (Abbildung 2.12) wird eine 3D-Kamera eingesetzt, mit der die Position und Bewegungen der Hand mithilfe des "Time-of-Flight"-Verfahrens in 3D-Koordinaten übersetzt wird. In dieser Methode wird die Distanz darüber ermittelt, wie lange das Licht zum gefilmten Objekt hin und wieder zurück benötigt.

2.4 GESTEN

Gesten sind im Allgemeinen Körperbewegungen, die als symbolische und nonverbale Kommunikationsform aufzufassen sind, und damit einen expressiven Charakter haben.

Unterscheidbar sind bewusst ausgeführte Gesten, die aktiv der Kommunikation dienen, und koverbalen Gesten, das heißt sprachbegleitenden Gesten. Weitere Unterscheidungen werden anhand der Eigenschaften und Verwendung von Gesten im folgenden Abschnitt behandelt.

2.4.1 Eigenschaften und Bedeutung von Gesten

Nach [Saffer, 2009] haben Gesten im Allgemeinen vergleichbare Eigenschaften, die je nach Sensortechnik in dem Gestenerkennungsprozess einbezogen werden. Sie umfassen die At-

¹⁹Quelle: <http://www.hhi.fraunhofer.de/de/departments/interactive-media-human-factors/overview/hand-interaction/exhibit/ipoint-presenter/> [Aufruf: 15.01.2011]

²⁰Quelle: <http://www.fit.fraunhofer.de/presse/10-07-13.html> [Aufruf: 15.01.2011]

tribute *Präsenz*, *Dauer*, *Position*, *Bewegung*, *Druck*, *Größe*, *Orientierung*, *Objekte*, *Anzahl erkannter Punkte*, *Sequenz* und *Anzahl der Nutzer*. Die *Präsenz* ist eine Grundvoraussetzung um Gesten zu erkennen. Mit der *Dauer* wird der zeitliche Rahmen vom Anfang bis zum Ende der Ausführung beschrieben. Die *Position* ist der Ausführungsort in einem Bezugssystem. In Touch-Interfaces ist das Bezugssystem die Eingabefläche, in dem Kontakte auf zweidimensionale Koordinaten abgebildet werden. Um den *Druck* von Eingaben zu erfassen sind entsprechende Sensoren notwendig, die die Stärke von Berührungen messen können. Unter *Bewegung* ist zum einen die Geschwindigkeit, zum anderen die Beschleunigung der Ausführung zu verstehen. Darüber hinaus lassen sich weitere Merkmale einer Bewegung finden, wie beispielsweise die Richtung der Bewegung. Die *Größe* von Kontakten wird in Touch-Interfaces als Indiz dafür genutzt, ob die Eingabe mit einem Stylus, dem Finger oder der Hand erfolgt. In Hinblick auf Point-Gesten und die Formulierung einer allgemeineren Eigenschaft für die Abmessungen der Eingabe, stellt die Form der Geste eine Erweiterung der Eigenschaften dar. Mit der *Orientierung* ist die Ausrichtung des Nutzers und der Hand während einer Eingabe gemeint. *Objekte*, wie zum Beispiel Tangibles, finden vorallem in Touch-Interfaces Anwendung. Unter der *Anzahl erkannter Punkte* sind in Touch-Interfaces die zeitgleich erkannten Kontakte und in Point-Interfaces die simultan erkannten Point-Gesten (auf das Interface gerichtete Zeigegesten) zu verstehen. Neben der parallelen Ausführung können Gesten auch als *Sequenz* nacheinander auftreten. Mit der Fähigkeit mehrere Gesten gleichzeitig zu erkennen, bietet es sich an, mehrere *Nutzer* in das Bedienkonzept einzubeziehen.

Weitere Merkmale von Gesten lassen sich unter Betrachtung des Gestenerkennungsprozesses ausmachen. Gesten können abhängig vom anwendungsspezifischen Kontext unterschiedlich interpretiert werden. Zur Ermittlung des Kontextes ist der Ausführungsort von Bedeutung. Je nachdem ob die Handlung über einem Objekt geschieht oder auf globaler Ebene, hat dies verschiedenen Auswirkungen. In [Kammer et al., 2010] wird zwischen applikationsweiter Handlung, objektbezogener Handlung und Ausführung auf einer Sammlung von Objekten unterschieden. Weiterhin wird zwischen Online- und Offline-Gesten unterschieden. Während Online-Gesten zur Eingabezeit erkannt werden, wird eine Offline-Geste erst nach vollendeter Ausführung interpretiert. Nach [Nardi, 2008] kann außerdem eine Unterscheidung zwischen einmaligen und kontinuierlichen, sowie zwischen statischen und dynamischen Gesten getroffen werden. Kontinuierliche Gesten vermitteln zu jedem Zeitpunkt der Ausführung den Status wie zum Beispiel die Position oder in einer Zoom-Geste den Abstand zwischen den Fingern. Einmalige Gesten hingegen werden entweder erkannt oder nicht. Statischen Gesten werden von [Malerczyk, 2009, S. 27] auch als Handposen bezeichnet. Entscheidend ist dabei, dass sich die Form der Geste nicht ändert. Eine dynamische Geste wird in [Malerczyk, 2009, S. 28] als eine "*Bewegung einer (starrten) Handpose im Raum*" beschrieben, kann aber auch ungeachtet der Position und Orientierung der Hand eine Bewegung der Finger- und Handgelenke sein.

Betrachtet wurden bisher Merkmale von Gesten, die in Bezug auf die Sensortechnik und

den Erkennungsprozess erklärt werden. Darüber hinaus können Gesten hinsichtlich ihres Bedeutungskontextes und ihrer Verwendung eingeordnet werden. Klassifikationsansätze nach [Pavlovic et al., 1997], [Wobbrock et al., 2009], [Karam und Schraefel, 2005] und [George und Blake, 2010] werden bereits in [Meißner, 2010, S. 16 u. S. 27] und [Kammer et al., 2010, S. 4] zusammengefasst und gegenübergestellt. Im Sinne einer begrifflichen Klärung werden auf dieser Grundlage folgende Arten von Gesten unterschieden (siehe Tabelle 2.1).

Gesten	Bedeutung	Terminologie in den Quellen
Deiktische Gesten	Zeigehandlungen, die die Auswahl oder Identifikation von Objekten möglich machen	deictic [Kammer et al., 2010] [Pavlovic et al., 1997] [Karam und Schraefel, 2005], bindings - objektgebunden [Wobbrock et al., 2009]
Nachahmende Gesten	Imitieren von Bewegungsabläufen oder Formen von Objekten	mimetic [Pavlovic et al., 1997], metaphorical [Wobbrock et al., 2009], lexikal [Kammer et al., 2010]
Symbolische Gesten	Beschreiben Zeichen und Symbole, die sich meist auf einen sprachlichen Gegenstand beziehen	referential [Pavlovic et al., 1997], symbolic [Wobbrock et al., 2009], semaphores [Karam und Schraefel, 2005], ikonik [Kammer et al., 2010]
Sprachbegleitende Gesten	Gesten, die sprachliche Äußerungen unterstützen und untermalen	gesticulation [Karam und Schraefel, 2005], modelizing [Pavlovic et al., 1997]
Manipulierende Gesten	Bewirken Veränderungen von Objekten	physical Wobbrock et al. [2009], manipulative [Pavlovic et al., 1997] [Kammer et al., 2010] [Karam und Schraefel, 2005], manipulations [George und Blake, 2010]
Navigierende Gesten	Änderung globaler Ansichten, Perspektiven und Zuständen einer Anwendung	navigational [Kammer et al., 2010]
Zeichensprache	Gebärdensprache und andere erlernte Gesten, die einem Regelwerk folgen	sign language [Karam und Schraefel, 2005], arbitrary nach J. L. Nespoulous (vgl. [Lee, 2010b])
Bedeutungslose Gesten	Gesten ohne wesentliche Information	unintentional movements [Pavlovic et al., 1997], abstract [Wobbrock et al., 2009]

Tabelle 2.1: Zusammenfassung der Arten von Gesten in Anlehnung an [Meißner, 2010] und [Kammer et al., 2010]

3 VERWANDTE ARBEITEN

Dieses Kapitel behandelt verwandte Themen zu multimodalen und gestenbasierten Bedienkonzepten. In Abschnitt 3.1 werden zunächst bekannte multimodale Anwendungen vorgestellt, die eine Gestensteuerung unterstützen. Im darauffolgenden Abschnitt wird die Verwendung etablierter Gesten in Touch- und Point-Interfaces behandelt. In Abschnitt 3.3 wird das Thema der Gestenbeschreibung aufgegriffen und das Konzept der GeForMT erläutert. Abschließend werden Systeme vorgestellt, die Touch- und Point-Interaktion miteinander kombinieren (Abschnitt 3.4).

3.1 MULTIMODALE ANWENDUNGEN

Die Forschung an multimodalen Benutzerschnittstellen, die die Steuerung über Gestik einschließen, begann mit Richard A. Bolt am MIT im Jahre 1982. [Bolt, 1980] In dem sogenannten PUT-THAT-THERE-System sitzt der Nutzer einer großen Projektion einer Datenlandschaft gegenüber. Mit Sprachbefehlen und simultan ausgeführten Zeigegesten können Objekte in der Darstellung verschoben oder erstellt werden. Abbildung 3.2 verdeutlicht den Aufbau. Die Steuerung folgt dem Namen des Systems. Mit folgender Eingabesequenz wird ein Objekt verschoben: "Put" - Zeigegeste - "That" - Zeigegeste - "There" (Ausdrücke in Anführungszeichen sind sprachliche Äußerungen des Nutzers). "Create a blue square"-Zeigegeste-"there" erstellt beispielsweise an der Stelle, auf die der Nutzer deutet, ein Objekt mit entsprechender Farb- und Formeigenschaft.

Mit dem Touristeninformationssystem GEORAL¹ wurde 1995 erstmals ein Touchscreen in Kombination mit Spracheingabe verwendet. Das System stellt ähnlich wie "Put-That-There" eine verbal geäußerte Frage des Nutzers mit semantischen Eigenschaften des berührten Ortes auf dem Interface in Beziehung, so dass die Frage in Bezug auf das Objekt interpretiert und beantwortet werden kann. Das Projekt SMARTKOM² der DFKI Saarbrücken lehnt an dieses Konzept an und verbindet Gestik, Mimik und Sprache zur Bearbeitung der Interaktionsaufgabe.

Im Trend der fortschreitenden Entwicklung natürlicher Benutzerschnittstellen stellte eine For-

¹vgl. www.dfki.de/~kipp/seminar/writeups/Haichao.pdf [Aufruf: 04.01.2011]

²Quelle: <http://www.smartkom.org> [Aufruf: 04.01.2011]



Abbildung 3.1: Multimodale gestenbasierte Anwendungen

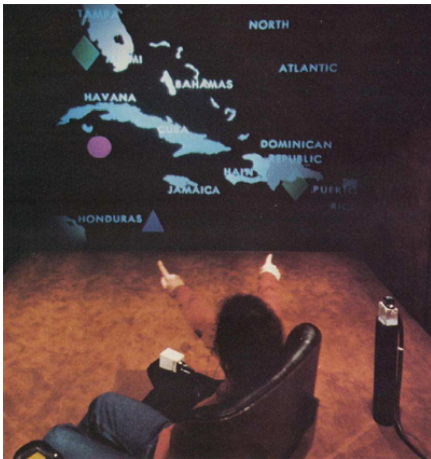


Abbildung 3.2: "Put-that-there"
[Bolt, 1980, S. 6]



Abbildung 3.3: mUltimo3D
[Wandke]

schungsgruppe des Heinrich-Hertz-Instituts das mUltimo3D als Computerarbeitsplatz der Zukunft vor, der in Abbildung 3.3 zu sehen ist. Neben der dreidimensionalen Darstellung durch ein stereoskopisches Display, erkennt das System Handgesten, Kopfposition, Blickbewegungen und Sprache. Die Interaktion mit dem System soll so auf natürliche Weise und ohne weitere Geräte möglich sein. Je Kopfausrichtung und Fokus auf das Display, wird die Visualisierung nachjustiert und es entsteht ein räumlicher Eindruck. Darüber hinaus werden aber auch die gewohnten Eingabegeräte wie Maus und Tastatur unterstützt.

Das von der EU geförderte Projekt AUGMENTED MULTI-PARTY INTERACTION (AMI)³ untersucht multimodale Technologien zur Unterstützung der Kommunikation im Kontext intelligenter Besprechungszimmer und Assistenten für Konferenzschaltungen. Neben dem Ziel die Zusammenarbeit zwischen den Teilnehmern von Meetings zu fördern und die Interaktion effektiver zu gestalten, sollen Mitschnitte des Meetings über die Ergebnisse Protokoll führen. Neben der Spracherkennung wird mitunter der Sprecher erkannt, wodurch Dialoge zugeordnet werden können und nachvollziehbar bleiben. Weiterhin wird Gestik zu Kommunikations- und Interaktionszwecken interpretiert. Genauso können auch handschriftliche Aufzeichnungen oder Skizzen zur Kommunikation beitragen. Alle gesammelten Informationen können zu hilfreichen Ergebnisdokumenten zusammengefasst werden.

Mit SQUIDY⁴ wird von der Universität Konstanz ein Konzept vorgestellt, wie sich Ein- und Ausgabekomponenten multimodaler Systeme grafisch in einem Editor verknüpfen lassen. Die Idee hinter dem Entwicklungswerkzeug ist, angelehnt an Pipeline- und Filtermetaphern, ein Datenflussdiagramm, welches die Komplexität des Aufbaus und der Schnittstellenkopplung versteckt. Es wird eine Vielzahl an Eingabeoptionen unterstützt: Gestik, Blickbewegung,

³Quelle: <http://www.amiproject.org/> [Aufruf: 15.12.2010]

⁴Quelle: <http://www.squidy-lib.de/> [Aufruf: 15.12.2010]

Spracheingabe, Multitouch und Tangibles, Stifteingabe, Laser Pointer und weitere. Die Eingabeströme können mit dem Editor auf einfache visuelle Ausgabegeräte gelenkt werden. Untertützt wird aber auch ein Mausemulator als Ausgabeknoten, mit welchem es ermöglicht wird vertraute Desktop-Anwendungen anzusteuern. (vgl. [König et al., 2009])

In Abbildung 3.1 sind die genannten multimodalen Anwendungen zusammengefasst.

3.2 VERWENDUNG VON GESTEN

Benutzerschnittstellen, die Gesten als Eingabeoption unterstützen, haben das Problem, dass sie dem Nutzer kommuniziert werden müssen. Ohne eindeutigen Hinweis weiß ein unerfahrener Nutzer in den wenigsten Fällen welche Bewegung auszuführen ist und wie sie interpretiert wird. Durch Werbemaßnahmen und der gezielten Vermarktung von Geräten, die Gesten unterstützen, ist es Großkonzernen wie Apple, Microsoft, Nintendo und weiteren Unternehmen gelungen einige einfach zu erlernende Gesten als intuitive Eingabeoption durchzusetzen. In den folgenden Abschnitten werden praxistaugliche Gesten zur Steuerung und Manipulation digitaler Inhalte bestimmt.

3.2.1 Standardgesten für Touch-Interfaces

In der Abbildung werden Gesten gezeigt, die sich zur Steuerung auf berührungsempfindlichen Geräten erfolgreich durchgesetzt haben und in den meisten Touch-Systemen wie z.B. Apples iPad⁵ oder dem Microsoft Surface⁶ Anwendung finden. Während TAP und HOLD gebraucht werden um Objekte zu selektieren oder zu aktivieren, werden DRAG/SLIDE, SPREAD/PINCH und ROTATE zur Manipulation, d.h. zur Änderung von Position, Größe und Orientierung, von Objekten verwendet. FLICK kommt eine gesonderte Rolle zu, da die Geste vorallem in einer Sammlung von Objekten oder auf gesamter Darstellungsfläche zum Einsatz kommt, um in dieser zu navigieren. Weitere Details zur Ausführung und Verwendung von Touch-Gesten werden in [Meißner, 2010, S. 43] erläutert.

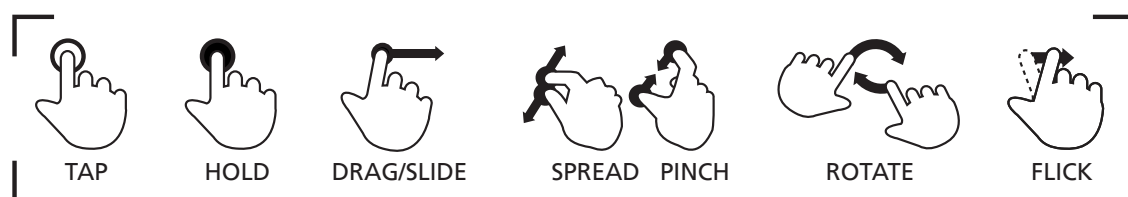


Abbildung 3.4: Standardgesten im Touch-Interface

3.2.2 Verwendung berührungsloser Gesten

In Anlehnung an die verbreiteten Standardgesten in Touch-Interfaces, werden in Systemen, die eine berührungslose Steuerung vorsehen, ähnliche Gesten verwendet, die dem Interaktionskonzept entsprechend angepasst werden. Als Beispiel können Anwendungen unter dem

⁵<http://www.apple.com/de/ipad/> [Aufruf: 20.12.2010]

⁶www.microsoft.com/surface/ [Aufruf: 20.12.2010]

Einsatz des iPoint Presenter⁷ des Fraunhofer Instituts genannt werden. Aus einer Zeigegeste (POINT) wird abgeleitet, auf welchen Ort auf der Ausgabefläche der Nutzer deutet. An der entsprechenden Bildschirmposition wird ein Cursor dargestellt. Es erfolgt also eine Projektion der Position und Orientierung der Fingerspitze auf die zweidimensionale Darstellungsebene. Auf dieser Basis können erkannte POINT-Gesten analog zu den Kontakten auf einem Touchscreen behandelt werden. SPREAD/PINCH und ROTATE werden hier ebenso für die Manipulation von Objekten eingesetzt. Unter längerem Halten einer Zeigegeste können die Gesten HOLD und DRAG erkannt werden. (vgl. [Langbein, 2009])



Abbildung 3.5: Standardgesten im Point-Interface

In einem aktuellen Forschungsprojekt⁸ des Fraunhofer Instituts werden die in Abbildung 3.5 veranschaulichten Gesten GRAB und RELEASE genutzt, um die Manipulation an einem Objekt einzuleiten. Hierzu ist es notwendig, die gesamte Hand und mehrere Finger in den Erkennungsprozess einzubeziehen. Die Hände werden zum Ergreifen zur Faust zusammengeballt. In diesem Zustand können ausgewählte Objekte wie gehabt verschoben, vergrößert oder gedreht werden. Die flache Hand signalisiert, dass das Objekt wieder losgelassen wird. Diese Art, digitale Objekte zu steuern, kommt der Interaktion mit realen Gegenständen sehr nahe.

Eine andere Variante, ein DRAG im Sinne der berührungslosen Interaktion auszuführen, ist der sogenannte Pinzettengriff (vgl. [Benko und Wilson, 2010]). Oftmals reichen Daumen und Zeigefinger aus, um einen Gegenstand zu ergreifen. In Zusammenhang mit POINT scheint die Geste zur Auswahl eines Objektes präziser zu sein, da mit Detektion der Fingerspitze eine eindeutigere Position angesprochen wird.

In [Epps et al., 2006] zeigt eine Studie, in der berührungslose Gesten in horizontal angelegten Interfaces analysiert werden, dass für die meisten Interaktionsaufgaben eine Zeigegeste angewandt wird. Insgesamt 20 Probanden wurden Screenshots einer Windows-Anwendung vorgelegt und unter Verwendung von Raum-Gesten dazu aufgefordert verschiedenste Aufgaben, wie zum Beispiel das Selektieren oder Verschieben von Icons, zu simulieren. Neben GRAB und RELEASE zählen zu weiteren häufig verwendeten Gesten die flache Hand, parallel und orthogonal zur Interface-Ebene. Letztere Handpose wird mit der Handkante zum

⁷Quelle: <http://www.hhi.fraunhofer.de/de/departments/interactive-media-human-factors/overview/hand-interaction/exhibit/ipoint-presenter/> [Aufruf: 04.01.2011]

⁸Quelle: <http://www.fit.fraunhofer.de/presse/10-07-13.html> [Aufruf: 04.01.2011]

Interface ausgeführt und zumeist zur Separierung oder Selektion von Text verwendet. Die Interpretation der parall ausgerichteten Hand, kann je nachdem ob die Finger angelegt oder gespreizt sind, unterschiedlich ausfallen. (vgl. [Epps et al., 2006])

In einer Aufzählung verwendbarer Handgesten in [Saffer, 2009, S. 210] wird die Bewegung mit der flachen Hand zum Interface als WAVE bezeichnet. In Anlehnung an die von SAFFER bezeichnete SLAP-Geste kann eine Bewegung mit der Handkante zum Interface bedeuten, dass ein oder mehrere Objekte aus dem sichtbaren Bereich geschoben werden. Sie erinnert jedoch auch an das Blättern eines Buches, so dass ebenso eine Navigation zwischen Sichten der Anwendung oder in einer Sammlung von Objekten denkbar wäre. In [Saffer, 2009, S. 210 ff.] wird eine Vielzahl weiterer Gesten vorgestellt, auf die jedoch in Hinblick auf die Relevanz von Gesten nach [Epps et al., 2006] nicht weiter eingegangen wird.

3.3 GESTENBESCHREIBUNG

Im Zuge der Entwicklung von Gestenerkennungssystemen entstehen nach und nach allgemeine Herangehensweisen zur Verarbeitung von Gesten. Die Verwendung von Frameworks erlaubt es Entwicklern komplizierte Erkennungsalgorithmen und implementierte Gesten für ihren Anwendungsfall zu nutzen und um neue Gesten zu erweitern. Obwohl dem Programmierer damit viel Aufwand erspart wird, muss er, um neue Gesten zu implementieren, die Auswertung von Eingaben selbst im Programmcode vornehmen oder aber dem System Muster antrainieren. Die Entkopplung einer einfachen textuellen oder visuellen Beschreibung vom Erkennungsprozess stellt im Hinblick dessen eine geeignete Lösung dar, die Entwicklung gestenbasierter Anwendungen zu vereinfachen.

In [Henzen, 2010, S. 34] werden Gestenbeschreibungssprachen vorgestellt, die bezüglich der Struktur der Gestendefinition als linear, baumstrukturiert oder netzartig typisiert werden. In linearen Beschreibungssprachen wird lediglich eine Abfolge von Bewegungsmustern dargestellt. Netzartige Beschreibungen verwenden hauptsächlich eine graphische Notationen in Form von Zustandsdiagrammen wie sie beispielsweise in [Lao et al., 2009] eingesetzt wird. Baumstrukturierte Sprachen erlauben die Zerlegung der Gesten in atomare Bestandteile und deren hierarchischen Aufbau zur Beschreibung komplexer Bewegungsabläufe. In dieser Sprachgruppe stellt sich insbesondere XML als bevorzugtes Format für eine allgemeine Gestenbeschreibung heraus. Eine grundlegende Unterscheidung von Gestenbeschreibungssprachen kann bezüglich ihrer Beschreibungsfähigkeit getroffen werden. Während Ansätze von [Lao et al., 2009], [Meyer, 2009] (GDML) und [Kammer et al., 2010] (GeForMT) die Formalisierung von Touch-Gesten fokussieren, wird in [Kopp et al., 2006] mit der Zeichnungssprache BML (Behaviour Markup Language) ein Konzept vorgestellt, mit dem jegliche Arten der Körperbewegung (Körper, Oberkörper, Beine, Handgesten, Gesicht, Augen- und Lippenbewegungen) in ihrem Zusammenwirken beschrieben werden. Im Rahmen dieser Arbeit stellt sie einen interessanten Ansatz für die Komposition verschiedener Aktionsmodalitäten dar. In BML steht jedoch die zeitliche Synchronisation sprachbegleitender Handlungen für die

Rekonstruktion und Animation virtueller menschlicher Gestalten im Vordergrund. Allgemein fehlt es noch an Ansätzen, in denen räumliche Bewegungen und berührungsempfindliche Gesten unterschieden werden und über zeitliche und räumliche Parameter miteinander verknüpft werden.

3.3.1 GeForMT

Während sich XML als bevorzugtes Format für eine allgemeine Gestenbeschreibung herausstellt, wird in [Kammer et al., 2010] vorallem der unnötige Overhead ("Syntactic Sugar") an dem deskriptiven Ansatz kritisiert. In dieser Hinsicht stellt die **Gesture Formalization for Multitouch (GeForMT)** mit der Einführung einer formalen Grammatik eine einfach zu überblickende und komprimierte Lösung dar, die folgende Ziele verfolgt:

- **Maschinenlesbarkeit** durch Nutzung von Standards, den Einsatz von Parsern und Verwendung in Frameworks
- **Menschenlesbarkeit** durch Text oder grafische Darstellung, verständliche Syntax und überblickbare Strukturen
- **Erlernbarkeit** durch erfassbare Komplexität, einfache Sprache und ableitbare Muster

Zur Formalisierung und Beschreibung von Multitouchgesten schlagen KAMMER ET AL. die Semiotik als Technik der Linguistik vor, welche alle Erscheinungsformen hinsichtlich der Darstellung und Interpretationen von Zeichen und Symbolen beschreiben kann. Die Syntax beschreibt diese Symbole und ihre Kombination. Auf Systemebene ist sie als Grammatik in EBNF definiert. Mit ihr kann die Ausführung einer Geste in der Fläche mit all ihren für ein Multitouch-System relevanten Eigenschaften beschrieben werden. Auf höherer Ebene müssen die spezifizierten Gesten semantisch in Beziehung gesetzt und im Kontext der Interaktion interpretiert werden. Die Grammatik ist nach [Kammer et al., 2010] als EBNF wie folgt beschrieben:

1:	complex	::=	gesture (operator gesture)* relation['gesture (operator gesture)*']
2:	gesture	::=	function('atomfocus(',')atomfocus)*') atomfocus(',')atomfocus)*
3:	relation	::=	'CROSS' 'JOIN' 'SYNC' 'SPREAD'
4:	atomfocus	::=	atom atom('focuslist')
5:	focuslist	::=	focus(',')focus)*
6:	focus	::=	'a' 's' 'o'[integer]
7:	function	::=	[integer]'F' [integer]'H' [integer]'B'
8:	operator	::=	'*' '+' ','
9:	atom	::=	identifier vector['_direction'] shape['_direction']['_rotation']
10:	identifier	::=	'MOVE' 'POINT' 'DEPOINT' 'HOLD'
11:	vector	::=	'LINE'
12:	shape	::=	'CIRCLE' 'SEMICIRCLE'
13:	direction	::=	'NORTH' 'N' 'NORTHEAST' 'NE' 'EAST' 'E' 'SOUTHEAST' 'SE' 'SOUTH' 'S' 'SOUTHWEST' 'SW' 'WEST' 'W' 'NORTHWEST' 'NW'
14:	rotation	::=	'CLOCKWISE' 'CW' 'COUNTERCLOCKWISE' 'CCW'
15:	integer	::=	[1-9] [0-9]*

Um eine Gestendefinition aus softwaretechnischer Sicht identifizieren zu können, wird die Syntax nach [Henzen, 2010] um Hilfsregeln erweitert, die den Gesten eine eindeutige Bezeichnung in Form eines Namens geben.

0.1:	definition	::=	name '=' complex (' complex)* ';' ;
0.2:	name	::=	([a-z][A-Z]) ([a-z][A-Z])*

Posen Beschreiben die Form des Kontaktes auf der Multitouch-Oberfläche, die von dem System erkannt wird. Beispielsweise steht 2F(...) für die Berührung der Oberfläche mit zwei Fingern, 1H(...) für das Erkennen einer Hand und 1B(...) für einen generischen Blob, um weitere Handformen oder Objekte zu beschreiben.

Atomare Gesten Atomare Gesten sind Gesten, die in ihrer Form und Ausführung nicht weiter aufgeteilt werden können. Zu ihnen zählen MOVE für eine Freiform, POINT zur Beschreibung eines TAP, DEPOINT für den Kontaktverlust, HOLD zur Beschreibung eines langanhaltenden Kontaktes auf einer Stelle, LINE für eine Linie, CIRCLE beschreibt einen vollen Kreis und SEMICIRCLE einen Halbkreis. Der Formbeschreibung der Geste wird eine Richtungsangabe angehängen. Für die einfache Geste LINE kann jede Himmelsrichtung der Windrose herangezogen werden, um den Vektor zu beschreiben. Zu den Kreisgesten SEMICIRCLE und CIRCLE müssen darüber hinaus Angaben zur Rotationsrichtung (CLOCKWISE oder COUNTERCLOCKWISE). Einige Beispiele atomarer Gesten mit ihren Richtungs-

eigenschaften sind zur Verdeutlichung in Abbildung 3.6 dargestellt.

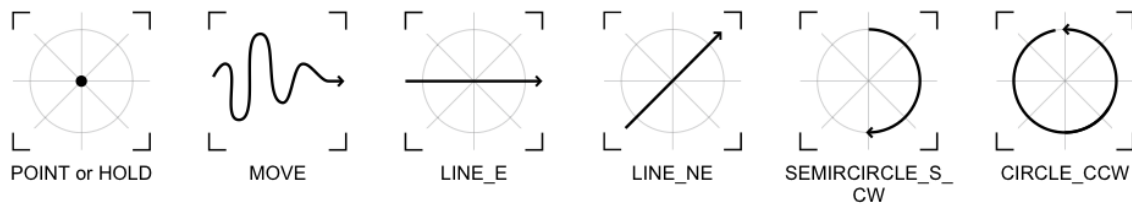


Abbildung 3.6: Beispiele atomare Gesten in GeForMT [Kammer et al., 2010, S. 3]

Kompositionsooperatoren Die Operatoren Komma, Stern- und Plus-Zeichen beschreiben die zeitliche Abfolge komponierter Gesten. Ein Komma drückt eine sequentielle Abfolge der Gesten aus. Stern- und Plus-Operatoren beschreiben eine parallele Ausführung von Gesten, wobei ein Stern-Operator zusätzlich den Anspruch an eine synchrone Ausführung erhebt. Nach [Henzen, 2010] sind die Operatoren um eine ODER-Verknüpfung zu erweitern, damit auch leichte Abweichungen in den Bewegungsmustern der Gesten möglich sind. Die Option wird mit dem Zeichen “|” ausgedrückt.

Fokus Jede Geste wird in einem bestimmten Kontext ausgeführt. Der Fokus beschreibt einen Teil dieses Kontextes, nämlich das grafische Element auf welchem es ausgeführt wird. Unterschieden wird zwischen Objekt, Objektmenge oder Applikation. Beispielsweise wird mit LINE(a) eine Linie beschrieben, die global ausgeführt wird.

Relationen Die Sprachkonstrukte CROSS, JOIN, SYNC und SPREAD beschreiben die örtliche Beziehung komponierter Gesten zueinander. Eckige Klammern umschließen den Verbund der Gesten. So kann zum Beispiel ausgedrückt werden, ob sich die Gesten überlappen, parallel ablaufen und sich aufeinander zu oder voneinander weg bewegen. Nach [Henzen, 2010] ist ein weiteres Sprachelement notwendig, mit dem bei einer sequentiellen Komposition zweier Gesten ausgedrückt werden kann, an welcher Stelle die zweite Geste anknüpft. CONNECT definiert, dass die zweite Geste dort beginnt, wo die vorangegangene Geste endet. Ein anderer Lösungsansatz ist die Verschachtelung von Gestenatomen. (siehe [Henzen, 2010, S. 79])

Nach [Henzen, 2010] ist es von Vorteil bereits in der Beschreibung einer Geste Angaben zum Auswertungszeitpunkt zu machen, das heißt eine Unterscheidung zwischen Online- und Offline-Geste zu treffen. Damit sollen “Disharmonien” vermieden werden und dem Gestendesigner mehr Freiheiten eingeräumt werden, auf den Erkennungsprozess Einfluss zu nehmen.

```
complex ::= gesture (operator gesture)* |
          relation['gesture (operator gesture)*'] ':' ['ON' | 'OFF']
```

Des Weiteren wird eine Vereinfachung der Komposition von Gesten vorgeschlagen, die es erlaubt den Bezeichner von Gestendefinitionen in der Beschreibung komplexer Gesten wie-

derzuverwenden. Dies vereinfacht vorallem die Beschreibung wiederholender Gesten wie das folgende Beispiel aus [Henzen, 2010] für das mehrmalige eines nördlich gerichteten Pfeils zeigt.

gesture	=	LINE_NE,LINE_NE,LINE_SE
complexgesture	=	gesture,gesture

Die Verknüpfung der formalen Definition einer Geste und der konkreten Implementierung erfordert ein weiteres Sprachkonstrukt, das die Objektklasse und auszuführende Methode spezifiziert. [Henzen, 2010] schlägt aus diesem Grund die Erweiterung einer Gestendefinition um die Objektklasse und Ausführungsmethode anhand des folgenden Beispiels vor. Es wird eine Bewegung auf einem Objekt der Klasse ObjectClass ausgeführt und ein Eventbezug zu der Methode moveObject hergestellt.

gesture	=	MOVE(Object) : ObjectClass -> moveObject();
---------	---	---

3.4 TOUCH- UND POINT-INTERAKTION

In diesem Teil der Arbeit werden verwandte Arbeiten vorgestellt, die sich mit dem Nutzerverhalten in Touch- und Point-Interfaces und deren Kombination auseinandersetzen.

3.4.1 Studie von Schick et al.

Nach [Meißner, 2010, S. 54] können in größeren Touch-Bedienoberflächen lange Interaktionswege entstehen, womit sich die Fehlerrate bezüglich der Benutzereingaben erhöht. Auch SCHICK ET AL. erkannten, dass es in überdimensionierten Systemen passieren kann, dass bestimmte Regionen nicht mehr erreichbar sind. In diesem Fall werden Aufwendungen des Nutzers nötig, um in geifbare Nähe des Bereichs zu kommen. Das hat zur Folge, dass sich der zeitliche Aufwand, um eine Aufgabe via Touch zu lösen, erhöht.

“However, when using touch for large-scale surfaces, not every point is reachable. Therefore, an extension is required that keeps the intuitivity provided by touch: pointing.”

[Schick et al., 2009]

In der Studie werden die Interaktionskonzepte als einzelne und in Kombination hinsichtlich der Bearbeitungszeit einer einfachen Aufgabe evaluiert. Der Versuchsaufbau besteht aus einem vertikal aufgestellten Touchscreen und mehreren RGB-Kameras, die über dem Display angeordnet sind. Das Display ist 4m breit und 1,5m hoch in 0,8m über dem Boden angeordnet. Die Bildschirmauflösung beträgt 4096x1536 px.

Aufgabe der 19 Probanden war es, verschiedenfarbige Blöcke von 300x300px in ihre entsprechend eingefärbten Bereiche in der Mitte des Displays einzuordnen. Auffällig ist dabei die Größe der manipulierbaren Objekte. Nach [Schick et al., 2009] ist diese notwendig, da mit dem Abstand der Kameras und einer geringeren Auflösung des Kamerabildes die Präzision geringer wird. Zunächst sollte die Aufgabe nur durch Berührung des Displays erledigt

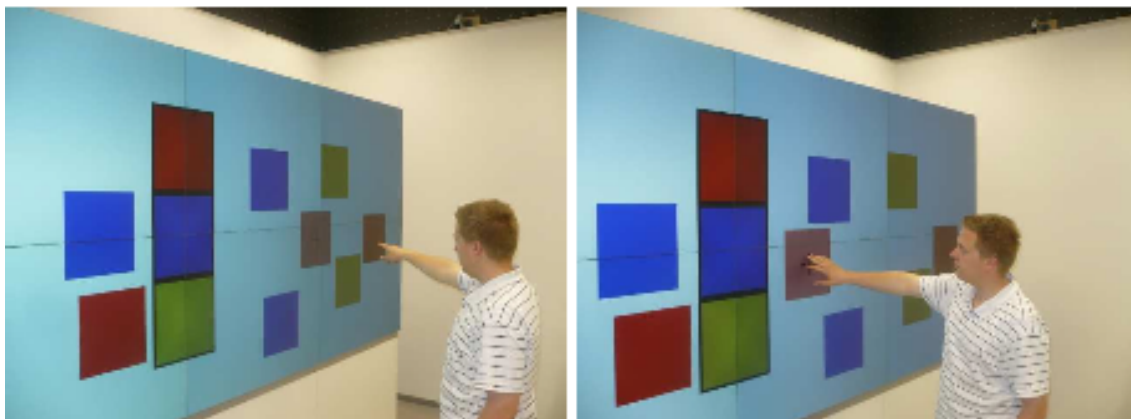


Abbildung 3.7: Touch- und Point-Interaktion in großflächigen Benutzeroberflächen [Schick et al., 2009]

werden. Anschließend durften die Nutzer mit einer Kombination der Interaktionstechniken verfahren. Dabei sollte es den Nutzern möglich sein zwischen den Eingabemodalitäten zu wechseln, ohne dass sie ihr Verhalten ändern müssen. Die Auswahl eines Objektes erfolgt mit ausgestrecktem Arm und einer Zeigegeste, die gehalten wird. Wird der Arm wieder heruntergenommen, dann verliert sich der Kontakt. Dieser Ansatz erlaubt die Nutzung der Interaktionstechniken auf ähnliche Weise. Um die Zeit zu erfassen, die ein Nutzer allein für die Interaktion braucht, beginnt die Messung mit der Auswahl des Objektes und endet mit dem Einordnen in den Zielbereich.

Die Auswertung der Daten ergab, dass die Zeit, die für das Einordnen eines Blockes unter Ausnutzung beider Interaktionstechniken benötigt wird, durchschnittlich um 0,5sec geringer ist, als wenn nur Touch verwendet werden darf. Weiterhin wird für die erhobenen Daten der Touch-Eingaben eine Standardabweichung von 0.88 ermittelt, während für die kombinierte Eingabemethode ein Wert von 0.54 festgestellt wird. Die Messwerte sind also bei einer kombinierten Eingabemethode weniger gestreut. Das bedeutet, dass die Interaktion mit den Objekten schneller und zuverlässiger ist. Eine anschließende Befragung der Teilnehmer hat ergeben, dass die kombinierte Bedienung bevorzugt wurde und die Objekte gegenüber Touch eindeutig besser erreichbar waren. Des Weiteren wird die Bedienung vergleichbar zu Touch als einfach und intuitiv eingeschätzt. (vgl. [Schick et al., 2009, S. 7])

3.4.2 Studie von Parker et al.

In [Parker et al., 2006] wird eine vergleichbare Studie beschrieben, die Touch- und Point-Interaktion in horizontal angeordneten Benutzeroberflächen untersucht. Hierbei wird jedoch nicht wie in [Schick et al., 2009] auf die kamerabasierte Erkennung von Hand und Arm zurückgegriffen, sondern es kommt ein spezielles Eingabegerät zum Einsatz. Das Gerät nennt

sich TractorBeam und ist ein kabelgebundener Stift, der wie ein Laser Pointer funktioniert.

Durch entsprechende Sensoren unterstützt das Eingabegerät sowohl Touch- als auch Point-Eingaben. Da die Eingabe mit dem TractorBeam die selben Merkmale aufweist, als wenn eine deiktische Geste mit dem Zeigefinger ausgeführt wird, kann sie für den Vergleich von fingerbasierten berührungssensitiven und berührungslosen Eingaben herangezogen werden. Um entfernte Regionen des Displays zu erreichen, muss sich der Akteur meist weit über den Tisch lehnen oder aufstehen. Ziel ist es, durch die hybride Eingabetechnik sowohl mit entfernten als auch mit nahen Objekten effizient zu interagieren. Abbildung 3.8 verdeutlicht die Funktionsweise.

Eine erste Messreihe hat zum Ziel, die Geschwindigkeit der beiden Eingabemodalitäten und deren Präzision zu vergleichen. Die Aufgabe der Probanden ist es mit der Point-Eingabetechnik kreisförmige Objekte, die in drei unterschiedlichen Winkeln (40° nach links, 0° , 40° nach rechts) und drei Entfernungen (195mm, 390mm, 780mm) angeordnet sind und sich außerdem in ihrer Größe (30mm, 60mm, 120mm) unterscheiden, zu selektieren. Abbildung 3.9 zeigt den Versuchsaufbau. Dabei kommt es häufig vor, dass Nutzer bei der Point-Geste über das Ziel hinauschnellen und danach eine korrigierende Bewegung machen. Die Ergebnisse zeigen, dass insbesondere mit kleineren Zielen mehr Korrekturbewegungen notwendig sind, um Objekt auszuwählen. Des Weiteren nimmt die Treffsicherheit mit größerer Distanz ab. Die Nutzung von Touch ist in diesem Fall die effizientere Lösung. Größere Zielbereiche lassen sich vergleichsweise schnell selektieren. Um mit nahen Zielen zu interagieren ist Touch die wesentlich schnellere Alternative. Die Gesamtauswertung ergibt jedoch, dass die Benutzeroberfläche mit Point gegenüber Touch genauso schnell und schneller bedient werden kann, ausgeschlossen kleine entfernte Ziele. (vgl. [Parker et al., 2006, S. 3]) Das Feedback der Teilnehmer zeigt, dass die Point-Interaktionstechnik bevorzugt wurde, vorallem weil sie komfortabler ist. Die Touch-Variante ist nach deren Aussagen anstrengender, wegen des permanenten Aufstehens und Streckens, um entfernte Ziele zu erreichen.

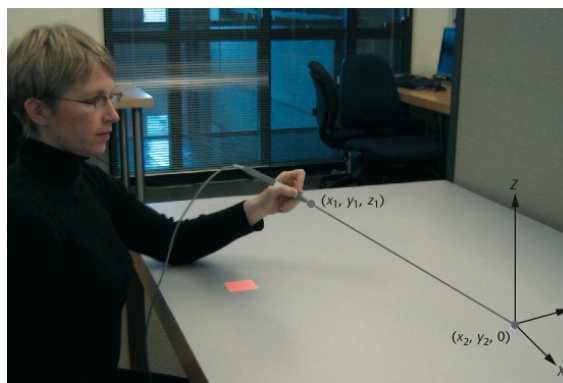


Abbildung 3.8: Touch- und Point-Interaktion mit dem TractorBeam [Parker et al., 2006]

In einem weiteren Experiment wird untersucht, welche Interaktionstechnik in Abhängigkeit von der Entfernung, Lage und Größe der Zielobjekte bevorzugt wird. Abbildung 3.10 zeigt die Zielbereiche. Es werden wieder drei Entfernungen (195mm, 390mm, 780mm) von der unteren Kante der Benutzeroberfläche und drei Zielgrößen (30mm, 60mm, 120mm) unterschieden. Die Lage eines Objekts ist einem Abstand von 530mm entweder links, mittig oder

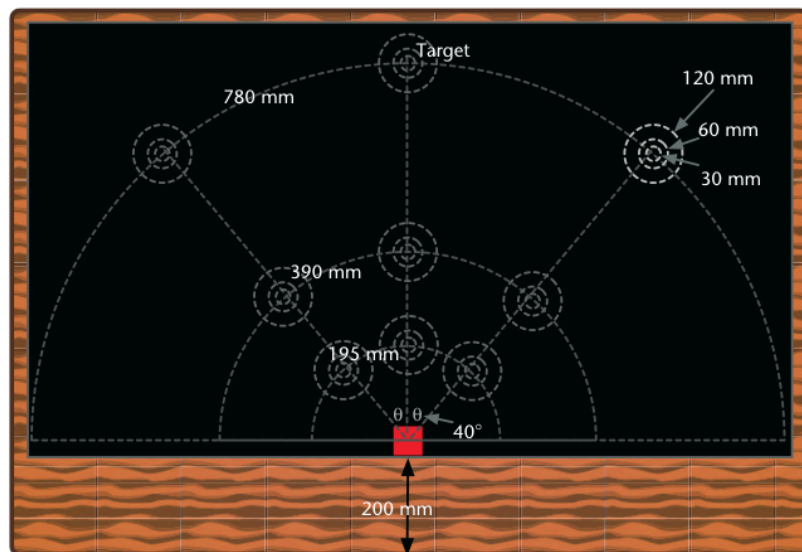


Abbildung 3.9: Aufbau des Experiments zur Evaluierung der Touch- und Point-Interaktionstechnik [Parker et al., 2006]

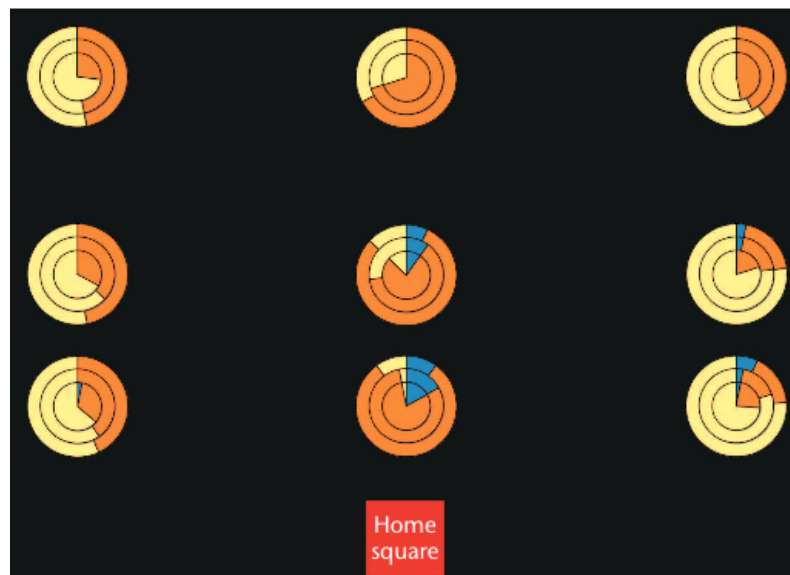


Abbildung 3.10: Bevorzugte Interaktionstechniken in Abhängigkeit von der Zielgröße, Entfernung und Region [Parker et al., 2006] - Touch (blau), Point (orange), "Reach-and-Point" (gelb)

rechts. Die 6 Teilnehmer (Rechtshänder) erhalten wieder die selbe Aufgabe. Das Auftreten der Objekte ist diesmal jedoch zufällig. Es wird dokumentiert, ob der Nutzer mit einer Berühreingabe oder Point-Geste reagiert. Bei der Point-Interaktion wird des Weiteren beobachtet, ob der Arm während der Auswahl des Objektes bewegt wird, um in die Nähe des Ziels zu kommen. In [Parker et al., 2006] wird diese Interaktion "reach-and-point" genannt. In Abbildung 3.10 ist die verhältnismäßige Nutzung der Interaktionstechniken für die jeweilige Position und jeweilige Zielgröße als Kreisdiagramm dargestellt. Blau gefärbte Bereiche stellen die Verwendung von Touch-Eingaben, orange die Point-Eingaben und gelb die Nutzung der "reach-and-point"-Technik dar. Die Auswertung zeigt, dass in den meisten Fällen, selbst in nahen Bereichen, die Point-Interaktion bevorzugt wurde. Das kann daran liegen, dass der abgestützte Arm eine komfortablere Interaktion zulässt. In entfernten Bereichen wurde vermehrt "reach-and-point" angewandt.

4 SYNTHESE

Unter Betrachtung der in Kapitel 3.1 vorgestellten multimodalen Anwendungen fällt auf, dass die Gestik sich in den Systemen hauptsächlich als sprachbegleitende Eingabeoption wiederfindet. Einer Verknüpfung der Touch- und Point-Schnittstellen wurde bisher keine besondere Beachtung geschenkt. Lediglich zwei Ansätze sind bekannt, die die Eingabemodalitäten als alternative Interaktionstechnik experimentell implementieren. (siehe Abschnitt 3.4) In Kapitel 4.1 wird auf der Grundlage der Ergebnisse der Studien zu den Interaktionskonzepten die Eignung von Touch- und Point-basierter Interaktion hinsichtlich der Ausrichtung und Dimension untersucht. Kapitel 4.1.2 beschäftigt sich weiterhin mit einem Formalisierungsansatz für berührungslose Gesten auf der Grundlage der in Kapitel 3.3.1 vorgestellten Gestenbeschreibungssprache GeForMT. Abschließend wird in Kapitel 4.3 die Beschreibung multimodaler Eingaben betrachtet.

4.1 KOMBINATION VON TOUCH- UND POINT-INTERAKTION

Während Touch-Gesten in einem zweidimensionalen Interaktionsraum ausgeführt werden und nur unter Berührung erkannt werden, bedient sich die berührungslose Interaktion einer dritten Dimension. Die Eingabebereiche orientieren sich nach Definition von Touch- und Point-Interface im Kapitel 2.2.2 an der visuellen Ausgabe. Abbildung 4.1 zeigt den prinzipiellen Aufbau der sensorischen Bereiche. Es wird illustriert, wie die sensorischen Bereiche zueinander angeordnet sind.

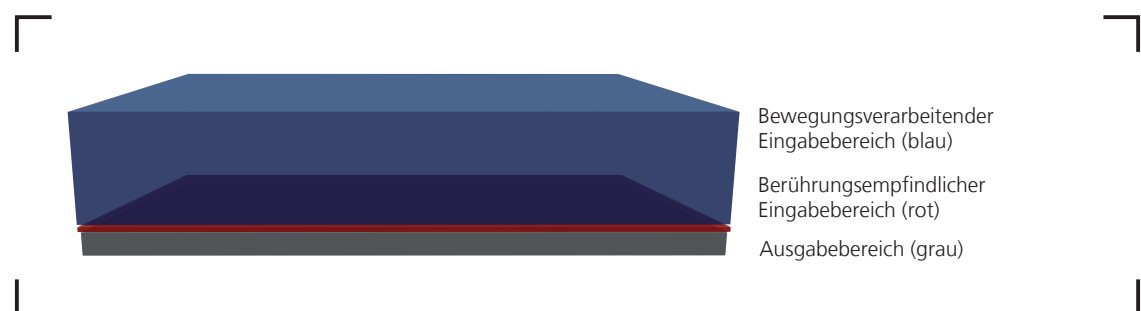


Abbildung 4.1: Anordnung von Ein- und Ausgabebereich

Auf unterster Ebene ist die Bildausgabe in der Fläche (grau) dargestellt. Der berührungsempfindlichen Eingabebereich (rot) deckt sich mit dem Ausgabebereich. Der virtuelle Raum, der mit der visuellen Darbietung digitaler Inhalte entsteht, täuscht je nach gestalterischer Ausprägung über die die Flächigkeit der Ausgabe hinweg. Demnach ist eine Herausforderung von Touch-Interfaces, unter der Einschränkung eines zweidimensionalen Interaktionsraum, in einer dreidimensionalen Darstellung zu navigieren. Der bewegungsverarbeitende Bereich (blau) bildet einen parallel zum Ausgabebereich orientierten dreidimensionalen Raum. Der Aufnahmebereich variiert je nach verwendeter Sensortechnik. Unter der Bedingung, dass die Geste in Bezug zum Ausgabebereich steht und insbesondere bei deiktischen Gesten eine Abbildung auf den Darstellungsbereich erfordert, wird der auswertbare Bereich als Quaderform abstrahiert. In Abschnitt 4.1.2 werden die Grenzen der sensorischen Bereiche bezüglich der Orientierung der Darstellungsebene genauer spezifiziert.

4.1.1 Dimensionen der Benutzeroberfläche

Ausgabegeräte wie LCD- und Plasmabildschirme oder auch Projektionssysteme werden durch technologische Fortschritte nicht nur größer, sondern können mittlerweile relativ preiswert erworben werden. Mit der Größe des Gerätes ergeben sich in dem Fall, dass Touch-Eingaben unterstützt werden jedoch Schwierigkeiten entfernte Regionen zu erreichen. Eine Lösung, dem Problem entgegenzuwirken, ist, die Entfernung und Größe der Zielbereiche von Bedienelementen an das Gerät anzupassen oder wie in Kapitel 3.4 dargelegt alternative Interaktionstechniken anzubieten.

Neben dem Problem der Erreichbarkeit ergeben sich hinsichtlich größerer Benutzeroberflächen weitere Probleme. Mit dem größeren Darstellungsbereich nimmt auch der Abstand des Nutzers zum Bildschirm zu. Der Grund ist, dass Anwender einen Überblick über den dargestellten Inhalt bevorzugen und damit von großen Displays zurücktreten. (vgl. [Malerczyk, 2009, S. 16]) Durch Kopf- und Augenbewegungen ist es möglich sich auch bei geringer Distanz in der Dargestellung zu orientieren, was auf Dauer jedoch belastend für die Gelenke ist und zur Ermüdung führt. Nutzer, welche einem überdimensionierten Multitouch-System gegenüberstehen, geraten so in den Zwiespalt: Durch Zurücktreten verlieren sie die Kontrolle, andernfalls den Überblick. Diesem Aspekt steht die geringe Präzision von Point-Gesten gegenüber. In [Schick et al., 2009] haben die manipulierbaren Objekte aus diesem Grund eine Größe von etwa 30x30cm. In dem Fall hat der Präzisionsverlust technische Ursachen, die auf den Informationsverlust bei der Abbildung des geringer auflösenden Kamerabildes auf die Darstellungsfläche zurückzuführen sind. Die Studie von PARKER ET AL. (siehe Abschnitt 3.4.2) zeigt jedoch, dass auch mit einer höheren Genauigkeit des Systems Schwierigkeiten bestehen, mit entfernten Objekte kleinerer Abmaße treffsicher zu interagieren.

Weiterhin spielt die Größe des Displays dahingehend eine Rolle, als dass die verwendbaren Gesten variieren. Kleinere mobile Touch-Geräte können maximal Finger-Eingaben erkennen. Als Point-Interface sind sie eher ungeeignet, da die Größe der dargestellten Informationen

entsprechend an das Gerät angepasst ist und die Treffsicherheit bei der Fokussierung kleiner Objekte wie bereits erwähnt gering ist. (vgl. Kapitel 3.4) Wie in Kapitel 2.3.3 vorgestellt, können mobile Geräte wiederum als Controller zum Einsatz kommen und damit zum Instrument handbasierter Eingaben werden. In größeren Geräten wird es möglich die Handerkennung und beidhändige Interaktion zu unterstützen. Unter diesen Voraussetzungen ist es unterdessen möglich, verschiedene Interaktionstechniken zu kombinieren. Zum einen können in Anlehnung an [Schick et al., 2009] und [Parker et al., 2006] entfernte Bereiche durch Point-Gesten und nahe Bereiche durch Touch bedient werden. Zum anderen können Gesten über Eingabetechniken hinweg komponiert werden, so dass die Aktion in einem neuen Kontext steht und dementsprechend anders interpretiert wird. Beispielsweise kann die linke Hand eine Berührung auf einem Objekt ausführen, während die rechte Hand eine Rotationsbewegung in den Raum macht, um das Objekt zu drehen. Digitale Tische und Wände geben so viel Platz, dass die Bedienung durch mehrere Nutzer möglich ist. Dieser Sachverhalt gilt für Touch- als auch Point-Interfaces. Wie im vorangegangenen Absatz beschrieben, müssen Objekte hinsichtlich ihrer Größe an die jeweilige Interaktionstechnik angepasst werden. Aus diesem Grund ist in Abhängigkeit des Abstands von Nutzer und Display die Darstellung zu adaptieren.

4.1.2 Ausrichtung der Benutzeroberfläche

Berührungsempfindliche Geräte können horizontal, vertikal oder diagonal angeordnet sein. Abbildung 4.2 illustriert die sensorischen Bereiche und den Interaktionsraum hinsichtlich der räumlichen Orientierung der Darstellungsfläche.

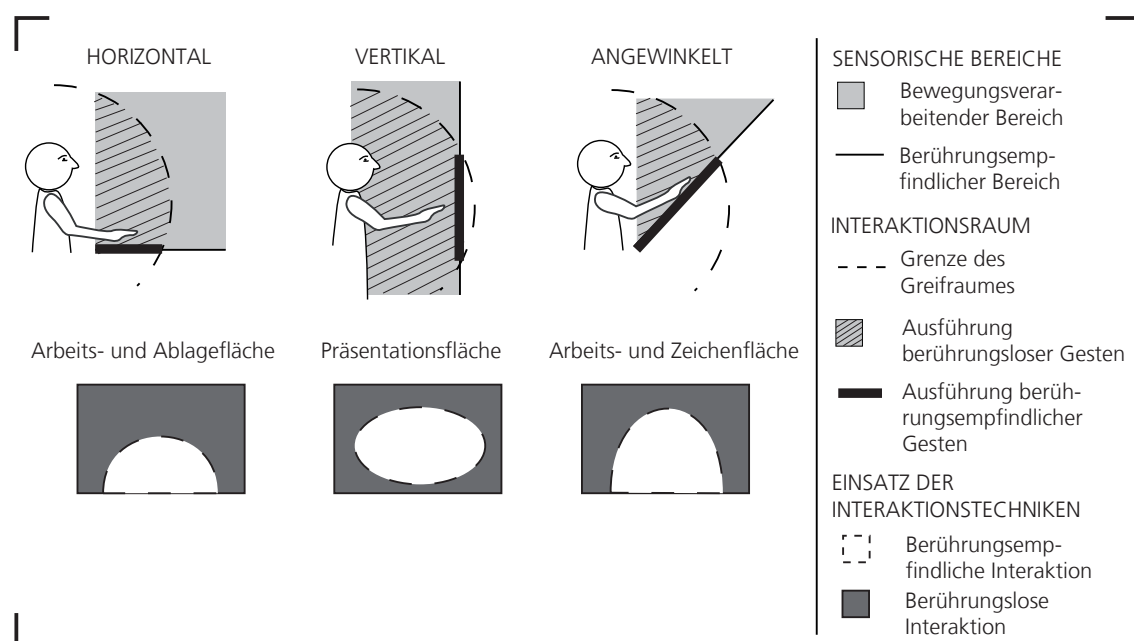


Abbildung 4.2: Interaktionsraum der berührungsempfindlichen und berührungslosen Gestensteuerung

Horizontale Flächen, wie Tische, dienen vorrangig der Ablage von Gegenständen, mit denen interagiert wird. In dieser Hinsicht sind derartige Eingabegeräte besonders für den Einsatz von Tangibles interessant. Da der Nutzer in der Regel an einem Tisch sitzt und seinen Arm auf der Oberfläche abstützen kann, bietet er gegenüber vertikalen Displays mehr Komfort. Nach [Parker et al., 2006] ist dies ein Faktor, der für die berührungslose Interaktion mit dem Interface spricht, da der Arm auf der Tischkante abgestützt bleiben kann. Abbildung 4.2 stellt den bewegungsverarbeitenden Bereich idealerweise als quadratische Form über dem Tisch dar. Er ist durch den höchsten erreichbaren Punkt über dem Eingabegerät und dessen Flächenabmaße begrenzt. Innerhalb des Raumes, der sich durch die Schnittmenge aus Greifraum und sensorischen Bereich ergibt, kann die berührungsempfindliche Schnittstelle problemlos bedient werden, sowie Gesten im bewegungsverarbeitenden Bereich ausgeführt werden. Für eine präzise und direkte Manipulation der Inhalte empfiehlt sich hier die Anwendung der Touch-Interaktion. Es bietet sich aber auch eine Komposition der Eingabemethoden an, sobald beide Hände eingesetzt werden. Regionen die außerhalb des Greifraumes liegen, können indirekt durch Zeigegesten bedient werden. Horizontale Systeme unterstützen die Kommunikation in der Gruppe besser als bei einer vertikalen Anordnung des Displays. (vgl. [Janssen, 2006, S. 88]) Dadurch, dass Systemnutzer sich um die Fläche herum anordnen, was eine Aufteilung in zugeordneten Arbeitsflächen zur Folge hat, sind die Teilnehmer zunächst gleichberechtigt. Findet dennoch eine Rollenverteilung statt, so ist ein Rollentausch zumindest flexibler. Das Interface hat die Zentrierung der Information und das kollaborative Arbeiten an digitalen Inhalten zum Ziel.

Vertikal angeordnete Systeme, wie digitale Tafeln oder Wände, fungieren nach [Janssen, 2006] als "zentralen Attraktions- und Kommunikationspunkt" und sind vorallem zur Präsentation und Moderation geeignet. Der sensorische Bereich für die Erfassung von Körperbewegungen begrenzt sich in der Tiefe. Das heißt bis zu einem gewissen Abstand zum Display kann berührungslos mit dem System interagiert werden. Hier zeigt sich das Problem des Touch-Interface, das bereits im vorigen Abschnitt erläutert wurde. Der Nutzer nimmt für einen besseren Überblick Abstand zum Display. Sobald das Touch-Display außer Reichweite ist, also aus dem Greifraum verschwindet, sind direkte Eingaben nicht mehr möglich. Aus diesem Grund bietet sich die berührungslose Interaktion als primäre Eingabetechnik an. Dem Nutzer bietet sich ein entsprechend großer Raum für die Ausführung. In einem kollaborativen Szenario begiebt sich ein Nutzer in die Rolle des Moderators, welcher die Kommunikation mit dem System dominiert, während anderen Teilnehmer eine passive Rolle einnehmen. (vgl. [Gallardo et al., 2010]) Mit der Erkennung deiktischer Gesten, kann es auch passiven Teilnehmern aus Distanz gelingen sich einzubringen und die Kommunikation zu fördern. Der bewegungsverarbeitende Bereich muss einen dementsprechend großen Bereich vor der Darstellungsfläche umfassen.

Der Vollständigkeit halber werden auch angewinkelte Systeme, die einem Architektentisch oder Pult gleich kommen, betrachtet. Das Interface stellt weder eine zentrale Arbeits- und

Ablagefläche, noch ein geeignetes Medium zur Präsentation dar. Da die Benutzeroberfläche optimal zum Anwender ausgerichtet ist, kann gegenüber dem horizontalen und vertikalen Interface eine größere Fläche direkt bedient werden. Dafür steht ein wesentlich geringerer Raum für die Ausführung berührungsloser Gesten zur Verfügung. Der sensorische Bereich, in dem berührungslose Gesten erfasst werden ist sowohl durch die erreichbare Höhe als auch den Nutzer begrenzt.

Zusammengefasst wird das Interaktionskonzept horizontaler Bedienoberfläche von der berührungsempfindliche Schnittstelle dominiert. Eine Erweiterung um berührunglosen Eingaben stellt sich als förderlich für die Interaktion mit entfernten Elementen heraus. Des Weiteren können durch beidhändige Interaktion Eingabetechniken kombiniert werden, womit eine neue Ebene der Interpretation von Gesten geschaffen wird. In vertikal angelegten Benutzeroberflächen dominiert hingegen die berührungslose Interaktion. Angewinkelte Systeme fördern die Touch-Interaktion mit einem optimierten greifbaren Bereich der Fläche. Nichtsdestotrotz sind hier vor allem Regionen an der oberen bzw. gegenüberliegenden Kante der Darstellungsfläche schwer zu erreichen. In diesem Fall gibt es jedoch nicht wie in horizontalen Systemen die Möglichkeit um das Display herumzugehen. Die Erweiterung einer berührungslosen Steuerung erscheint hier sinnvoll, wird aber als Spezialfall erachtet. Die Point-Interaktion erfordert eine Größenanpassung von Bedienelementen, die nicht im Greifraum des Nutzers liegen. In Abbildung 4.2 sind diese Bereiche des Interface tiefgrau eingefärbt. Ein weiterer Ansatz ist, durch Zeigegesten fokussierte entfernte Elemente zum Nutzer heranzuholen, so dass ein direkter Zugriff wieder möglich wird.

4.2 FORMALISIERUNG RÄUMLICHER HANDGESTEN

Der Formalisierungsansatz GeForMT von KAMMER ET AL. bezieht sich auf die Beschreibung von Touch-Gesten. In Anbetracht der Feststellung, dass Gesten in Point-Interfaces ähnliche Bewegungsmuster aufweisen (vgl. Abschnitt 3.2.2), gibt der Ansatz der Semiotik Potenziale her, die für eine formale Beschreibung räumlicher Gesten aufgegriffen werden können. Im Folgenden werden in Abschnitt 4.2.1 motorische Aspekte von Handbewegungen betrachtet, die für die Ansätze einer formalen Beschreibung räumlicher Handgesten in den darauffolgenden Abschnitten herangezogen werden.

4.2.1 Bewegungsfreiheit der Hand Hand-Arm-Koordination

Bewegungen mit der Hand sind in gewisser Weise von Armbewegungen abhängig. Insbesondere die Verlagerung der Hand - forwards, rückwärts, links, rechts, aufwärts wie abwärts - ist ohne die Gelenke des Ellenbogens und der Schulter nicht zu bewerkstelligen. Mit ausgestrecktem Arm ist die Grenze gesetzt, bis zu der die Hand in einer festen Position des Agierenden positioniert werden kann (Greifraum). Mit ausgestrecktem Arm ist es möglich Handgesten besonders hervorzuheben, eindeutig erkennbar zu machen oder gar eine andere Bedeutung zu geben. In Schick et al. [2009] dient der ausgestreckte Arm beispielsweise als

Indikator für eine Zeigehandlung. Das typische Melden, ein ausgestreckter Arm mit erhobenem Zeigefinger, kann vergleichsweise anders interpretiert werden, wenn der Arm gebeugt ist und die Geste direkt vor dem Körper ausgeführt wird.

Weiterhin ist eine Drehung der Hand um die vertikale Achse nur dadurch möglich, dass Elle, Speiche und dessen Fixierung durch Sehnen und Muskeln ein flexibles Gebilde ergeben. Wird nur der Unterarm involviert, ist eine Drehung der Hand um etwa 180° möglich. Unter Hinzunahme des Oberarms kann eine Drehung um fast 360° um die Armachse ausgeführt werden. Dies wird insbesondere bei ausgestrecktem Arm als anstrengend empfunden, da es die Sehnen und Muskeln stark beansprucht.

Handgelenk

Die Hauptbewegung, die über das Handgelenk ausgeführt wird, ist die Aufwärts- und Abwärtsdrehung des Handtellers sind. Aufwärts ist es mit bis zu 90° möglich die Hand orthogonal zur Unterarmachse zu stellen. Auch abwärts ist das Umlappen der Hand bis zu dieser Gradzahl möglich. Um diesen Winkel zu erreichen, ist starke Dehnung der Sehnen notwendig. Aus diesem Grund wird meist der Arm in die Bewegung involviert um die Hand in die richtige Position zu bringen. In der Ruheposition, das heißt wenn die Hand in einer Linie mit dem Unterarm ist, kann der Handteller seitlich geneigt werden. Nach außen ist eine Neigung bis etwa 45° möglich, nach innen deutlich weniger. Diese Bewegung kann beim Winken aus dem Handgelenk heraus beobachtet werden.

Daumen

Durch das Daumensattelgelenk ist es möglich den Daumen bis zu einem rechten Winkel in Ebene oder quer zum Handteller abzustellen. Unter Ausnutzung aller Gelenke kann der Daumen in die Handfläche hineingelegt werden und von den Fingern umschlossen werden. Dieser Fall stellt eine Variante dar, eine Faust zu machen. Grundlegend können vier verschiedene Posen des Daumens festgestellt werden: angelegt, abgestellt, orthogonal zur Handfläche und in die Handfläche gelegt.

Finger

Die Finger verfügen über jeweils drei Gelenke, mit denen Bewegungen ausgeführt werden, die hauptsächlich der Krümmung des jeweiligen Fingers in Richtung der Handinnenfläche dienen. Dabei ist zu beobachten, dass bei der Krümmung eine Abhängigkeit zwischen Fingerendgelenk und Fingermittelgelenk besteht. Die Fingerspitze krümmt sich automatisch mit. Aus diesem Grund wird das Fingerendgelenk als nicht relevant für eine Beschreibung von Handgesten bewertet. Für die Krümmungsbewegung eines Fingers können folgende drei Unterscheidungen festgehalten werden: Es wird nur das Fingermittelgelenk gekrümmt oder es wird nur das Fingergrundgelenk gekrümmt und der Finger selbst bleibt gestreckt oder beide Gelenke werden in die Handfläche hinein gekrümmt.

Mit dem Fingergrundgelenk ist darüber hinaus eine leichte Neigung zu den Seiten möglich.

Die Finger können dadurch voneinander abgespreizt werden. Wie in Abschnitt 3.2.2 gezeigt, stellt die Spreizung der Finger bereits eine relevante Handform dar. Auf dieser Grundlage können weitere Handformen festgestellt werden, in denen alle Fingerglieder involviert sind und eine synchrone Ausführung erfolgt.

4.2.2 Atomare Gesten

Mit den atomaren Gesten POINT, DEPOINT, HOLD und MOVE werden in der Spezifikation von GeForMT punkthafte Eingaben beschrieben. Sie können teilweise und mit veränderter Bedeutung für die Beschreibung der berührungslosen Interaktion übernommen werden. In dem Sinne ist die POINT-Geste eine deiktische Handlung, die sich durch eine relative und kontinuierliche Ausführung charakterisiert. Während POINT die Präsenz einer Handlung vermittelt, wird mit DEPOINT eine Abbruchhandlung registriert. In dem Fall wird keine Hand im bewegungsverarbeitenden Bereich registriert. HOLD zeichnet sich dadurch aus, dass eine Geste über einen definierten Zeitraum in ihrer Position gehalten wird. Damit ist es möglich statische Gesten zu beschreiben. Die formale Beschreibung einer willkürlichen Bewegung wird mit der kontinuierlichen Erkennung und Abbildung von POINT-Eingaben überflüssig.

4.2.3 Formalisierung von Handformen

Handformen beschreiben ausschließlich den statischen Zustand der Hand und beruhen auf den allgemeinen Gesteneigenschaften Größe und Form (vgl. Kapitel 2.4.1). In GeForMT werden diese Eigenschaften durch die EBNF-Regel *function* beachtet. Die Erweiterung der Funktionen um weitere Handformen bietet sich also für die Formalisierung statischer Raum-Gesten an. Nach den in Kapitel 3.2.2 vorgestellten Handposen und der Untersuchung von Bewegungsmöglichkeiten der Hand im vorangegangenen Kapitel, sind folgende in Tabelle 4.1 dargestellten Handformen relevant.

Handform	Beschreibung
FINGER (F)	Austrecken eines Fingers.
FLAT HAND (H_FLAT)	Die Hand ist flach und gestreckt. Die Finger und der Daumen sind dabei angelegt und ebenfalls gestreckt.
SPREAD HAND (H_SPREAD)	Die Hand ist flach und gestreckt. Finger und Daumen sind auseinander gespreizt und gestreckt.
FIST (H_FIST)	Die Hand ist zu einer Faust geballt, indem alle Finger gekrümmt in die Handfläche gelegt werden. Der Daumen wird entweder von den Fingern umschlossen oder außen angelegt.
CURVED HAND (H_CURVED)	Die Hand ist rund geformt. Finger und Daumen sind leicht gekrümmt. Es spielt dabei keine Rolle ob die Fingerglieder angelegt oder gespreizt sind.
ANGLED HAND (H_ANGLED)	Alle Fingerglieder stehen im rechten Winkel zur Handfläche.

Tabelle 4.1: Unterscheidung von Handformen

In Anlehnung an die Formalisierung von Touch-Gesten, wird grundlegend zwischen hand- und fingerbasierter Eingabe unterschieden. Ein generischer Blob, beschrieben durch `1B(..)`, ist eine Touch-spezifische Eingabe, die für die Erkennung beliebiger Objektformen auf einem berührungsempfindlichen Eingabegerät bestimmt ist. Eine Einschränkung der Objektformen wird in GeForMT nicht vorgenommen, womit die Auswertung und Interpretation einer beliebigen Form dem Framework überlassen wird. An der Stelle wird der Vorschlag gemacht die Blobs zu spezifizieren, indem der Funktion eine Bezeichnung der charakteristischen Form zugesetzt wird. Beispielsweise kann die charakteristische Form einer Faust oder Handkante auf dem Touchscreen durch `1B_FIST(..)` und `1B_EDGE(..)` oder auch Objektformen wie beispielsweise `1B_ELLIPSE(..)` oder `1_SQUARE(..)` beschrieben werden.

Der beschriebene Ansatz kann auf berührungslose Gesten übertragen werden. Blobs sind in Point-Interfaces nicht definiert und kommen daher nicht in Frage. Stattdessen wird vorgeschlagen, die Definition der Handgeste (`1H(..)`) um einen Sprachzusatz zu erweitern. Die Syntax würde mit den folgenden EBNF-Ausdrücken adaptiert werden. Fettgedruckte Ausdrücke markieren im Folgenden Änderungen gegenüber der GeForMT-Syntax.

```
function ::= [integer]'F' | [integer]'H'[_form]
form      ::= 'FLAT' | 'SPREAD' | 'FIST' | 'CURVED' | 'ANGLED'
```

Da das Schlüsselwort `SPREAD` bereits für die Angabe von Relativbeziehungen verwendet wird, muss in dem Fall eine Anpassung der Namensgebung vorgenommen werden, um verwirrende Dopplungen zu vermeiden. Da es im englischen Sprachgebrauch keine treffendere Bezeichnung für das Auseinanderspreizen der Fingerglieder gibt, wird der Vorschlag unterbreitet `APART` als Bezeichnung für die Formalisierung der gegensätzlichen Bewegungsrichtung zu verwenden.

```
relation ::= 'CROSS' | 'JOIN' | 'SYNC' | 'APART' | 'CONNECT'
```

4.2.4 Formalisierung von Handstellungen

Eine Handstellung beschreibt die Ausrichtung der Handfläche im Raum, unabhängig von der Handform und -bewegung. Unter Betrachtung der Freiheitsgrade der Hand (siehe Abschnitt 2.3.2) und der Bewegungsmöglichkeiten über das Handgelenk- und Armbewegungen, können sechs verschiedene Handstellungen identifiziert werden, die sich auf die Ausrichtung der Handfläche beziehen. Sie werden in Tabelle 4.2 zusammengefasst.

Die Stellungen der Hand beziehen sich dabei nicht auf die Interfaceebene, sondern sind aus der Perspektive des Agierenden beschrieben. Dieser Ansatz wird als leicht nachvollziehbar erachtet, führt allerdings dazu, dass die selbe Gestendefinition gegenüber einer variierenden Anordnung der Benutzeroberfläche eine unterschiedliche Bedeutungen erhält. Beispielsweise würde die Geste `1H_FLAT_FRONT(HOLD)` in einem horizontal angelegten Interface bedeuten, dass die Hand senkrecht zum Display ausgerichtet ist. In Bezug zu einem vertikal ausgestellt-

Handstellung	Beschreibung
IN	Die Handfläche ist nach innen gerichtet.
OUT	Die Handfläche ist nach außen gerichtet.
FRONT	Der Handrücken ist dem Agierenden zugewandt, die Handfläche dementsprechend seiner Umgebung.
REAR	Die Handfläche ist auf den Agierenden gerichtet.
TOP	Die Handfläche ist nach oben ausgerichtet
BOTTOM	Die Handfläche ist nach unten ausgerichtet.

Tabelle 4.2: Unterscheidung von Handstellungen

ten Display bedeutet der Ausdruck, dass die Handfläche auf das Interface zugewandt und parallel gestellt ist. In einer vereinfachten Alternativlösung können drei Stellungen der Hand in Relation zum Display unterschieden werden:

- Die Handfläche ist parallel zum Display angeordnet.
- Die Handkante ist zur Benutzeroberfläche ausgerichtet.
- Die Hand ist so ausgerichtet, dass bei einer flachen Hand die Fingerspitzen dem Interface zugewandt sind.

In Anlehnung an die Syntax von GeForMT können die unterschiedenen Handstellungen des ersten Vorschlags mit folgenden Regeln ausgedrückt werden. Dabei wird das Sprachelement `function` um die erläuterten Optionen der neuen Regel `posture` erweitert.

```

gesture    ::= function['_posture']('atomfocus(',')atomfocus(*)') |
              atomfocus(',')atomfocus(*)
posture    ::= 'IN' | 'OUT' | 'FRONT' | 'REAR' | 'TOP' | 'BOTTOM'
```

4.2.5 Rolle von Finger und Daumen

Diskutabel ist, ob der Daumen wie in GeForMT als einfacher (Finger-)Kontakt ausgewertet wird oder ob ihm eine gesonderte Funktion zukommt. Gesten wie der nach unten oder oben gerichtete Daumen, um damit eine Bewertung auszudrücken oder Richtungsweisungen sind Beispiele, die für eine Unterscheidung von Daumen und Fingern sprechen. Ein erster Lösungsvorschlag, um diese Gesten zu formalisieren wäre die Hinzunahme der Funktion `1T(..)`, wobei `T` für `THUMB` steht. Nach [Epps et al., 2006] kommen, wenn auch seltener, Handposen in 'L'- oder 'C'-Form vor. Die 'L'-Geste setzt sich aus dem ausgestreckten Zeigefinger und einem senkrecht dazu stehenden ausgestreckten Daumen zusammen, während die 'C'-Geste auf einer rundgeformten oder angewinkelten Handform (siehe 4.2.3) basiert und der Daumen dabei abgestellt wird. Laut [Epps et al., 2006] kommen sie besonders insbesondere in Kontextmenüs und Zoomfunktionen zum Einsatz. Auf Grundlage dieser Beispiele lässt sich der Daumen als eine Erweiterung von Finger- und Handeingaben abstrahieren. Ein Alternativvorschlag ist daher lediglich zu unterscheiden, ob der Daumen abgespreizt ist oder nicht. Die

EBNF-Regeln können in dem Fall wie folgt aussehen:

```
gesture ::= function['_posture']['+'extension']('atomfocus(',')atomfocus')*
        | atomfocus(',')atomfocus'*
function ::= [integer]'F' | [integer]'H'[_form]
extension ::= 'T' | 'THUMB'
```

Folgende Beispiele zeigen, wie die exemplarischen Gesten mit der Syntax beschrieben werden:

```
ThumbUp    = 1H_FIST_IN+T(HOLD(o));
ThumbDown  = 1H_FIST_OUT+T(HOLD(o));
L          = 1F+THUMB(HOLD);
C          = 1H_CURVED+THUMB(HOLD);
```

4.2.6 Formalisierung räumlicher Handbewegungen

Eine weitere Eigenschaft einer Geste ist nach Abschnitt 2.4.1 die Bewegung. Sie beschreibt die Dynamik in der Ausführung hand- und fingerbasierter Eingaben. Dabei spielt vorallem die Bewegungsrichtung eine tragende Rolle. In [Kammer et al., 2010] werden Bewegungsrichtungen dynamischer Touch-Gesten über acht unterscheidbare Himmelsrichtungen beschrieben. Dieser Ansatz wird für berührungslose Gesten in den dreidimensionalen Raum überführt.

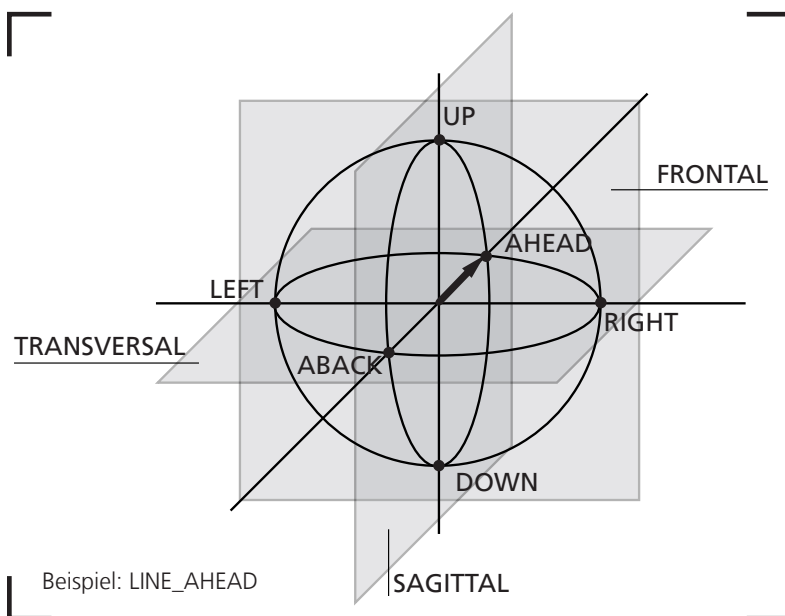


Abbildung 4.3: Raumebenen und Bewegungsrichtungen

Abbildung 4.3 zeigt die drei Raumebenen, in denen jeweils Bewegungen stattfinden können. Diskutabel ist dabei, ob eine Unterscheidung von acht Richtungen pro Ebene notwendig ist. In Abschnitt wurde erläutert, dass die berührungslose Interaktionstechnik weniger präzise ist. Aus diesem Grund werden im Wesentlichen sechs Richtungsangaben unterschieden, die in

Tabelle 4.3 zusammengefasst und in Abbildung 4.3 visualisiert sind. Wie bei der Benennung der Handstellungen in Abschnitt 4.2.4 ist der Nutzer das Bezugssystem.

Handbewegung	Beschreibung
UP	Aufwärtsbewegung der Hand.
DOWN	Abwärtsbewegung der Hand.
LEFT	Verlagerung der Hand nach links.
RIGHT	Verlagerung der Hand nach rechts.
AHEAD	Vorwärtsverlagerung der Hand. Die Bewegung erfolgt geradezu vom Agierenden weg.
ABACK	Rückwärtsverlagerung der Hand. Ausführung zum Agierenden hin.

Tabelle 4.3: Unterscheidung von Handbewegungen

Um Bewegungsrichtungen in Bezug zur Bedienoberfläche zu beschreiben, kann alternativ eine einfache Erweiterung der Regel *direction* um die Schlüsselwörter *AHEAD* und *ABACK* erfolgen, um Bewegungen zum Interface hin oder weg zu erklären. Die acht Himmelsrichtungen würden in dem Fall für eine genauere Definition der Bewegung in frontaler Ebene beibehalten werden.

Nach [Kammer et al., 2010] deckt die Komposition der drei atomaren Ausführungsformen *LINE*, *CIRCLE* und *SEMICIRCLE* die meisten Zeichen und Symbole in einer Ebene ab. Im Raum bieten sich drei Ebenen an. Der formale Ausdruck *LINE_AHEAD* beschreibt beispielsweise eine geradlinige Bewegung vom Körper weg. Die Beschreibung von (Halb-)Kreisbewegungen erfordert weitere Anpassungen der Grammatik, da mit der bloßen Angabe einer Richtung nicht bestimmt werden kann, in welcher Ebene die Bewegung ausgeführt wird. Im Beispiel *SEMICIRCLE_RIGHT* kann eine Kreisbewegungen beispielsweise über eine Aufwärts-, Abwärts-, Vorwärts- oder Rückwärtsverlagerung der Hand erfolgen, womit sich vier Varianten ergeben. Die zusätzlichen Sprachelemente *CLOCKWISE* (CW) und *COUNTERCLOCKWISE* (CCW) lösen das Problem nur teilweise und der Uhrzeigersinn ist in der Sagittalebene uneindeutig. Unter Angabe der Raumebene (*FRONTAL*, *SAGITTAL* oder *TRANSVERSAL*) können die Uneindeutigkeiten aufgelöst werden. Eine Alternative zum Uhrzeigersinn ist die Angabe einer tendenziellen Bewegungsrichtung. Der Ausdruck *SEMICIRCLE_SAGITTAL_AHEAD* erklärt beispielsweise eine Vorwärtsbewegung in der sagittalen Ebene. Mit *CIRCLE_FRONTAL_RIGHT* wird eine Kreisbewegung in frontaler Ebene rechts herum beschrieben. Von wo die Geste beginnt, kann hiermit jedoch nicht erklärt werden. Die Syntaxregeln sehen nach diesem Lösungsvorschlag (mit Agierenden als Bezugssystem) wie folgt aus:

```

atom      ::= identifier| vector['_direction']| shape['_plane']['_direction']
vector    ::= 'LINE'
shape     ::= 'CIRCLE' | 'SEMICIRCLE'
direction ::= 'UP' | 'DOWN' | 'LEFT' | 'RIGHT' | 'AHEAD' | 'ABACK'
plane     ::= 'FRONTAL' | 'SAGITTAL' | 'TRANSVERSAL'

```

4.2.7 Zusammenfassung

Die Verwendung des Ansatzes der GeForMT für die Beschreibung räumlicher Handgesten erfordert die Unterscheidung von Handformen und Handstellungen sowie die Überführung der Richtungsangaben von Bewegungen in den dreidimensionalen Raum. Die atomaren Gesten POINT, DEPOINT und HOLD beziehen sich lediglich auf die Erfassung von Bewegungen im sensorischen Bereich, wobei unter POINT eine kontinuierliche und willkürliche Bewegung zu verstehen ist. In Tabelle 4.4 sind die Formalisierungsvorschläge der vorangegangenen Abschnitte zusammengefasst.

Unterscheidbare Eigenschaften	Adaptierung	Formalisierungsansätze
Atomare Gesten	Adaption der Semantik	POINT, DEPOINT, HOLD
Handformen	Unterscheidung von finger- und handbasierter Eingabe	F(..), H(..)
	Unterscheidung statischer Handposen	H_FLAT, H_SPREAD, H_CURVED, H_FIST, H_ANGLED
Daumen	Alternative 1: Hinzunahme des Daumens aufbauend auf Finger- und Handposen	+THUMB
	Alternative 2: Unterscheidung von Daumen und Finger	T(..)
Handstellungen	Alternative 1: Ausrichtung der Handfläche in Relation zum Agierenden	IN, OUT, FRONT, REAR, TOP, BOTTOM
	Alternative 2: Ausrichtung der Handfläche in Relation zum Display	frontal, senkrecht mit Handkante zum Interface, senkrecht mit Fingerspitzen zum Interface
Bewegung	Alternative 1: Überführung linearer Bewegungen in den Raum	LEFT, RIGHT, UP, DOWN, AHEAD, ABACK
	Alternative 2: Bewegungen in Relation zum Display	Himmelsrichtungen in frontaler Ebene und Erweiterung um AHEAD und ABACK
	Übertragung pfadbasierter (Halb-)Kreisbewegungen auf Raumebenen	FRONTAL, TRANSVERSAL, SAGITTAL und tendenzielle Richtungsangabe

Tabelle 4.4: Zusammenfassung der Formalisierungsansätze

Die folgenden Beispiele zeigen, wie die in Abschnitt 3.2.2 beschriebenen Standardgesten demnach syntaktisch aufgebaut werden. Dabei wurden die unter "Alternative 1" gekennzeichneten Ansätze verwendet.

Point	=	1F(POINT(o));
Grab	=	1H_FIST_FRONT(HOLD(o));
PincerGrasp	=	1H_CURVED_FRONT+T(HOLD(o)) 1F_FRONT+T(HOLD(o));
Wave	=	1H_FLAT_FRONT(LINE_LEFT);
Slap	=	1H_FLAT_IN(LINE_LEFT);

Mit den vorgestellten Erweiterungen der Sprache können komplexe Bewegungen und Konfigurationen der Hand beschrieben werden, wobei in Anbetracht der verwendeten Technologie und Methoden der Handerkennung abzuwägen ist, ob eine genaue Unterscheidung von Handformen und Handstellungen in einem Point-Interface notwendig ist. Die entsprechenden formalen Ausdrücke sind daher optionale Parameter, um die Geste genauer zu spezifizieren. Die unter "Alternative 2" aufgeführten Ansätze stellen ein vereinfachtes Erweiterungskonzept der Sprache dar, das eine Beschreibung der Gesten in Bezug auf das Display vorsieht. Die Stellung der Hand spielt dabei eine weniger bedeutende Rolle und könnte auch als Handform abstrahiert werden. Für die Beschreibung von Bewegungsrichtungen wird in frontaler Ebene das Konzept der Windrose beibehalten und lediglich um Attribute erweitert, die eine Bewegung in die Tiefe ausdrücken. Demnach sind weniger Anpassungen der Syntax erforderlich.

Profil	Spracherweiterung	Gestenerkennung
2D-Projektion	Atomare Gesten	Erkennung deiktischer Handlungen und Punktprojektion auf die zweidimensionale Darstellungsebene
3D-Position	Bewegungspfade im Raum	Ermittlung der Distanz zum Interface und 3D-Koordinatenberechnung
Handposen	Handformen	Erkennung handinterner Gelenkstellungen, Auswertung von Konturen und Formen
Handposen erweitert	Handstellungen	Erkennung und Auswertung von Arm- und Handgelenkbewegungen

Tabelle 4.5: Profile der berührungslosen gestenbasierten Interaktion

Prinzipiell können mit der erarbeiteten Spezifikation vier Grade der Unterstützung (Profile) der berührungslosen Interaktion in einem Point-Interface bestimmt werden. Tabelle 4.5 zeigt die Profile neben der jeweiligen sprachlichen Spezifikation aus Tabelle 4.4 und den Anforderungen an die Gestenerkennung des Systems. Zeigesysteme, wie beispielsweise der iPoint Presenter oder Zeigeeinstrumente wie die Wiimote, werten Eingaben als Punkte aus, die auf die zweidimensionale Darstellungsebene projiziert werden. Gesten sind in dem Fall nur in der frontalen Ebene möglich. Unter der Voraussetzung, dass das System Tiefeninformation

nen ausgewertet, können Bewegungen im dreidimensionalen Raum beschrieben werden. Die Erkennung von Handformen erfordert in videobasierten Systemen eine Merkmalsextraktion oder entsprechende Sensorik, die die handinterne Stellung der Gelenke erfasst. Um die Ausrichtung der Hand erkennen zu können, sind Armbewegungen in den Erkennungsprozess zu involvieren.

4.3 FORMALISIERUNG MULTIMODALER BENUTZEREINGABEN

Im Sinne einer Verknüpfung der Eingabetechniken wird im Folgenden eine Lösung beschrieben, wie die Sprache erweitert werden kann um berührungslose und berührungsempfindliche Gesten zu komponieren. Hierfür muss die Ausführungszeit zweier Gesten in Beziehung gesetzt werden. In BML (vgl. [Kopp et al., 2006]) werden zeitliche Abhängigkeiten über Synchronisationspunkte (sync-points) beschrieben. Die Ausführung einer Geste wird in dem Ansatz in Phasen unterteilt. Deren Start- und Endzeitpunkt werden in Elementen, die weitere Aktionsmodalitäten beschreiben, unter Angabe der ID eingebunden. Die Komposition wird damit über den Bezeichner einer Geste realisiert. Um welche Aktionsmodalität es sich handelt, wird in BML anhand des Elementnamens (Tag-Name) bestimmt. In der GeForMT-Syntax existieren für die Beschreibung der zeitlichen Abfolge von Gesten die Kompositionsoperatoren (vgl. [Kammer et al., 2010]). Sie lassen sich für die Synchronisation multimodaler Benutzereingaben heranziehen.

Komposition	Zeichen	Beschreibung
Option		Aktionen sind als alternative Eingabetechniken beschrieben
Sequenz	,	Aktionen treten in sukzessiver Abfolge auf
Synchronität	*	Aktionen beginnen und enden zeitgleich
Asynchronität	+	Aktionen werden simultan ausgeführt, müssen aber weder zeitgleich beginnen noch enden

Tabelle 4.6: Kompositionsoperatoren für die modalitätsbezogene Verknüpfung

In Tabelle 4.6 sind die Kompositionsoperation, die für die Verknüpfung der Modalitäten bestimmt sind, zusammengefasst. Dabei wird dem Vorschlag von [Henzen, 2010] einen ODER-Operator einzuführen zugestimmt. Mit diesem Ausdruck ist es möglich die selbe Geste über verschiedene Interaktionstechniken auszuführen. Es erfolgt eine Abbildung alternativer Interaktionstechniken auf die selbe Interaktionsaufgabe. Die gestenbasierte Benutzerschnittstelle ist somit über mehrere Kanäle bedienbar (siehe Ziele multimodaler Benutzerschnittstellen in Abschnitt 4.6). Für die Komposition der Gesten wird weiterhin zwischen sequentieller Abfolge und paralleler Ausführung unterschieden. Die simultane Ausführung berührungsloser und berührender Gesten erfordert beide Hände für die Interaktion. In dieser Hinsicht liegt der Mehrwert in der Vielfalt der abbildbaren Funktionen und einer natürlicheren Interaktion mit den Inhalten.

Für die Identifizierung der Modalität ist angelehnt an BML ein weiterer Bezeichner notwendig. In Anlehnung an die Definition von Namensräumen der Auszeichnungssprache XML kann eine Zuordnung über einen Sprachzusatz in der Syntax-Regel definition realisiert werden. Mit diesem Vorschlag würde der Definition einer Touch-Geste die Bezeichnung TOUCH, gefolgt von dem Zeichen ':' vorangestellt werden. Dieser Ansatz kann auch herangezogen werden, um eine doppelte Vergabe von Namen aufzulösen. Dieser Fall würde vor allem dann auftreten, wenn das Framework eine Gestenbeschreibung aus mehreren Quellen heranzieht. Dies wurde bisher nicht realisiert, stellt aber ein vorstellbares Konzept mit dem Ziel der Austausch- und Erweiterbarkeit dar. Die Namensraumbezeichnung könnte über die Syntax festgelegt werden oder über eine entsprechende Regel frei wählbar sein. Weiterhin kann die Wiederverwendung des Bezeichners einer Geste (name) zur Lesbarkeit und Übersichtlichkeit der Definitionen beitragen. Dies wurde auch in [Henzen, 2010] empfohlen, um Formulierungen zu verkürzen (siehe Kapitel 3.3.1). Die Verknüpfung zweier Aktionsmodalitäten erfordert somit nur noch drei Sprachelemente in einer Definition. Das folgende Beispiel beschreibt die Verknüpfung zweier Gesten wie sie in dem Prototypen der Kartenanwendung realisiert wird (siehe Kapitel 5.2).

TOUCH:TouchGesture	=	HOLD;
POINT:PointGesture	=	1H_FLAT(LINE_AHEAD);
Composite	=	TOUCH:TouchGesture + POINT:PointGesture;

In Hinblick darauf, dass weitere Aktionsmodalitäten wie Kopfgesten oder Gestik und Mimik des Gesichts bis hin zur Sprache mit dem Ansatz der Semiotik beschrieben werden könnten, wird zunehmend Kontextwissen über den Agierenden und dessen Interaktionsverhalten angereichert. Auf dieser Basis ist es möglich die Intention des Nutzers früher und sicherer zu erfassen (siehe Abschnitt 2.3.2 für Interpretationsmöglichkeiten von Körperbewegungen). Weitere Vorteile wurden mit den Zielen multimodaler Benutzerschnittstellen in Kapitel 2.2.3 beschrieben.

5 MULTIMODALER SHOWCASE

Unabhängig von der Formalisierung räumlicher Handgesten und multimodaler Benutzereingaben ist im praktischen Teil der Arbeit eine kombinierte Gestensteuerung auf einem Multitouch-Tisch prototypisch umzusetzen. Immer häufiger kommen Kartendienste zur Anwendung, die verschiedene Informationen in einen geographischen Bezug setzen, wie zum Beispiel Adressen von Dienstleistern, Fotografien zu den Aufnahmeorten auf der Welt, Aufenthaltsorte von Personen oder Nachrichten. Die Navigation in einer Kartenanwendung umfasst im Wesentlichen die folgenden Funktionen:

- **Verschieben** - Der Ausschnitt der Karte wird geändert.
- **Rotation** - Die Karte wird gedreht, womit sich die Himmelsrichtung ändert.
- **Zoom** - Der Kartenausschnitt wird kleiner oder größer gewählt.
- **Perspektive** - Es kann zwischen Draufsicht und perspektivischer Sicht (Immersion) gewählt werden.

Weitere Funktionen wie die Auswahl eines Objektes oder das Setzen einer Route sind anwendungsspezifisch. Sie haben gegenüber der vier Funktionen der Navigation das Ziel Objekte der Karte zu manipulieren. Bekannte Kartendienste sind Google Maps¹, Microsoft Bing Maps², Yahoo Maps³ und Open Street Maps⁴. Google Earth (Google) und Virtual Earth (Microsoft) implementieren jeweils ein erweitertes Kartenmodell in 3D.

Ziel ist es, die Steuerung der Karte mit den Interaktionstechniken Touch und Point zu realisieren. Zum Verschieben, Rotieren und Zoomen der Ansicht werden im Touch-Interface die Gesten DRAG/SLIDE, ROTATE und SPRAD/PINCH verwendet. (siehe Abschnitt 3.2.1) Alternativ soll das Verschieben über die berührungslose Geste GRAB und das Zoomen durch die Höhenveränderung der flachen Hand über dem Tisch möglich sein. Die Einstellung der Perspektive wird mit einer Kombination der Interaktionstechniken umgesetzt. Dazu ist mit der einen Hand eine Touch-Geste (HOLD) und mit der anderen Hand asynchron dazu eine WAVE-Geste auszuführen.

¹URL: <http://maps.google.de/> [Aufruf: 12.01.2011]

²URL: <http://www.bing.com/maps/> [Aufruf: 12.01.2011]

³URL: <http://www.bing.com/maps/> [Aufruf: 12.01.2011]

⁴URL: <http://www.openstreetmap.de/> [Aufruf: 12.01.2011]

5.1 TECHNOLOGISCHER AUFBAU

Für die Erkennung berührungsempfindlicher Eingaben fällt die Wahl der Sensortechnik auf das Microsoft Surface⁵ System (Version 1.0) - ein horizontal angelegtes, optisches Touchscreen, das bereits ein auf dem Betriebssystem Windows Vista basiertes Computersystem integriert. Touch-spezifische Komponenten, wie die Bildverarbeitung zur Auswertung von Kontakten auf der Eingabefläche und grafische Bedienelemente, werden mit dem Surface SDK geliefert, das auf Vista aufbaut. Mit den Abmaßen von 56x53x107 cm (Breite x Höhe x Tiefe) und einer Displaygröße von 30" stellt der Surface ein geeignetes Basissystem dar, das um weitere Sensortechnik erweitert werden kann. Für die Verarbeitung berührungsloser Bewegungen wird mit dem Einsatz von Infrarotkameras eine markerbasierte Handerkennung vorgezogen. Als Gerät kommt die Wiimote zum Einsatz, die eine IR-Kamera integriert und sich als kabelloses und kompaktes Instrument relativ leicht an ein Stativ montieren lässt. Unter einem Kameraöffnungswinkel von 45° und 33,75° können maximal 4 Trackingpunkte erfasst werden. Weiterhin unterscheidet die Wiimote vier Distanzen zur Kamera. (vgl. [Peitz und Podewski, 2009]) Für die Umsetzung einer Zoom-Geste über den Abstand zur Bedienoberfläche sind diese Angaben zu ungenau. Es werden zwei Alternativen für die Distanzberechnung in Betracht gezogen:

Alternative 1: Es sind zwei parallel ausgerichtete Wiimotes über dem Touch-Tisch angebracht. Aus dem Abstand der Positionen, die ein Punkt in der Abbildung auf zwei unterschiedlichen Bildebenen einnimmt, wird die Entfernung zur Kamera ermittelt. In [Cuypers et al., 2009] und [Krejpowicz und Nguyen, 2009] wird mit dieser Methode verfahren, um 3D-Koordinaten zu errechnen. Mit dieser Lösung ist eine Kalibrierung durchzuführen. Das sogenannte Korrespondenzproblem (siehe [Krejpowicz und Nguyen, 2009]) stellt eine weitere Herausforderung in dieser Variante dar. Mit jedem Punkt, der zusätzlich erkannt werden soll, steigt der Aufwand diesen in den zwei Kamerabildern zuzuordnen.

Alternative 2: Es wird eine Kamera über dem Touch-Tisch angebracht. Die relative Höhenänderung der Eingabe ergibt sich aus der Änderung des Abstandes zwischen den erkannten Punkten. Je größer der Abstand der Punkte zueinander, desto näher sind die Infrarotquellen an der Kamera. Diese Methode geht davon aus, dass mehrere Infrarotquellen erfasst werden und dass sie in einem festen Abstand zueinander angeordnet sind. Da im folgenden vier Infrarotquellen zum Einsatz kommen und die relative Höhenänderung für eine Zoom-Geste im Raum ausreichend ist, stellt diese Vorgehensweise eine geeignete Alternative dar.

Die Wiimote wird so über dem Tisch montiert, dass das Kamerabild die gesamte Fläche des Touchscreens abdeckt. Aus den Abmaßen und dem Kamerawinkel lässt sich eine Mindesthöhe von 54 cm über dem Display ableiten. Abbildung 5.2 a) des nächsten Abschnitts zeigt

⁵URL: <http://www.microsoft.com/surface/> [Aufruf: 12.01.2011]

den Aufbau.

Die vier Infrarot-LEDs werden so auf dem Handrücken angebracht, dass mit der Anzahl der erkannten Punkte drei für die Kartenanwendung relevante Gesten abstrahiert werden können. Wird nur ein Punkt erfasst, kann davon ausgegangen werden, dass die Hand zur Faust geballt wurde. Die WAVE-Geste setzt voraus, dass alle Punkte erkannt werden. Zwei Punkte können eine deiktische Geste abstrahieren. Die Zeigehandlung wird in dem Prototypen allerdings nicht umgesetzt. Abbildung 5.1 zeigt die LED-Markierungen an der Hand.



Abbildung 5.1: Vier Infrarot-LEDs als Markierungen für die Handerkennung

5.2 IMPLEMENTIERUNG

Für die Anbindung und Integration eines Kartendienstes wird auf die Bibliothek InfoStrat.VE⁶ zurückgegriffen. Sie bietet eine Bing Maps 3D Karte als WPF-Control und bindet die Touch-Gesten bereits an die Steuerung der Karte. Für die Anbindung der Wiimote kommt die WiimoteLib⁷ in der Version 1.7 zum Einsatz. Über diese Schnittstelle werden die via Bluetooth übertragenen Wiimote-Daten eventbasiert an die Kartenanwendung übermittelt.

Abbildung 5.2 zeigt die grafische Benutzeroberfläche der Anwendung. Die Karte erstreckt sich über den gesamten Darstellungsbereich. Auf linker Seite informieren Kontrolldaten über den Batteriezustand und den Status der Infrarotsensorik der Wiimote, sowie die Registrierung von Gesten. In einer ausgereiften Lösung würden diese Debuginformationen nicht angezeigt werden, so dass allein die Karte wie in Abbildung d) angezeigt werden würde. Abbildung 5.2 b) zeigt, wie berührungslos ein Zoom ausgeführt werden kann. Die Hand wird dabei flach über den Tisch gehalten und eine Bewegung zum Display hin oder von der Oberfläche weg ausgeführt. Über den Abstand der vier erkannten Referenzpunkte an der Hand wird die relative Distanz, sprich die Höhenänderung, festgestellt. In Abbildung 5.2 c) ist dargestellt, wie die Verschiebung der Karte mit einer GRAB-Geste in jede Richtung funktioniert. Der Ausschnitt der Karte verschiebt sich mit der Bewegung. Ein Beispiel für die Komposition der Gesten unterschiedlicher Interaktionstechniken zeigt Abbildung 5.2 d). Mit der linken Hand wird die Touch-Oberfläche berührt. Die rechte Hand führt simultan eine Bewegung in Richtung obere Kante des Displays um in die perspektivische Ansicht zu gehen. Eine Bewegung in Richtung unterer Kante des Displays führt wieder zu einer Draufsicht.

⁶URL: <http://bingmapswpf.codeplex.com/> [Aufruf: 12.01.2011]

⁷URL: <http://wiimotelib.codeplex.com/> [Aufruf: 12.01.2011]

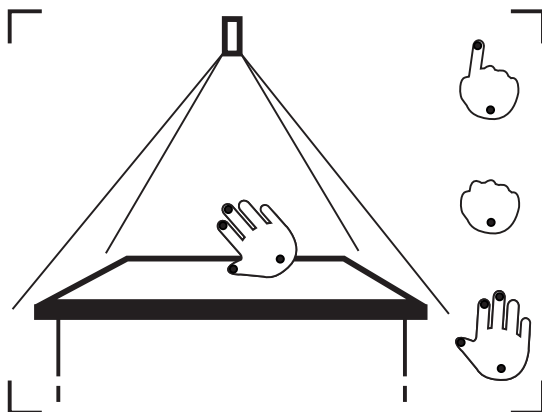


Abbildung a)

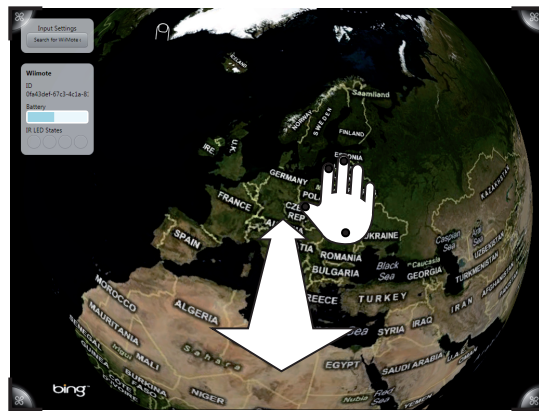


Abbildung b)



Abbildung c)

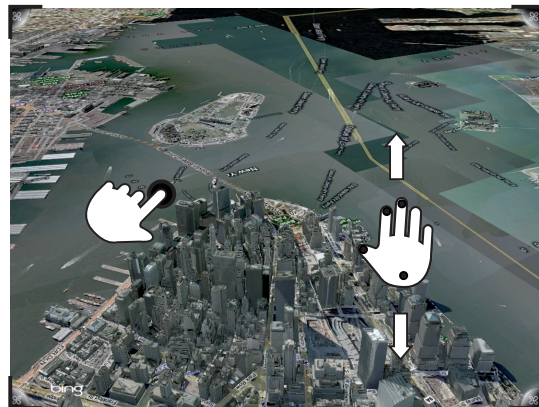


Abbildung d)

Abbildung 5.2: a) Technologischer Aufbau und Gesten, b) Zoomen der Karte mit flacher Hand, c) Verschieben der Karte mit der GRAB-Geste, d) Einstellung der Perspektive durch Berührung des Touchscreens bei simultaner Ausführung einer WAVE-Geste

5.3 HERAUSFORDERUNGEN

Indem die Kamera den Bereich über dem Multitouch-Tisch abfilmt, deckt der bewegungsverarbeitende Bereich den berührungsempfindlichen Eingabebereich mit ab. Touch-Interaktion erfolgt dabei in einer Bildebene, die sich durch einen festen Abstand zur Kamera definiert. Es stellt sich dabei das Problem heraus, dass Aktionen des Nutzer zunächst als potentielle Raum-Geste registriert werden. Konkret bedeutet das, dass der Nutzer mit der Absicht eine Touch-Geste auszuführen, zunächst in den Interaktionsraum für berührungslose Eingaben greift, was zur Folge hat, dass die Bewegung zur Benutzeroberfläche hin bereits als Geste fehlinterpretiert wird. Die Herausforderung der Gestenerkennung besteht darin, die Intention des Nutzers richtig zu erkennen. Ein Lösungsansatz ist, in Analogie zu den Clippingebenen in der 3D-Computergrafik, die das Sichtvolumen einer 3D-Szene begrenzen, den bewegungsverarbeitenden Bereich einzuschränken. Das bedeutet, dass bis zu einer gewissen Höhe über dem Tisch keine Bewegungen verarbeitet werden. Dabei wird davon ausgegangen, dass Bewegungen, die mit Touch-Eingaben einhergehen, im Nahbereich der Bedienoberfläche auftreten. Aus ergonomischer Sicht ist diese Lösung insbesondere für die Komposition der beiden Interaktionstechniken unvorteilhaft, da die Interaktionsbereiche weit auseinanderliegen. Die korrekte Klassifikation und Merkmalsextraktion hinsichtlich unterscheidbarer Eigenschaften von Gesten liegt letztlich bei einem intelligenten Gestenerkennungsalgorithmus. Im Prototypen werden unkontrollierte Bewegungen daran festgemacht, dass ihre Ausführungsgeschwindigkeit gegenüber kontrollierten gestenbasierten Eingaben höher ist.

Das Auftreten unbeabsichtigter Gesten kann als ein generelles Problem der Gestenerkennung gesehen werden. Dabei beschreibt der vorige Abschnitt nur einen Aspekt. Unabhängig vom Interaktionsraum können bedeutungslose Gesten, das heißt Gesten ohne konkrete Nutzerintention, im gesamten vom System sensorisch verarbeiteten Bereich auftreten. Hinsichtlich Touch-Gesten ist hier beispielsweise auch eine ungewollte Verknüpfung von Touch- und Freihandgesten denkbar, die ohne Clipping-Bereich auftreten kann. Um dieses Problemfeld zu umgehen ist bei der Implementierung eines Gestenerkennungssystems vor allem eine intelligente Definition erlaubter Gesten notwendig, so dass das Design der Gesten ungewollte "Zufallsgesten" von vornherein bestmöglich ausschließt. Dies liegt im Verantwortungsreich des Interaktionsdesigners.

Unabhängig von der fehlerhaften Interpretation von Gesten kann vor allem die verwendete Hardware eine Fehlerquelle darstellen. So ist beim Wiimote-Controller etwa ein Flackern der aufgenommenen Punkte zu erkennen, was im einfachsten Fall dazu führt, dass eigentlich vollständig ausgeführte Gesten vom System als unterbrochen und damit abgebrochen erkannt werden. Um dieses Problem zu umgehen wurde im Prototyp eine Zeitspanne (in Form einer Anzahl betrachteter Kamera-Frames) festgelegt, innerhalb welcher ein Flackern einzelner Infrarotquellen nicht als Abbruch einer Geste interpretiert wird, vorausgesetzt, der flackernde Punkt tritt vor Ablauf der definierten Zeitspanne wieder in Erscheinung. Die kon-

krete Dauer dieser Zeitspanne wäre als Maximalwert zu messen, den ein Punkt längstens flackert. Aus Komplexitätsgründen wurde im Prototyp allerdings darauf verzichtet und eine feste Zeitspanne vorgegeben.

6 ZUSAMMENFASSUNG UND AUSBLICK

Ausgangspunkt dieser Arbeit war die Problematik, dass die Interaktion mit berührungsempfindlichen Benutzerschnittstellen eine große Nähe zur Eingabefläche und ihren abgebildeten Informationen erfordert, wodurch die Interaktionswege etwa durch Ausstrecken des Armes oder Änderung der Position des Nutzers länger und somit der Bedienaufwand erhöht sind. Darüber hinaus ist eine berührungsgebundene Eingabe zum Beispiel aus hygienischen Gründen nicht in jedem Fall erstrebenswert. Ziel der Arbeit war die Erarbeitung eines erweiterten Interaktionskonzeptes, das berührungslose Eingaben zulässt. In den Grundlagen wurden dazu die Begriffe Touch- und Point-Interface geprägt und unter dem Blickwinkel multimodaler Benutzerschnittstellen in Beziehung zueinander gesetzt. Der Begriff der Multimodalität wurde somit als Synthese verschiedener Ausdrucksformen der Körpersprache verstanden. In Abschnitt 2.3.2 wurden dazu Interpretationsmöglichkeiten von Körperbewegungen vorgestellt. Hinsichtlich der finger- und handbasierten Eingabe wurden in den Grundlagen technologische Aspekte der Handerkennung als auch Eigenschaften von Gesten erläutert.

Anhand bekannter Anwendungsbeispiele wurden erfolgreich eingesetzte Gesten bestimmt und ihr Gebrauch bezüglich der verwendeten Interaktionstechnik erläutert. In Kapitel 3.4 wurden verwandte Arbeiten vorgestellt, die ein multimodales Interaktionskonzept unter der selben Problemstellung wie diese Arbeit verfolgen und die Interaktionstechniken Touch und Point im Vergleich und in Kombination evaluieren. Das Verhalten der Nutzer zeigte, dass die Kombination der Eingabemethoden ein effizienteres Arbeiten mit digitalen Objekten begünstigt und die Interaktion als angenehmer empfunden wurde. Mit den Studien werden allerdings auch Probleme der berührungslosen Interaktion aufgedeckt wie die geringe Präzision und Treffsicherheit bei der Auswahl von Objekten. Ausgehend von diesen Ergebnissen wurde die Kombination von Touch- und Point-Interface unter verschiedenen Dimensionen und Ausrichtungen der Benutzeroberflächen betrachtet. Während mobile Endgeräte zum Instrument für die Ausführung von Gesten werden, zeigt sich in größeren Displays die Notwendigkeit einer Adaption der Darstellung an die Position und Distanz des Nutzers. Es zeigt sich außerdem eine Dominanz der Touch-Interaktion in horizontalen Eingabegeräten, während vertikal angelegte Benutzeroberflächen vorallem für eine berührungslose Bedienung geeignet sind.

In Hinblick auf eine formale Beschreibung von Gesten wurde die Gestenbeschreibungsspra-

che GeForMT mit ihrer Spezifikation und ihren Zielen als verwandte Arbeit vorgestellt. Mit den in Kapitel 4.2 erarbeiteten Formalisierungsansätzen zeigt sich das Erweiterungspotenzial dieser Sprache hinsichtlich der Verwendung für Gesten im dreidimensionalen Raum. Sie umfassen die Definition von räumlichen Bewegungsrichtungen, Handformen, den Daumen und Handstellungen. Unter Berücksichtigung technologischer und methodischer Aspekte der Gestenerkennung wurden vier aufeinander aufbauende Profile unterschieden, die den Erweiterungsgrad auf die Leistungsfähigkeit des Systems abstimmen. Der Erweiterungsgrad wird mit der optionalen Angabe der spezifizierten Merkmale der Geste bestimmt. Des Weiteren wurde in Kapitel 4.3 ein Lösungsvorschlag erarbeitet, der im Rahmen dieser Arbeit eine Verknüpfung von Modalitäten in Betracht zieht. Die wachsende Komplexität der Syntax ist dabei mit erhöhten Anforderungen an die Beschreibungsfähigkeit abzuwägen. Befragungen oder alternative Methoden der Evaluation können in weiterführenden Arbeiten darüber Aufschluss geben in wie weit die Gestenbeschreibung intuitiv und anwendbar ist. Auf dieser Grundlage kann eine Entscheidung über die Integration der vorgestellten Lösungsansätze und Alternativen erfolgen und weitere Anpassungen vorgenommen werden, um den Zielvorstellungen bestmöglich gerecht zu werden. Der nächste Schritt wäre die Umsetzung der erweiterten Gestenbeschreibung in einem Framework aufbauend auf der Arbeit von [Henzen, 2010].

Ein gestenbasiertes Interaktionskonzept, das alternative Eingabetechniken anbietet, wurde am Beispiel einer digitalen Karte realisiert. Der Prototyp sieht eine abstrahierte, markerbasierte Handgestenerkennung vor. Zukünftige Weiterentwicklungen der Anwendung sollten idealerweise auf einer videobasierten markerlosen Gestenerkennung aufbauen, um die Notwendigkeit einer körpergebundenen Eingabehardware zu umgehen. Dies ist jedoch mit vielen Herausforderungen verbunden, die sich auf Bildverarbeitungsprozesse beziehen. In dem Fall sind Methoden der Bildsegmentierung, Merkmalsextraktion, Objektklassifikation und Bildanalyse zu bewerten. Die Gestaltung der Benutzeroberfläche der Kartenanwendung konzentriert sich auf die wesentlichen Informationen. Das ist zum einen die Karte selbst und ortsgebundene Objekte, die darauf platziert sind. Interaktionsformen wie Menüstrukturen, Schaltflächen und Schieberegler, die die Funktionen der Karte kapseln werden mit dem Einsatz gestenbasierter Interaktionstechniken überflüssig. Mit dem Verschwinden dieses Navigationsbildes zeigt sich aber auch die Relevanz anwendungsspezifische Funktionen und Gesten geeignet zu vermitteln. Im einfachsten Fall sind die Gesten intuitiv anwendbar oder verinnerlicht worden. Für komplexere Gesten, die zunächst erlernt werden müssen, sind geeignete Hilfesysteme zu entwickeln. Des Weiteren sind, insbesondere für die berührungslose Interaktion, Möglichkeiten des Feedbacks zu untersuchen, um den Nutzer darüber zu informieren, dass eine Eingabe im bewegungsverarbeitenden Bereich erfolgt. Im Prototypen beschränkt sich die Rückmeldung auf die Abbildung registrierter Punkte, die jedoch nicht sehr genau ist. Denkbar wäre hier die zukünftige Implementierung einer Kalibrierungsfunktion, sowie die Anwendung von Distanzberechnungsverfahren mit einer höheren Genauigkeit um die absolute räumliche Position abzubilden.

LITERATURVERZEICHNIS

- [Benko und Wilson 2010] Benko, Hrvoje ; Wilson, Andrew D.: Multi-Point Interactions with Immersive Omnidirectional Visualizations in a Dome. In: *Displays*. ITS'10, Saarbrücken, Germany : ACM, Nov. 2010, S. 19–28
- [Blake 2010] Blake, Joshua: *NUIs reuse existing skills (updated NUI definition)*. <http://nui.joshland.org/2010/04/nuis-reuse-existing-skills.html> [Auf-ruf: 03.11.2010]. April 2010
- [Blattner und Dannenberg 1992] Blattner, M. M. ; Dannenberg, R. B.: *Multimedia Interface Design*. ACM Press, 1992
- [Boles 1998] Boles, Dipl.-Inform. D. (Hrsg.): *Begleitbuch zur Vorlesung Multimedia-Systeme*. Oldenburg : Carl von Ossietzky Universität Oldenburg, <http://www-is.informatik.uni-oldenburg.de/~dibo/teaching/mm/buch/main.html>, Dez. 1998
- [Bollhoefer et al. 2009] Bollhoefer, Klaas W. ; Meyer, Kerstin ; Witzsche, Rosina: *Microsoft Surface und das Natural User Interface (NUI)*. Feb. 2009
- [Bolt 1980] Bolt, Richard A.: *“Put-That-There”: Voice and Gesture at the Graphics Interface*. ACM. 1980
- [Chua et al. 2002] Chua, Chin-Seng ; Guan, Haiying ; Ho, Yeong-Khing: *Model-based 3D hand posture estimation from a single 2D image*. <http://haiyingguan.org/HandPostureEstimation.aspx>. 2002
- [Cuypers et al. 2009] Cuypers, Tom ; Eede, Tanja V. den ; Ligot, Stijn ; Francken, Yannick ; Hermans, Chris ; Arickx, Frans ; Bekaert, Philippe: *STEREOWIISION: STEREO VISION WITH WIIMOTES*. 2009
- [Dix et al. 1993] Dix, A. ; Finlay, J. ; Abowd, G. ; Beale, R.: *Human-Computer Interaction*. Prentice Hall, 1993

- [Epps et al. 2006] Epps, Julian ; Lichman, Serge ; Wu, Mike: *A Study of Hand Shape Use in Tabletop Gesture Interaction*. ACM. April 2006
- [Foley et al. 1992] Foley, J. D. ; Dam, A. van ; Feiner, S. K. ; Hughes, J. F.: *Computer Graphics: Principles and Practice, Second Edition*. Addison-Wesley Publishing Company, Inc., 1992
- [Gallardo et al. 2010] Gallardo, Daniel ; Julià, Carles F. ; Jordà, By S.: Interactive Surfaces and Tangibles. In: *Crossroads The ACM Magazine for Students* 16 (2010), Nr. 4, S. 23–30
- [Geng et al. 2001] Geng, Weidong ; Elistratov, Vladimir ; Kolesnik, Marina ; Kulesa, Thomas ; Strauss, Wolfgang: Perceptual user interface for human-computer interaction. In: *The information society landscape* Fraunhofer Institute for Media Communication IMK, MARS-Exploratory Media Lab, St. Augustin, Germany (Veranst.), 2001, S. 69–73
- [George und Blake 2010] George, R. ; Blake, J.: *Objects, Containers, Gestures, and Manipulations: Universal Foundational Metaphors of Natural User Interfaces*. Meyer, A. S. *Gesture Recognition. Website*. Retrieved: 2010-06-23. Available at: <http://wiki.nuigroup.com/>. CHI 2010, Atlanta, Georgia, USA. April 2010
- [Groh 2005] Groh, Rainer: *Das Interaktions-Bild - Theorie und Methodik der Interfacegestaltung*. TUDPress, 2005
- [Han 2010] Han, Jefferson Y.: *Multi-Touch Interaction Research*. <http://cs.nyu.edu/~jhan/ftirtouch/> [Aufruf: 16.07.2010]. Juli 2010
- [Hedicke 2000] Hedicke, Volkmar: Multimodalität in Mensch-Maschine-Schnittstellen. In: *Mensch-Maschine-Systemtechnik*. Symposium Publishing GmbH Düsseldorf, 2000, Kap. 7
- [Henzen 2010] Henzen, Christin: *Prototypische Konzeption eines Frameworks basierend auf einer formalen Beschreibungssprache basierend auf einer formalen Beschreibungssprache*, TU Dresden, Diplomarbeit, 2010. – Diplomarbeit in Arbeit
- [Herczeg 2006] Herczeg, Michael: *Interaktionsdesign - Gestaltung interaktiver und Multimedialer Systeme*. Oldenburg Wissenschaftsverlag GmbH, 2006
- [Janssen 2006] Janssen, Doris: Multimodale Interaktion in interaktiven Räumen. In: *Interaktion in komplexen Informationsräumen - Visualisierung, Multimodalität, Kooperation*. Jürgen Ziegler and Wolfgang Beinhauer, Oldenbourg Verlag, 2006, Kap. 3.5
- [Kadera 2007] Kadera, Stefan: *Informationsdesign – eine kurze Einführung*. <http://blog.seibert-media.net/2007/09/25/informationsdesign-eine-kurze-einfuehrung/> [Aufruf: 03.01.2011]. Sept. 2007

- [Kammer et al. 2010] Kammer, Dietrich ; Taranko, Severin ; Keck, Mandy ; Wojdziak, Jan ; Groh, Rainer: *Towards a formalization of multitouch gestures*, Technische Universität Dresden, Professur Mediengestaltung, Dissertation, 2010
- [Kantel 2001] Kantel, Jörg: *Industrieroboter*. <http://www.kantel.de/robot/folio100.html>. Dec. 2001
- [Karam und Schraefel 2005] Karam, Maria ; Schraefel, M.C.: *A taxonomy of Gestures in Human Computer Interaction*. ACM. 2005
- [König et al. 2009] König, Werner A. ; Rädle, Roman ; Reiterer, Harald: *Squidy: A Zoomable Design Environment for Natural User Interfaces*. April 2009
- [Kopp et al. 2006] Kopp, Stefan ; Krenn, Brigitte ; Marsella, Stacy ; Marshall, Andrew N. ; Pelachaud, Catherine ; Pirker, Hannes ; Thórisson, Kristinn R. ; Vilhjálmsson, Hannes: *Towards a Common Framework for Multimodal Generation: The Behavior Markup Language*. 2006
- [Krejpowicz und Nguyen 2009] Krejpowicz, Marcus ; Nguyen, Song T.: *Implementierung einer Bibliothek zur einfachen Nutzung eines 3D- Tracking-Systems unter Verwendung von zwei Wiimotes*. 2009
- [Langbein 2009] Langbein, Dr.-Ing. R.: Mensch-Computer-Interaktion: Berührungslos Computer steuern. In: *KEM Sonderheft* (2009), June
- [Lao et al. 2009] Lao, Songyang ; Heng, Xiangang ; Zhang, Guohua ; Ling, Yunxiang ; Wang, Peng: A Gestural Interaction Design Model for Multi-touch Displays. In: *People and Computers XXIII – Celebrating people and technology* National University of Defence Technology (Veranst.), British Computer Society, 2009, S. 440–446
- [Lee 2010a] Lee, Johnny C.: *Chung Lee - Projects -Wii*. <http://johnnylee.net/projects/wii/> [Aufruf: 15.07.2010]. Juli 2010
- [Lee 2010b] Lee, Johnny C.: In Search of a Natural Gesture. In: *Crossroads The ACM Magazine for Students* 16 (2010), Nr. 4, S. 11–14
- [Malerczyk 2009] Malerczyk, Cornelius: *Intuitive Interaktion durch videobasierte Gesten-erkennung*. Aug. 2009
- [Meißner 2010] Meißner, Ronny: *ENTWICKLUNG NEUARTIGER BEDIENELEMENTE FÜR BERÜHRUNGSEMPFINDLICHE SYSTEME*. Mai 2010
- [Meyer 2009] Meyer, A. S.: *Gesture Recognition*. http://wiki.nuigroup.com/Gesture_recognition [Aufruf: 09.08.2010]. Nov. 2009

- [Nardi 2008] Nardi, Alessandro D.: *Grafiti - Gesture Recognition mAnagement Framework for Interactive Tabletop Interfaces*, University of Pisa, Dissertation, Dec. 2008
- [Norman und Nielsen 2010] Norman, Donald A. ; Nielsen, Jakob: *Gestural Interfaces: A step backwards in Usability*. http://jnd.org/dn.mss/gestural_interfaces_a_step_backwards_in_usability_6.html [Aufruf:14.12.2010]. 2010
- [P. Chojeci 2008] P. Chojeci, Heinrich-Hertz-Institut: *Jenseits von Multi-Touch-Eingabegeräten - Die berührungslose "multipointing" Interaktion*. CeBit 2008. 2008
- [Parker et al. 2006] Parker, J. K. ; Mandryk, R. L. ; Inkpen, K.M.: *Integrating Point and Touch for Interaction with Digital Tabletop Displays*. IEEE Computer Graphics and Applications. 2006
- [Pavlovic et al. 1997] Pavlovic, V. ; Sharma, R. ; Huang, T.: Visual interpretation of hand gestures for human-computer interaction: A review. In: *Pattern Analysis and Machine Intelligence*, 1997, S. 677 –695
- [Peitz und Podewski 2009] Peitz, Alexander ; Podewski, Nils: *Untersuchung von Möglichkeiten der Mensch-Maschine-Interaktion unter Verwendung von 3D-Tracking mit der Nintendo™ Wii-Remote*. 07 2009
- [Rötting und Seifert 2005] Rötting, Matthias ; Seifert, Katharina: Multimodale Mensch-Maschine Interaktion. In: *Beiträge zur Mensch-Maschine-Systemtechnik aus Forschung und Praxis* Bd. 1. Symposium Publishing GmbH Düsseldorf, 2005, Kap. Themenkreis: Multimodalität, S. 283–300
- [Saffer 2009] Saffer, Dan: *Designing Gestural Interfaces*. O'Reilly Media, Inc., 2009
- [Schacht 2010] Schacht, Marie: *Natürliche Interaktion mit Tangible User Interfaces*. 2010
- [Schick et al. 2009] Schick, Alexander ; Camp, Florian van de ; Ijsselmuiden, Joris ; Stiefelhagen, Rainer: *Extending Touch: Towards Interaction with Large-Scale Surfaces*. ITS. Nov. 2009
- [Wandke] Wandke, Hartmut: *Multimodalität*. http://www3.psychologie.hu-berlin.de/ingpsy/Teaching/Seminar/Video-Seminar/Inhalt_der_Kassetten/Multimodalit_C3_A4t.htm [Aufruf: 11.01.2010]
- [Wang und Popovic 2009] Wang, Robert Y. ; Popovic, Jovan: *Real-Time Hand-Tracking with a Color Glove*. 2009
- [Wickey und Alem 2007] Wickey, Aiden ; Alem, Leila: *Analysis of Hand Gestures in Remote Collaboration: Some Design Recommendations*. Nov. 2007

[Wikipedia 2010] Wikipedia: *Interaction Design*. http://de.wikipedia.org/wiki/Interaction_Design [Aufruf: 03.01.2011]. Nov. 2010

[Wobbrock et al. 2009] Wobbrock, Jacob O. ; Morris, Meredith R. ; Wilson, Andrew D.: *User-Defined Gestures for Surface Computing*. April 2009

ABBILDUNGSVERZEICHNIS

2.1	Terminologische Zusammenhänge am Beispiel "Auswahl eines Textfeldes"	6
2.2	Modell einer Benutzerschnittstelle [Groh, 2005]	9
2.3	Technische Komponenten eines multimodalen Systems nach HEDICKE	12
2.4	Ziele der multimodalen Mensch-Maschine-Interaktion	13
2.5	Erkennung und Interpretation von Körperbewegungen	16
2.6	Freiheitsgrade eines dreidimensionalen Objektes im Raum	17
2.7	21 Freiheitsgrade der Hand	17
2.8	Farbliche Markierungen an der Hand [Chua et al., 2002]	19
2.9	Handschuh mit farbigem Muster [Wang und Popovic, 2009]	19
2.10	LED- und Reflektoren [Peitz und Podewski, 2009]	19
2.11	iPoint Presenter	20
2.12	Forschungsprojekt des FIT	20
3.1	Multimodale gestenbasierte Anwendungen	24
3.2	"Put-that-there"	25
3.3	mUltimo3D	25
3.4	Standardgesten im Touch-Interface	26
3.5	Standardgesten im Point-Interface	27
3.6	Beispiele atomarer Gesten in GeForMT	31
3.7	Touch- und Point-Interaktion in großflächigen Benutzeroberflächen [Schick et al., 2009]	33
3.8	Touch- und Point-Interaktion mit dem TractorBeam	34
3.9	Aufbau des Experiments zur Evaluierung der Touch- und Point-Interaktionstechnik	35
3.10	Bevorzugte Interaktionstechniken in Abhängigkeit von der Entfernung und Größe der Zielbereiche	35
4.1	Anordnung von Ein- und Ausgabebereich	37
4.2	Interaktionsraum der berührungsempfindlichen und berührungslosen Gestensteuerung	39
4.3	Raumebenen und Bewegungsrichtungen	46

5.1	Vier Infrarot-LEDs als Markierungen für die Handerkennung	55
5.2	Technologischer Aufbau und Interaktion mit der Kartenanwendung	56

TABELLENVERZEICHNIS

2.1	Zusammenfassung der Arten von Gesten	22
4.1	Unterscheidung von Handformen	43
4.2	Unterscheidung von Handstellungen	45
4.3	Unterscheidung von Handbewegungen	47
4.4	Zusammenfassung der Formalisierungsansätze	48
4.5	Profile der berührungslosen gestenbasierten Interaktion	49
4.6	Kompositionsoperatoren für die modalitätsbezogene Verknüpfung	50