

Masterarbeit

Sebastian Gregor

Seamless Interaction - Entwicklung von Tangible
Interaction im Context von Smart Homes

Sebastian Gregor

Seamless Interaction - Entwicklung von Tangible
Interaction im Context von Smart Homes

Masterarbeit eingereicht im Rahmen der Masterprüfung
im Studiengang Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. rer. nat. Gunter Klemke
Zweitgutachter : Prof. Dr. rer. nat. Kai von Luck

Abgegeben am 2. Februar 2011

„From the isolation of our workstations we try to interact with our surrounding environment, but the two worlds have little in common. How can we escape from the computer screen and bring these worlds together?“ [Wellner u. a. \(1993\)](#)

Sebastian Gregor

Thema der Arbeit

Seamless Interaction - Entwicklung von Tangible Interaction im Context von Smart Homes.

Stichworte

Nahtlose Interaktion, Mensch - Computer Interaktion, Anfassbare Benutzerschnittstellen, Anfassbare Interaktionsformen, Intelligente Wohnumgebungen, Umgebungszintelligenz

Kurzzusammenfassung

Tangible Interaction ist ein Forschungsgebiet, das sich mit physischer Interaktion mit Computern beschäftigt. Dabei werden sowohl physische Benutzerschnittstellen und die Manifestation von Daten als auch die Einbettung von Benutzerschnittstellen in die tägliche Umgebung und den Context der Benutzer betrachtet. Neben weiteren ist diese Form der Mensch-Computer-Interaktion ein Ansatz für die Interaktion von Benutzern mit intelligenten Umgebungen. Diese Arbeit beschäftigt sich mit der Entwicklung von Tangible Interaction Artefakten im Kontext von intelligenten Wohnumgebungen. Anhand zweier Tangible Interaction Experimente werden die Herausforderungen und der Ablauf der Entwicklung von Tangible Interaction Artefakten dargestellt. Aufbauend darauf wird ein Designprozess für Tangible Interaction Artefakte entwickelt und die dafür notwendigen Techniken und Herangehensweisen genauer betrachtet.

Sebastian Gregor

Title of the paper

Seamless Interaction - Developing Tangible Interaction in context of smart homes

Keywords

Seamless Interaction, Human-Computer Interaction, Tangible User Interfaces, Tangible Interaction, Smart Homes, Ambient Intelligence

Abstract

Tangible Interaction is a research area that deals with physical human-computer interaction. This includes both physical user interfaces and the manifestation of data, as well as the integration of user interfaces in the daily environment. Among others this form of Human-Computer interaction is an approach for users to interact with intelligent environments. This thesis deals with the development of Tangible Interaction artifacts in the context of smart living environments. The challenges and the process of development of Tangible Interaction artifacts are shown by describing the implementation of two experiments. Based on this results the thesis describes a design process for tangible interaction artifacts and will analyse the necessary techniques and approaches.

Inhaltsverzeichnis

Abbildungsverzeichnis	8
1 Einleitung	10
1.1 Motivation	10
1.2 Thema der Arbeit	11
1.3 Aufbau	11
2 Tangible Interaction	13
2.1 Forschungsumfeld	13
2.1.1 Ubiquitous Computing und Pervasive Computing	13
2.1.2 Augmented Reality	14
2.1.3 Ambient Intelligence	14
2.2 Der Begriff „tangible“	15
2.3 Verwandte Gebiete	16
2.3.1 Graspable User Interfaces	16
2.3.2 Embodied User Interfaces	16
2.3.3 Tangible Computing	17
2.3.4 Tangible User Interfaces	18
2.4 Tangible Interaction	21
2.4.1 Mehrwert	24
2.4.2 Einschränkungen	26
2.5 Zusammenfassung	27
3 Tangible Interaction Elemente	29
3.1 Smart Homes	29
3.1.1 Living Place Hamburg	30
3.2 Experiment: Hamburg Cubical	31
3.2.1 Konzept	32
3.2.2 Proof of Concept	33
3.2.3 Prototyp	33
3.2.4 1. Verfeinerung des Prototypen	41
3.2.5 2. Verfeinerung des Prototypen	45
3.2.6 Zusammenfassung	47

3.3	Experiment: Hamburg SITab	48
3.3.1	Interaktionskonzept	50
3.3.2	Verwandte Arbeiten	54
3.3.3	Umsetzung	56
3.3.4	Zusammenfassung	64
3.4	Fazit	65
4	Designmethodik	68
4.1	Designmethoden	68
4.1.1	Interaction Design	68
4.1.2	User-Centered Design	69
4.1.3	Ethnographie und Contextual Design	70
4.1.4	Analytisches Design	72
4.2	Design von Tangible Interaction	73
4.2.1	Anforderungsanalyse	75
4.2.2	Prototyping	77
4.2.3	Konzeptionelles Design	79
4.2.4	Physisches Design	84
4.2.5	Evaluation	86
4.3	Zusammenfassung	87
5	Fazit und Ausblick	89
	Literaturverzeichnis	92

Abbildungsverzeichnis

2.1	TUI-Konzept (Quelle: Ishii (2008b) - S. 473)	18
2.2	MCRpd - Modell (Quelle: Ullmer und Ishii (2000))	19
2.3	TUI - Aktorenmodell (Quelle: Ishii (2008a) - S. XXI)	20
2.4	Tangible Interaction Framework (Quelle: Hornecker (2008))	22
3.1	Schematischer Aufbau Living Place Hamburg (Quelle: Living Place Hamburg Team)	31
3.2	Interaktionskonzept	32
3.3	Arduino Mega (Quelle: Arduino)	34
3.4	Arduino - Entwicklungsumgebung	35
3.5	Blockschaltbild erster Prototyp	36
3.6	Arduino Duemilanove (Quelle: Arduino)	36
3.7	MMA7260Q (Quelle: Sparkfun)	37
3.8	LISY300AL (Quelle: Sparkfun)	37
3.9	XBee (Quelle: Sparkfun)	38
3.10	Erste Prototyp mit geöffneten Gehäuse	38
3.11	Flussdiagramm Treibersoftware	39
3.12	Cubical JSON Format	40
3.13	Versuchsaufbau	41
3.14	Gehäuse-Prototypen	41
3.15	Blockdiagramm Hardware des Prototypen	42
3.16	Sensoren (Quelle: Sparkfun)	43
3.17	PCB-Layout (Oberseite)	44
3.18	PCB-Layout (Unterseite)	44
3.19	Platine (Unterseite)	45
3.20	Platine (Oberseite)	45
3.21	Platine (Oberseite)	46
3.22	Platine (Unterseite)	46
3.23	3D Modell des Gehäuses (rechts) und gedrucktes Gehäuse mit Platine (links)	47
3.24	Interaktionskonzept	50
3.25	Siftable Aufbau (Quelle: Merrill u. a. (2007))	54
3.26	Beispielanwendung (Quelle: Merrill u. a. (2007))	54

3.27	Microsoft Prototyp (Quelle: Hinckley (2003b))	55
3.28	Blockschaltbild	57
3.29	Arduino Pro Mini (Quelle: Arduino)	57
3.30	USB zu Seriell Wandler (Quelle: Sparkfun)	57
3.31	IRDA Transceiver	58
3.32	erster Prototyp	59
3.33	Treibersoftware	60
3.34	SITab JSON Format	60
3.35	SITab IR Datenformat	61
3.36	Platine (Oberseite)	62
3.37	Platine (Unterseite)	62
3.38	Blende für Infrarot-Transceiver	62
3.39	Prototypischer Rahmen	63
3.40	Flussdiagramm Testanwendung	63
3.41	Anwendungstest „Skalieren von Bildern“	64
4.1	Tangible Interaction Design	74
4.2	TI - Design: Anforderungsanalyse	75
4.3	TI - Design: Prototyping	77
4.4	TI - Design: konzeptionelles Design	79
4.5	TI - Design: physisches Design	84
4.6	physisches Design	84
4.7	TI - Design	86

1 Einleitung

1.1 Motivation

Seit Anfang der Neunziger Jahre prägen Trends wie Ubiquitous Computing, Pervasive Computing und Augmented Reality¹ die Entwicklung und Forschung in der Informatik. Nicht mehr der Umgang mit einem Computer sondern die Informationen, Dienste und Kommunikationsmöglichkeiten die durch einen Computer bereitgestellt werden stehen im Fokus der Benutzer. Ein gutes Beispiel hierfür sind die modernen Tablet Computer wie z. B. das iPad². Diese Art der Computer hat nichts mehr mit dem Computer, wie er zum Ende des zwanzigsten Jahrhunderts gesehen wurde, gemein. Die Interaktion findet nicht mehr über die Maus und Tastatur statt. Der Bildschirm steht nicht mehr separat vom Computer getrennt auf einem Tisch, sondern er ist mit dem Computer in einem Gehäuse verbaut. Dieses Gehäuse ist mit einer Diagonalen von zehn Zoll und wenigen Zentimetern Höhe für die mobile Verwendung konzipiert. Durch diesen Aufbau verschwindet der Computer sinnbildlich hinter dem Bildschirm und der Benutzer hat das Gefühl nur noch ein Anzeigelement in den Händen zu haben. Dies nimmt der Benutzer jedoch nicht mehr als solches, sondern je nach Situation als z. B. Zeitung, interaktives Nachrichtenportal, mobiles Kommunikationsportal oder interaktiven Bilderrahmen wahr. Die Interaktion findet dabei über das Berühren oder Bewegen des Gerät statt. Dadurch hat der Benutzer das Gefühl, direkt mit den Informationen und Inhalten und nicht mehr mit dem Computer an sich zu arbeiten.

Dieses beschriebene „Verschwinden“ der Computer aus der Wahrnehmung der Benutzer hat sich in der Autoindustrie bereits vollzogen. In modernen Fahrzeugen sind eine Vielzahl unterschiedlicher, in das Fahrzeug eingebetteter Computer verbaut, die den Benutzer bei der Bedienung des Fahrzeugs unterstützen. Verwendet z. B. ein Benutzer bei der Fahrt das Bremspedal, so erwartet er das sich die Geschwindigkeit des Fahrzeugs verringert. Dabei wird jedoch nicht mehr, wie zu Beginn der Fahrzeugentwicklung, die Kraft mit der ein Benutzer das Bremspedal betätigt über Seilzüge auf die Bremsen übertragen. Vielmehr wird die Größe der Auslenkung des Bremspedals durch Sensoren erfasst und so der Wunsch des Fahrers zur Verringerung der Geschwindigkeit an ein Netzwerk von Computern im Fahrzeug übertragen. Diese Computer ermitteln über Sensoren die aktuelle Geschwindigkeit der

¹Zur genaueren Erläuterung dieser Begriffe siehe Kapitel 2.1.

²Das iPad ist ein Tablet PC der Firma [Apple](#)

einzelnen Räder, den Zustand der Umgebung, sowie den Fahrhahununtergrund. Mit diesen Werten passen diese Computer während des gesamten Bremsvorgangs die Bremskraft, die über die Bremsen auf die einzelnen Räder wirkt, so an, dass der Bremsvorgang optimal abläuft ohne das eines der Räder blockiert. Für den Fahrer des Fahrzeugs bleibt dieser Vorgang jedoch vollkommen transparent.

Solche, sich dem Benutzerkontext adaptierende System können auch in anderen Umgebungen zum Einsatz kommen. Während der Einsatz in Fahrzeugen bereits heute Stand der Technik ist, stehen die Entwicklungen für den Einsatz in Wohnumgebung, sog. Smart Homes³, erst am Anfang. Dies in der Informatik als „Ambient Intelligence“⁴ bezeichnete Forschungsgebiet beschäftigt sich mit zahlreiche Herausforderungen. Eine davon ist die Frage, wie der Benutzer in das System eingreift, wenn sie nicht so verhält wie er es sich vorstellt. Sicherlich sind eine Universalfernbedienung, die Verwendung von Spracheingaben oder berührungsempfindlichen Bildschirmen als Eingabeelemente denkbare Lösungsansätze für dieses Problem. Eine weiterer Lösungsmöglichkeit ist die Verwendung von physischen Elementen der Wohnumgebung als Eingriffsmöglichkeit für den Benutzer.

1.2 Thema der Arbeit

Die in der Motivation beschriebenen Form der Interaktion mit Computern, über berühr- und greifbaren Benutzerschnitteln oder durch physische Manipulation des Computers selbst, wird als Tangible Interaction bezeichnet und ist Gegenstand dieser Arbeit ist. Der Forschungsbe- reich Tangible Interaction beschäftigt sich mit der Entwicklung, Konzeption und Verwendbar- keit von berühr- und greifbaren Benutzerschnittstellen und der physischen Interaktion mit rechnerbasierten Artefakten. Der Fokus dieser Arbeit liegt auf der Entwicklung solcher Ele- mente, insbesondere im Kontext von intelligenten Wohnumgebungen. Dabei steht jedoch nicht die Entwicklung eines einzelnen Artefaktes oder Systems, sondern die Erstellung einer Entwicklungsmethodik zum Design von Tangible Interaction Elementen im Vordergrund.

1.3 Aufbau

Diese Arbeit ist in vier Abschnitte gegliedert. Im ersten Abschnitt, der das zweite Kapitel umfasst, wird der Begriff Tangible Interaction definiert. Dazu wird eine Übersicht über das Forschungsumfeld und verwandte Themenbereiche gegeben. Im Anschluss wird der Begriff „Tangible Interaction“ beschrieben und es werden Vorteile, aber auch Einschränkungen bei

³siehe Kapitel 3.1

⁴siehe Kapitel 2.1

der Verwendung von Tangible Interaction erläutert. In darauf folgendem Kapitel werden zwei Experimente vorgestellt, die in der Arbeitsgruppe Seamless Interaction entstanden sind. Im ersten Teil dieses Kapitels werden Smart Homes definiert und das Living Place Hamburg als Ausprägung eines Smart Home Labores vorgestellt. Danach werden die in dem Kontext Living Place Hamburg entstandenen Projekte „Hamburg Cubical“ und „SITab“ beschrieben. Den Abschluss des Kapitels bildet ein Fazit über die vom Autor in den beiden Projekten gemachten Erfahrungen und aufgetretenen Herausforderungen. Der dritte Abschnitt befasst sich damit, eine Methodik für die Entwicklung von Tangible Interaction Elementen zu definieren. Zu diesem Zweck werden im ersten Teil unterschiedliche Entwicklungsmethodiken betrachtet. Danach wird aus den in Abschnitt zwei gemachten Erfahrungen und den betrachteten Entwicklungsmethodiken heraus eine Vorgehensweise für die Entwicklung von TI definiert sowie die einzelnen Arbeitsschritte und benötigten Werkzeuge genauer betrachtet. Im letzten Abschnitt dieser Arbeit wird ein abschließendes Fazit gezogen, die gewonnenen Erkenntnisse resümiert und ein Ausblick auf die weiteren möglichen Schritte gegeben.

2 Tangible Interaction

Tangible Interaction oder Tangible User Interfaces gewinnen im Bereich Human-Computer Interaction immer mehr an Bedeutung. Im folgenden Kapitel wird ein Überblick über die bisherige und aktuelle Entwicklungen in diesem Gebiet aufgezeigt. Dazu wird zunächst eine Übersicht über das Forschungsumfeld, aus dem heraus sich Tangible Interaction entwickelt hat, gegeben und der Begriff „Tangible“ definiert. Nach einer Übersicht über verwandte Forschungsrichtung und Teilbereiche wird der Begriff „Tangible Interaction“ erläutert. Den Abschluss des Kapitels bildet eine Übersicht über die Vorteile und Einschränkungen bei der Verwendung von Tangible Interaction.

2.1 Forschungsumfeld

Tangible Interaction oder Tangible User Interfaces haben ihren Ausgangspunkt in der Notwendigkeit, andere Bedienkonzepte als das in den 90er Jahren dominierende GUI - Konzept zu entwickeln. Die Grundlage dieser Notwendigkeit waren Entwicklungen und Trends in der Informatik, die Versuchten die Virtualität der Computerwelt mit der realen Welt der Benutzer zu verbinden oder zumindest in Teilen zu kombinieren. Um einen Einblick in diese Entwicklung zu geben, werden im folgenden Abschnitt die Forschungsrichtungen beschrieben, die als Ausgangspunkt für die Entwicklung von Tangible Interaction gesehen werden können.

2.1.1 Ubiquitous Computing und Pervasive Computing

Der Begriff Ubiquitous Computing wurde von Mark Weiser ([Weiser \(1991\)](#)) geprägt. Weiser postuliert, dass Computer allgegenwärtig (engl. ubiquitous) sein werden. Computer sind in diesem Zusammenhang ein breites Spektrum an Geräten, die vom klassischen Mainframe und PC, über Chipkarten und eingebettete Systeme bis hin zu intelligenten Heizungssteuerungen reichen. Die Computer werden dabei nicht mehr als solche wahrgenommen, sondern ihre Funktionen rücken in den Vordergrund. Weiser verdeutlicht dieses Prinzip am Beispiel der Schrift. Die Schrift wird als solche nicht mehr wahrgenommen, sondern man nimmt nur noch die Informationen wahr, die mit der Schrift vermittelt wird.

Pervasive Computing wird oft als Synonym für Ubiquitous Computing verwendet. Im Unterschied zu Ubiquitous Computing, welches „eher in akademisch-idealistischer Weise als eine unaufdringliche, humanzentrierte Technik“ (Mattern (2001)) verstanden wird, ist der „Begriff 'Pervasive Computing' mit einer leicht unterschiedlichen Akzentuierung geprägt“ (Mattern (2001)). Das primäre Ziel von Pervasive Computing ist, in kurzfristigen Zeiträumen allgegenwärtige Informationsverarbeitung in wirtschaftlichen Szenarien und Geschäftsprozessen nutzbar zu machen.

2.1.2 Augmented Reality

Augmented Reality (AR) ist ein Forschungsbereich, der aus dem Gebiet Virtual Reality (VR) entstanden ist. In VR begibt sich der Benutzer in eine komplett synthetische und immersive Umgebung (s. a. Milgram und Kishino (1994)). In dieser Umgebung nimmt der Benutzer die Realität und Welt um ihn herum nicht mehr wahr. Anstatt die reale Welt komplett zu ersetzen, beschäftigt sich AR damit, die reale Welt durch digitale Inhalte zu vermehren (engl. augmented) oder zu annotieren (vgl. Azuma (1997)). Bei der Wahrnehmung der realen Welt und den Zusatzinformationen besteht eine Echtzeitinteraktion. Die Zusatzinformationen können dabei sowohl als 2D, als auch als 3D Annotationen visualisiert werden.

2.1.3 Ambient Intelligence

Ambient Intelligence (Aml), kann mit Umgebungsintelligenz übersetzt werden. Es erforscht und beschreibt Konzepte, mit dem sich dem Benutzer anpassbare und Benutzerwünsche antizipierende Umgebungen entwickeln lassen. Dies soll mit Hilfe von allgegenwärtigem Computing (s. a. Kapitel 2.1.1), allgegenwärtiger, vernetzter Kommunikation und intelligenten Benutzerschnittstellen verwirklicht werden (Hollatz (2010)).

Das „Umgebungs-“ aus Umgebungsintelligenz deutet auf die Verteilung, Allgegenwärtigkeit und Transparenz von Aml Systemen hin. Verteilung bezieht sich auf die Dezentralität des Systems, Allgegenwärtigkeit bedeutet, dass die Dienste und elektronischen Bauteile überall präsent sind und Transparenz spiegelt die Unsichtbarkeit und Unaufdringlichkeit solcher Systeme wieder. Mit dem Begriff Intelligenz soll aufgezeigt werden, dass die Umgebung die Benutzer in ihrem Umfeld und ihre Verhaltensweisen erkennt und sich ihnen anpasst. Aarts (2004) beschreibt fünf Kerneigenschaften von Ambient Intelligence Systemen als:

- **Embedded**
Die unterschiedlichen Geräte sind eingebettet in die Umgebung.
- **Context aware**
Das System erkennt den Benutzer und die Situation, in der sich der Benutzer befindet.

- **Personalisiert**
Das System kann sich den Benutzerwünschen und -anforderungen anpassen.
- **Adaptiv**
Es passt sich dem Benutzer(-verhalten) an.
- **Antizipativ**
Das System antizipiert die Wünsche des Benutzers ohne bewussten Eingriff durch den Benutzer.

2.2 Der Begriff „tangible“

Tangible ist ein englisches Wort und lässt sich mit berührbar, greifbar, handfest oder dinghaft übersetzen. Bei den Begriffen „Tangible Interaction“ oder „Tangible User Interfaces“ wird die Greifbarkeit und physische Manifestation einer Benutzerschnittstelle und die physische Interaktion betont. Beim Greifen oder Berühren von Gegenständen spielt der Tastsinn eine bedeutende Rolle. Dieser menschliche Sinn ist im Gegensatz zu allen anderen Sinnen der einzige aktive (Hornecker, 2004). Während ein gesunder Mensch mit dem gesamten Körper Gegenstände ertasten kann, ist es ihm nur mit der Hand möglich, Dinge zu greifen. Dabei lassen sich z. B. Form, Struktur, Temperatur und Beschaffenheit der Oberfläche ertasten.

Das Greifen von Gegenständen ist eine motorische Aktivität, die die interne kognitive Verarbeitung eines Menschen einfacher, schneller und verlässlicher machen kann (Fitzmaurice, 1996). Fitzmaurice erläutert ein Experiment, in dem Testpersonen beim Tetris¹ spielen beobachtet wurden. Das Experiment zeigte, dass die Personen eher den Rotationsknopf benutzen, als erst nachzudenken und die Rotation mentale Modell des Körpers gedanklich zu vollziehen. Diese Vorgehensweise erlaubte den Spielern:

- frühzeitig neue Informationen zu gewinnen,
- die Belastung beim mentalen Rotieren zu verringern,
- Erinnerung an die Form aus ihrem Gedächtnis zu extrahieren,
- den Typ des Objektes einfacher zu identifizieren,
- die Passform des neuen Bauteils zu erkennen.

¹Tetris ist ein Computerspiel, bei dem einzelne, vom oberen Rand eines rechteckigen Spielfelds herunterfallende, aus Quadraten zusammengesetzte Formen von einem Spieler in sich am unteren Rand bildende, horizontale Reihen eingepasst werden müssen. Dazu können sie jeweils um 90 Grad im Uhrzeigersinn gedreht werden.

Diese Vorteile erlaubten es den Benutzern bis zu sieben Mal schneller Bauteile in die richtige Passform zu dirigieren. Kirsh (1995) beschreibt dieses Vorgehen als „complementary strategy“. Durch diese Methode hat ein Mensch das Potential mit physischen Artefakten sehr schnell zu interagieren und durch diese seine Umgebung zu beeinflussen und zu manipulieren.

2.3 Verwandte Gebiete

Es gibt eine Vielzahl von Entwicklungen und Forschungsrichtungen die sich mit Teilbereichen von Tangible Interaction beschäftigen. Im folgenden Abschnitt werden diese Gebiete beschrieben.

2.3.1 Graspable User Interfaces

Der Begriff Graspable User Interfaces (Graspable UI) wurde in Fitzmaurice u. a. (1995) geprägt und in Fitzmaurice (1996) genauer definiert. Graspable UIs bieten dem Benutzer den Zugriff auf vielfältige, spezialisierte Eingabegeräte (sog. Handles), welche als dedizierte physische Eingabegeräte dienen. Diese Eingabegeräte sind an digitale Funktionen (Fitzmaurice spricht hier auch von „Graspable Funktionen“) oder grafische Repräsentationen gebunden. Durch die greifbaren Handles sind diese Funktionen physisch manipulierbar und ermöglichen eine räumliche Anordnung. Jede Funktion ist dabei festen Ein- und Ausgabewerten zugeordnet, die jeweils einen eigenen Platz im physischen Raum einnehmen. Dadurch wird eine voneinander unabhängige, simultane und dauerhafte Auswahl von Objekten möglich, was eine beidhändige oder eine gleichzeitige Bedienung durch mehrere Benutzer erlaubt. Neben dieser „space-multiplexed“ Ein- und Ausgabefunktionalität beschreibt Fitzmaurice die simultane Benutzung und Sichtbarkeit von Objekten, die Verwendung von spezialisierten Werkzeugen in eindeutiger Form, die Möglichkeit der räumlichen Anordnung (wie z. B. Position, Orientierung und Nähe) der Geräte untereinander und die räumliche Anordnung der Arbeitsgegenstände im Arbeitsbereich als Hauptmerkmale von Graspable User Interfaces.

2.3.2 Embodied User Interfaces

„By treating the body of the device as part of the user interface - an embodied user interface - we can go beyond the simulated manipulation of a GUI and allow the user to really directly manipulate an integrated physical-virtual device.“ Fishkin u. a. (2000)

Wie man aus dem einleitenden Zitat erkennen kann, charakterisieren Embodied User Interfaces eine Art von Schnittstellen, bei denen alle Aspekte der Interaktion über im Gerät eingebettete Sensoren und Aktoren erfolgen. Im Gegensatz zu Tangible User Interfaces (vgl. Kapitel 2.3.4), bei denen der „tangible“ - Teil als Interface zu einem Computer dient, sind bei Embodied User Interfaces die Computer ein Teil des Interfaces. Dadurch werden die physischen Objekte und die virtuelle Repräsentation zu einer Einheit vereint. Dies impliziert, dass die Computer selbst physisch greifbar, tangible sind. In [Fishkin u. a. \(2000\)](#) werden diese Art der Computer charakterisiert als:

- portabel und greifbar,
- entwickelt, um eine limitierte Anzahl von Aufgaben zu erfüllen,
- enthalten die benötigten Arbeitsmittel und verkörpern so die Aufgabe, für die sie entwickelt wurden,
- die physische Form so gewählt, dass diese Aufgaben vom Benutzer leicht und natürlich erledigt werden können,
- metaphorisch verwandt mit ähnlichen nicht-Computer - Artefakten.

Zu dieser Klasse von Computern gehören z. B. PDAs, Tablets oder moderne Mobilfunktelefone, die heute bereits über eine Vielzahl von Sensoren verfügen. Die Annotation von weiteren Sensoren oder die Verwendung der bereits integrierten Sensoren ermöglicht eine intuitive Interaktion mit dem Gerät und eine leichte Bedienung der Funktionen (vgl. [Harrison u. a. \(1998\)](#)).

2.3.3 Tangible Computing

In [Dourish \(2001\)](#) prägt Dourish den Begriff Tangible Computing. Unter diesem Begriff fasst er die bisherigen Forschungsergebnisse aus dem Bereich des Ubiquitous Computing, der Virtual Reality und der Augmented Reality, sowie der Tangible User Interfaces (s. Kapitel 2.3.4) zusammen. Die Kernfrage von Physical Computing ist nach Dourish das Verhältnis zwischen Handlung und Raum, in dem diese ausgeführt wird. Dabei spielt die Ausnutzung von materiellen und physischen Einschränkungen (vgl. Affordances), sowie die Konfigurierbarkeit (im Sinne einer Individualisierbarkeit) der Arbeitsumgebung von Benutzern eine besondere Rolle. Dourish beschreibt drei Hauptmerkmale, die die Interaktion mit Tangible Computing Elementen beschreiben:

- **kein singulärer Ort der Kontrolle oder Interaktion**

Es gibt keinen singulären Punkt und kein singuläres Gerät (ähnlich einer Maus bei GUIs) für die Interaktion. Vielmehr kann dieselbe Aktion auf mehrere Geräte verteilt oder durch das Zusammenspiel mehrerer Geräte durchgeführt werden.

- **keine sequentielle Interaktion**

Tangible Computing erzwingt keine sequentielle oder modale Interaktion. Es wird keine Reihenfolge von bestimmten Interaktionen vorgegeben und es können auch parallele Interaktionen durchgeführt werden.

- **Ausnutzung physikalischer Eigenschaften**

Durch die gezielte Ausnutzung von physikalischen Eigenschaften beim Entwurf soll dem Benutzer eine geeignete Verwendungsweise aufgezeigt werden. Der Benutzer soll durch die Interaktion geleitet werden, in dem die aktuelle physische Konfiguration der Geräte auf den nächsten Schritt hinweist.

2.3.4 Tangible User Interfaces

Der Begriff Tangible User Interface (TUI) wurde 1997 durch Hiroshi Ishii vom MIT Media Lab geprägt. Ishii beschreibt eine zweigespaltene Interaktion zwischen dem Computer und dem Benutzer (Ishii und Ullmer (1997)). In der bis dato gebräuchlichen GUI-Metapher werden die Augen und Ohren des Menschen repräsentiert durch Monitore und Lautsprecher zu „Windows to digital Worlds“, während der Körper und damit auch der Sinn des Fühlens in der Realität bei Maus und Tastatur bleibt. Ullmer und Ishii schlagen vor, digitale Informationen durch physische Objekte zu repräsentieren (Ullmer und Ishii (2000)). Dadurch können digitale Informationen durch die Interaktion mit der Repräsentation manipuliert werden, wodurch es zu einer Aufhebung der Trennung von (digitalem) Datenraum und (physischem) Körperraum kommt.

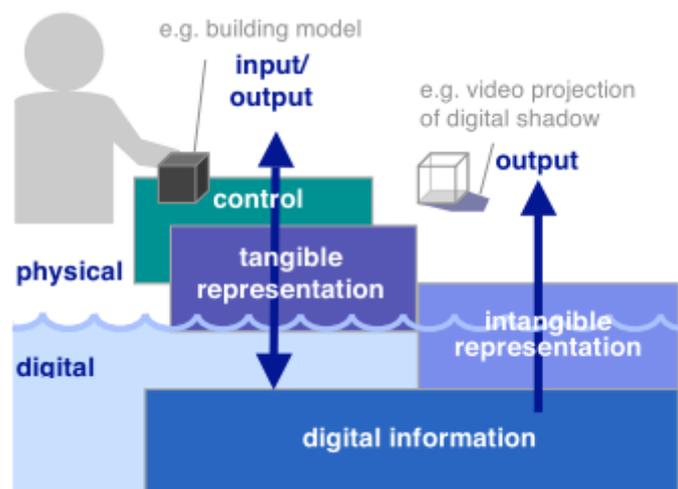


Abbildung 2.1: TUI-Konzept (Quelle: Ishii (2008b) - S. 473)

In Abbildung 2.1 wird das TUI-Konzept von Ishii dargestellt. Es besteht aus einer Abwandlung des bekannten Model View Controller (MVC)² - Konzeptes. Das View aus MVC wird in einen tangible - Teil und einen intangible - Teil unterteilt. Der tangible - Teil bezieht sich auf den physischen Raum durch eine Abbildung in Objekten. Der intangible - Teil steht für die digitale Abbildung innerhalb der Software. Die enge Kopplung der beiden Teile bilden das eigentliche Control.

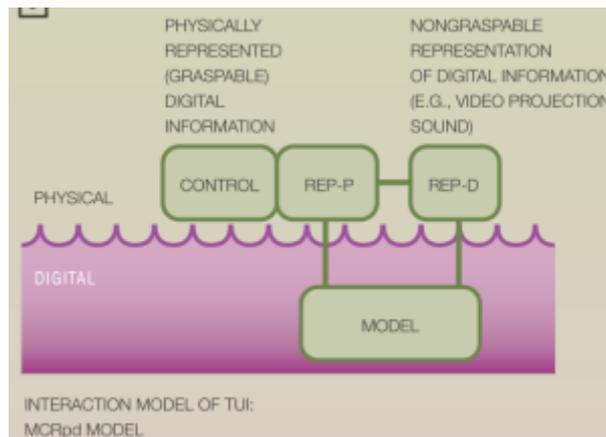


Abbildung 2.2: MCRpd - Modell (Quelle: [Ullmer und Ishii \(2000\)](#))

Ullmer und Ishii präzisieren das „model-control-representation (physical and digital)“ (MCRpd) - Modell (siehe Abb. 2.2) als Interaktionsmodell für TUIs ([Ullmer und Ishii \(2000\)](#)). Die Autoren nennen drei Hauptmerkmale für TUIs:

- die Kopplung der physischen Repräsentation mit dem Modell,
- die Verkörperung von Mechanismen zur interaktiven Kontrolle,
- die wahrgenommene Kopplung von physischer Repräsentation und digitaler Repräsentation,

Die Kopplung zwischen physischer Repräsentation und digitalem Modell ist eine Haupteigenschaft dieses Interaktionsmodells. So können durch Manipulation der Repräsentation auch die Daten des Model manipuliert werden. Die physische Repräsentation funktioniert dabei als eine interaktive, physische Kontrolleinheit. Die Bewegung der physischen Objekte und ihre Anordnung oder Einführung in andere Objekte dient als Hauptkontrollmittel. Das dritte Merkmal bezieht sich auf die Balance zwischen physischer und digitaler Repräsentation. Durch diese Balance der digitalen Repräsentationen wird es möglich, mittels des Computersystems dynamische Informationen darzustellen.

²MVC → Beschreibung hier!

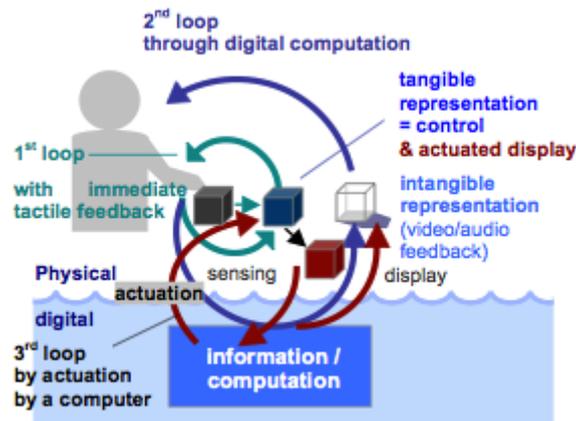


Abbildung 2.3: TUI - Aktorenmodell (Quelle: Ishii (2008a) - S. XXI)

In Ishii (2008a) beschreibt Ishii die Double Interaction Loop, die Persistenz, die Zusammenlegung von Ein- und Ausgaberaum, die Space-Multiplexed Eingabe und die Spezialisierung als Haupteigenschaften von TUIs.

Double Interaction Loop

Ishii beschreibt zwei Interaktionsschleifen. In der ersten Schleife, die er als „passiv haptic feedback loop“ bezeichnet, erhält der Benutzer durch die haptische Rückmeldung über seinen Tastsinn eine sofortige Rückmeldung, dass er ein Objekt bewegt hat. Dabei entsteht weder ein durch den Computer verschuldeter Verzug noch muss der Benutzer auf die zweite Interaktionsschleife warten. Die zweite Schleife, der sog. „digital feedback loop“, beinhaltet das Erfassen der Objektbewegung durch Sensoren, das Verarbeiten im Computer und die Rückmeldung über das Ergebnis der Manipulation mittels visuelles oder auditives Feedback. In Abbildung 2.3 beschreibt Ishii eine dritte Interaktionsschleife, die durch das Hinzufügen von computergesteuerten Aktoren ermöglicht wird. Dadurch wird dem Computer eine Feedback - Möglichkeit über z. B. eine Veränderung von internen Modellen oder Daten gegeben.

Persistenz

Durch ihren physischen Aufbau sind TUIs im physischen Raum persistent. Persistenz bedeutet in diesem Zusammenhang die Eigenschaft, dass ein physisches Artefakt, wenn es nicht von außen beeinflusst wird, seinen Zustand behält. Eine Änderung der Eigenschaften und des Zustandes eines Objektes erfolgt nur, wenn es von einer externen Kraft beeinflusst und auf diese Weise z. B. die Lage oder Orientierung des Objektes verändert wird. Aufgrund einer engen Kopplung der physischen Repräsentationen und der darunterliegenden Daten verkörpern TUIs den digitalen Zustand des Systems.

Zusammenlegung von Ein- und Ausgaberaum

Um eine nahtlose Verknüpfung der physischen und digitalen Welt zu erreichen, wird eine möglichst enge Zusammenführung des Ein- und Ausgaberaums angestrebt. Dies gehört zu den Schlüsselkonzepten von TUIs. GUIs benutzen die Maus und die Tastatur als Eingabe- und den Bildschirm als Hauptausgabegerät für die digitalen Daten. Diese räumliche Unterbrechung wird durch das TUI-Konzept (vgl. Abbildung 2.1) aufgehoben.

Space-Multiplexed Eingabe

Jedes Element einer TUI beansprucht seinen eigenen physischen Raum. Dadurch, dass jedes Element als dedizierter Kontroller fungiert, ist eine beidhändige und gleichzeitige Eingabe möglich.

Spezialisierung

TUIs werden oft gezielt für eine spezielle Aufgabe entworfen. Im Gegensatz zu allgemeinen Werkzeugen, bei denen für die unterschiedlichen Aufgaben oft Kompromisse gemacht werden müssen, um andere nicht zu behindern oder gar unmöglich zu machen, sind TUIs auf die Aufgaben in einer bestimmten Domäne ausgelegt und auf diese zugeschnitten.

2.4 Tangible Interaction

Tangible User Interface ist ein fester Leitbegriff auf einer Vielzahl von Tagungen, Forschungsbeiträgen und Veröffentlichungen. In der von dem HCI-Umfeld geprägten Definition versteht Ishii unter dem Begriff TUI eine Schnittstelle zu einem Computer, die die Interaktion zurück in die reale Welt holt (Ishii und Ullmer (1997)). Hornecker kritisiert, dass der Begriff Interface oder Schnittstelle an sich eine eingeschränkte Sichtweise widerspiegelt, da er aus einer Ingenieurstradition stammt und zumeist mit Software sowie technischen Ein- und Ausgabegeräten umgesetzt wird (Hornecker (2008)). Die Verwendung des Begriffs impliziert eine Trennung zwischen dem Interface und der darunterliegenden Funktionalität, wodurch die Gestaltungsfragen als reine Oberflächengestaltung erscheinen. Entwicklungen in anderen Design- und Forschungsbereichen sowie der interaktiven Kunst erweitern jedoch die Perspektive und beschäftigen sich mit Dingen, die nicht in die Definition von TUIs fallen, weil sie nicht als „Computing“ verstanden werden. Die unterschiedlichen Sichtweisen unterteilt Hornecker und Buur (2006) in:

- **Data-centered view**

Diese datenzentrierte Sichtweise stammt vor allem aus dem Forschungsbereich Human-Computer Interaction und ihr prominentester Vertreter ist das TUI-Konzept (vgl. Kapitel 2.3.4). Der Fokus liegt hier besonders auf den unterschiedlichen Möglichkeiten der Verknüpfung und Repräsentation von Daten mit physischen Objekten.

- **Expressive-Movement-centered view**

Dieser Ansatz stammt vor allem aus dem Produktdesign und geht über die eigentliche Formgebung und das Erscheinungsbild von Produkten hinaus. Vielmehr beschäftigt sich dieser Ansatz mit der körperlichen Interaktion mit Produkten. Dabei stehen die möglichen Bewegungen, die körperlichen Fähigkeiten und die Ausdrucksweise von Bewegungen bei der Interaktion mit auf bestimmten Domains zugeschnittenen Produkten im Vordergrund.

- **Space-centered view**

Interaktive Flächen nehmen bei Architekten und in der interaktiven Kunst eine immer größere Bedeutung ein. Unter diesen interaktiven Flächen versteht man physische Artefakte oder Räume, die auf Bewegungen von Besuchern reagieren. Dabei wird die Bewegung des gesamten Körpers oder sogar der Artefakte selbst als interaktives Eingabegerät verwendet.

Tangible Interaction versucht die beschriebenen Ansätze zu vereinigen ([Hornecker \(2005\)](#)) und fokussiert die Dynamik der Interaktion sowie die körperlich-motorische Interaktion ([Buur u. a. \(2004\)](#) und [Larsen u. a. \(2007\)](#)), anstatt der Entwicklung einer Schnittstelle (zu einer bereits vorhandenen Maschine) und der Übermittlung von Informationen. Fernaeus spricht in diesem Zusammenhang über einen Wechsel von einer „Daten-zentrierten“ hin zu einer „Aktions-zentrierten“ Perspektive und sieht Tangibles eher als eine Ressource für menschliche Aktionen, Kontrolle, Kreativität und soziale Aktionen ([Fernaeus u. a. \(2008\)](#)).

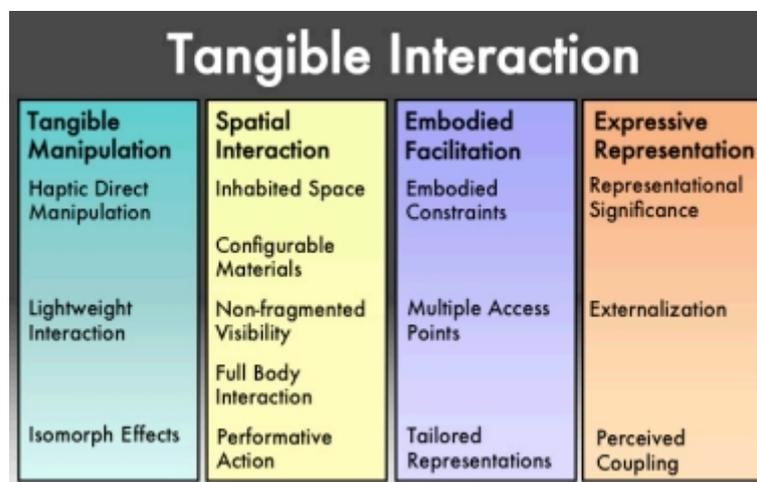


Abbildung 2.4: Tangible Interaction Framework (Quelle: [Hornecker \(2008\)](#))

In [Hornecker \(2008\)](#) stellen die Autoren ein Framework für Tangible Interaction vor. Dieses fokussiert das Erleben der Interaktion durch den Benutzer und die Verflechtung von materiell physischen und sozialen Aspekten bei der Interaktion. Dieses Framework ist um vier Leitmotive aufgebaut (s. Abb. 2.4):

Tangible Manipulation

Tangible Manipulation bezieht sich auf die physische Manipulation von Artefakten, die mit digitalen Daten und rechnerbetonten Ressourcen verknüpft sind. Tangible Manipulation teilt sich in drei Teilbereiche auf. *Haptic Direct Manipulation* beschreibt die direkte Manipulation von Objekten, die gleichzeitig Interface, Interaktionsobjekt und Interaktionengerät sind. *Lightweight Interaction* bezieht sich auf das konstante Feedback, das der Benutzer erhält und ihm ermöglicht, in kleinen Schritten vorzugehen und seine Ideen schnell auszudrücken und zu testen. Mit *Isomorph Effects* ist die Direktheit der Manipulation mit dem Artefakt und den entsprechenden Objekten und Effekten gemeint.

Spatial Interaction

Tangibles befinden sich im physischen Raum, nehmen einen physischen Platz ein und die Benutzer müssen sie im physischen Raum bewegen und manipulieren. Dadurch gehen Benutzer und Artefakte, als auch die Artefakte untereinander und zu ihrer Umgebung eine räumliche Beziehung ein, die das Verständnis vom Raum beeinflusst und diesen „bevölkern“. Im Framework wird dies als *Inhabited Space* umschrieben. Während sich *Configurable Materials* auf die (Neu-)Anordnung von signifikanten Artefakten bezieht, bedeutet *Non-fragmented Visibility* das Erfassen der gesamten Szenerie bei Tangible Interaction. So kann man gleichzeitig jemandem bei der Manipulation oder beim Gestikulieren folgen und dabei selbst interagieren und die andere Personen nimmt dies wahr. Das physische Interagieren mit Artefakten wird mit dem ganzen Körper durchgeführt und hier als *Full-Body Interaction* bezeichnet. Das Konzept der *Performative Action* deutet auf den kommunikativen Effekt bei der Durchführung einer Aktion hin. So findet eine Kommunikation durch eine Bewegung statt, während man diese ausführt.

Embodied Facilitation

Die Konzepte von Embodied Facilitation beinhalten *Embodied Constraints*, *Multiple Access Points* und *Tailored Representation*. Hinter *Embodied Constraints* verbirgt sich der physische Aufbau und die Konfiguration des Systems, mit dem Aktivitäten unterstützt oder limitiert werden können. Verteilte Kontrolle über mehrere Zugriffspunkte ist ein weiteres Konzept und wird als *Multiple Access Points* bezeichnet. Hierbei wird allen Benutzern die Möglichkeit gegeben, nachzuvollziehen was passiert und die Kontrolle über für sie interessante Objekte und Dinge zu erlangen. Unter *Tailored Representation* verstehen die Autoren, dass Tangibles auf die Fähigkeiten und Erfahrungen von bestimmte Benutzergruppen abgestimmt sind. Dadurch sollen neuen Benutzern als auch Experten geeignete Interaktionsmöglichkeiten und Funktionen angeboten werden.

Expressive Representation

Ein ausdrucksloses oder zeitlich nur kurz relevantes Tangible Interface wird von den

Benutzern als „nicht-tangible“ wahrgenommen. Hornecker bezeichnet dies als *Representational significance*. Zusammen mit *Externalization* und *Perceived Coupling* gehört es zu den zentralen Konzepten dieses Themas (Hornecker (2008)). *Externalization* bezeichnet die Möglichkeit, Tangibles als Denkhilfe benutzen zu können, in dem Benutzer mit ihnen kommunizieren. Auf diese Weise sollen sie den Denkprozess unterstützen. Die Prozess- und Bearbeitungsschritte sollen hierbei jederzeit nachvollziehbar sein. Das dritte Konzept *Perceived Coupling* steht für die klare und für den Benutzer jederzeit ersichtliche Kopplung von physischer und digitaler Repräsentation. Ziel ist es, die funktionellen Abläufe für den Benutzer jederzeit nachvollziehbar zu machen.

2.4.1 Mehrwert

Verminderung der kognitiven Last

Klemmer u. a. (2006) beschreiben die tiefe Verbindung zwischen motorischen Aktionen und dem menschlichen Denkprozess. So können Interaktionsstile wie Tangible Interaction, die den Benutzer wenig in ihren Bewegungen einschränken, diesen beim Denken und Kommunizieren unterstützen. Bei der Verwendung von Gesten als Eingaben wird die Verwendung des motorischen Gedächtnisses eines Benutzers unterstützt. Durch das motorische oder kinetische Gedächtnis können Menschen die Position und Bewegungen ihrer Muskeln erkennen, speichern und wieder abrufen. Während GUIs für eine Vielzahl von unterschiedlichen Funktionen immer die gleichen motorischen Aktionen erfordern, erlauben es TI Artefakte, Funktionen an ganz bestimmte Aktionen zu knüpfen und dadurch die kognitive Last bei der Interaktion zu vermindern. Shaer und Hornecker (2009) zeigen, dass TI Artefakte epistemische³ Aktionen unterstützen. Desweiteren helfen diese Aktionen, sich an bisherige Wege zu erinnern und das Gedächtnis zu unterstützen.

Situiert in der physischen Welt

Eine hervorstechende Eigenschaft von Tangibles ist, dass sie in der gleichen Welt wie der Benutzer verhaftet sind und wie Dourish schreibt, dass Sie die gleiche Welt bevölkern wie ihre Benutzer (vgl. Dourish (2001)). D.h. auch, dass sich die Bedeutung und Funktion von Tangibles je nach Kontext verändern kann. Außerdem macht die physische Greifbarkeit eines Tangible das Ausrichten, Platzieren und Anpassen an die Bedürfnisse des Benutzers möglich.

³Unter epistemische Aktionen versteht man Aktionen bei der ein Objekt manipuliert wird um dessen Funktionsweise und Einsatzmöglichkeiten zu verstehen (vgl. auch Kapitel 2.2).

Parallele Eingabe und Zusammenarbeit

Bei grafischen Benutzerschnittstellen können Funktionen nur zeitlich aufeinanderfolgend verändert werden. Im Gegensatz dazu funktioniert jede tangible Repräsentation als ein dedizierter Controller. Somit kann ein Benutzer beidhändig Manipulationen vornehmen und darüberhinaus können mehrere Benutzer, wenn mehrere Artefakte als Eingabemöglichkeiten zur Verfügung stehen, gleichzeitig Manipulationen vornehmen (s. a. [Ishii \(2008a\)](#)). Dadurch, dass Tangibles für jeden Zuschauer und Benutzer sichtbar sind, unterstützen sie die Koordination der Gruppe beim Ausführen von Aufgaben und Tätigkeiten und die Zusammenarbeit innerhalb einer Gruppe (vgl. [Shaer und Hornecker \(2009\)](#)). Nicht zuletzt können Tangible Artefakte von einer Person zu einer anderen weiter gereicht werden, wodurch Aufgaben und Funktionen geteilt oder übergeben werden können.

Direktheit der Interaktion

Ishii beschreibt, dass Tangibles ein persistentes Mapping vom Interface zu den darunterliegenden Daten ermöglicht ([Ishii \(2008a\)](#)). Dadurch werden die Daten durch das Tangible instanziiert und in physischer Form repräsentiert. Die Tangibles sind in diesem Fall Repräsentation und Eingabegerät zugleich, was ein direkteres Manipulieren als z. B. bei GUIs, mit der eine Eingabe über eine Maus und Monitor erfolgt, ermöglicht.

Verwendung von Affordances und Beschränkungen

Die Möglichkeit, physische Affordances (siehe Kapitel [4.2.4](#)) und Beschränkungen nutzen zu können, ist einer der großen Vorteile bei Tangible Interaction. So können Formen, Farben, Material und Gewicht dem Benutzer zur Ausführung von Handlungen verleiten. [Ferscha u. a. \(2007\)](#) untersuchen den Einfluss von Affordances bei der Entwicklung einer als TUI ausgeprägten Fernbedienung. Die Autoren kommen zu dem Schluss, dass die Form und Größe einen signifikanten Einfluss auf Verwendbarkeit einer TUI haben. Gleichzeitig kann die Ausnutzung von physischen Beschränkungen dazu genutzt werden, dass ein Tangible Artefact in einer korrekten Weise verwendet wird. So kann die Form eines Objektes den Benutzer „zwingen“, es nur in einer bestimmten Lage zu verwenden oder es nur mit bestimmten anderen Objekten zu kombinieren.

Spezialisierung

Fitzmaurice erläutert in [Fitzmaurice \(1996\)](#), dass es sich bei Tangibles meist um spezialisierte, genau auf eine Aufgabe zugeschnittene, Artefakte handelt. Diese Spezialisierung hat

den Vorteil, dass die Affordance (siehe Kapitel 4.2.4) und die physischen Eigenschaften so gewählt werden können, dass sie für die Ausführung der Aufgaben und Funktionen des Tangibles gezielt ausgenutzt werden können. Couture u. a. (2008) zeigen, dass spezialisierte Geräte bessere Resultate hervorbringen können als allgemein gültige Geräte.

2.4.2 Einschränkungen

Skalierbarkeit

Eine Einschränkung bei Tangible Interaction kann die Skalierbarkeit der TI Elemente sein. Dies zeigt sich in dem Fakt, dass Tangible Artefakte einen physischen Raum einnehmen. In diesem hat kein weiteres Artefakt mehr Platz. Da der physische Raum an einem Arbeitsplatz oder einer Installation begrenzt ist, können nicht beliebig viele Artefakte platziert und verwendet werden. Desweiteren kann es passieren, dass Tangibles zur Seite geräumt werden müssen, um ein anderes Tangible Artefakt zu nutzen. Eine weitere Einschränkung in der Skalierbarkeit betrifft die Komplexität der Tangibles in Form, Umfang und in ihrer Funktion. In Form und Umfang sind Tangibles vor allem durch ihre Größe und ihr Gewicht stark eingeschränkt. Dies hängt von den physischen Fähigkeiten der Benutzer sowie der Form der Artefakte ab (s. a. Ferscha u. a. (2007)).

Flexibilität

Bei grafischen Benutzerschnittstellen können Objekte leicht erzeugt, dupliziert, verändert oder gelöscht werden. Auch die Form und Größe kann durch das Herein- oder Herauszoomen verändert werden. Diese Möglichkeiten fehlen bei TI Artefakten aufgrund der festen physischen Beschaffenheit von tangiblen Objekten. Dadurch ist es einem System nur sehr beschränkt möglich, die physikalischen Eigenschaften von Tangibles zu verändern, um zum Beispiel andere Funktionen sichtbar zu machen oder den Zustand von Daten und deren Änderung zu zeigen.

Vielseitigkeit

Spezialisierung kann sowohl als Vorteil oder Nachteil vom Tangible Artefakt gesehen werden, da sie zur Einschränkung der Vielseitigkeit führt. Tangibles verlieren viele ihrer Vorteile, wie z. B. das gezielte Ausnutzen von physischen Affordances und Beschränkungen, wenn Nutzer sie als ein vielseitig verwendbares Artefakt zur Steuerung oder Manipulation von Daten verwenden möchten. Auch die oben beschriebene fehlende Flexibilität spricht gegen die Verwendung von Tangible Artefakts als allgemein verwendbare Eingabegeräte.

Ermüdung

Nach [Kirk u. a. \(2009\)](#) ist das Greifen, Anheben und Bewegen von Objekten bei der Tangible Interaction die primäre Interaktionsform. Dafür muss ein Benutzer unterschiedliche und im Umfang mehr und größere Bewegungen durchführen, als es z. B. mit einer Maus notwendig ist. Aus diesem Grund ist es laut [Shaer und Hornecker \(2009\)](#) wichtig, bei der Entwicklung auch die Ergonomie und Langzeitbelastung bei der Verwendung von Tangibles zu beachten, da ein Benutzer sonst bei deren Benutzung schnell ermüden kann. Aber nicht nur der Benutzer, auch das Material eines Tangible kann „ermüden“ und so z. B. instabil werden.

Verlust

[Kirk u. a. \(2009\)](#) beschreiben die Gefahr eines Verlustes oder Verlegens von TI Artefakten. Durch einen Verlust kann die Funktionsweise eines Systems eingeschränkt werden oder ganz verloren gehen. Diese Einschränkung wird durch die oben beschriebenen beschränkte Möglichkeit, Tangibles ohne weiteres in einem kurzen Zeitraum neu zu erzeugen oder zu kopieren noch verstärkt.

2.5 Zusammenfassung

Tangible Interaktion bietet die Möglichkeit und die Notwendigkeit einer im Vergleich zur Bedienung einer grafischen Benutzerschnittstelle vollkommen neuen Interaktionsform zwischen Benutzer und Computer. Diese Notwendigkeit besteht zum großen Teil durch die Weiterentwicklung von Computern und durch Trends wie Ubiquitous Computing, Augmented Reality oder Ambient Intelligence. Der Begriff „tangible“ lässt sich mit berührbar, greifbar, handfest oder dinghaft übersetzen und verdeutlicht die physische Natur von TI Artefakten. Beim Greifen, Berühren und Manipulieren von Tangible Interaction Objekten wird der Tastsinn des Menschen angesprochen. Das Hantieren mit Gegenständen ist tief verwurzelt in unsere Denkweise und mindert die kognitive Last beim Ausführen von Aufgaben.

Bereits 1995 hat Fitzmaurice mit seinen Graspable User Interfaces die ersten Untersuchungen zu greifbaren Benutzerschnittstellen veröffentlicht. Auf diese Arbeit baut das TUI - Konzept von Ishii auf. Ishii erweitert das für GUIs übliche MVC-Paradigma zu seinem MCRpd - Interaktionsmodell. Das Hauptmerkmal dieses Konzeptes sind physische Repräsentationen für digitale Daten, über die diese manipuliert und kontrolliert werden können.

Während Graspable UI und TUIs sich hauptsächlich mit Interfaces zu (entfernten) Computern und mit der Manipulation und Kontrolle von Daten auf diesen beschäftigen, erfolgt bei Embodied User Interfaces die Interaktion mit Computern, wie z. B. Tablet PCs, PDAs oder

moderne Mobilfunktelefonen, durch eingebettete oder angebrachte Sensoren des Computers. Dies ist auch ein Teilaspekt von Tangible Computing. Dieser von Dourish geprägte Begriff bezeichnet ein Forschungsgebiet, das sich mit der Benutzung von in unserer alltäglichen, physischen Umgebung eingebettete Computer beschäftigt. Hauptaugenmerk liegt dabei auf der Interaktion und dem Verhältnis zwischen Handlung und Raum, in dem diese ausgeführt wird.

Tangible Interaction versucht, die hier vorgestellten Ansätze und Strömungen aus anderen Design- und Forschungsbereichen zu integrieren. Das Hauptaugenmerk liegt dabei sowohl auf der Interaktionsform, der Dynamik der Interaktion als auch auf der körperlich - motorischen Interaktion mit computerisierten Artefakten. Dabei werden soziale, technische und ästhetische Aspekte betrachtet. Der Einsatz dieser Form der Interaktion bietet einen Mehrwert, kann aber auch zu Einschränkungen bei der Benutzung führen. Die Verminderung der kognitiven Last, die Situiertheit in der physischen Welt, die Direktheit der Interaktion, die Verwendbarkeit von Affordances und physikalischen Einschränkungen und die Möglichkeit, dass ein System über Artefakte zeitgleich von mehreren Benutzern oder von einem Benutzer beidhändig manipuliert werden kann, zählen zu den Mehrwerten von TI. Auch die Spezialisierung von TI Artefakten kann als Mehrwert aber auch, neben der Skalierbarkeit, der Flexibilität, der Gefahr des Verlustes von Artefakten und der Ermüdung des Benutzers bei der Verwendung, als Einschränkung bei der Verwendbarkeit von TI als Interaktionsform gesehen werden.

3 Tangible Interaction Elemente

In diesem Kapitel werden zwei Experimente im Bereich Tangible Interaction beschrieben. Beide sind im Rahmen des Living Place Hamburg (siehe Kapitel [3.1.1](#)) entstanden und dienen als Beispiel für unterschiedliche Arten von TI Artefakten.

Bei der Betrachtung des ersten Experimentes wird die Entwicklung des Hamburg Cubical (s.a. Kapitel [3.2](#)) aufgezeigt. Dabei handelt es sich um einen mit unterschiedlichen Sensoren bestückten Würfel, der als Eingabe- und Interaktionsmedium für Smart Homes konzipiert wurde. Während dieses Experiment die Entwicklung eines praktischen Eingabemediums zeigt, werden im zweiten Experiment mögliche Dienstleistungen und Funktionen aufgezeigt, die durch die Erweiterung von bereits bestehenden Geräten möglich werden. Zu diesem Zweck wurde mit dem SITab (s. Kapitel [3.3](#)) ein Rahmen entwickelt, durch den Tablet PCs, Smartphones oder ähnliche Geräte mit Sensoren erweitert werden können. Dies ermöglicht diesen Geräten ihre unmittelbare Nachbargeräte zu erkennen, mit diesen Daten auszutauschen und so dem Benutzer neue Funktionalitäten zur Verfügung zu stellen.

Die aus diesen beiden beispielhaften Experimenten erkennbaren Herausforderungen bei der Entwicklung von TI Artefakten werden am Ende des Kapitels (s. Kapitel [3.4](#)) zusammengefasst.

3.1 Smart Homes

Unter Smart Homes versteht man eine „intelligente“ Wohnumgebung, die den Benutzer durch in die Umgebung eingebetteten, unaufdringlichen, computergestützten Geräten bei der Ausführung von Aufgaben unterstützt. Dabei kommt Technologie zum Einsatz, die auf natürliche Art und Weise mit den Bewohnern interagiert bis hin zu dem Punkt, an dem sie implizit wird ([Meyer und Rakotonirainy, 2003](#)). Damit solche Ambient Intelligence Systeme (s.a. Kapitel [2.1.3](#)) diese Funktion ausüben können, müssen die Gefühle und Intentionen des Benutzers erfasst werden, der eigenen und der Zustand der mit ihnen kooperierenden Systeme bekannt und das Verhalten autonom an den Kontext der Benutzer angepasst werden.

Solche unterstützenden Systeme wurden vor allem im Bereich des Computer Supported Cooperative Work (CSCW) untersucht (Kirstgen, 2010). Die Ergebnisse lassen sich nur bedingt in den Bereich der Smart Homes übertragen. Hindus (1999) beschreibt drei Hauptgründe:

- **Homes are not workplaces**

Wohnungen sind primär nicht dazu konzipiert, Technologien zu beherbergen. Sie werden zum großen Teil nicht professionell geplant, installiert oder gewartet. Im Gegensatz zu Unternehmen gibt es keine professionellen Administratoren, die Probleme mit der Technik lösen oder technologische Konzepte erstellen und umsetzen. Desweiteren findet man in Wohnungen im Gegensatz zu Arbeitsplätzen, was die Expertise angeht, eine heterogene Benutzergruppe.

- **Consumers are not knowledge workers**

Die Motivationen, aus denen heraus Entscheidungen getroffen werden, unterscheiden sich bei Wohn- und Arbeitsplätzen drastisch. Während bei Arbeitsplätzen Entscheidungen vor allem aus Überlegungen zur Produktivitätssteigerung heraus getroffen werden, spielen bei Wohnbereichen eher ästhetische, modische oder Image - Gründe eine Rolle.

- **Families are not organizations**

Während Firmen eine feste Hierarchie und klare Entscheidungsstrukturen haben, funktioniert die Entscheidungs- und Wertebildung im Heimbereich anders. Hier werden Entscheidungen ohne Einschränkung getroffen und können je nach Bewohner variieren. Dies betrifft beispielsweise die Einteilung von Raum und Zeit, sowie Ort und Art und Weise der Aktivitäten.

Die oben beschriebenen Gründe zeigen, welche speziellen Anforderungen an Geräte und Bedienelemente gestellt werden, die für ein Smart Home konzipiert werden. Neben der Benutz- und Verwendbarkeit, sind vor allem die soziale Akzeptanz, die Wahrung der Privatsphäre, sowie geringe Kosten und kein administrativer Aufwand wichtige Anforderungen bei der Entwicklung. Dabei ist es wichtig, neben diesen Punkten auch auf die für die Benutzererfahrung (engl. User Experience) wesentlichen Aspekte wie Schönheit, Spaß oder Wohlbefinden zu achten. Daraus lässt sich erkennen, dass es bedeutsam ist, die Forschung und Entwicklung in einem Kontext durchzuführen, der dem der Endbenutzer entspricht. Aus diesen Gründen sind Forschung und Entwicklung in speziell dafür vorgesehenen Forschungslaboren unerlässlich.

3.1.1 Living Place Hamburg

Das Living Place (LP) Hamburg ist ein Labor für Untersuchungen im Bereich Smart Home und Ambient Intelligence (Rahimi und Vogt, 2011). Es ist als eine 140 Quadratmeter Ein-

zimmerwohnung mit Koch-, Schlaf-, Wohn-, Arbeits- und Essbereichen und separatem Bad konzipiert (vgl. Abbildung 3.1). Das seit 2009 entwickelte Konzept, an dem Entwickler aus unterschiedlichen Forschungsbereichen in interdisziplinären Teams arbeiten, trennt sich von der vorherrschenden Idee, dass es für unterschiedliche Funktionen jeweils einen separaten Raum gibt (Rahimi und Vogt, 2010). Anstelle eines Wohnzimmers zum Wohnen, eines Schlafzimmers zum Schlafen, einer Küche zum Kochen etc. soll sich das LP Hamburg als intelligente Wohnumgebung der Situation des Bewohners anpassen.

Im LP sollen Experimente in einer Umgebung, die einer Alltagsumgebung sehr ähnlich ist, durchgeführt werden. Zu diesem Zweck ist die Wohnung mit sechs fest installierten und zwei beweglichen Kameras ausgestattet. Mitsamt den installierten Mikrofonen und dem Aufzeichnungsequipment können Usability - Untersuchungen durchgeführt werden. Die dazu nötigen Steuerpulte befinden sich im Kontrollraum, der direkt an die Wohnung grenzt. Von diesem ist es möglich, alle Geräte und Experimente im LP zu steuern. Dafür kann die nötige Verkabelung direkt vom Kontrollraum aus in einem Doppelboden unter dem Wohnbereich verlegt werden.

3.2 Experiment: Hamburg Cubical

Im folgenden Abschnitt wird der Hamburg Cubical beschrieben. Der Hamburg Cubical ist vom Autor in Zusammenarbeit mit der Seamless Interaction Arbeitsgruppe des Ubicomp - Labors der Hochschule für Angewandte Wissenschaften Hamburg entwickelt worden (Gregor u. a., 2009).

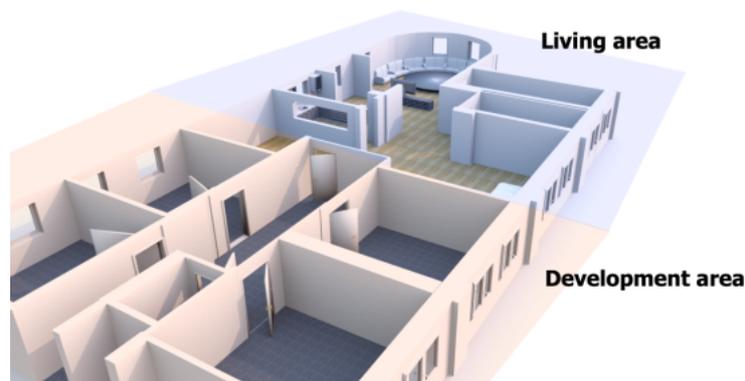


Abbildung 3.1: Schematischer Aufbau Living Place Hamburg (Quelle: [Living Place Hamburg Team](#))

3.2.1 Konzept

Der Hamburg Cubical soll die Idee einer Fernbedienung in einer intelligenten Umgebung umsetzen. Die Hauptanforderungen an die neue Fernbedienung sind vor allem eine handbuchfreie, einfache und intuitive Bedienung (s. Kapitel 4.2.3). Um dies umsetzen zu können, wurde die Fernbedienung als Tangible Interaction Element konzipiert (Gregor u. a., 2009).

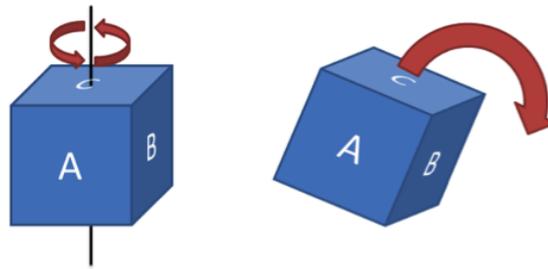


Abbildung 3.2: Interaktionskonzept

Ausgangspunkt der Überlegungen stellte ein einfacher Drehknopf dar. Ein solcher Knopf ist den meisten Menschen vertraut und eine simple Möglichkeit, die Werte einer Funktion zu verändern. Er lässt genau zwei Interaktionen zu: rechts- und links drehen. Aufgrund dieser beschränkten Anzahl von Manipulationsmöglichkeiten ist es einfach, ein mentales Modell (vgl. Kapitel 4.2.3) zu bilden, wie man ein solches Objekt manipulieren kann. Dies macht die Bedienung intuitiv und einfach beherrschbar, weist aber auch gleichzeitig Beschränkungen bei der Funktionsvielfalt auf, da nur eine Funktion hinterlegt werden kann. Um diese Restriktionen zu überwinden, wurde der Würfel als geometrische Grundform für die weiteren Überlegungen gewählt. Durch diese lassen sich sechs unterschiedliche Funktionen annotieren und manipulieren. Durch die Veränderung der Lage des Würfels wird dessen Semantik verändert und durch Links- bzw. Rechts - Rotation (analog zum Drehknopf) eine Kontrolle auf die Funktion ausübt, die der jeweiligen Lage zugeordnet ist (s. a. Abb. 3.2).

Für die Untersuchung des vorgestellten Bedienkonzeptes sind andere Formen, wie beispielsweise ein Oktaeder, nicht ausgeschlossen. Ferscha u. a. (2007) hat unterschiedliche Formen auf ihre Tauglichkeit untersucht und kommt zum Schluss, dass ein Würfel gut geeignet ist. Durch die einfache Form ist es für den Benutzer leichter, eine Annahme über die Funktionsweise zu treffen. Um dem Benutzer beim Aufbau eines mentalen Verständnisses über die Funktionsweise zu unterstützen, soll der Würfel sowohl bei der Auswahl als auch bei der Manipulation eine direkte Rückmeldung geben.

3.2.2 Proof of Concept

Zur Validierung des gewählten Interaktionsmodelles wurde ein einfacher Nutzertest durchgeführt. Als Eingabegerät wurde der Controller einer Wii ¹ verwendet. Durch ein C# - Programm² konnten die Werte des Wii - Controllers ausgelesen und ausgewertet werden.

Als erster Benutzertest wurde ein Stehtisch implementiert, der farblich beleuchtet und mit Hilfe eines Wii - Controllers manipuliert werden konnte. Dieser Aufbau wurde auf einer Feier installiert und beliebige Besucher dieser Feier konnten mit Hilfe des Controllers die Farbe des Tisches ändern. Der Test hat gezeigt, dass die Interaktion von jedem Benutzer verstanden wurde und die Benutzer in der Lage waren, die Farbe des Tisches ohne besondere Erklärungen zu manipulieren und ihre gewünschte Farbe einzustellen.

3.2.3 Prototyp

Zur Evaluation des Konzeptes wurde ein einfacher Prototyp erstellt. Um möglichst schnell unterschiedliche Sensoren und Möglichkeiten für Feedback zu testen sowie die Funktionen iterativ verbessern und erweitern zu können, wurde die Open-Source Hardwareplattform Arduino verwendet. Diese Plattform erlaubt einen sehr schnellen Aufbau von mikrocontrollerbasierten Hardware - Prototypen.

Arduino

Arduino ist ein Open-Source-Projekt auf der Basis von Wiring ([Baragan, 2004](#)). Wiring wurde 2003 von Barragán als Prototyping - Tool zum Einstieg von Künstlern in die elektronische Kunst entwickelt. Es basiert auf einer Mikroprozessorplattform mit einem AVR - Mikrocontroller und einer Entwicklungsumgebung (IDE) auf Basis von Processing ([Reas und Fry, 2005](#)). Durch die enge Verzahnung von Hardware und IDE ist es möglich, den Benutzern, die bei der Mikrocontrollerprogrammierung nötigen Konfigurationsschritte bei der Inbetriebnahme und Programmierung zu erleichtern. So lassen sich Programme für die Arduino - Hardware schreiben, ohne dass der Entwickler über spezielle Kenntnisse vom verwendeten Mikrocontroller oder fundierte Erfahrungen bei der Programmierung von Mikrocontrollern verfügen muss. Als Hardware kommen bei Arduino 8 - Bit Mikroprozessorsysteme³ zum Einsatz. Zum Zeitpunkt der Erstellung dieser Arbeit existiert eine Vielzahl unterschiedlicher Arduino

¹Die Wii ist eine Spielkonsole der Firma Nintendo (s. a. ([Nintendo](#))). Sie lässt sich durch drahtlose Fernbedienungen steuern, die auf die Bewegungen der Benutzer erkennt und diese zur Steuerung von Spielen benutzt.

²C# ist eine Programmiersprache der Firma Microsoft

³Es handelt sich um Chips der ATmega Baureihe der Firma [ATMEL](#).



Abbildung 3.3: Arduino Mega (Quelle: [Arduino](http://www.arduino.cc))

- kompatibler Hardware, wie beispielsweise der in [Abbildung 3.3](#) gezeigten Arduino Mega. Über eine USB-Schnittstelle kann die Arduino - Hardware an einen PC angeschlossen werden. Damit ein Mikroprozessorsystem mit dem Arduino kompatibel ist, muss es über einen speziellen Bootloader⁴ verfügen. Dieser Bootloader ermöglicht das Programmieren des Mikroprozessorsystems durch die Arduino Entwicklungsumgebung, ohne dass eine spezielle Programmierhardware benötigt wird. Die Hardwareplattform ist modular aufgebaut und kann durch sog. Shields, die auf den Arduino gesteckt werden, erweitert werden.

Die Arduino Entwicklungsumgebung (siehe [Abbildung 3.4](#)) ist in Java⁵ geschrieben und besteht aus zwei Hauptelementen:

- Source Code Editor:
Mit dem auf der Processing - Entwicklungsumgebung basierenden Source Code Editor kann der Benutzer Programme, die sog. Sketche editieren, kompilieren und auf die Arduino - Hardware laden. Alle notwendigen Einstellungen, wie die verwendete Hardware und Schnittstelle, lassen sich vom Benutzer einstellen. Dies hat den Vorteil, dass der Benutzer weder die bei der Mikrokontrollerprogrammierung üblichen Make-files, noch Command - Line Argumente für den Compiler bearbeiten bzw. editieren muss. Zum Kompilieren der erstellten Sketche durch den AVR-GCC⁶ - Compiler reicht ein Klick auf die entsprechende Schaltfläche. Der Source Code Editor enthält desweiteren einen Serial Monitor, mit dem eine serielle Verbindung zwischen Hardware und dem Editor hergestellt werden kann. Damit können Ausgaben von der Hardware empfangen und Befehle zur Hardware gesendet werden. Diese Funktionalität erlaubt ein

⁴spezielle Software zum Laden eines Programmes in den Speicher eines Mikrokontrollers während der Initialisierungsphase

⁵objektorientierte Programmiersprache der Firma Sun Microsystems - <http://java.sun.com>

⁶AVR-GCC ist der Name der Compiler-Suite des GNU-Projekts - <http://gcc.gnu.org/>

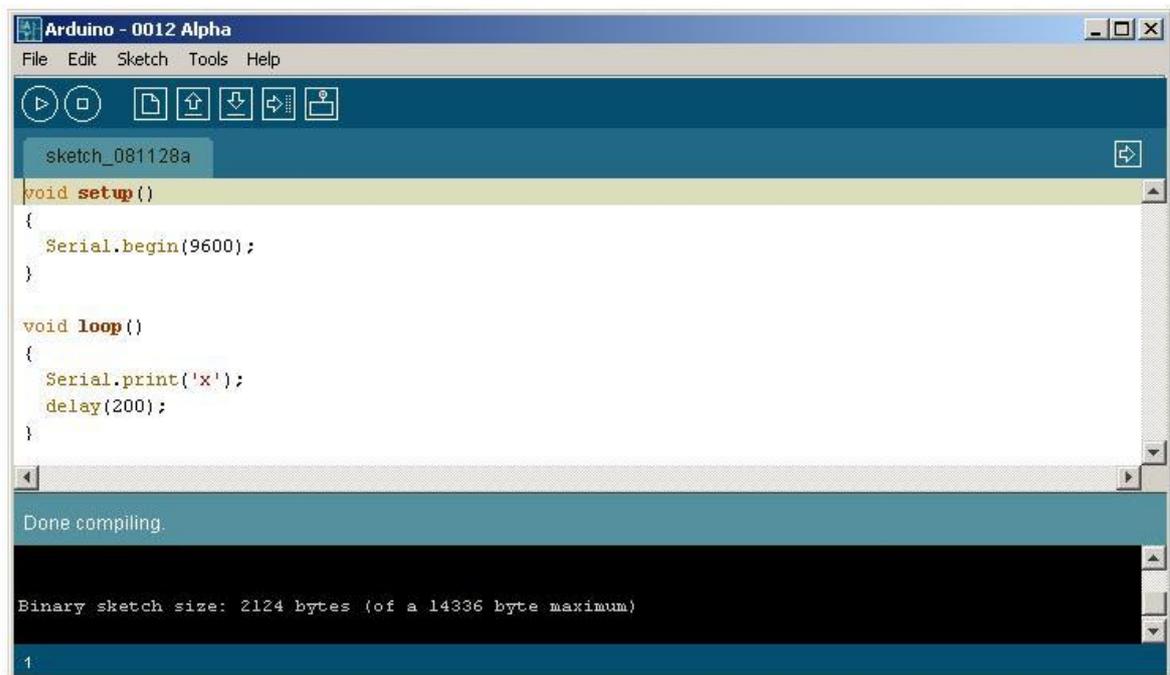


Abbildung 3.4: Arduino - Entwicklungsumgebung

einfaches Debuggen des Programmes auf Hardwareebene mittels Ausgabebefehlen im Programmcode.

- Core Library:
Die Core Library besteht aus einer Reihe von AVR C/C++ - Funktionen. Diese Funktionen kapseln die mikrokontrollerspezifischen Punkte, wie z.B. das Manipulieren von Registern bei der Programmierung. Die Core Library hat die Möglichkeit, andere C/C++ - Bibliotheken einzubinden. Mit dieser Funktion lassen sich anwendungsspezifische Bibliotheken wie z.B. für die Ansteuerung eines Servomotors schreiben und der Arduino Entwicklungsumgebung hinzufügen. Diese Bibliotheken lassen sich über ein einfaches Menu im Source Code Editor in ein Programm einbinden und verwenden.

Die Core Library und der Source Code Editor stehen auf der Arduino Webseite für Windows, MacOS X und Linux zum Download zur Verfügung. Nach dem Entpacken ist die Entwicklungsumgebung sofort funktionsfähig, ohne das weitere Komponenten installiert werden müssen.

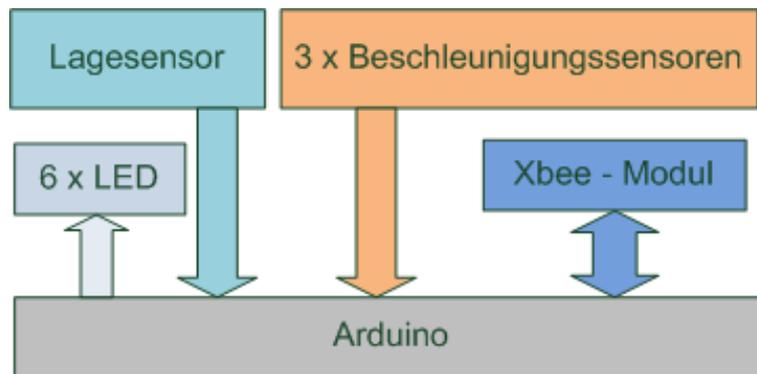
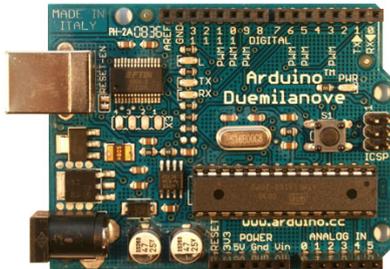


Abbildung 3.5: Blockschaltbild erster Prototyp

Aufbau des Prototypen

Abbildung 3.5 zeigt das Blockschaltbild des ersten Prototyps. Dieser ist mit unterschiedlichen Lage- und Beschleunigungssensoren ausgestattet. Als Rückmeldung an den Benutzer wurde sowohl mit taktilen als auch mit visuellen Feedback-Möglichkeiten experimentiert. Der Prototyp kann sowohl per USB-Kabel aber auch per Batterie und Funk betrieben werden. Abbildung 3.10 zeigt den ersten Prototypen, der mit folgender Hardware realisiert wurde:

Prototyping Plattform



Als Plattform für die Entwicklung des Prototypen wurde der Arduino Duemilanove (vgl. Abbildung 3.6) verwendet. Diese Arduino - Plattform wird mit einer Versorgungsspannung von 5 Volt und mit einer Taktrate von 16 MHz betrieben.

Abbildung 3.6: Arduino Duemilanove (Quelle: [Arduino](#))

Lagesensor

Als Lagesensor wurde ein Drei-Achsen-Bewegungssensore der Firma Freescale Semiconductor Inc. (siehe [Freescale](#)) eingesetzt. In dem MMA7260QT sind drei Beschleunigungssensoren für die x-, y- und z-Achse integriert. Der Chip (IC) wird mit einer Versorgungsspannung von 3,3 Volt betrieben und kann auf vier unterschiedliche Messbereiche eingestellt werden. Für den Prototypen wurde ein Breakoutboard für den MMA7260QT der Firma [Sparkfun](#) verwendet auf dem bereits ein Lowpass - Filter integriert ist (vgl. [Abbildung 3.7](#)).



Abbildung 3.7: MMA7260Q

(Quelle:
[Sparkfun](#))

Sensoren für Drehbeschleunigung



Abbildung 3.8: LISY300AL

(Quelle:
[Sparkfun](#))

Um die Drehbeschleunigung für jede Achse zu ermitteln, wurden 3 Gyroskope der Firma ST Microelectronics ([Elektronics](#)) verwendet. Der LISY300AL misst maximal einen Bereich von 300 Grad pro Sekunde, wobei der IC eine zur Winkelrate analoge Spannung herausgibt. Für den Prototypen wurden Breakoutboards von ([Sparkfun](#)) verwendet, auf denen bereits ein Low-Pass Filter integriert ist (vgl. [Abbildung 3.8](#)).

Funk

Für die Datenübertragung per Funk wurde das ZigBee Protokoll (siehe [802.15 Working Group](#)) verwendet. Als Hardware kam das 2,4 GHz XBee - Modul der Firma Digi zum Einsatz (Abbildung 3.9). Diese Modul benutzt den 802.15.4 Protokollstack, vereinfacht die Benutzung jedoch so stark, dass das Modul mit einfachen seriellen Kommandos benutzt werden kann. Dadurch kann das Modul mit Hilfe des seriellen Ports einer MCU betrieben werden. Die XBee - Module unterstützen sowohl eine Punkt-zu-Punkt als auch eine Punkt-zu-Allen oder Punkt-zu-Mehreren Verbindung.



(Quelle:
[Sparkfun](#))

Stromversorgung

Wenn der Prototyp nicht über USB an einen Rechner angeschlossen ist, wird ein Lithium-Polymer-Akkumulator als Energiequelle verwendet. Zum Einsatz kommt hier ein Lithium Pack von [Liquidware](#) zum Einsatz. Dieser Akkumulator kann direkt unter einen Arduino Duemilanove montiert werden und liefert bei 5 Volt Spannung 2200mAh Strom. Damit konnte der Prototyp ca. 24 Stunden betrieben werden.

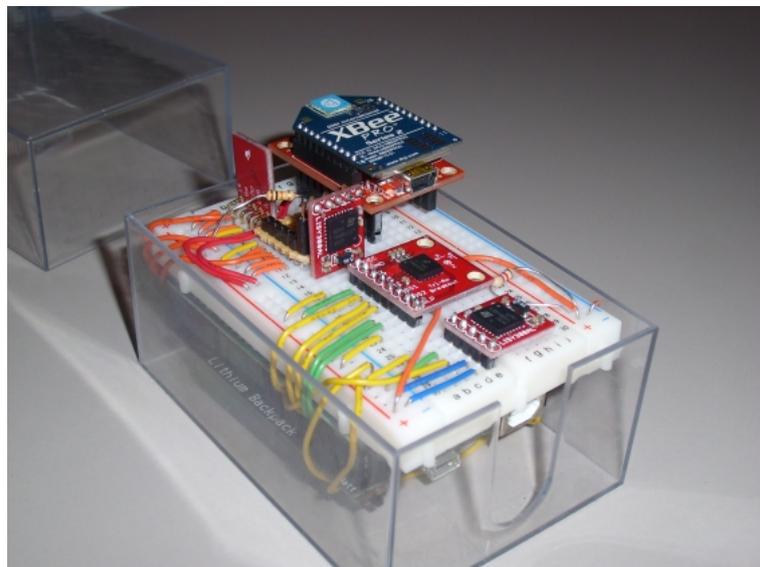


Abbildung 3.10: Erste Prototyp mit geöffneten Gehäuse

Software des Prototypen

Treiber

Der Treiber des Prototypen wurde mit der Arduino IDE in der Programmiersprache C geschrieben. Der Treiber liest alle zwanzig Millisekunden die Werte der Sensoren aus und speichert die jeweils letzten fünf Sensorwerte in ein Array. Mit diesen Werten wird alle einhundert Millisekunden ein gleitender Mittelwert gebildet und an den PC übertragen. In diesem Intervall werden ebenfalls die Farbwerte der RGB - LEDs neu gesetzt. Die dafür nötigen Daten erhält der Treiber über das Kommunikationsprotokoll von der Serversoftware. Wenn der Treiber seit der letzten Aktualisierung keine neuen Werte von der Serversoftware erhalten hat, werden die alten Farbwerte verwendet. Abbildung 3.11 zeigt den logischen Ablauf des Treiberprogrammes als Flussdiagramm.

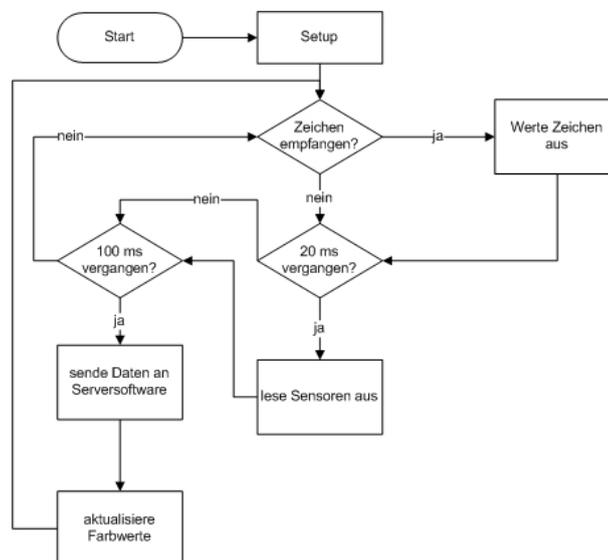


Abbildung 3.11: Flussdiagramm Treibersoftware

Serversoftware

Die Serversoftware wurde von Matthias Vogt als C# - Programm realisiert. Die Software empfängt die Sensorwerte vom Cubical, wertet diese aus und triggert ggf. Funktionen, die Dienste oder Aktoren steuern und schickt neue Farbwerte an den Cubical. Beim Start der Software kann eine Konfigurationsmodus durchlaufen werden. Dabei wird der Cubical auf die entsprechende Seite gelegt und die Werte des Lagesensors werden über eine GUI einer Funktion zugeordnet. Da die Werte des Lagesensors schwanken und der Cubical auch auf einer nicht hundertprozentig ebenen Fläche liegen kann, wird die Funktion einem Wertebereich zugeordnet, der sich um den ermittelten Wert aus dem Konfigurationsmodus gruppiert. Die einzelnen Funktionen und

Dienste, die mit dem Cubical gesteuert werden und in welcher Art und Weise wie diese manipuliert werden können, ist in der Serversoftware fest implementiert.

Kommunikationsprotokoll

Damit die Serversoftware und der Treiber des Cubicals Daten austauschen können, wurde ein Kommunikationsprotokoll festgelegt. Die Werte des Cubicals werden im JSON⁷ - Format an die Serversoftware gesendet (s. Abb. 3.12). Um die Daten auswerten zu können, wird ein JSON - Parser verwendet. Diese Art Parser steht für eine Vielzahl von Betriebssystemen zur Verfügung. Das macht eine schnelle und einfache Entwicklung von Software möglich, die den Cubical als Eingabemedium verwenden kann. Für das Setzen der Farbwerte verwendet die Treibersoftware des Cubicals ein einfaches Protokoll, welches aus drei Bytes pro RGB-Mischfarbe besteht. Das erste Byte kann aus den ASCII - Werten für r, g oder b bestehen und zeigt an welcher RGB-Farbwert gesetzt werden soll. Das zweite Byte muss per Definition der ASCII - Wert für den Kleinbuchstaben v sein und das dritte Byte ist dann ein Farbwert zwischen 0 und 255.

```
["BX": "[BXVAL]", "BY": "[BYVAL]", "BZ": "[BZVAL]", "LX": "[LXVAL]", "LY": "[LYVAL]", "LZ": "[LZVAL]"]
```

[BXVAL], [BYVAL] und [BZVAL] entspricht den aktuellen Messwerten zur Drehbeschleunigung für X-, Y- und Z-Achse
[LXVAL], [LYVAL] und [LZVAL] entspricht den aktuellen Messwerten zur Lage für X-, Y- und Z-Achse

Abbildung 3.12: Cubical JSON Format

Nutzertest

Mit dem oben beschriebenen Prototypen wurde eine erster Testaufbau erstellt. Als Feedback für den Benutzer wurden in jede Würfelseite farbige LEDs integriert. Ist eine Lage mit einer Funktion verbunden, so wird es dem Benutzer mit dem Aufleuchten der LED für diese Würfelseite angezeigt. Als Funktionen wurden die Lautstärkeregelung eines PCs und die Farbgebung und das Dimmen einer Lampe implementiert und unterschiedlichen Seiten des Würfels zugeordnet. Die Lautstärke des PCs und analog dazu die Helligkeit und Farbe der Lampe konnten durch Rechts- und Linksdrehung verändert werden. Abbildung 3.13 zeigt den Versuchsaufbau, bei dem drei der sechs Würfelseiten mit Funktionen belegt wurden.

⁷JSON steht für JavaScript Object Notation und ist ein leichtgewichtiges Format zum Datenaustausch (s. a. [Crockford](#)).



Abbildung 3.13: Versuchsaufbau

Eine Reihe von zufällig ausgewählten Benutzern, die nicht zum Entwicklungsteam gehörten, wurden vor die Aufgabe gestellt, die unterschiedlichen Funktionen zu testen. Den Testpersonen wurde erläutert, welche Funktionen vorhanden sind und dass sie per Drehung des Tangibles manipulierbar sind. Hingegen wurde den Probanden weder mitgeteilt welche Lage welcher Funktion entspricht noch was die Drehung in einer konkreten Lage bewirkt. Es stellte sich heraus, dass die Benutzer über die Farben der LEDs schnell die Funktionsweise erkannten und sich das

Verknüpfen der Funktionen mit der aufleuchtenden Farbe leicht einprägen ließ. Die Manipulation der einzelnen Funktionen wurde auf Anhieb verstanden und erfolgreich vorgenommen. Von den Probanden wurde neben fehlenden Symbolen auf dem Würfel auch die wenig ergonomische Form kritisiert.

3.2.4 1. Verfeinerung des Prototypen

Nach den ersten absolvierten Tests mit dem Prototypen wurde die Bauform verändert. Gründe hierfür waren vor allem die Notwendigkeit, unterschiedliche Größen evaluieren zu können und die Zuverlässigkeit der Testhardware zu verbessern. Zu diesem Zweck wurde ein Platinen - Layout und unterschiedliche Gehäuse - Prototypen erstellt.

Gehäuse - Prototypen

Um experimentell die richtige Größe für das Gehäuse herausfinden zu können, wurden Gehäuse - Prototypen erstellt. Damit die Prototypen möglichst einsatznah getestet werden konnten, wurde ein Material gesucht, das stabil und doch lichtdurchlässig ist. Die Wahl fiel auf mattes Plexiglas, einem bruchfesten, halb-transparentem Kunststoff. Mit diesem Kunststoff wurden drei Cubicals mit den Kantenlängen 5, 6.5 und 8 Zentimetern erstellt (vgl. Abb. 3.14). Auf die Flächen konnten nach Bedarf auf Folie aufgedruckte Symbole aufgebracht werden. Der oben be-

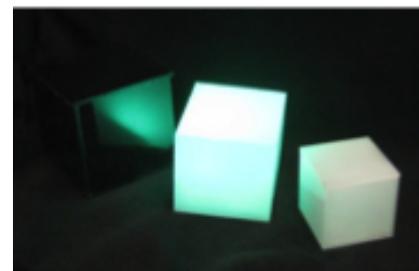


Abbildung 3.14: Gehäuse-Prototypen

beschriebene Versuch wurde mit diesen Tangible und neuen Probanden wiederholt. Bei diesen Untersuchungen stellten sich Würfel mit einer Kantenlänge von 5 Zentimetern und 6.5 Zentimetern für die Testpersonen als am angenehmsten in der Handhabung heraus.

Hardware - Layout

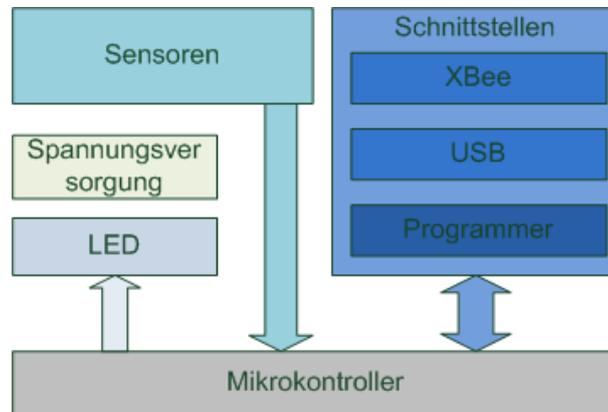


Abbildung 3.15: Blockdiagramm Hardware des Prototypen

Beim Entwurf der Hardware wurde darauf geachtet, dass Platine und Bauteile kompatibel zu der Arduino Plattform sind. Dadurch mussten nur marginale Änderungen an der Treibersoftware vorgenommen werden. Die Hardware des Cubicals besteht aus fünf logischen Teilen (vgl. farbige Markierung in Abbildung 3.15):

Mikrokontroller

Als Mikrokontroller (MCU) wurde der Atmega328P der Firma Atmel verwendet. Dabei handelt es sich um eine MCU mit 8-Bit RISC Architektur ([Atmel](#)). Da der Prototyp mit einer Versorgungsspannung von 3,3 Volt betrieben wird, kann der Mikrokontroller ausschließlich mit einer Taktrate von acht MHz betrieben werden.

Stromversorgung

Die Stromversorgung kann sowohl über Batterie als auch über USB geschehen. In beiden Fällen wird die Spannung durch eine LDO⁸ Spannungsregler auf 3,3 Volt Gleichspannung geregelt. Diese Spannung wurde gewählt, um den Prototypen mit einem Lithium-Polymer-Akkumulator (LiPo) mit einer Nennspannung von 3,7 Volt und 1100 Mili-Ampere-Stunden (mAh) betreiben zu können. Ein größerer Akkumulator, der eine höhere Nennspannung und größeren Leistung hat, kam aus Gewichtsgründen und aufgrund der größeren Form nicht in Frage.

⁸LDO ist die Abkürzung für Low-Drop Out. Dies bedeutet, dass zwischen Ein- und Ausgangsspannung nur eine geringe Abfallspannung entsteht.

Sensoren

Als Sensorplattform wurden fertige Platinen der Firma Sparkfun verwendet (vgl. Abbildung 3.16), auf welchen bereits ein Drei-Achsen Lagesensor und Beschleunigungssensoren für jede der drei Achsen aufgebracht worden sind (Sparkfun). Die Sensoren liefern je nach Lage, bzw. Beschleunigung eine analoge Spannung. Diese Spannung wird von den internen ADC des Mikrokontroller abgefragt. Die Platine kann per Steckverbinder auf die Hauptplatine gesteckt werden.



Abbildung 3.16: Sensoren
(Quelle:
Spark-
fun)

LED

Um den Prototypen mit unterschiedlichen Farben illuminieren zu können, wurden an zwei Ecken der Hauptplatine RGB - LEDs plziert. Durch den Anschluss an drei PWM⁹-Kanäle des Atmega328, lassen sich so eine Vielzahl unterschiedlicher Farben erzeugen.

Schnittstellen

Es gibt drei Schnittstellen, über die mit der Platine kommuniziert werden kann. Als Programmierschnittstelle steht ein sechspoliger In-System-Programmer (ISP) - Anschluss zur Verfügung. Mit einem ISP - Programmer kann über eine Programmiersoftware ein Programmcode in den Atmega geladen werden. Beim Prototypen wird diese Schnittstelle nur für das Programmieren der MCU mit dem Arduino-Bootloader (s. Kapitel 3.2.3) verwendet. Zur Kommunikation mit einem PC wurde eine USB - Schnittstelle und ein XBee Modul auf der Platine integriert. Der XBee kann über Steckverbinder direkt auf die Unterseite der Platine gesteckt werden. Für die USB - Kommunikation wurde eine FTDI FT232RL als Treiberchip vorgesehen(FTDI). Dieser Chip wandelt die Signalpegel des Atmega zu USB - konformen Signalen und verarbeitet die Protokoll - spezifischen Kontrollsequenzen.

PCB - Layout

⁹PWM ist die Abkürzung für Pulsweitenmodulation und beschreibt ein Verfahren bei dem bei konstanter Frequenz das Taktverhältnis moduliert wird.

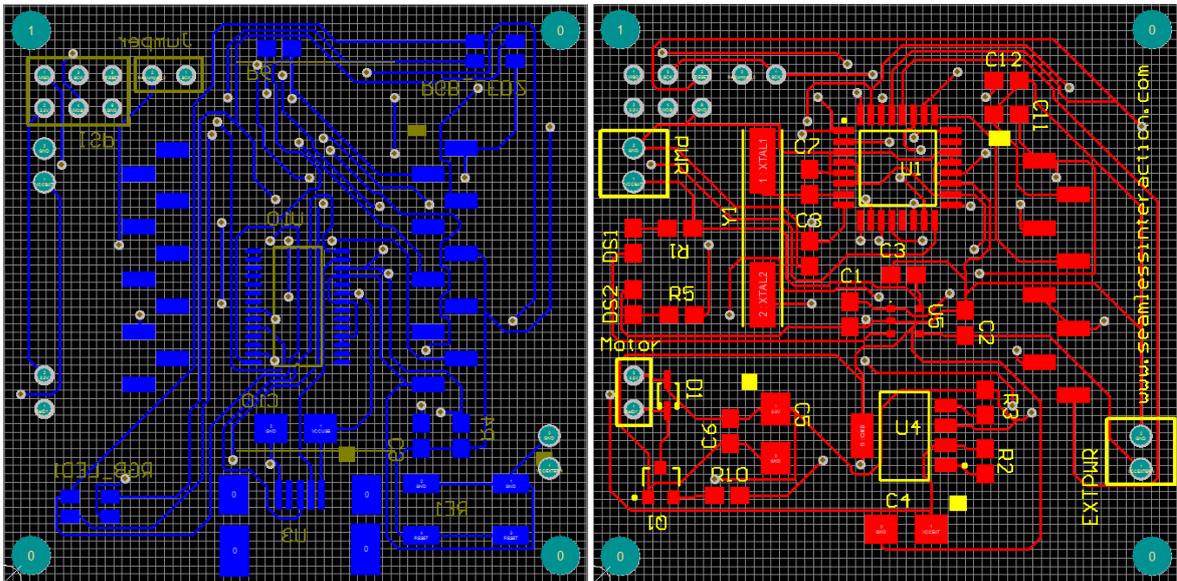


Abbildung 3.17: PCB-Layout (Oberseite)

Abbildung 3.18: PCB-Layout (Unterseite)

Mit den beschriebenen Komponenten wurde ein Layout für eine doppelseitige PCB¹⁰ - Platine mit den Ausmaßen 50 x 50 Millimeter erstellt. Auf der Oberseite (vgl. Abbildung 3.17) erkennt man mittig die Pads für FTDI - Treiber (U10) mit dem am unteren Rand erkennbaren Platz für den USB - Steckverbinder (U3). Umrahmt wird der FTDI - Treiber von den Steckverbindern für das XBee Modul. In der rechten oberen und der linken unteren Ecke sieht man jeweils vier Pads für die LEDs.

Auf der Unterseite (s. Abbildung 3.18) kann man die Plätze für den Mikrokontroller (beschriftet mit U1), den acht MHz Quarz (Y1), den Spannungsregler (U4), sowie rechts neben Mikrokontroller und Spannungsregler die Pads für Steckverbinder der Sensoren erkennen. In den Abbildungen 3.19 und 3.20 ist die fertig bestückte Platine zu sehen.

¹⁰PCB steht für Printed Circuit Board, einer Leiterplatte, auf der Leitungen und Lötstellen für Komponenten aufgetragen sind.

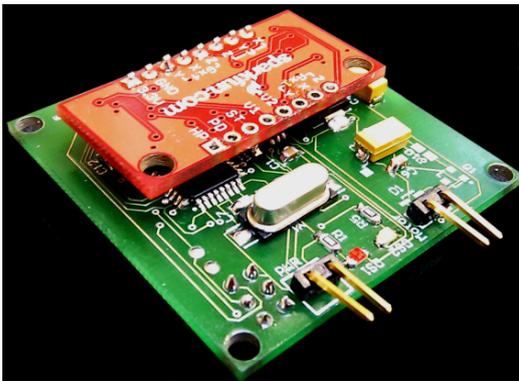


Abbildung 3.19: Platine (Unterseite)

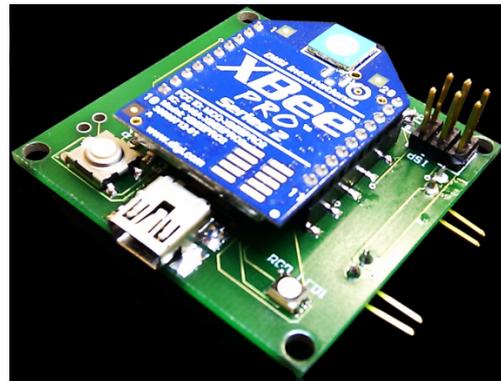


Abbildung 3.20: Platine (Oberseite)

3.2.5 2. Verfeinerung des Prototypen

Die erste Überarbeitung des Prototypen ließ ein wesentlich stabileres Testen und Entwickeln zu. Beim Testen mit dem Prototypen wurden jedoch einige Nachteile sichtbar:

- Die **Bauform** des Prototypen wies einige Nachteile auf. Nachdem das Breakoutboard mit den Sensoren auf der einen Seite und das XBEE Funkmodul auf der anderen Seite angebracht waren, stellte sich die Platine als **zu hoch** heraus. Da die LEDs auf der Platine aufgebracht sind, wurden die beiden kleineren Gehäuseprototypen nur auf einer Seite beleuchtet.
- Die **fehlende** Möglichkeit der **Befestigung** der Platine auf den Gehäuseprototypen ließ die Platine bei schnellen Bewegungen rutschen und verfälschte die Messwerte. Das Kleben der Platine in das Gehäuse erwies sich als unbrauchbare Lösung, da zum Austauschen der Batterien die Platine entfernt werden musste.
- Die **LEDs**, die auf der Platine aufgebracht wurden, hatten **nicht genug Leuchtkraft** und wurden **falsch positioniert**. Bei aufgestecktem XBee Funkmodul war die Ausleuchtung zu ungleichmäßig.
- Die **Akkumulatoren** mussten häufig **zum Aufladen ausgetauscht** werden, weil sie nicht an der Platine geladen werden konnten.

Um diese Nachteile korrigieren zu können, wurde folgende Verbesserungen vorgenommen:

- Um die *Höhe der Platine zu verringern*, wurden die **Lage- und Beschleunigungssensoren direkt auf die Platine appliziert**.

- Um die Ausleuchtung der LEDs zu verbessern, wurden **LEDs mit besseren physikalischen Eigenschaften** ausgesucht und zwei **LEDs mittig auf der Platine** auf der gleichen Seite wie die Sensoren platziert. Durch diese Anordnung kann sich das Licht zu allen Seiten gleichmäßig ausbreiten und eine homogene Ausleuchtung des gesamten Würfels (außer der Unterseite) ist möglich.
- Um die Akkumulatoren nicht jedes Mal zum Laden wechseln zu müssen, wurde eine **Ladefunktion auf der Platine integriert**. Dazu wurde der MAX1555 von Maxim als IC aufgebracht. Diese IC kann die verwendeten Lithium-Polymer Akkumulatoren laden, ohne dass die Platine im Betrieb unterbrochen werden muss. Damit wurde die **Voraussetzung zur induktiven Ladung der Akkumulatoren** geschaffen.
- Zur Verbesserung der Befestigung der Platine wurde ein **neues Gehäuse konzipiert**. Dieses wurde im Rapid Prototyping - Verfahren hergestellt ([MuP/RP, 2010](#)). Bei diesem Verfahren wird aus einem 3D-Modell¹¹ heraus ein Objekt gedruckt. Bei dem Modell wurden vier Stifte für die Aufnahme der Platine und zum Fixieren der Akkumulatoren vorgesehen (vgl. [Abb. 3.23](#)).

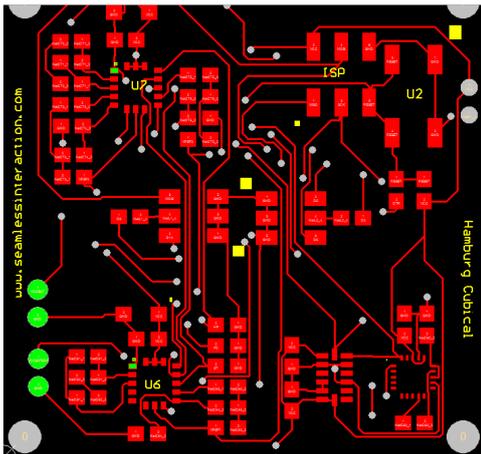


Abbildung 3.21: Platine (Oberseite)

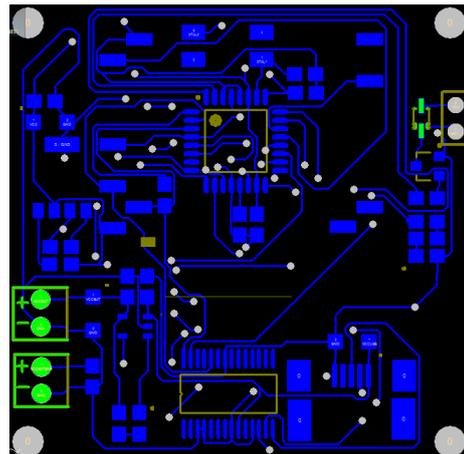


Abbildung 3.22: Platine (Unterseite)

Mit diesen Änderungen wurde ein neues zweiseitiges Platinen-Layout mit dem Mikrocontroller, dem FTDI Treiber, dem Spannungswandler, dem IC zum Laden der Akkumulatoren und den Steckverbindern auf der Unterseite (s. [Abb. 3.21](#)), sowie den Sensoren und LEDs auf der Oberseite (s. [Abb. 3.22](#)) erstellt.

¹¹Das Modell wurde von Thomas Schulz hergestellt.

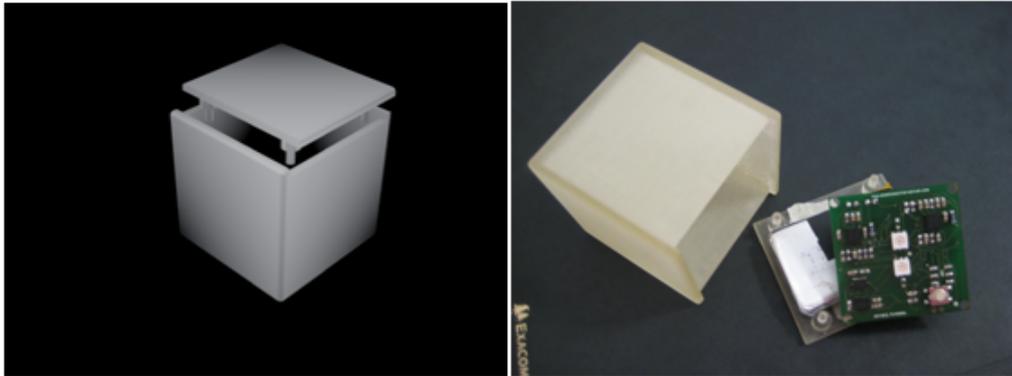


Abbildung 3.23: 3D Modell des Gehäuses (rechts) und gedrucktes Gehäuse mit Platine (links)

3.2.6 Zusammenfassung

Der Hamburg Cubical wurde als Tangible Interaction Bedienelement konzipiert. Es handelt sich um einen Würfel mit fünf Zentimeter Kantenlänge. Über in den Würfel eingebaute Sensoren lässt sich die Lage des Würfels und die Beschleunigung bei einer Rotation des Würfels ermitteln. Durch diesen Aufbau und die Annotation jeder Lage mit einer (digitalen) Funktion ist es möglich, diese (digitale) Funktion zu manipulieren. Dies geschieht, indem ein Benutzer über die Lage die gewünschte Funktion auswählt und über eine Rotation des Würfels die Funktion manipuliert.

Die Möglichkeit der Manipulation von Daten über die Rotation eines physischen Objektes wurde mit einem Wii - Controller getestet. Zu diesem Zweck wurden auf einer Firmenmesse ein Tisch mit LEDs und einer Husse ausgestattet. Über den Wii - Controller ließ sich nun die Farbe der LEDs und damit die Farbe der Husse verändern. Bei diesem Aufbau stellte sich die Interaktionsmetapher als für die einfache Manipulation von Funktionen geeignet heraus.

Zur weiteren Entwicklung wurde mit Hilfe der Arduino - Plattform ein einfacher Prototyp erstellt. Dabei wurde auf eine Arduino Duemilanove mit einem LiPo - Akkumulator, ein XBee Funkmodul, sowie Lage- und Beschleunigungssensoren appliziert. Dieser Aufbau wurde in ein einfaches Gehäuse in Form eines Hexaeders verpackt und mit farbigen LEDs ausgestattet. Mit diesem Würfel wurde ein Versuchsaufbau erstellt, der aus einem Laptop und einer RGB - Lampe bestand. Mit dem Cubical Prototypen ließen sich die Helligkeit, die Farbe und die Lautstärke des Laptops verändern. Ein Test mit einer Reihe zufällig ausgewählter Benutzer zeigte die Tauglichkeit des Prototypen bei der Manipulation von einfachen Funktionen.

Um die Handhabbarkeit und die Zuverlässigkeit des Prototypen zu verbessern, wurde dieser überarbeitet. Es wurde eine zweiseitige Platine mit den Maßen fünf mal fünf Zentimeter

und unterschiedlich große Gehäusetyperen aus lichtdurchlässigem, mattem Plexiglas erstellt und weitere Tests vorgenommen. Dabei stellte sich ein Würfel mit der Kantenlänge von 5,5 Zentimeter und abgerundeten Kanten als für die Benutzer angenehm zu verwendbare Form heraus. Technisch stellte sich die zu hohe Bauform, die fehlende Befestigungsmöglichkeit der Platine am Gehäuse, die zu schwachen und falsch positionierten LEDs und die fehlende Auflademöglichkeit für die Akkumulatoren als Nachteile des überarbeiteten Prototypen heraus. Um diese Nachteile zu beseitigen und weitere Tests durchführen zu können, wurde der Prototyp noch einmal verfeinert. Zu diesem Zweck wurden ein neues Layout für die Platine und ein neues Gehäuse mit 5,5 Zentimeter Kantenlänge im Rapid Prototyping Verfahren erstellt. Durch die Überarbeitung konnten die beschriebenen Nachteile des vorherigen Prototypen verbessert werden.

Mit dem Hamburg Cubical wurde prototypisch eine einfache Art der Steuerung und Manipulation von Funktionen geschaffen. Die Entwicklung erfolgte mittels iterativ verbesserten Prototypen. Diese Art der Entwicklung erlaubt es, schon in einem frühen Entwicklungsstadium gleichzeitig die Verwendbarkeit der gewählten Interaktion von Benutzern zu untersuchen und die technische Realisierung mit unterschiedlichen Techniken zu testen.

Obwohl bei der Entwicklung Prototypen verwendet wurden, lag deren Fokus zu stark auf technischen Details und der Untersuchung der technischen Umsetzbarkeit. Der Einsatz von Prototypen bei der Untersuchung der unterschiedlichen Materialien und Gehäuseformen in Verbindung mit Usability - Tests kam aus Sicht des Autors bei der bisherigen Entwicklung zu kurz. Dieser Punkt wird bei der zukünftigen Entwicklung eine größere Rolle einnehmen müssen. Inwieweit Form und Materialien die technischen und funktionalen Eigenschaften des Hamburg Cubicals beeinflussen, ist hierbei eines der Hauptmerkmale, die es zu untersuchen gilt.

Bei der technischen Entwicklung wird in den nächsten Schritten die Entwicklung einer induktiven Ladefunktionalität für die Akkumulatoren, die Implementation von Energiespartechiken und die Verbesserung der Stabilität und der Konfigurationsmöglichkeiten durch den Anwender im Vordergrund stehen.

3.3 Experiment: Hamburg SITab

In den letzten Jahren haben sich mit den Smartphones¹² und Tablets¹³ neue Geräteklassen im Computermarkt etabliert. Dabei handelt es sich um Geräte, die für den mobilen Einsatz

¹²Smartphones sind leistungsstarke Mobiltelefone, die über eine Vielzahl von erweiterten Diensten wie z. B. E-Mail Programmen verfügen.

¹³Tablets sind in ihrer Größe und Leistungsfähigkeit zwischen Smartphone und Notebook einzugliedern.

entwickelt wurden und oft weder eine Maus noch eine Tastatur als Eingabemöglichkeit besitzen. Stattdessen verfügen diese Geräte über einen berührungsempfindlichen Bildschirm als Interaktionsmöglichkeit¹⁴, wobei sowohl Stifte als auch die Finger einer Person als Eingabemedium dienen. Desweiteren besitzen viele dieser Geräte Lage- und Beschleunigungssensoren.

Durch diesen Aufbau sind neue Formen der Interaktion mit diesen Geräten, wie z. B. die Eingabe mittels durch die Finger ausgeführte Gesten s. a. (Gehn, 2010) möglich. Dabei sind die Bildschirme von Smartphones mit einer Diagonalen von fünf Zoll und Tablets mit einer Größe von sieben bis zehn Zoll im Vergleich zu Notebooks relativ klein.

Diese relativ kleinen Displays erschweren nicht nur die Eingabe von Daten und Konfiguration des Gerätes, sie führen auch dazu, dass Daten, wie z. B. Bilder, Filme oder Textdokumente, sehr klein und dadurch nur schwer erkennbar dargestellt werden. Um Details erkennen oder Texte lesen zu können, müssen Ausschnitte vergrößert werden und es können jeweils nur Teile in einer akzeptablen Größe betrachtet werden.

Eine weitere Einschränkung ergibt sich dadurch, dass es keine nahtlose Synchronisation der Daten von zwei Geräten gibt. Dabei verwenden viele Nutzer zwei oder mehr der beschriebenen Geräte, was unweigerlich zu einer Fragmentierung der Daten des Nutzers auf mehrere Geräte führt. Der Abgleich und die Synchronisation von Daten stellt eine Herausforderung dar. Um eine Datei von einem Gerät auf ein anderes zu kopieren, muss eine Verbindung zwischen beiden aufgebaut werden. Dafür stehen unterschiedliche Möglichkeiten zur Verfügung:

- Austausch über einen Datenträger wie z. B. USB Stick oder SD Karte.
- Aufbau einer drahtlosen Verbindung zu einem Zielgerät.
- Aufbau einer kabelgestützten Verbindung von einem Gerät zum anderen.

Nachdem eine Verbindung aufgebaut ist, muss die Datei entweder manuell oder automatisch über ein Synchronisationsprogramm auf das Zielsystem kopiert werden. Der beschriebene Ablauf bringt eine Reihe von Schwierigkeiten mit sich. Betrachtet man den Verbindungsaufbau per Funk, so müssen zumindest eine der folgende Bedingungen erfüllt sein, um eine Verbindung aufzubauen und Daten auszutauschen:

- 1 Beide Geräte müssen in einem Funknetzwerk erreichbar sein.
- 2 Auf beiden Geräten muss die Berechtigung für das jeweils andere Gerät zum Austausch von Daten eingerichtet sein.
- 3 Das Austauschformat muss von beiden Geräten akzeptiert werden.
- 4 Auf beiden Geräten muss ausreichend Speicherplatz vorhanden sein.

¹⁴abgesehen von der Spracheingabe bei Smartphones

Während Punkt drei und vier hier nicht weiter betrachtet werden sollen, bringen Punkt eins und zwei eine Vielzahl von Konfigurationsschritten für den Benutzer mit sich. Diese Konfigurationsschritte werden dadurch erschwert, dass die nötigen Eingaben auf einem (wie oben beschrieben) kleinen Touch - Bildschirm durchgeführt werden müssen. Dies ist eine oft umständliche Prozedur, die im Vergleich zum eigentlichen Austausch der Daten viel Zeit in Anspruch nimmt.

3.3.1 Interaktionskonzept

Um die im vorherigen Abschnitt beschriebenen Nachteile ausgleichen zu können, ist es notwendig, die Geräte in ihrer Umwelt zu verorten. Wenn es jedem Gerät möglich ist, seine direkte Nachbarn zu erkennen, lassen sich völlig neue Formen der Interaktion damit durchführen. Abbildung 3.24 zeigt zwei mögliche Interaktionen. Die linke Seite der Abbildung stellt

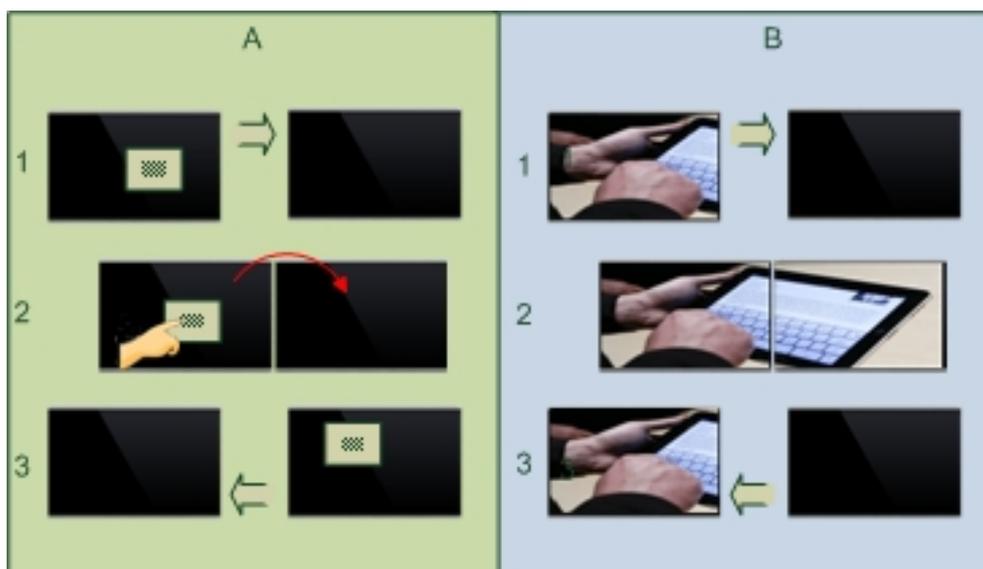


Abbildung 3.24: Interaktionskonzept

das Verschieben eines Datenobjektes von einem Tablet zu einem anderen dar. Die rechte Seite veranschaulicht, wie ein Bild skaliert auf zwei Tablets angezeigt wird.

In den folgenden beiden Szenarien wird die Verwendung eines solchen Konzeptes genauer betrachtet:

Verschieben von Daten

Das Verschieben von Daten von einem Gerät auf ein anderes (Abbildung 3.24 linke Seite A) lässt sich in drei Abschnitte gliedern:

- 1 Im ersten Abschnitt bauen beide Geräte eine Verbindung zueinander auf. Zu diesem Zweck werden sie unmittelbar nebeneinander platziert. Beide Tablets müssen nun in der Lage sein zu erkennen,
 - * dass sich ein anderes Gerät in unmittelbarer Nähe befindet,
 - * wo sich das andere Gerät befindet,
 - * wie die Identität des anderen Gerätes ist,
 - * ob und wie Daten mit dem anderen Gerät ausgetauscht werden können.

Mit diesen Informationen sind beide Geräte in der Lage, eine Verbindung zueinander aufzubauen.

- 2 Im zweiten Schritt können die Daten von einem Gerät auf ein anderes übertragen werden. Dazu soll die Datei mit einer Geste an den Rand des Gerätes gezogen werden. Wenn sich ein anderes Gerät an diesem Rand befindet, dann soll die Datei zu diesem Gerät transferiert werden und am Rand des Zielgerätes erscheinen. Zu diesem Zweck muss jedes Gerät erkennen,
 - * dass die Daten zu einem anderen Gerät übertragen werden sollen,
 - * zu welchem Gerät die Daten übertragen werden sollen,
 - * wie die Verbindung zu diesem Gerät aufgebaut werden kann,
 - * wann die Daten übertragen sind.

- 3 In diesem Schritt wird die Verbindung zwischen den beiden Geräten getrennt. Dies geschieht, indem man die beiden Geräte physisch voneinander trennt. Um dies durchzuführen, müssen beide Geräte erkennen, dass sie sich voneinander entfernt haben.

Durch diesen Aufbau ist es möglich, den sinnbildlichen Begriff des Verschiebens von Daten von einem Computer zu einem anderen durch eine Geste zu bewerkstelligen. In dieser Geste erfasst man ein Dokument mit einem Finger und schiebt es durch das Gleiten des Fingers über die Oberfläche des Touchbildschirms zum Rand des Gerätes auf ein Gerät, was direkt daneben liegt.

Skalieren von Bilddaten

Das Skalieren von Bildern ist in Abbildung 3.24 auf der rechten Seite dargestellt. Es lässt sich ebenfalls in drei Abschnitte gliedern:

- 1 Der erste Schritt geschieht analog mit dem ersten Schritt beim Szenario „Verschieben von Daten“.

2 Beim zweiten Schritt wird die Anzeige der Bilddaten so skaliert, dass sie auf beiden Geräten angezeigt wird. Dabei kann man sich zwei mögliche Szenarien vorstellen:

* *automatisches Skalieren*

Beim automatischen Skalieren werden Bilddaten bei der Verbindung von zwei Geräten automatisch so verändert, dass sie auf der größeren Anzeigefläche optimal dargestellt werden.

* *Darstellen ausgeblendeter Bereiche*

Beim Darstellen von ausgeblendeten Bereichen werden Bereiche von Bilddaten, die über den anzeigbaren Bereich eines Gerätes „hinausragen“ würden, automatisch auf einem anderen Gerät angezeigt, sobald beide miteinander verbunden sind.

Um diese beiden Szenarien abbilden zu können, müssen beide Geräte erkennen,

- * dass Ausschnitte von Bilddaten skaliert dargestellt werden sollen,
- * wie die Verbindung zu einem Zielgerät aufgebaut werden kann,
- * wieviele Geräte bereits miteinander interagieren und ein Bild anzeigen,
- * welcher Bereich auf welchem Gerät skaliert werden soll,
- * ob ein Gerät seinen Bildbereich neu skalieren oder einen anderen Bildbereich anzeigen muss.

3 Beim dritten Schritt wird die Verbindung zwischen den beiden Geräten getrennt. Jedes der Geräte zeigt dann eine Ansicht der Bilddaten, wie sie vor der Verbindung mit dem anderen Gerät bestanden hat.

Durch diesen Ablauf ist es möglich, auf einem Gerät Bilddaten zu zoomen und Details eines Bildes zu betrachten und dennoch durch das Aneinanderlegen mit anderen Geräten eine Übersicht über das gesamte Bild zu behalten. Dadurch lassen sich große Bilddaten auf den (im Vergleich zu Laptops oder herkömmlichen Computerbildschirmen) kleineren Bildschirmen in einer adäquaten Skalierung betrachten.

Technische Herausforderungen

Aus den im vorherigen Kapitel beschriebenen Szenarien und dem Interaktionskonzept lassen sich drei technische Herausforderungen identifizieren:

- **Erkennen von Nachbargeräten**

Das Erkennen und Identifizieren von Nachbargeräten ist die Hauptvoraussetzung für die Umsetzung des oben vorgestellten Interaktionskonzeptes. Viele der Geräte aus den beschriebenen Geräteklassen verfügen über Funkmöglichkeiten und GPS oder Näherungssensoren. Alle diese Technologien ermöglichen eine Positionserkennung der Geräte. Sie sind allerdings zu ungenau, um zentimetergenau zu erkennen, ob sich ein Gerät neben einem anderen befindet. Desweiteren lassen die genannten Sensoren keine Erkennung der Ausrichtung eines Gerätes zu. Dies ist aber Voraussetzung bei der Erkennung, an welcher Geräteseite sich an anderes Gerät befindet und wie dieses ausgerichtet ist.

- **Identifizieren von Nachbargeräten**

Nachdem ein Gerät ein Nachbargerät erkannt hat, so muss es dieses Geräte identifizieren können. Dies ist für den Verbindungsaufbau, der Kommunikation und der Synchronisation von Daten, die von beiden Geräten verwendet werden, wichtig.

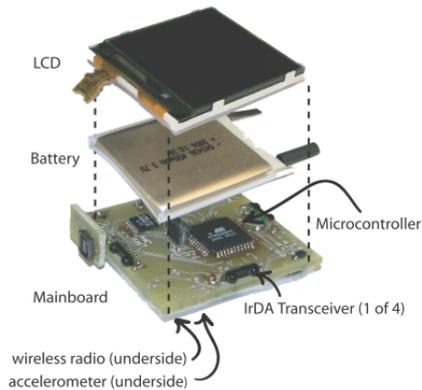
- **Datenaustausch und Synchronisation**

Um die Geräte auf die wie im Konzept beschriebene Weise verwenden zu können, müssen die Geräte auf einer gemeinsamen Datenbasis arbeiten oder zumindest Daten austauschen können. Die Synchronisation der Daten ist vor allem dann wichtig, wenn von mehreren Geräten eine Bilddatei angezeigt wird. So muss z. B. ein auf einem Gerät angezeigter Bildbereich verändert werden, wenn ein neues Gerät zu der Gruppe von anzeigenden Geräten hinzukommt.

Während der Datenaustausch und die Synchronisation der Daten eine Herausforderung ist, die auf der Ebene der Anwendungssoftware gelöst werden kann, müssen bei den anderen beiden Herausforderungen zusätzliche Sensoren an ein Gerät annotiert werden. Die im folgenden Abschnitt beschriebenen Projekte haben sich bereits mit ähnlichen Herausforderungen beschäftigt.

3.3.2 Verwandte Arbeiten

Siftables



Die Idee der Siftables entstand aus den Beobachtungen der Fähigkeiten von Menschen, kleine Objekte zu händeln, zu sortieren und mit ihnen zu interagieren (Merrill u. a., 2007). Dabei benutzen die Menschen beide Hände aktiv und agieren so äußerst effektiv. Um diese Fähigkeiten zu nutzen, entwickelten die Autoren kleine 36 Quadratzentimeter große Plättchen mit einer Höhe von einem Zentimeter. Diese bestehen im Wesentlichen aus einem LCD-Bildschirm, einem Funkchip, einem 3-Achsen Bewegungssensor, einem Mikrocontroller

Abbildung 3.25: Siftable Aufbau und vier Infrarot (IR) - Transceiver (s. a. Abb. 3.25) und (Quelle: Merrill u. a. (2007)) werden mit einer Batterie betrieben.

Siftables erkennen die Bewegungen, die mit ihnen ausgeführt werden und tauschen diese Daten per Funknetzwerk untereinander aus. Dadurch lassen sich mit mehreren Siftables Gesten im physischen Raum durchführen, wodurch sich ebenfalls der Zustand des Gesamtsystems aller Siftables oder nur einer Auswahl von Siftables ändern kann. Aus dieser Form der Interaktion heraus prägten die Entwickler den Begriff des „Sensor Netzwerk User Interface“ (SNUI).



Abbildung 3.26: Beispielanwendung (Quelle: Merrill u. a. (2007))

An jeder der vier Seiten eines Siftables sind IR - Transceiver angebracht, deren Reichweite auf einen Zentimeter begrenzt ist. Mit ihnen lassen sich direkt benachbarte Siftables erkennen und identifizieren. Auf diese Weise können unter den direkten Nachbarn koordinierte Funktionen ausgeführt werden. Abbildung 3.26 zeigt als Beispiel für eine solche Funktion, wie man mit Siftables rechnen kann.

Microsoft Research „Synchronous Gestures“

Hinckley stellt ein Konzept für die Ausführung von synchronen Gesten für mobile Geräte vor (Hinckley, 2003b). Dabei werden an die Geräte eine Platine mit zwei-Achsen-Beschleunigungssensoren, Berührungssensoren, Abstandssensoren und Lichtsensoren angebracht (s. Abb. 3.27). Die Sensoren der jeweiligen Geräte bauen über ein IEEE 802.11¹⁵ Funknetzwerk eine Verbindung zueinander auf. Über diese Verbindung werden die durch die Platine erfassten Daten zusammen mit einem Zeitstempel übertragen. Stößt man zwei Geräte aneinander, so kann mittels der Daten erkannt werden, mit welcher Seite der jeweiligen Geräte diese zusammengestoßen wurden. Auf diese Weise konnte der darstellende Bereich eines Gerätes durch den darstellenden Bereich eines anderen Gerätes erweitert werden.

In Hinckley (2003a) erweitert Hinckley die beschriebenen Prototypen so, dass durch die Verbindung von zwei Geräten ein kollaborativer Arbeitsplatz über diese beiden Geräte entsteht. Auf diese Weise entsteht ein „öffentlicher“ Arbeitsraum. In diesem können an beiden Tablets Änderungen an Daten vorgenommen werden, welche dann sofort mit dem anderen Tablet synchronisiert werden. Die Synchronisation der Tablets erfolgt per Funk über die IEEE 802.11 Netzwerkschnittstelle der Geräte.

Möchte ein Benutzer Daten seines Tablets nicht mit den Daten des anderen Tablets synchronisiert haben, so kann er sich einen „privaten“ Arbeitsbereich schaffen. Dies geschieht, indem der Benutzer seine Hand in die Nähe des Randes seines Tablets legt. Dies wird durch die Touch- und IR - Sensoren erkannt und ein kleiner Bereich des Bildschirms am Rande

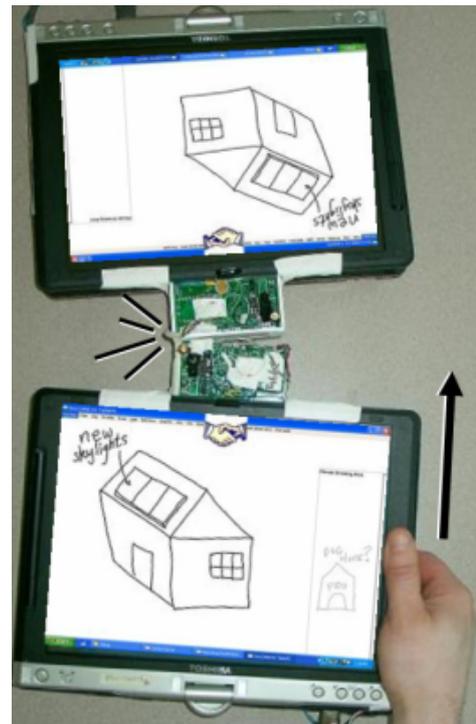


Abbildung 3.27: Microsoft Prototyp
(Quelle: Hinckley (2003b))

¹⁵802.11 ist eine von dem Institute of Electrical and Electronics Engineers (IEEE) herausgegebene Norm für die Kommunikation in Funknetzwerken.

des öffentlichen Bereiches wird als privater Arbeitsbereich eingerichtet. Die Daten dieses Arbeitsbereiches werden nicht mit dem anderen Tablet synchronisiert.

3.3.3 Umsetzung

Die beiden in Kapitel 3.3.2 vorgestellten Projekte nutzen unterschiedliche Techniken, um Nachbargeräte zu erkennen und Daten auszutauschen. Siftables verwenden auf kurzer Distanz IR - Transceiver, um direkte Partner zu erkennen und zu identifizieren. Der Austausch von Anwendungsdaten erfolgt hingegen über ein Funknetzwerk, welches zwischen den einzelnen Siftables aufgebaut wird. Das Microsoft Research Projekt verwendet Bewegungssensoren, um eine Zusammenlegung von zwei Geräten zu erkennen. Die Anwendungsdaten werden bei diesem Projekt über ein IEEE 802.11 Funknetzwerk gesendet.

Der Ansatz des zweiten vorgestellten Projektes hat den Nachteil, das bei dieser Form der Nachbarschaftserkennung diese nur mit zwei Geräten durchgeführt werden kann. Bei dem Siftable - Projekt können zwar beliebig viele Geräte miteinander kooperieren, allerdings verwenden Siftables ein proprietäres Format für die Nachbarschaftserkennung per IR - Transceiver und die drahtlose Kommunikation.

Bei der technischen Konzeption des SITabs wurde deswegen ein Weg gewählt, der Ansätze aus beiden Projekten verbindet. Für die Nachbarschaftserkennung und -identifizierung sollen IR - Transceiver verwendet werden. Da die Mehrzahl der mobilen Geräte, wie Smartphone oder Tablets, über eine IEEE 802.11 kompatible Netzwerkschnittstelle verfügen, sollen die Anwendungsdaten über diese Schnittstelle gesendet werden. Die dazu nötigen Verbindungsinformationen sollen über die IR - Schnittstelle ausgetauscht werden.

Damit unterschiedliche Modelle von mobilen Geräten verwendet werden können, sollen die nötigen Sensoren in einem Rahmen untergebracht werden können, der in unterschiedlichen Formen hergestellt werden kann. Ein solcher Rahmen enthält in allen vier Seiten eingebettete, nach außen gerichtete IR - Transceiver. Da der überwiegende Teil aller Smartphones und Tablets über einen USB - Schnittstelle verfügt, ist es sinnvoll, die Kommunikation mit dem Rahmen über diese Schnittstelle zu implementieren.

Prototyp

Zur Überprüfung der technischen Machbarkeit wurde ein Prototyp erstellt. Dieser wurde mit der im Kapitel 3.2.3 vorgestellten Arduino Plattform realisiert. Neben der Stromversorgung und einer USB - Schnittstelle zur Kommunikation mit dem Gerät, wurde für die Erkennung von Nachbargeräten für jede Geräteseite ein IR - Transceiver vorgesehen (s. a. Abb. 3.28).

Der Prototyp wurde mit einem Steckboard und darauf aufgesteckten Bauteilen realisiert:

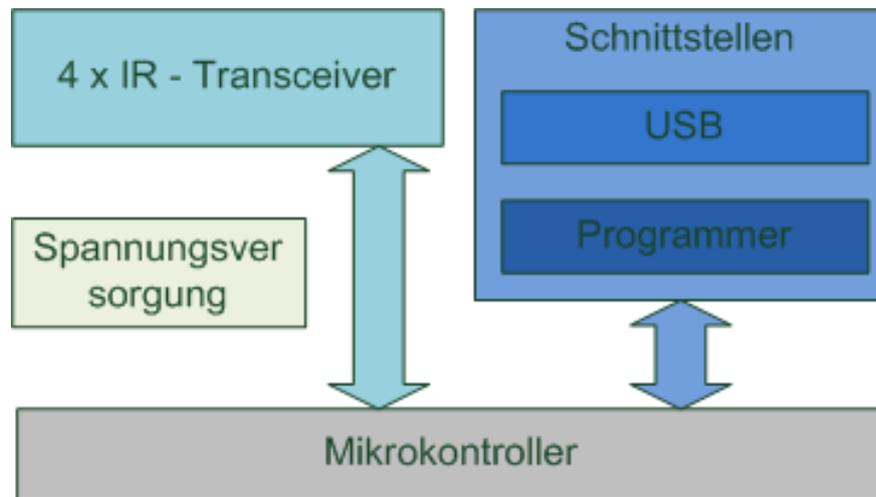


Abbildung 3.28: Blockschaltbild

Arduino Pro Mini

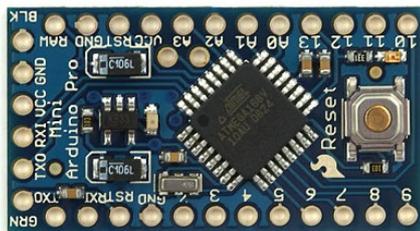


Abbildung 3.29: Arduino Pro Mini
(Quelle: [Arduino](#))

Als Mikrokontroller - Plattform wurde ein Arduino Pro Mini eingesetzt (s. Abb. 3.29). Diese Version der Arduino Plattform (vgl. Kapitel 3.2.3) ist mit einem Atmega328P als Mikrokontroller bestückt und wird mit einer Versorgungsspannung von 3,3 Volt und einer Taktfrequenz von acht Megahertz betrieben. Zum Aufbringen auf das Board wurden auf der Unterseite des Arduino Steckkontakte gelötet.

FTDI USB zu Seriell Wandler

De Arduino Pro Mini verfügt selbst nicht über eine USB - Schnittstelle. Um per USB mit dem Arduino kommunizieren zu können, wird eine FTDI FT232RL USB zu Seriell Wandler von [Sparkfun](#) verwendet (s. Abb. 3.30). Über diesen können neue Programmcodes auf den Arduino geladen und Anwendungsdaten vom Arduino empfangen und zum Arduino gesendet werden. Gleichzeitig dient der Wandler auch als Stromversorgung für den Prototypen.



Abbildung 3.30: USB zu Seriell Wandler (Quelle: [Sparkfun](#))

MAX3100

Der MAX3100 ist ein Universal Asynchronous Receiver Transmitter (UART) IC der Firma [Maxim](#), der über eine SPI mit dem Arduino kommuniziert. Beim Prototypen wird der MAX3100 als Treiber und Buffer für die IR - Transceiver verwendet. Dafür bietet dieser einen IrDA¹⁶ kompatiblen Modus an, in dem mit einer Geschwindigkeit von 115200 Kilobaud (kBd) Daten über einen IR - Transceiver gesendet werden können. Die Send- und Empfangsdaten werden in einem FIFO¹⁷ - Buffer zwischengespeichert, bis sie entweder gesendet werden können oder vom Mikrokontroller abgerufen werden. Um den MAX3100 als Treiber für die Infrarot Transceiver benutzen zu können, muss ein Quarz mit 3.6864 MHz an den IC angeschlossen werden.

TFBS6711

Als IR - Schnittstelle kommt der TFBS6711 Transceiver der Firma [Vishay](#) zum Einsatz. Dieser ist mit den Ausmaßen von sechs Millimeter Breite, drei Millimeter Tiefe und zwei Millimeter Höhe sehr klein und kann deswegen gut in die Rahmen verbaut werden. Für die Testzwecke wurde eine einfache Platine erstellt, die auf die Steckplatine gesteckt oder per Buchsenleiste angeschlossen werden konnte (s. Abb. 3.31).



Abbildung 3.31: IRDA Transceiver

GL5528

Um zu erkennen, ob sich ein Gerät auf einer Seite eines Tablets befindet und um die IR - Transceiver für die Nachbarschaftserkennung zu aktivieren, wurde der Fotowiderstand GL5528 verwendet. Der Fotowiderstand hat je nach einfallender Lichtstärke unterschiedliche Widerstandswerte, welche vom Arduino gemessen werden können. Der Einsatz des Fotowiderstandes ist notwendig, damit nur Geräte erkannt werden, die unmittelbar nebeneinander liegen. Mit den IR - Transceiver alleine ließ sich dies nicht bewerkstelligen, da die Sendeleistung nicht so genau eingestellt werden konnten.

In Abbildung 3.32 ist der Prototyp, bestehend aus Steckplatine und den beschriebenen Komponenten zu sehen. Nicht zu sehen sind die Adapterplatinen mit den IR - Transceiver. Man erkennt jedoch die vier Steckkontakte (im oberen Drittel der Steckplatine), an denen die Adapterplatinen angeschlossen werden können.

¹⁶IrDA steht für Infrared Data Association und ist eine Vereinigung zur Standardisierung von Infrarot - Transceivern und Protokollspezifikationen.

¹⁷FIFO ist eine Abkürzung für den englischen Begriff First In - First Out. Damit wird ein Verfahren zur Speicherung von Daten bezeichnet, bei dem die Elemente, die zuerst gespeichert wurden, auch zuerst wieder aus dem Speicher entnommen werden.

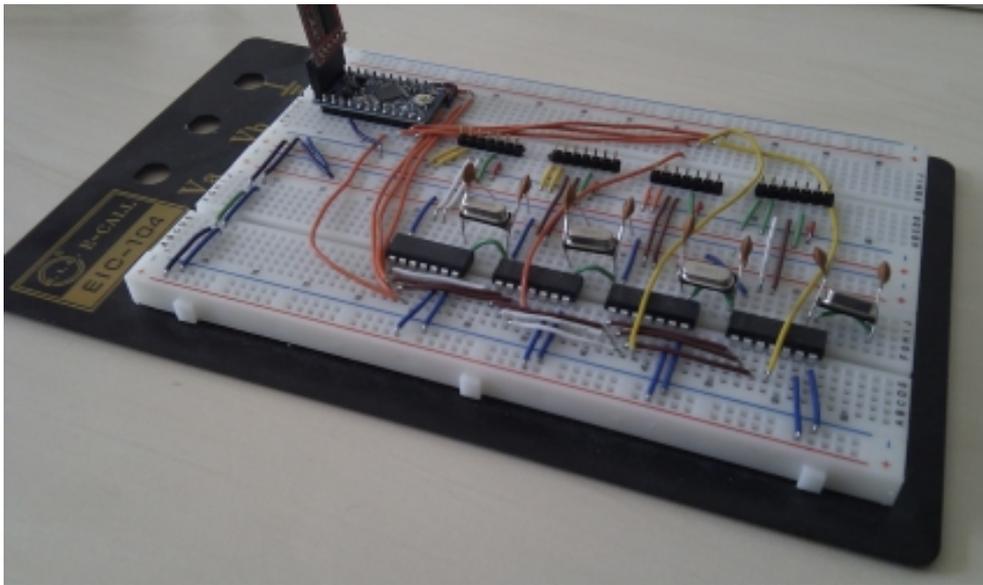


Abbildung 3.32: erster Prototyp

Software

Abbildung 3.33 zeigt ein Flussdiagramm der Treibersoftware für den Prototypen. Während des Setups werden ein Identifikator (ID) und die Internetprotokoll (IP) - Adresse des Gerätes konfiguriert. Danach werden periodisch die Werte der Fotowiderstände gemessen. Verändern sich diese Werte absolut über einen Schwellenwert hinaus, so wird anhand der Polarität festgelegt, ob sich ein potentiell Nachbargerät an dieser Geräteseite befindet. Danach werden auf den Seiten, auf denen ein potentiell Nachbargerät erkannt wurde, die FIFO-Buffer der MAX3100 abgefragt. Wenn eine gültige Geräte - ID erkannt wurde, so wird diese als für diese Seite erkanntes Geräte erfasst. Unabhängig davon wird bei jeder der abgefragten Seiten auch die eigene ID und IP gesendet. Nachdem alle Seiten abgearbeitet wurden, wird die Liste mit den eingetragenen IDs und IPs per USB - Schnittstelle an das Host - Gerät gesendet.

Zur Kommunikation mit dem Hostgerät über die USB - Schnittstelle wird genau wie beim Hamburg Cubical (s. Kap. 3.2) JSON verwendet. Abbildung 3.34 zeigt den Aufbau einer JSON Nachricht. Es werden für alle Kanten jeweils die Werte der ID und die IP der Nachbargeräte übertragen.

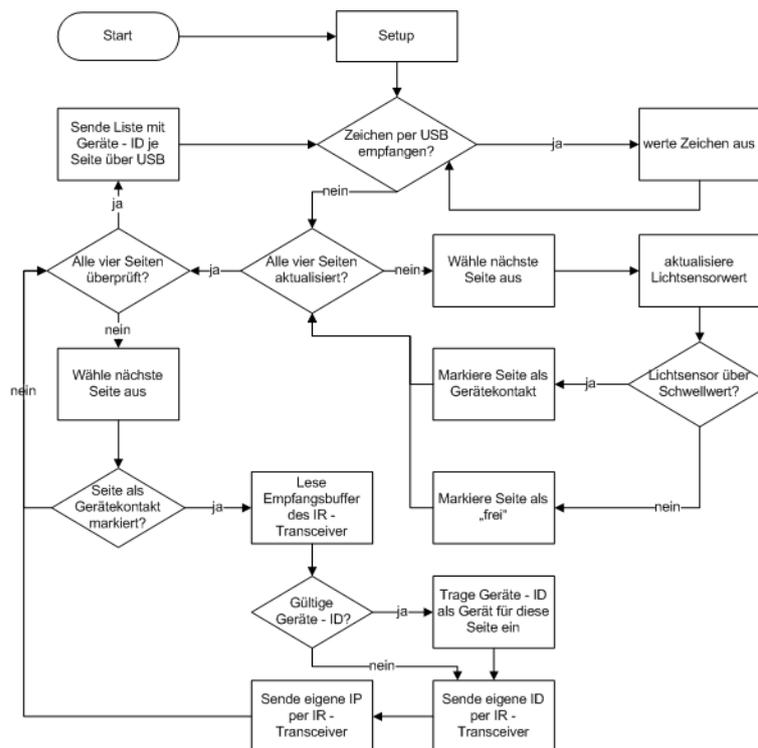


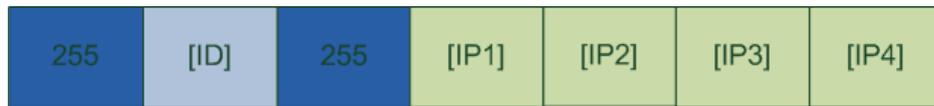
Abbildung 3.33: Treibersoftware



Abbildung 3.34: SITab JSON Format

Für die Kommunikation der ID und IP Adresse über die IR - Schnittstelle wird ein einfacheres Protokoll gewählt. Es werden sieben Byte übertragen (s. Abb. 3.35), wobei das erste und dritte Byte immer dem Wert 255 entsprechen. Das zweite Byte entspricht der ID und die vier letzten Bytes entsprechen den vier Tupeln der IPv4¹⁸ Adresse des Hostgerätes.

¹⁸Ist eine Notation von IP Adressen, die aus vier Zahlen, die jeweils zwischen 0 und 255 liegen und mit einem Punkt getrennt sind, besteht.



[ID] entspricht dem Wert der ID des Gerätes
 [IP1], [IP2], [IP3] und [IP4] entsprechen den Tupeln der IPv4 Adresse des Gerätes

Abbildung 3.35: SITab IR Datenformat

Verfeinerung des Prototypen

Um die Interaktion mit einem Tablet testen zu können, wurde parallel zur Entwicklung der beschriebenen Software ein Layout für eine PCB - Platine erstellt. Mit der Platine konnte ein Prototyp für einen Rahmen erstellt werden, der zu Testzwecken an zwei Viliv X70 Tablets montiert wurde (s. Abb. 3.41).

Um die für den ersten Prototypen entwickelten Treiber wiederverwenden zu können, wurde darauf geachtet, dass die Bauteile mit Arduino kompatibel sind:

Mikrokontroller

Als Mikrokontroller kam der bereits im Kapitel 3.2.4 beschriebene Atmega328P der Firma [ATMEL](#) (in derselben Konfiguration) zum Einsatz.

Stromversorgung

Die Stromversorgung erfolgt über die USB - Schnittstelle. Die Spannung wird durch einen LDO Spannungsregler auf 3,3 Volt Gleichspannung geregelt.

MAX3100

Die im vorherigen Abschnitt beschriebenen MAX3100 kommen auch bei dem verfeinerten Prototypen zum Einsatz. Damit der Prototyp möglichst in einem kleinen Formfaktor hergestellt werden kann, wurden die ICs in SMD¹⁹ Bauform verwendet.

Drei-Achsen-Beschleunigungssensoren

Als Drei-Achsen-Beschleunigungssensor wurde der ADXL345 von Analog Devices verwendet. Dieser Sensor - IC wird über das I²C²⁰ - Interface an den Atmega328P angeschlossen. Über diese Schnittstelle können die Beschleunigungswerte für die x-, y- und z-Achse des Sensors ausgelesen werden. Durch diesen Sensor ist es möglich

¹⁹SMD ist die Abkürzung für „surface-mounted device“ und bedeutet „oberflächenmontierbares Bauelement“. Dabei handelt es sich um Bauelemente, die mittels lötfähiger Anschlussflächen direkt auf eine Leiterplatte gelötet werden.

²⁰I²C steht für Inter-Integrated Circuit. Es handelt sich hierbei um einen seriellen Datenbus.

zu erkennen welches der Nachbargeräte zuletzt bewegt wurde. Dies ist wichtig, wenn z. B. automatisch entschieden werden soll, welcher Bildbereich auf welchem Gerät angezeigt werden soll.

Schnittstellen

Neben der XBee - Funkschnittstelle verfügt der Prototyp über die im Kapitel 3.2.4 vorgestellten Schnittstellen.

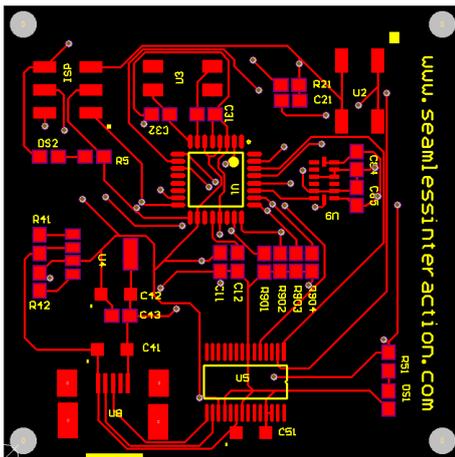


Abbildung 3.36: Platine (Oberseite)

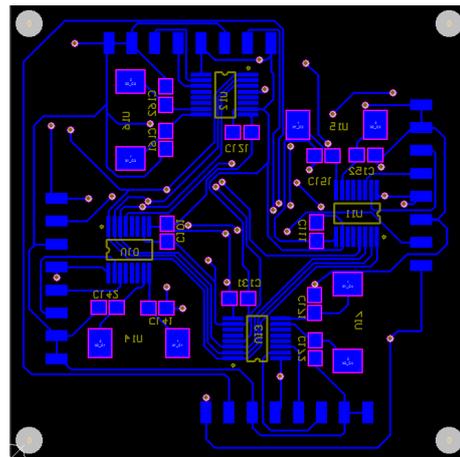


Abbildung 3.37: Platine (Unterseite)

Mit den oben beschriebenen Komponenten wurde ein zweiseitiges Layout für eine PCB Platine erstellt. Auf der Oberseite (s. Abb. 3.36) sind ein Atmega328P (U1), die USB - Schnittstelle (U8) mit dem FTDI - Treiber (U5) und ein Drei-Achsen-Beschleunigungssensor (U9) platziert. Auf der Unterseite der Platine (s. Abb. 3.37) sind die Infrarot - Treiber (U10, U11, U12 und U13) samt Quarze (U14, U15, U16 und U17) und Steckverbinder zu den Infrarot-Transceivern (jeweils acht Pads in Reihe am Rand der Platine) zu erkennen.

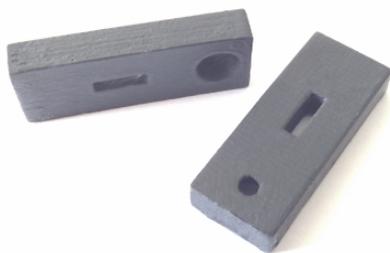


Abbildung 3.38: Blende für Infrarot-Transceiver

Die erstellte Platine wurde in einem Gehäuse Prototypen eingepasst. Dazu wurde aus mehreren Schichten Pappe ein Unterbau für das Viliv X70 erstellt (s. Abb. 3.39). Dieser Unterbau kann direkt unter dem X70 befestigt werden und somit einen Rahmen simulieren. Um die Infrarot-Transceiver in den Prototypen einpassen zu können, wurde im Rapid Prototyping - Verfahren eine Blende gedruckt. Diese Blende ist 35 Millimeter breit, vier Millimeter tief und zehn Millimeter hoch und besitzt Aussparungen für die Infrarot-LED und -Diode des Transceivers, sowie für den Lichtsensor (vgl. Abb. 3.38).

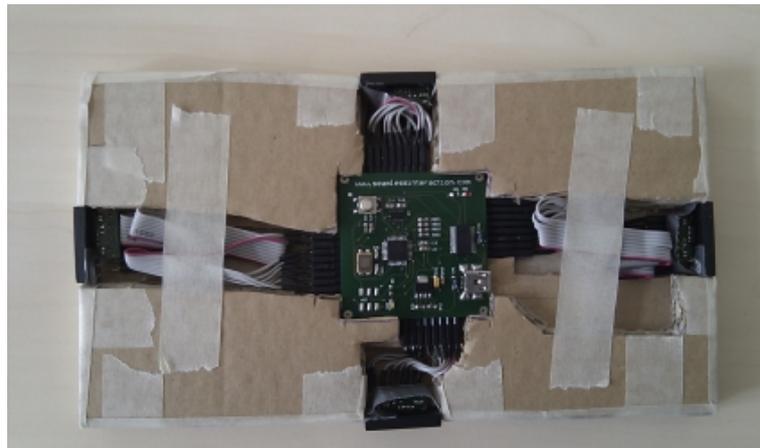


Abbildung 3.39: Prototypischer Rahmen

Testapplikation

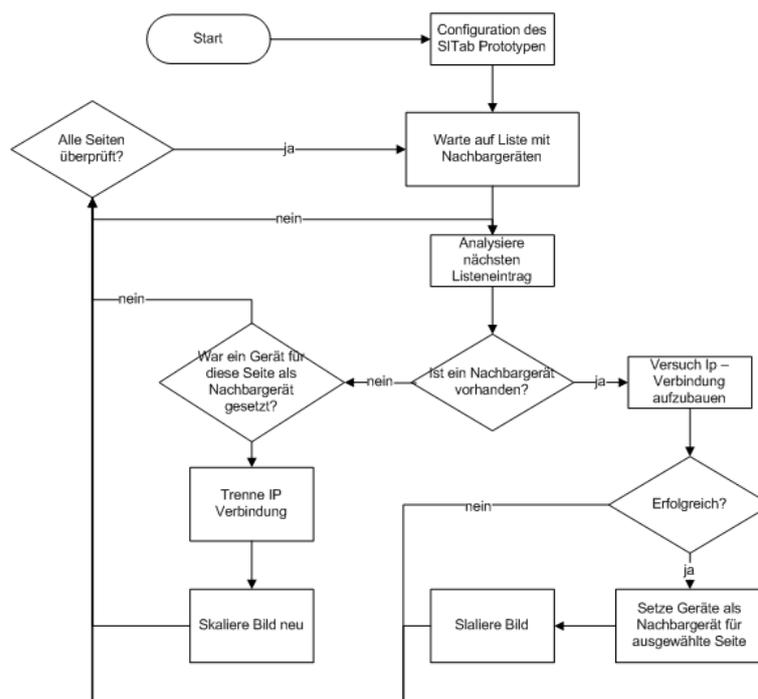


Abbildung 3.40: Flussdiagramm Testanwendung

Es wurde eine Testapplikation implementiert, die das in Kapitel 3.3.1 vorgestellte Anwendungsszenario „Skalieren von Bilddaten“ umsetzt. Die Anwendung wurde in der Programmiersprache C# umgesetzt. In Abbildung 3.40 sieht man ein Flussdiagramm der Anwendung. Man kann erkennen, dass die periodisch empfangenen Daten des SITab Prototypen

nach Empfang analysiert werden. Dabei wird die Liste mit den Nachbargeräten sequentiell durchlaufen und überprüft, ob ein Nachbargerät erkannt wurde. Wenn ein Nachbar erkannt ist und noch keine IP - Verbindung zu diesem besteht, so wird versucht ein IP - Verbindung herzustellen. Über diese Verbindung können dann die Bilddaten ausgetauscht und die Skalierung des Bildes auf den einzelnen Geräten ausgehandelt werden. Wenn keine Nachbargerät auf eine Geräteseite erkannt wird, allerdings noch eine IP - Verbindung zu einem Gerät, das vorher als Nachbargerät erkannt wurde, besteht, dann wird diese Verbindung abgebaut und die Skalierung des Bildes neu berechnet.

Die so erstellte Anwendung wurde auf beiden Tablets gestartet (s. Abb. 3.41) . Die Interaktion mit der Anwendung zeigte, dass sich das vorgestellte Anwendungsszenario und das Interaktionskonzept mit dem Prototypen grundsätzlich umsetzen lassen.



Abbildung 3.41: Anwendungstest „Skalieren von Bildern“

3.3.4 Zusammenfassung

Mit dem beschriebenen SITab Konzept können mobile Geräte wie Smartphones und Tablets erweitert werden. Dadurch ist es möglich, dass sie Geräte in ihrer unmittelbaren Nachbarschaft erkennen und mit ihnen kommunizieren. Auf diese Weise können Geräte selbst als Eingabe verwendet werden. Dadurch ist es möglich, durch Gesten und die physische Positionierung der Tablets Daten auszutauschen und so z.B. Bilddaten auf mehreren Geräten verteilt anzuzeigen.

Im Laufe des Experimentes wurden zwei Prototypen erstellt. Der erste Prototyp diente der Verifizierung der technischen Machbarkeit und bestand aus einer Steckplatine auf der ein Arduino Pro Mini, sowie die MAX3100 ICs als Treiber für IR - Transceiver aufgesteckt wurden. Mit dem zweiten Prototyp sollte die Umsetzung des Interaktionskonzeptes nachgewiesen

werden. Dazu wurde eine zweiseitige PCB - Platine erstellt und mit den nötigen Komponenten bestückt. Diese Platine wurde in einen prototypischen Rahmen eingebettet und an einem Tablet befestigt. Über eine USB - Verbindung konnte das Tablet Daten von der Platine erhalten und war so in der Lage direkte Nachbarn zu erkennen und zu identifizieren.

Für einen ersten Test wurde eine Anwendung erstellt, mit der die Bilddaten eines Tablets automatisch auf eine anderes Tablet skaliert angezeigt werden konnten.

In dem Projekt konnte die grundsätzliche technische und konzeptionelle Machbarkeit der beschriebenen Interaktionsform bestätigt werden. Um den Nutzen und die Funktionsweise des Konzeptes ausführlich analysieren zu können, müssen allerdings noch weitere Tests vorgenommen werden. Vor allem Benutzertests mit Nutzern außerhalb der Entwicklergruppe fehlen bisher. Um diese Tests durchführen zu können, muss der Funktionsumfang der Testanwendung erweitert und mehr Testgeräte konstruiert werden. Dies wird auch der nächste Schritt in der Entwicklung sein.

Neben der Erweiterung der Anwendungssoftware ist ein weiterer Schritt die Verfeinerung des Rahmens. Hierzu ist es notwendig, einen verbesserten Rahmen zu konzipieren und in unterschiedlichen Formen zu erstellen. Dadurch können andere Testgeräte als das Viliv X70 mit dem Rahmen ausgestattet werden und es können z. B. Geräte mit unterschiedlich großen Displays miteinander getestet werden. Die Anordnung der IR - Transceiver im Rahmen kann verändert werden. Der jetzige Aufbau ist empirisch als der für die ersten Tests am besten geeignete ermittelt worden. Eine andere Anordnung oder Anzahl von IR - Transceivern kann aber zu einer Verbesserung der Nachbargeräteerkennung oder neuen Formen der Interaktion führen.

3.4 Fazit

In diesem Kapitel wurden zwei Experimente im Bereich Tangible Interaction vorgestellt, die im Kontext des Living Place Hamburg entstanden sind. Sie dienten dazu, mit dem im Kapitel 2 vorgestellten Tangible Interaktion Konzept zu experimentieren und diese Interaktionsform zu analysieren.

Beide Projekte sind empirisch entstanden, um die technische Umsetzbarkeit einer Idee auszuprobieren. Es wurde weder zu Beginn der Projekte eine Vorgehensweise festgelegt, noch wurde sich bei der Entwicklung an eine Vorgehensweise orientiert. Bei beiden Projekten wurde mit der iterativen Verbesserung von Prototypen gearbeitet. Mit den neuen Prototypen wurden entweder neue technische Komponenten und Funktionen getestet oder bereits bestehende Funktionen verbessert und stabiler implementiert.

Bei den Projekten zeigte sich, dass bei der Entwicklung von Tangible Interaction Artefakten eine Vielzahl von Herausforderungen auftreten:

- **Design von Hardware und Software**

Eine Herausforderung beim Entwickeln von TI - Elementen ist der fehlende Standard der Hardware für Sensoren, Aktoren und Kontrolleinheiten. Ein Grund hierfür ist vor allem die Eigenschaft, dass es sich bei TI - Artefakten um spezialisierte, auf die Ausführung von bestimmten Aufgaben und Funktionen zugeschnittene, Geräte handelt. Aus diesem Grund kommen unterschiedliche Techniken und Technologien zum Einsatz, von der jede eine spezialisierte Hardware und speziell auf diese Hardware und Funktion zugeschnittene Programmierung benötigt. Dies macht die Entwicklung sehr kosten- und zeitintensiv und erfordert viele Eigenentwicklungen von Hard- und Software, was wiederum zu Problemen bei der Integration von einmal erstellten Komponenten in neue Projekte führt.

- **Erwartete Persistenz**

Bei der längeren und anhaltenden Benutzung von TI - Systemen erwartet der Benutzer, dass diese Systeme und die darunterliegenden Daten persistent sind. Bei der Benutzung des Hamburg Cubicals zur Einstellung der Helligkeit einer Lampe möchte der Benutzer, dass diese sich nicht ändert, auch wenn der Benutzer kurzfristig den Würfel auf eine andere Seite dreht, um z. B. die Musik leiser zu stellen. Das häufig verwendete Sense-Act-Modell²¹ und der auf Events basierende Softwareaufbau (Shaver und Jacob, 2009) kann kontinuierliche Benutzerinteraktionen jedoch nicht abbilden und erfassen.

- **Handbuchfreie Bedienung**

Eine Herausforderung, die besonders auch im Bereich Smart Homes und bei der Akzeptanz von neuen Systemen wichtig ist, ist die Notwendigkeit, Elemente möglichst ohne das vorherige Lesen eines Handbuches verwenden zu können. Gerade im Fall eines notwendigen Eingriffs in das Verhalten und die Funktionsweise einer intelligenten Wohnung (s. Kapitel 3.1) ist es oft unmöglich und akzeptanzhemmend, erst eine Bedienungsanleitung lesen zu müssen, bevor man eine Aktion ausführen kann. Häufig spricht man in diesem Zusammenhang auch von der Notwendigkeit, Elemente und Funktionen intuitiv bedienen zu können.

- **Kontextabhängige Interaktion**

Das Verhalten von physischen Objekten wird nicht nur durch ihre physikalischen Eigenschaften sondern auch durch den Kontext, in dem eine Handlung statt findet, bestimmt.

²¹Mit Sense-Act-Modell ist ein Vorgehen gemeint, bei dem Werte von Sensoren ermittelt werden. Bei der Messung von vorher definierten Werten oder Wertebereichen wird automatisch auf einer vorher definierten Art und Weise reagiert und festgelegte Aktionen werden ausgelöst.

So kann sich beispielsweise Verhalten und Funktionen ändern, wenn zwei Objekt verbunden, bzw. in eine andere Umgebung gebracht wird. Aus diesem Grund ist es nötig, das Verhalten für jeden möglichen Nutzerkontext zu definieren. Der Entwickler muss dem Benutzer über das Objekt vermitteln, welche Aktionen in welchem Kontext von Bedeutung sind.

- **Parallelität**

Die Parallelität der Eingaben eines Benutzers oder die Möglichkeit der gleichzeitigen Benutzung von Tangible Artefakten eines Systems durch mehrere Benutzer stellt eine große Herausforderung dar. Da mehrere Zugangspunkte zu einem System vorhanden sein können (vgl. Kapitel [2.4](#)), müssen Möglichkeiten der räumlichen und zeitlichen Koordination von Aktionen bedacht werden.

4 Designmethodik

Viele der bisherigen Entwicklungen im Bereich TI sind, genau wie die im vorherigen Kapitel beschriebenen, empirisch entstanden. Hintergrund waren häufig das Experimentieren mit neuen Techniken und das Ausprobieren von technisch Machbarem (Shaer und Hornecker, 2009). Aus den experimentellen Erfahrungen heraus wird in diesem Kapitel ein strukturierter Ansatz für die Entwicklung von TI aufgezeigt, mit dem sich die im vorherigen Kapitel beschriebenen Herausforderungen (s. Kapitel 3.4) bewältigen lassen.

Am Anfang dieses Kapitels werden Designmethoden vorgestellt, die sich mit ähnlichen Thematiken auseinandersetzen und die den hier beschriebenen Entwicklungsprozess mit geprägt haben. Im Anschluss folgt die Übersicht über den Entwicklungsansatz und eine detaillierte Beschreibung der einzelnen Arbeitsschritte und Vorgehensweisen.

4.1 Designmethoden

4.1.1 Interaction Design

„In the next fifty years, the increasing importance of designing spaces for human communication and interaction will lead to expansion in those aspects of computing that are focused on people, rather than machinery. The methods, skills, and techniques concerning these human aspects are generally foreign to those of mainstream computer science, and it is likely that they will detach (at least partially) from their historical roots to create a new field of ,interaction design.“ Winograd (1997)

Interaction Design ist eine Teildisziplin der Informatik, die sich mit der Beziehung zwischen Benutzern und interaktiven Produkten beschäftigt. Wie Terry Winograd in dem vorangegangenen Zitat beschreibt, stellt Interaction Design die Benutzer und Benutzerszenarien in den Mittelpunkt der Entwicklung. In der Literatur gibt es eine Vielzahl von unterschiedlichen Definitionen von Interaction Design. Jonas Löwgren geht in seinem Artikel von zwei unterschiedlichen Ansichtsweisen des Interaction Design aus, die in der Forschung und praktischen Anwendung zusammenfließen (Löwgren). Auf der einen Seite spricht Löwgren von der Sicht der Interaction Design als Design Disziplin, auf der anderen Seite als eine Erweiterung von

Mensch-Maschinen Interaktion oder Human-Computer Interaction (HCI). HCI ist eine Disziplin, deren Ursprung bereits auf das Jahr 1960 zurückgeht und die sich mit unterschiedlichen Themen aus Ingenieurwissenschaften und Verhaltensforschung, sowie deren Zusammenwirken beschäftigt (ACM). In der Informatik liegt der Fokus auf der Interaktion zwischen Mensch und Computer. Nach Löwgren liegt das Hauptaugenmerk von HCI auf den instrumentellen Qualitäten, wie Benutzbarkeit (engl. Usability) und Nutzen von digitalen Produkten und Diensten. Dabei werden überwiegend berufsbedingte oder aufgabenorientierte Gebrauchssituationen mit dem Fokus auf individuelle Benutzer und deren Absichten betrachtet. Fragen, wie ein Mensch Informationen aufnimmt und wie er diese verarbeitet, nehmen eine zentrale Rolle ein. Bei Interaction Design als Erweiterung von HCI, werden aus der HCI-Forschung gewonnene Erkenntnisse und Werkzeuge bereits im Designprozess mit eingebunden, anstatt nach dem Designentwurf auf Usability-Probleme hinzuweisen. Hinzu kommt, dass die Benutzung von digitalen Produkten und Diensten sich stark verändert hat. Von einem reinen arbeitsrelevanten Einsatz erfolgt ein immer stärker werdender Wandel hin zu einem Einsatz in der Informations- und Unterhaltungsindustrie. Dadurch fließen in die Betrachtungen immer mehr nicht technische, ästhetische und ethische Qualitäten mit ein und werden zu gleichen Teilen betrachtet.

4.1.2 User-Centered Design

Beim User-Centered Design befindet sich der Benutzer von der Analyse der Benutzeranforderungen bis hin zum Testen und Evaluieren im Mittelpunkt des Entwicklungsprozesses. User-Centered Design lässt sich durch drei Merkmale charakterisieren (Rogers und Sharp, 2002, S. 285):

- **Fokus auf Benutzer und deren Aufgaben**

Dieses Merkmal beinhaltet das Wissen über das Spektrum an Benutzern und die Aufgaben der Benutzer. Dieses Wissen wird durch Beobachtungen des Benutzers in seinem täglichen Umfeld beim Erledigen von Aufgaben, dem Studieren der Aufgaben und seiner Handlungsabläufe sowie der Einbeziehung des Benutzers in den Entwicklungsprozess gewonnen.

- **Empirische Untersuchungen**

Empirische Untersuchungen dienen schon früh in der Entwicklung dazu, die Leistungsfähigkeit und Reaktion von Benutzern zu erfassen. Im späteren Entwicklungsprozess nutzt man sie zum Beurteilen der Leistungsfähigkeit interaktiver Prototypen im Zusammenspiel mit den Benutzern.

- **Iterative Entwicklung**

Diese Art der Entwicklung dient dazu, bei Benutzertests gefundene Probleme zu beheben und mit neuen Prototypen die Lösungen dieser Probleme neu zu evaluieren.

Auf diese Weise wiederholt sich ein Zyklus von „Entwicklung - Tests - Auswertung - Neuentwicklung“.

Dieses Einbeziehen der Benutzer in die Entwicklung hat den Vorteil, dass die Erwartungen an ein neues Produkt realistisch sind. Lässt man die Benutzer mit einem Produkt arbeiten bevor es fertig ist, so verstehen sie eher was das Produkt leisten kann. Desweiteren betrachten sich die Benutzer als Mitbesitzer des Produktes, wodurch seine Akzeptanz steigt.

Zur Evaluation und zum Testen der Benutzbarkeit und Funktionsweise des Produktes durch den Benutzer kommt bei User-Centered Design das Usability Engineering zum Einsatz (Aarts und Marzano, 2003, S. 42). Diese Form der Untersuchungen dient der Analyse von Benutzeranforderungen, dem Testen der Verwendbarkeit von Konzepten, sowie der Implementierung und Evaluation interaktiver Systeme.

Participatory Design

Participatory Design ist eine Form des User Centered Design, welches den Benutzer in alle Phasen des Designprozesses mit einbezieht (Rogers und Sharp, 2002, S. 306). Dabei werden die Benutzer als Partner verstanden und arbeiten mit den Designern/Entwicklern in Teams zusammen. Der Vorteil bei dieser Zusammenarbeit ist, dass der Benutzer den Kontext, in dem das neue System verwendet werden soll, am besten kennt. Der Entwickler oder Designer hat die Aufgabe, ein breiteres Spektrum an Möglichkeiten und neue, dem Benutzer unbekanntes Ideen mit in die Kooperation einzubringen.

4.1.3 Ethnographie und Contextual Design

Ethnographie ist eine Methode, die ursprünglich aus der Anthropologie stammt. Anthropologie legt seinen Schwerpunkt auf ein detailliertes Verständnis einer Kultur. Dies soll durch intensive und langzeitliche Beobachtung der Menschen geschehen, bei der der Beobachter in die zu untersuchende Kultur eintaucht (Dourish, 2001, S. 57-60). Dabei geht es nicht nur um die Beobachtung, sondern auch das Betrachten von Aktionen aus der Sicht des Beobachteten und das tiefe Verständnis, warum und in welchem Kontext Aktionen durchgeführt werden und was der Beobachtete dabei erlebt.

Obwohl die Ethnographie aus der Anthropologie stammt, ist es auch eine Basis für Feldversuche in der Soziologie und Psychologie. In diesen Feldern wird Ethnographie dazu verwendet die soziale Organisation von Aktivitäten zu untersuchen. Das Ziel hierbei ist es, eine Abfolge und Ordnung in den Aktivitäten zu bestimmen (Rogers und Sharp, 2002, S. 288).

Contextual Design

Contextual Design integriert die Erfahrungen und Interpretationen aus ethnografischen Feldversuchen in das Design von softwarebasierten Produkten. Designer und Entwickler beobachten Personen beim Ausführen ihrer Tätigkeiten und führen Interviews, sog. Contextual Inquiry, durch. Diese unterscheiden sich durch ihren geringeren Umfang und der geringeren Intensität von ethnografischen Feldversuchen. Die so gewonnenen Erkenntnisse sollen direkt in die Formulierung und Artikulation von funktionalen Anforderungen (vgl. Kapitel 4.2.1) für das zu entwickelnde System führen. Die gemachten Beobachtungen werden in unterschiedliche Perspektiven und dazugehörige Modelle gebündelt:

- **workflowmodel**

Im Ablaufmodell werden die in den Arbeitsprozessen beteiligten Personen und die Kommunikation und Koordination unter diesen Personen aufgelistet.

- **sequence model**

Das Abfolgmodell listet die detaillierten Arbeitsschritte und Abläufe zur Erreichung eines Zieles auf. Desweiteren werden auch die Auslöser für die einzelnen Schritte festgehalten.

- **artifact model**

In diesem Modell werden die benötigten oder hergestellten Werkzeuge und Hilfsmittel aufgezeigt. Von jedem Werkzeug wird ein annotiertes Bild oder eine annotierte Zeichnung angefertigt.

- **cultural model**

Das kulturelle Modell zeigt die Einschränkungen eines Systems aufgrund von Organisations- oder Firmenkulturen auf. Dieses Modell soll die Haupteinflussfaktoren auf die Arbeit aufzeigen. Hierzu können Gruppen, Rangfolgen oder Kleiderordnungen gehören.

- **physical model**

Dieses Modell zeigt die physische Struktur einer Arbeit oder eines Arbeitsumfeldes. Dazu gehören ein physischer Plan der Arbeitsumgebung, sowie Kommunikationsinfrastruktur und Verknüpfung von Arbeitsmitteln. Das Modell beinhaltet physikalische Eigenschaften und Einschränkungen der einzelnen Komponenten.

Das explizite Ziel dieser Technik ist es, das Wissen über eine aktuelle Situation in Wissen über eine zukünftige Situation mit dem neuen System zu verwandeln (Steen u. a., 2007).

4.1.4 Analytisches Design

„Design is analytic when it is broken down into its elemental components, with each of these components being analyzed individually.“ [Edge und Blackwell \(2009\)](#)

Der von Edge und Blackwell beschriebene analytische Designansatz für TUIs hat seinen Ursprung im rationalen Design Prozess ([Edge und Blackwell, 2009](#)). Hier wird Design als ein logischer, rationaler Weg vom Problem zu Lösung betrachtet. Der Prozess kann als eine rationale, progressive Verfeinerung von einem Designkonzept hin zu einem bedeutungsvollen Design beschrieben werden. Dabei werden vier Abschnitte durchlaufen:

- **Context analysis** (engl. für Kontextanalyse):
Die Kontextanalyse identifiziert die Aktivitäten, die von einer TUI unterstützt werden können. In diesem Prozess findet eine Verfeinerung des Designkonzepts in unterschiedliche Designmöglichkeiten statt. Bei diesem Prozess werden vier Aspekte betrachtet:
 - Der *strukturelle Kontext* betrachtet die Verteilung von Aktivitäten zwischen Benutzern, Artefakten und einem räumlichen Bereich.
 - Der *prozedurale Kontext* zeigt, wie Aktivitäten über die Zeit eingeleitet, koordiniert und komplettiert werden.
 - Der *kognitive Kontext* ermittelt die kognitiven Anforderungen von Aktivitäten im Vergleich zu den kognitiven Möglichkeiten der Benutzer.
 - Der *soziale Kontext* betrachtet die sozialen Anforderungen von Aktivitäten.
- **Activity analysis** (engl. für Aktivitätenanalyse):
Die Aktivitätenanalyse untersucht wie die Eigenschaften der TUI, die die (in der Kontextanalyse) identifizierten Aktivitäten unterstützen können. Hier werden unterschiedliche Fragestellung und Eigenschaften, wie z. B. inwieweit eine dreidimensionale Form benötigt wird, inwieweit die physische Form die digitale Form der Daten wiedergeben muss oder wie anfällig das System bei einer physischen Lageänderung der TUI-Elemente ist, untersucht.
- **Mapping analysis** (engl. für Abbildungsanalyse):
In der Abbildungsanalyse wird die Abbildung vom digitalen in den physischen Raum und damit ein strukturelles Design generiert. Hier gibt es drei unterschiedliche Arten der Abbildungen, die untersucht werden. Die Erste ist das sog. *spatial mapping*. Diese Abbildung betrachtet, in welcher Art und Weise die physische Konfiguration von Objekten durch einen Computer interpretiert wird. Das *action mapping* betrachtet im Bezug

auf die Direktheit und Kompatibilität die Auswirkungen von physischen Aktionen auf digitale Daten, während das *attribute mapping* den Zusammenhang von physikalischen Eigenschaften mit den Eigenschaften von digitalen Informationen untersucht.

- **Meaning analysis** (engl. für Bedeutungsanalyse):

Bei der Bedeutungsanalyse wird das strukturelle Design in ein bedeutungsvolles Design gewandelt. Hier wird die physisch-digitale Abbildung mit Bedeutung gefüllt, die der Benutzer verstehen und adaptieren kann. Dabei werden vier Punkte untersucht:

- Die *visuelle Bedeutung* zeigt, inwieweit die äußere Erscheinung eines TUI Objektes die möglichen Aktionen anzeigt.
- Die *haptische Bedeutung* beschreibt, wie die haptische Erfahrung beim Benutzer der TUI Objekte die Durchführbarkeit der Aktionen unterstützt.
- Die *funktionale Bedeutung* bezeichnet, inwieweit die physische Form jedes Objektes die konzeptionelle Funktion zeigt.
- Die *relationale Bedeutung* meint, die Art und Weise, in der die räumliche Konfiguration die konzeptionelle Funktion andeutet.

4.2 Design von Tangible Interaction

Aus den im ersten Teil dieses Kapitels beschriebenen Designansätzen und den im Kapitel 3 gemachten Erfahrungen wird in diesem Abschnitt eine Herangehensweise für das Design von Tangible Interaction Elementen entwickelt. Dabei handelt es sich um einen iterativen Zyklus von Design, Evaluation und Redesign, in dem Prototyping eine zentrale Rolle einnimmt (s. Abb. 4.1).

Der Einsatz von Prototypen in allen Bereichen des Prozesses erlaubt es, schnell unterschiedliche Designrichtungen zu testen und den Benutzer früh in den Entwicklungsprozess einzubinden. Dies ist vor allem bei der Entwicklung von Tangible Interaction Artefakten wichtig, da hier die Form einen Einfluss auf die Funktion und Interaktion mit dem Artefakt hat (s. a. Kap. 4.2.4). So erlauben es sog. Affordances (s. Kap. 4.2.4), den Benutzer zu der korrekten Benutzung des Artefaktes zu verleiten. Dadurch kann ein Entwickler dem Benutzer kommunizieren, welche Aktion in welchem Kontext sinnvoll ist. Da dies allerdings sehr stark vom Benutzerkontext und den Möglichkeiten und Einschränkungen in diesem Kontext abhängt, ist es notwendig, die Designansätze bereits frühzeitig im Benutzerkontext zu untersuchen. Hierzu ist der Einsatz von Prototypen unerlässlich.

Zu Beginn der Entwicklung werden in der Anforderungsanalyse die Kernpunkte ermittelt. Diese müssen klar und verständlich formuliert werden. Unter der Verwendung von einfachen

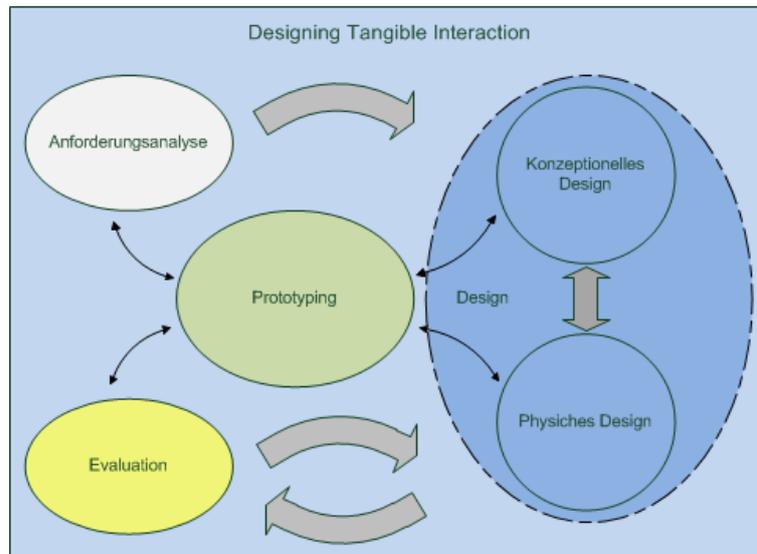


Abbildung 4.1: Tangible Interaction Design

Prototypen können hier Szenarien und Anwendungsfälle definiert werden. Die Anforderungsanalyse ist kein für sich abgeschlossener Bereich, bei dem die ermittelten Anforderungen fix sind. Es kann sein, dass sich bei der Evaluation des Designs zeigt, dass die Anforderungen nicht vollständig oder sogar falsch ermittelt wurden und dass es notwendig wird, diese zu korrigieren.

Die Designphase lässt sich in die Abschnitte konzeptionelles und physisches Design unterteilen. Diese sind jedoch nicht klar voneinander zu trennen und beeinflussen sich gegenseitig. Während der konzeptionellen Designphase werden die Anforderungen analysiert. Aufgrund dieser Analyse können Interaktionsmodelle und -metaphern entwickelt werden. Dabei kommt es zu einer Verfeinerung der Anforderungen zu unterschiedlichen Designmöglichkeiten. Diese Designmöglichkeiten werden im physischen Design zu einem greifbarem Design weiterentwickelt. Dabei handelt es sich um die Umsetzung der unterschiedlichen Ansätze zu Prototypen. Mit diesen kann u. a. untersucht werden, wie die Abbildung vom physischen in den digitalen Raum vorgenommen werden kann, welche Materialien für die Implementation geeignet sind, welche Form die Interaktion und Funktion unterstützt und welche Technologien und Techniken verwendet werden können.

Die Evaluation spielt in allen vorgestellten Bereichen eine große Rolle. Sie erlaubt es die Korrektheit der Anforderung, die Verwendbarkeit von erstellten Modellen oder die Eignung von Entwürfen zu überprüfen. Zu diesem Zweck können unter Verwendung der Prototypen unterschiedliche Evaluationsparadigmen angewendet werden. Die Erkenntnisse aus der Evaluation können und sollen sofort wieder in die Verfeinerung und Weiterentwicklung des Designs und der verwendeten Prototypen verwendet werden.

In den folgenden Abschnitten werden die für die einzelnen Abschnitte und die jeweils nötigen Techniken genauer erläutert.

4.2.1 Anforderungsanalyse

Die Anforderungsanalyse steht am Anfang eines TI - Entwicklungsprozesses (vgl. Abbildung 4.2). Anforderungen beschreiben, wie ein TI - Element aussehen und welche Funktionen es besitzen soll. Die Aufgabe dieser Analyse ist es, die Anforderungen so klar, präzise und eindeutig wie möglich darzustellen. Im folgenden Abschnitt werden die für die Entwicklung von TI - Artefakten zu beschreibenden Anforderungen vorgestellt. Um diese Anforderungen ermitteln zu können, müssen Daten gesammelt, ausgewertet und übersichtlich formuliert werden.

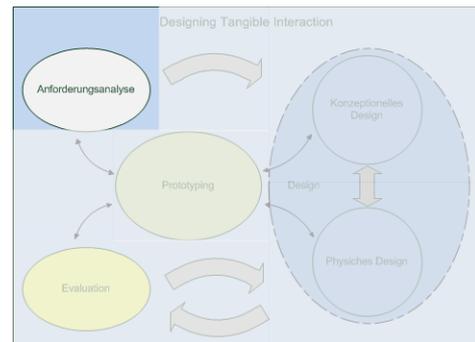


Abbildung 4.2: TI - Design: Anforderungsanalyse

Arten der Anforderungen

Traditionell werden Anforderungen in funktionale und nichtfunktionale Anforderungen gegliedert. Während funktionale Anforderungen beschreiben, welchen Funktionsumfang ein System haben soll, beschreiben nicht-funktionale Anforderung qualitative und quantitative Eigenschaften eines Systems. Bei der Entwicklung von TI Elementen sind die Ermittlung von bestimmten nichtfunktionalen Anforderungen sehr wichtig:

- **Kontext**

Kontextanforderungen beschreiben, in welchem Kontext / Umfeld Objekte benutzt werden sollen. Das schließt sowohl die physische Umgebung und deren Einschränkungen als auch die Einschränkungen für den Benutzer ein. Desweiteren soll hier das soziale Umfeld des TI Systems betrachtet werden und ob es durch potentiell einen oder mehrere Benutzer verwendet werden soll.

- **Benutzeranforderungen**

Unter Benutzeranforderung wird festgelegt, wie die Benutzergruppe zusammengesetzt ist. Handelt es sich um eine dem Leistungsvermögen nach homogene Benutzergruppe oder muss von unterschiedlichen Graden der Expertise der Benutzer ausgegangen werden? Gleichzeitig soll die Häufigkeit der Verwendung der Elemente hier beschrieben werden.

- **Datenanforderungen**

Die Datenanforderungen sollen den Datenumfang und die Art der Daten beschreiben. Hierbei ist es wichtig den Datentyp, die Genauigkeit und die Festlegung der Persistenz zu charakterisieren.

- **Benutzbarkeit**

Die Benutzbarkeit wird durch die vorher beschriebenen Punkte stark beeinflusst. Bei diesen Anforderungen müssen Punkte wie die geforderte Effektivität, die Effizienz, Sicherheit und Erlernbarkeit beschrieben werden. Eine Charakterisierung der Verwendung des Systems als Werkzeug oder im Zusammenhang mit anderen Systemen findet hier statt.

Um die beschriebenen Anforderungen ermitteln zu können, bieten sich unterschiedliche Verfahren, wie Fragebögen, Interviews, Dokumentationen, Workshops oder (Ethnografische) Feldversuche an. Diese Verfahren unterscheiden sich stark im Aufwand, dem Detailgrad der ermittelten Daten, dem Risiko, falsche Daten zu ermitteln, dem Umfang der Kenntnisse über den Ablauf von Aufgaben und dem kognitiven Prozess bei der Ausführung (Rogers und Sharp, 2002, S. 211ff). Eine generelle Aussage, welche dieser Methoden in welchem Umfang verwendet werden sollen, kann hier nicht getroffen werden, da dies von Entwicklung zu Entwicklung variiert.

Aufbereitung

Die ermittelten Anforderungen müssen für die Verwendung in der Designphase aufbereitet werden. Da diese immer im Kontext mit der Benutzung gesehen werden sollten, bietet es sich an, die Anforderungen durch Szenarien und Anwendungsfälle zu verdeutlichen.

Szenarien und Anwendungsfälle sind Beschreibungen, die sich mit der Sicht der Benutzer auf das System befassen. Szenarien beschreiben die Aktivitäten oder Aufgaben von Benutzern. Dabei dienen sie der Beschreibung

- des Kontextes, in dem ein System verwendet werden soll,
- der Anforderungen,
- Bedürfnisse des Benutzers bei der Verwendung des Systems.

Sie helfen ein Verständnis dafür zu entwickeln, was Anwender für Aktivitäten ausführen und warum sie das tun. Dadurch lassen sich mit Szenarien die Anwendergruppe bestimmen. Anwendungsfälle fokussieren im Gegensatz zu Szenarien die Interaktion zwischen Benutzer und System. Sie sind mit einem Benutzer verknüpft und helfen die Ziele des Benutzers bei der Verwendung des Systems zu erfassen.

4.2.2 Prototyping

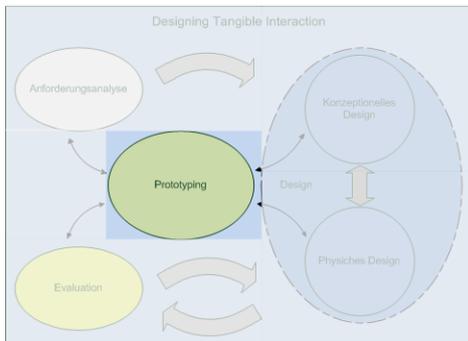


Abbildung 4.3: TI - Design: Prototyping

Ein Prototyp ist eine konkrete Umsetzung eines Bestandteils oder eines gesamten Systems. Je nach Disziplin werden unterschiedliche Bedeutungen mit dem Begriff Prototyp assoziiert. In der Informatik ist ein Prototyp oft eine Teilimplementierung eines Softwaresystems, in der bestimmte Techniken und Prozesse evaluiert werden. Im Zusammenhang mit der Entwicklung von Tangible Interaction sind Prototypen eine physisch greifbare Hypothese über die Funktion und Gestalt von Tangible Interaction Elementen. Diese Tangibles können sowohl mit Funktionen versehene Objekte als auch Objekte ohne Funktion sein und

in Originalgröße oder skaliert implementiert werden.

Es gibt unterschiedliche Strategien, um mittels Prototypen Funktionen, Hypothesen über Interaktionen, Techniken und Technologien untersuchen und evaluieren zu können. [Beaudouin-Lafon und Mackay \(2008\)](#) unterscheidet hier zwischen „horizontalen“, „vertikalen“, „aufgabenorientierten“ und „szenariobasierten“ Prototypen. Diese Strategien unterscheiden sich sowohl um Umfang des Prototypen, als auch in der Sichtweise auf den Prototypen. Diese Strategien können auf unterschiedlicher Art und Weise implementiert werden, wobei man die unterschiedlichen Ansätze in „offline“ und „online“ Prototypen gruppieren kann.

Offline Prototyping

Unter Offline Prototypen fasst man Prototypen zusammen, die keine Software oder computergestützte Rechenleistung benötigen. Prototypen dieser Art werden oft in den Anfängen des Entwicklungsprozesses verwendet und sind meist einfach und günstig herzustellen:

- **Diagramme und Sketche**

Diagramme und Sketche sind die schnellste Form des Prototyping. Nur mit einem Stift und Papier lassen sich schnell unterschiedliche Design- und Interaktionsmöglichkeiten skizzieren. Der Nachteil hierbei ist jedoch, dass sich vor allem Interaktionen und Auswahlmöglichkeiten nur schlecht darstellen lassen. Dennoch ist diese Form des Prototyping vor allem bei der Kommunikation von Ideen und der Ideenfindung sehr effektiv.

- **Storyboards**

Storyboards dienen zur Präzisierung einer Idee. Mit ihnen lassen sich Was-Wenn-Szenarien erstellen und innerhalb eines Entwicklerteams und zu externen Personen

kommunizieren. Sie dienen dazu, Auswahlmöglichkeiten und unterschiedliches Systemverhalten aufzuzeigen und den Kontext, in dem etwas benutzt werden soll, zu vermitteln.

- **Mock ups**

Mock ups sind physische Prototypen ohne Funktionen. Sie bieten einen Überblick über das physische Design, wie z. B. wo ein Knopf sitzt und wie die Ausmaße des Objektes sind. Mock ups können zum besseren Verständnis und zum Test der eingesetzten Materialien und als Vergleichsobjekte von unterschiedlichen physischen Aufbauten dienen. Dadurch können Affordances (s. a. Kapitel 4.2.4) getestet und Aufschlüsse über die Handhabbarkeit gewonnen werden.

- **Wizard of Oz**

Um Benutzer mit einem System arbeiten zu lassen, bevor es fertig implementiert ist, wird ein Wizard of Oz Prototyp verwendet. Diese Art des Prototypen, dessen Name aus einem gleichnamigen Film von 1939 stammt, lässt Benutzer mit einem teilweise funktionierenden System arbeiten. Die noch nicht oder teilweise implementierten Funktionen werden von einem Entwickler im Hintergrund so gesteuert, wie es das System machen würde. Desweiteren kann der Entwickler oft auch Funktionen des Systems überschreiben oder abändern. Auf diese Weise lassen sich frühzeitig Benutzertests in einem einsatznahen Kontext durchführen.

Online Prototyping

Diese Art der Prototypen benötigen Software und computergestützte Rechenleistung. Sie werden verwendet, um Funktionsweisen zu erproben und zu evaluieren. Online Prototypen können sowohl interaktiv, als auch nicht-interaktiv sein. Bei nicht-interaktiven Systemen geht es vor allem darum, bestimmte Funktionsabläufe und physisches Verhalten zu testen. Interaktive Prototypen hingegen eignen sich besonders für die Evaluierung mit Benutzern.

Bei Prototypen von Tangible Interaktion Artefakten werden oft Mikrocontroller und unterschiedliche Sensoren und Aktoren benötigt. Das Verbauen von solchen Elementen und das Programmieren ist jedoch sehr kostspielig und zeitintensiv. Erschwerend kommt hinzu, dass die erstellten Prototypen nur schwer wiederverwendet, umgebaut oder weiterentwickelt werden können. Eine Abhilfe bieten hier sog. Physical Computing Plattformen und Prototyping Toolkits (Gregor, 2009). Mit diesen Plattformen können schnell Prototypen erstellt, erweitert und verändert werden. Desweiteren können vorherige Entwicklungen übernommen und einfacher portiert werden. Dadurch werden die Schwierigkeiten und Hemmschwellen bei der Entwicklung von voll funktionsfähigen Prototypen gemindert (Shaer und Hornecker, 2009). Im Kapitel 3 wurde bereits Arduino als eine dieser Plattformen vorgestellt. Sie hat

sich bei den beschriebenen Experimenten für die Entwicklung von Prototypen als gut geeignet herausgestellt.

4.2.3 Konzeptionelles Design

Beim konzeptionellen Design wird der erste Schritt von der Anforderung hin zu einem Designansatz vorgenommen (s. a. Abb. 4.4). Dabei geht es vor allem darum, das Verhalten des Produktes bei der Interaktion mit dem Benutzer festzulegen und unterschiedliche Ideen und Entwicklungsansätze zu testen. Um der in Kapitel 3.4 beschriebene Herausforderung „Handbuchfreie Bedienung“ bewältigen zu können, ist es wichtig, intuitiv bedienbare Artefakte zu entwickeln. Bei der dazu notwendigen Entwicklung von intuitiven Interaktionskonzepten ist der Einsatz von mentalen Modellen und Metaphern hilfreich.

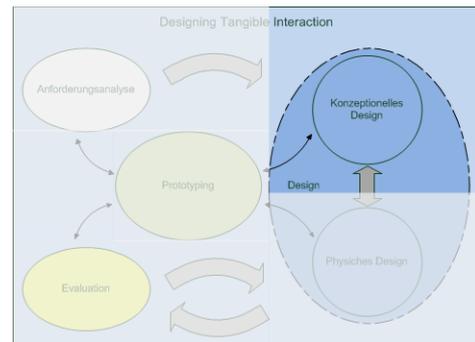


Abbildung 4.4: TI - Design: konzeptionelles Design

Mentale Modelle

Der Begriff „Mentales Modell“ wurde von [Johnson-Laird \(1980\)](#) eingeführt. Das Konzept von internen Modellen wurde aber bereits 1943 von Craik in „The Nature of Explanation“ vorgestellt ([Staggers und Norcio, 1993](#)). In diesem beschreibt er, dass Menschen externe Events in interne Modelle umwandeln, welche durch Interaktionen und externe Eindrücke erstellt werden. [Staggers und Norcio \(1993\)](#) beschreiben die weite Verbreitung von mentalen Modellen in der HCI. Es existieren unterschiedliche Begriffe, wie konzeptionelles Modell, mentales Modell, kognitives Modell, Komponentenmodell oder kausales Modell, um den Themenbereich „Mentales Modell“ zu beschreiben. Norman präzisiert den Begriff „Mentale Modelle“. In [Norman \(1988\)](#) definiert er drei unterschiedliche Modelle eines zu beschreibenden Objekts:

- **Konzeptionelles Modell**
Das konzeptionelle Modell beschreibt, wie Designer ein Objekt begrifflich erfassen.
- **Benutzermodell oder mentales Modell**
Das Benutzermodell wird vom Benutzer entwickelt und beschreibt wie die Funktionen eines Objektes ablaufen.
- **Systemabbild**
Unter Systemabbild versteht Norman die physische Umsetzung des konzeptionellen Modells, die reale Funktionsweise des Objektes.

In dieser Auflistung entspricht das Benutzermodell dem mentalen Modell des Benutzers, welches von diesem auf Basis des Systemabbildes erstellt wird. Es basiert wiederum auf dem konzeptionellen Modell. Damit ist das Systemabbild entscheidend bei der Kommunikation zwischen Designer und Benutzer. Die beiden stimmen jedoch nicht unbedingt überein, da die konzeptionellen Modelle eine exakte, konsistente und komplette Beschreibung eines Objektes durch den Designer oder Entwickler ist, während das mentale Modell bei der Anwendung durch den Benutzer von diesem erstellt wird.

Menschen benutzen mentale Modelle,

- um die Funktion eines neuen Systems oder Objektes zu erlernen,
- um Probleme beim Umgang mit einem System oder Objekt zu lösen,
- wenn ein Benutzer ein System reflektiert, rationalisiert oder seine Funktionsweise zu erklären versucht.

Zu diesem Zweck werden laut [Carroll und Reitman Olsen \(1987\)](#) drei unterschiedliche mentale Modelle verwendet:

- **„Surrogates“**
Surrogates sind konzeptionelle Analysen, in denen das Ein- und Ausgabeverhalten eines Zielsystems nachgebildet wird. Dabei wird nicht versucht, den Verarbeitungsweg des Zielsystems zu übernehmen. Da die internen Prozesse des mentalen Modells nicht mit den internen Prozessen des Zielsystems isomorph sind, ist es zwar geeignet, sobald ein Nutzer eine neue Situation verstehen will, auf der andere Seite aber ungeeignet, wenn der Benutzer das Verhalten eines Systems verstehen will.
- **„Metaphor“**
Ein metaphorisches Modell ist ein direkter Vergleich zwischen einem Zielsystem und einem Ausgangssystem, welches der Benutzer bereits kennt. Diese Art der Modelle sind einfacher zu konstruieren und zu erlernen als Surrogates, weil sie Erklärungen zu der Funktionsweise eines Systems enthalten. Es ist allerdings schwer zu sagen, welche Metaphern ein Benutzer verwendet. So muss ein Benutzer Stahl kennen, wenn er den Ausdruck „Es ist stahlhart.“ verstehen will. Desweiteren sind domainübergreifende Metaphern sehr selten, so dass ein domainübergreifender Austausch dieser Art von Modellen schwierig ist.
- **„Glass boxes“**
Glass box bezeichnet ein Modell, das Elemente der beiden zuvor beschriebenen Modelle enthält. Es ist dem Surrogates - Modell in dem Maß ähnlich, als dass dieses Modell ein perfekter Mime des Zielsystems ist. Dem metaphorischen Modell ähnelt es, in dem es semantische Interpretationen zulässt. Glass boxes sind mentale Strukturen, die nicht durch den Benutzer entdeckt, sondern ihm beigebracht werden.

Alle hier beschriebenen Modelle haben jedoch Nachteile gemein ([Staggers und Norcio, 1993](#)):

- Mentale Modelle sind nicht komplett und akkurat.
- Die Verarbeitungsmöglichkeiten von mentalen Modellen sind bei Menschen eingeschränkt.
- Modelle sind nicht persistent.
- Mentale Modelle haben keine festgelegten Grenzen, so dass ähnliche Objekte und Funktionen verwechselt werden können.
- Mentale Modelle sind nicht wissenschaftlich in dem Sinn, dass Funktionsweisen ohne Belege als gegeben hingenommen werden.
- Mentale Modelle sind nicht unbedingt effektiv. Oft führen Benutzer zusätzliche physische Aktionen aus, die bei einer genauen Analyse des Modells vermieden hätten werden können, weil sie dadurch die Komplexität der mentalen Verarbeitung reduzieren können.

Bei der Verwendung, der Beschreibung und Analyse von mentalen Modellen muss man sich dieser Probleme bewusst sein.

Metapher

„Metaphors become natural models that allow us to take familiar, concrete objects and experiences and re-cast them onto unknown or abstract concepts or things, giving them structure and meaning“ [Erickson \(1995\)](#)

Wie man aus dem Anfangszitat erkennen kann, sind Metaphern natürliche, kognitive Modelle, die es Menschen erlauben, bekannte Objekte und Erfahrungen auf unbekannte oder abstrakte Konzepte oder Dinge zu projizieren, um ihnen Bedeutung und Struktur zu geben. [Saffer \(2005\)](#) beschreibt eine Metapher als ein sprachliches, visuelles oder auditives Konstrukt, in dem eine Sache (die Quelle) auf eine andere Sache (das Ziel) verweist und Aspekte dieser Sache repräsentiert. Das Begriffssystem des Menschen ist grundlegend metaphorisch. Demnach basieren menschliche Aktionen auf Metaphern, da sie zum Schlussfolgern benutzt werden. Auch [Fishkin](#) beschreibt, dass Menschen Metaphern benutzen, um die Welt begrifflich zu fassen ([Fishkin, 2004](#)). Abstrakte Konzepte wie Zeit, Raum, Zustände, Veränderungen, Kausalität oder Aktionen sind alle metaphorischen Ursprungs.

In der HCI und dem Interaction Design ist der Einsatz von Metaphern weit verbreitet. Sowohl in der Forschung, der Entwicklung als auch bei der Konzeption von UIs werden Metaphern

eingesetzt ([Frøkjær und Hornbæk, 2002](#)). Saffer beschreibt, wie Metaphern bei der Entwicklung und Konzeption helfen, Probleme neu zu definieren sowie als Recherche - Werkzeug, Inspirationsquelle und Kommunikationsmittel dienen können ([Saffer, 2005](#)).

Bei Tangible Interaction Elementen helfen Metaphern vor allem den Benutzern, neue Interaktionen oder Interaktionsstile zu entdecken und sind besonders geeignet bei der Einführung von neuen Interaktionskonzepten. Dabei entstehen beim Einsatz von Metaphern in diesem Zusammenhang auch Gefahren. Wenn die von den Entwicklern vorgesehene Metapher dem Benutzer nicht verständlich vermittelt wird, so bildet dieser eigene Metaphern, die nicht unbedingt adäquat und geeignet sein müssen.

Intuitivität

In vielen Konzepten und Projekten der HCI wird Intuitivität oder intuitive Interaktion als eines der Hauptziele genannt ([Hurienne und Israel, 2007](#)). Intuitive Interaktion mit einem Benutzerinterface oder Gerät stellt eine wichtige Voraussetzung für die handbuchfreie Benutzung dar. Dies ist vor allem bei der Entwicklung von Tangible Interaction wichtig, weil es hier im Gegensatz zu reinen Softwaresystemen oft nur schwer möglich ist, dem Benutzer Hilfetexte oder ähnliches anzubieten.

Für den Begriff an sich gibt es keine allgemeingültige Definition, im Allgemeinen wird Intuition oder intuitives Verhalten jedoch mit einer unterbewussten Sicherheit bei Handlungen oder der Handhabung von unbekanntem Gegenständen assoziiert. Blackler hat unterschiedliche Definitionen und Arbeiten untersucht und beschreibt, dass alle Definition in Basisbestandteilen übereinstimmen ([Blackler, 2006](#)). So beruht Intuition hauptsächlich auf empirischem Wissen. Es ist ein Prozess, der von Personen nicht wahrgenommen wird und der im Unterbewusstsein abläuft. Der beschriebene Prozess ist dabei wesentlich schneller als die übrigen kognitiven Prozesse.

Ist eine Situation neu, arbeiten Menschen auf einem wissensbasiertem Level. Es handelt sich dabei um einen analytischen Prozess, bei dem konzeptionelles Wissen verwendet wird ([Blackler u. a., 2002](#)). Während dieser Überlegungen müssen Menschen eine Menge von intuitiven Hinweisen beachten, welche das Aufrufen von erfahrungsbasierten Regeln aus dem Gedächtnis veranlassen. Daraus kann man schließen, dass Menschen nur intuitiv handeln können, wenn sie Erfahrungen gemacht haben, auf denen sie ihre Überlegungen aufbauen können.

Ein solches auf Erfahrungen basierendes Wissen teilen [Hurienne und Israel \(2007\)](#) in vier unterschiedliche Kategorien ein:

- **Angeborenes Wissen**

bezeichnet durch die Aktivierung von Genen oder durch vorgeburtliche Entwicklungen

erlerntes Wissen. Vor allem Reflexe und instinktives Verhalten fallen in diese Kategorie.

- **Sensormotorisches Wissen**

besteht als allgemeinem Wissen, das in sehr früher Kindheit erworben wird und von diesem Zeitpunkt an immer wieder bei der Interaktion mit der physischen Welt verwendet wird. Auf Wissen dieser Kategorie basieren wissenschaftliche Begriffe wie Affordances und Bild-Schemata.

- **Kulturelles Wissen**

basiert auf kulturellen Gegebenheiten und ist dementsprechend von Kultur zu Kultur unterschiedlich.

- **Expertenwissen**

spiegelt das spezielle Wissen wieder, was Menschen sich bei der Ausführung ihres Berufes aneignen.

In jedem dieser Bereiche sammelt und speichert ein Mensch laut Hurlienne unterschiedlichstes Wissen über Werkzeuge ab. Im sensormotorischen Bereich werden vor allem Wissen über primitive Werkzeuge wie Steine, Stäbe oder Stöcker verankert, während man im kulturellen Bereich vor allem Werkzeuge, die allgemein verwendet werden, wie Telefone oder Lampen, findet. Im expertisen Bereich finden sich hingegen Werkzeuge die man in seinem Spezial- oder Expertengebiet verwendet. Wichtig dabei ist zu erkennen, dass Personen aus dem gleichen Spezialgebiet Wissen über unterschiedliche Werkzeuge haben können.

Bei der Entwicklung von handbuchfreien und intuitiv zu bedienenden Benutzerschnittstellen und TI Artefakten ist es wichtig, die bisher beschriebenen Basiseigenschaften von intuitivem Handeln zu beachten. Aus diesem Grund beschreibt [Blackler u. a. \(2003\)](#) folgende Kriterien für die Entwicklung und das Design von intuitiv verwendbaren Geräten:

- Verwendung von familiären Symbolen / Wörtern für bekannte Funktionen,
- Verdeutlichung der Funktionsweise von weniger bekannten Funktionen,
- Verwendung eines konsistenten Mappings zwischen Gerät und Funktion,
- ausreichend groß skalierte Knöpfe und Bedienelemente, die klar ersichtlich und mit ausreichendem Freiraum zwischen ihnen angebracht sind.

Diese Punkte sollen bei der Entwicklung von intuitiv zu bedienenden Geräten und Benutzerschnittstellen als Anhaltspunkte dienen.

4.2.4 Physisches Design

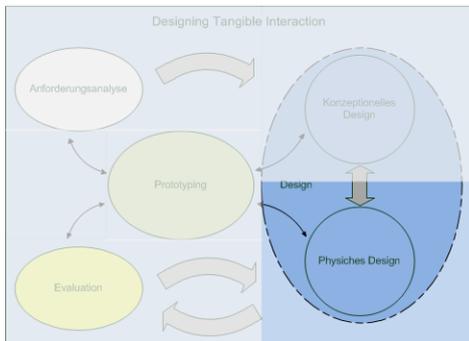


Abbildung 4.5: TI - Design: physisches Design

Frens (2006) beschreibt, dass die Funktion, die Form und die Interaktion als die drei Haupteigenschaften von interaktiven Produkten angesehen werden können. Bei anderen Interaktionsformen spielen die Wechselwirkungen zwischen diesen drei Eigenschaften keine große Rolle. Dies lässt sich am Beispiel eines Terminals zum Kauf von Bahntickets gut veranschaulichen. In Bezug auf die Interaktion und Funktion spielt die Form eine eher untergeordnete Rolle. Die Interaktion mit dem Terminal wird über Buttons realisiert und die Funktion über Software abgebildet.

Alle diese Komponenten können verändert werden, ohne dass eine andere Komponente im großen Maße beeinflusst wird. Bei Tangible Interaction hingegen stehen diese Eigenschaften in Wechselwirkung zueinander (s. Abb. 4.6). Mit den Auswirkungen dieser Wechselwirkungen und ihrem Einsatz in der Entwicklung beschäftigt sich der Abschnitt „physisches Design“ (vgl. Abb. 4.5).

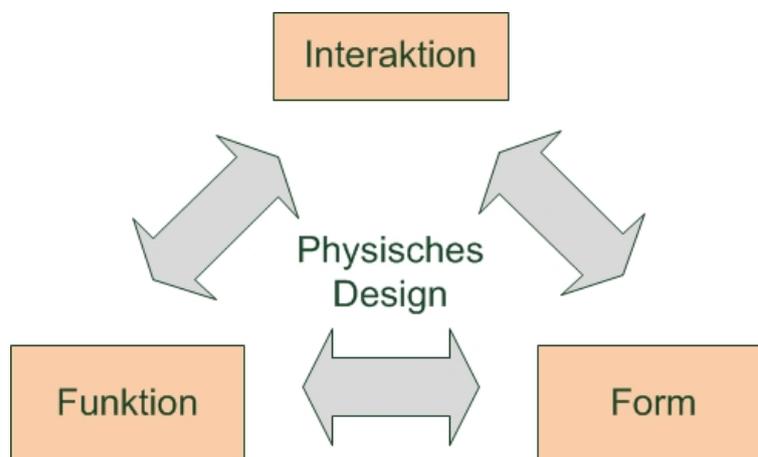


Abbildung 4.6: physisches Design

Bei der Wahl der Form sind vor allem ästhetische Faktoren, verwendete Materialien, sowie Gewicht und Form Kriterien, die untersucht und festgelegt werden müssen. Der Bereich Funktion zielt auf das Verstehen der Hauptfunktionalität und die mögliche Unterteilung in Unterfunktionen. Bei der Interaktion werden vor allem mögliche Einschränkungen, verwendbare Interaktionsmetaphern und die Möglichkeiten für Feedback untersucht.

Darüber hinaus bestehen zwischen diesen Faktoren Wechselwirkungen:

- *Funktion vs. Interaktion*

Die Gestaltung der Interaktion ist ebenfalls für die angestrebte Funktion wichtig. Wie muss die Interaktion gestaltet werden, um die Funktionen optimal ausführen zu können? Welche Funktionen erfordern welche Arten der Interaktion? Dies sind u. a. Fragen, die geklärt werden müssen und die zeigen, wie sich diese beiden Faktoren beeinflussen können.

- *Interaktion vs. Form*

Daneben besteht auch ein Zusammenhang zwischen Form und Interaktion. Die Form kann die Interaktion mit einem Objekt erleichtern oder erschweren, aber auch die Interaktion kann eine Form erforderlich machen (Ferscha u. a., 2007; Wensveen, 2005).

- *Form vs. Funktion*

Die Form eines Artefaktes kann Rückschlüsse auf deren Funktion zulassen. Beim Design ist es wichtig eine Form zu finden, die die gewollte Funktion in einem bestimmten Kontext am besten umschreibt. Als Beispiel hierfür wird in der Literatur häufig ein Hammer angeführt. Die Form des Hammers unterstützt und weist auf dessen Funktion hin und „verleitet“ den Benutzer zum richtigen Benutzen. Gibson umschreibt dieses Phänomen mit den Begriff „Affordances“.

Affordance

Der Begriff „Affordances“ stammt ursprünglich aus der Wahrnehmungstheorie und wurde von J.J. Gibson geprägt ((Gibson, 1977) nach (Norman, 1999)). Gibson beschreibt Affordances als Handlungsaufforderungen bzw. -angebote von Gegenständen an einen Benutzer. Diese natürliche Beziehung, die nicht immer explizit sichtbar, bekannt oder erwünscht sein muss, wurde 1999 von Norman in die HCI eingeführt (Shaer und Hornecker, 2009). Norman beschreibt Affordances als Eigenschaften von Gegenständen oder der Umgebung, die für Menschen relevant sind, weil sie bestimmte Handlungen ermöglichen (Norman, 1988). Diese Umgebungseigenschaften reflektieren die möglichen Verbindungen zwischen Akteuren und Gegenständen. In Norman (1999) unterteilt Norman Affordances in:

- **perceived (engl. für erkennbare oder wahrnehmbare) Affordances**

Damit sind solche Affordances gemeint, die für den Benutzer visuell wahrnehmbar sind. Als Beispiel nennt Norman grafische Elemente einer GUI, wie z.B. Buttons. Diese sind bei grafischen Bedienelementen die einzigen Affordances, die vom Designer kontrolliert werden können, so dass sie besonders interessant für ihn sind.

- **real (engl. für wirkliche oder physische) Affordances**

Dies sind laut Norman die 'richtigen' Affordances, die nicht unbedingt sichtbar sein

müssen und von Benutzern unterbewusst wahrgenommen werden. Zu ihnen gehören z. B. die physische Form, das Material eines Gegenstandes oder dessen Gewicht.

Djajadiningrat u. a. (2002) greift der Ansatz von Norman nicht weit genug. In ihrer Ausarbeitung kritisieren die Autoren, dass in der HCI Gefühle und Intentionen ausgeklammert werden und zu stark auf motorische Fähigkeiten eingegangen wird. So motiviert nicht nur die Paßform eines Objekts in der Hand, sondern auch die Erwartung einer ästhetischen Interaktion, zum Ergreifen und „verführt“ so zur Interaktion (Hornecker, 2004).

4.2.5 Evaluation

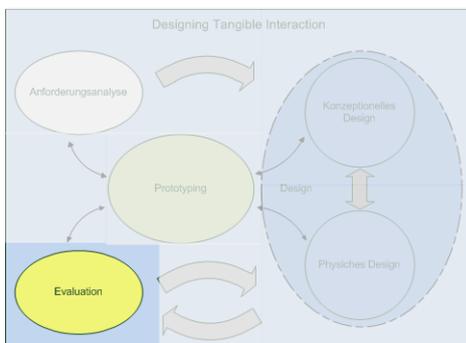


Abbildung 4.7: TI - Design

Die Evaluation (vgl. Abb. 4.7) dient in allen Teilen des Entwicklungsprozesses dazu, die Benutzeranforderungen zu verstehen und mit den Ansätzen und Entwicklungen zu vergleichen. Dabei sollen Probleme erkannt und eine Einschätzung über die Qualität und Nutzbarkeit von Entwicklungen gewonnen werden. Es gibt eine Vielzahl von unterschiedlichen Arten, wie man Evaluationen durchführen kann. Es gibt vier Hauptparadigmen, die alle bei der Evaluation von Tangible Interaction Artefakten zum Einsatz kommen können (Rogers und Sharp, 2002):

- **„Quick and dirty“**

Mit dieser Art der Evaluation kann ein Entwickler schnell Rückmeldung über eine Idee oder ein Konzept bekommen. Dabei beobachtet ein Entwickler Benutzer bei der Verwendung eines Prototypen oder führt informelle Gespräche mit ihnen. Diese Art der Evaluation kann an beliebigen Orten durchgeführt werden und erfordert meist keine Vorbereitungen. Der Entwickler steuert weder einen Test noch übt er eine Kontrolle über die Situation oder den Benutzer aus. Dabei kann der Entwickler quantitative und informelle Beschreibungen seiner Beobachtungen in Form von Notizen, Szenarien oder Berichten anfertigen und zu anderen Entwicklern kommunizieren.

- **Usability Untersuchungen**

Im Gegensatz zur „Quick and dirty“ Evaluation werden bei Usability Untersuchungen eine Reihe von Tests mit mehreren Benutzern einer Zielgruppe durchlaufen. In diesen Tests muss ein Benutzer eine oder mehrere Aufgaben erfüllen. Während des gesamten Ablaufes übt der Entwickler die Kontrolle auf den Ablauf aus. Bei der Durchführung der Aufgaben wird der Benutzer beobachtet und meist gefilmt. Dabei ist es üblich, dass der Benutzer bei der Untersuchung seine Gedanken und Beweggründe laut

ausspricht. Nach den Untersuchungen können zusätzliche Interviews mit dem Benutzer gemacht oder Fragebögen von ihm ausgefüllt werden. Usability Untersuchungen werden in Laboren durchgeführt und dienen dazu, die Leistung der Benutzer bei der Durchführung bestimmter Tätigkeiten zu ermitteln

- **Feldstudien**

Feldstudien sind Evaluationen, die am Einsatzort und im natürlichen Kontext des Benutzers durchgeführt werden. Die Daten und Beobachtungen können dabei in Form von Notizen, Fragebögen, Beobachtungen der Nutzer und ethnografischen Beobachtungen erfasst und zu einem späteren Zeitpunkt analysiert werden. Die Beobachtungen liefern qualitativ hochwertige Daten über die Einsetzbarkeit und Leistungsfähigkeit der Entwicklungen im Benutzerkontext.

- **Prediktive Tests**

Im Gegensatz zu den bisher vorgestellten Evaluationparadigmen sind bei prediktiven Evaluationen keine Benutzer involviert. Vielmehr werden die Untersuchungen von Experten durchgeführt. Dabei werden sowohl theoretische Modelle als auch Heuristiken über den „typischen“ Benutzer als Grundlage für die Tests verwendet. Dabei können Probleme aus Expertensicht und, indem man z. B. eine Aufgabe mit unterschiedlichen Modellen durchführt, quantitative Aussagen zur Durchführung einer Aufgabe ermittelt werden.

Die auf diese Weise gesammelten Informationen sollen im Entwicklungsprozess direkt in die Verbesserung der Anforderungen oder des Designs einfließen.

4.3 Zusammenfassung

In diesem Kapitel wurde eine Designmethodik für die Entwicklung von Tangible Interaction Artefakten beschrieben. Die hier beschriebene Vorgehensweise resultiert aus dem physischen Wesen von Tangible Interaction. Die physikalischen Eigenschaften, Art der Interaktion und damit auch die Funktion von TI Elementen werden durch ihre Umgebung und den Kontext in dem sie benutzt werden stark beeinflusst. Aufgrund dieser Wechselwirkung reicht eine Entwicklung die ausschließlich auf Benutzermodellen und Benutzermodellierung basiert nicht aus. Vielmehr müssen alle entwickelten Modelle mit den Benutzern zusammen im Kontext der Benutzung getestet werden. Aus diesem Grund ist eine benutzerzentrierte Entwicklung, bei der mit Hilfe von Prototypen die entwickelten Konzepte evaluiert und weiterentwickelt werden, für das Design von TI unerlässlich.

Die in diesem Kapitel entwickelte Methode besteht aus einem iterativen Zyklus von Design, Evaluation und Redesign. Dabei nehmen Prototypen und die iterative Verbesserung dieser

in allen Entwicklungsphasen eine zentrale Rolle ein. Als Grundlage der Überlegungen für die beschriebene Entwicklungsmethode dienen die in dem Kapitel 3 gemachten Erfahrungen, sowie Ansätze aus dem Interaction Design, User - Centered Design, Contextual Design und Analytischen Design.

Zu Beginn des beschriebenen Zyklus steht die *Anforderungsanalyse*, in der die funktionalen und nichtfunktionalen Anforderungen ermittelt werden. Ziel ist es, unterschiedliche Szenarien und Anwendungsfälle zu generieren. Diese dienen im Entwicklungsprozess als Grundlage für das *konzeptionelle Design*. In diesem Abschnitt der Designmethodik werden unterschiedliche Interaktions- und Funktionskonzepte entwickelt, sowie unterschiedliche Interaktionsmodi und -formen getestet. Mit Hilfe von Prototypen werden im *physischen Design* die entwickelten Designmöglichkeiten und -konzepte zu greifbaren Entwürfen weiterentwickelt. Dadurch lässt sich das Zusammenspiel von Form, Funktion und Interaktion und die Einflüsse dieser Faktoren aufeinander evaluieren. Die *Evaluation* erfolgt durch das Testen der Prototypen mit der Benutzergruppe und im Idealfall in dem Kontext in dem das System von den Benutzern verwendet werden soll. Die so gemachten Erfahrungen und Erkenntnisse können direkt in die Weiterentwicklung der Prototypen und die Verbesserung der Konzepte einfließen. Da es sich bei Tangible Interaction Artefakten um auf spezielle Anwendungen zugeschnittene und physikalischen Gesetzen und Einflüssen unterliegende Elemente handelt, ist diese Art der ständigen Evaluation, Anpassung und Weiterentwicklung der Prototypen sehr wichtig.

5 Fazit und Ausblick

Das Thema dieser Arbeit beschäftigt sich mit einer Form der Mensch - Computer Interaktion durch physisch greifbare Artefakte und Benutzerschnittstellen. Die Überlegung zu einer solchen Interaktion und die Notwendigkeit hierfür haben ihren Ausgangspunkt in Trends und Entwicklungen der Informatik wie Ubiquitous Computing und Pervasive Computing, Augmented Reality und Ambient Intelligence. Die Entwicklung führt zu immer kleiner werdenden Computern, mit denen nicht mehr über eine Maus und Tastatur interagiert wird und die ganz in die Umgebung des Benutzers integriert werden. Dadurch nimmt der Benutzer den Computer nicht mehr als technisches Gerät, sondern nur noch die Dienste, die ihm mittels diese Computers angeboten werden, wahr.

Das Forschungsgebiet, welches sich mit dieser Form der Interaktion beschäftigt, wird als Tangible Interaction bezeichnet. Tangible Interaction fasst unterschiedliche Ansätze und Herangehensweisen, wie Graspable User Interfaces, Embodied User Interfaces, Tangible Computing und Tangible User Interfaces zusammen. Tangible Interaction als Interaktion mit Artefakten, entfernten Computern oder intelligenten Umgebungen bietet sowohl Vorteile als auch Einschränkungen. Diese Arbeit hat die Situiertheit in der physischen Welt, die Möglichkeit der parallelen Eingabe und Zusammenarbeit, die Direktheit der Interaktion, die Möglichkeit der Verwendung von physischen Affordances und Beschränkungen sowie die Spezialisierung der Artefakte auf spezielle Einsatzgebiete als Vorteile aufgezeigt. Dem stehen Einschränkungen bei der Skalierbarkeit, der Flexibilität und Vielseitigkeit, die schnelle Ermüdung der Benutzer, der Verlust von Artefakten und die Spezialisierung gegenüber.

Es wurden mit den Experimenten Hamburg Cubical und SITab Beispiele der Tangible Interaction vorgestellt. Hierbei handelt es sich um Entwicklungen, die im Rahmen des Living Place Hamburg, einem Smart Home welches als Labor im Bereich Ambient Intelligence konzipiert worden ist, entstanden sind.

Hinter Hamburg Cubical verbirgt sich die Entwicklung eines Artefaktes, welches als Fernsteuerung für und als Eingriffsmöglichkeit in intelligente Wohnumgebungen erstellt wurde. Es handelt sich um einen Plexiglas - Würfel bei dem ein Benutzer durch die Veränderung der Lage eine Funktion auswählen und durch das horizontale Drehen diese Funktion manipulieren kann. Dabei kommt der Cubical ohne symbolische Bedienelemente, wie Knöpfe oder Touch-Displays, aus. Das zweite Experiment steht symbolisch für die Möglichkeit, durch die Erweiterung von Gegenständen und Geräten mit Sensoren neuen Funktionen und Diensten

zu entwickeln und zugänglich machen zu können. Dazu wurde ein Rahmen zur Erweiterung von Tablet PCs, Smartphones oder ähnliche Geräte mit Sensoren entwickelt. Dadurch ist eine Verortung der Geräte in ihrer Umgebung möglich, um so zum Beispiel Nachbargeräte zu erkennen und mit diesen Daten austauschen zu können. Als Anwendungsbeispiel wurde so die Skalierung von Bildern über mehrere Geräte gezeigt.

Das Hauptaugenmerk dieser Arbeit lag jedoch nicht in der Entwicklung solcher Artefakte. Vielmehr dienten diese zur Vorbetrachtung und Veranschaulichung von Herausforderungen bei der Entwicklung dieser Geräte. Die Arbeit hat das Design von Hardware und Software, eine erwartete Persistenz bei der Benutzung von TI Artefakten, die Notwendigkeit der handbuchfreien Bedienung, die Abhängigkeit der Interaktion vom Kontext der Benutzung, sowie die mögliche Parallelität der Eingabe als Herausforderungen untersucht und hervorgehoben. Um diese Herausforderungen lösen zu können, wurden die notwendigen Schritte zur Konstruktion von TI Artefakte und die dafür notwendigen Techniken und Vorgehensweisen betrachtet und in einen Designprozess überführt. Dabei handelt es sich um einen benutzerzentrierten, iterativen Designprozess, in dem Prototyping eine zentrale Rolle einnimmt.

Bei den Prototypen handelt es sich im Kontext dieser Arbeit um physisch greifbare Hypothesen über die Funktion und Gestalt von Tangible Interaction Artefakten. Ihr Einsatz in diesem Prozess erlaubt es, schnell unterschiedliche Designrichtungen zu testen und den Benutzer früh in den Entwicklungsprozess einzubinden. Diese Einbeziehung des Benutzers in den Entwicklungsprozess und die Abstimmung des Prozesses auf den Benutzerkontext ist zwingend erforderlich, da die physikalischen Eigenschaften, die Art der Interaktion und damit auch die Funktion von TI Artefakten durch den Kontext, in dem sie benutzt werden, und durch Wechselwirkungen untereinander stark beeinflusst werden.

Die nächsten Entwicklungen werden zeigen wie gut der hier entwickelte Designprozess für die Entwicklung von TI Artefakten geeignet ist oder ob an manchen Stellen Veränderungen vorgenommen werden müssen. Da es sich um eine reale Wohnung handelt, ist das Living Place Hamburg als Entwicklungsumgebung für TI Artefakte, welche im Kontext von Smart Homes verwendet werden sollen, sehr gut geeignet. So kann während der Evaluation auf das integrierte Usability Labor zurückgegriffen werden. Es müssen jedoch bei zukünftigen Entwicklungen Konzepte und Verfahren definiert werden, die einen effektiven Einsatz des Usability Equipment für die Evaluation der Prototypen und Konzepte während der Anwendung durch den Benutzer in einem möglichst natürlichen Benutzerkontext möglich machen. Desweiteren müssen beim Prototyping unterschiedliche Plattformen untersucht werden. Die bei den vorgestellten Projekten verwendete Arduino Plattform ist gut geeignet im Anfangsstadium des Designprozesses um Konzepte, Sensoren oder Aktoren evaluieren zu können, stößt jedoch aufgrund der begrenzten Leistung bei der Entwicklung von leistungsstärkeren Prototypen an ihre Grenzen. Bei zukünftigen Entwicklungen wird es notwendig, leistungsstärkere und flexiblere Prototyping Plattformen zu testen oder zu entwickeln.

Zu Beginn dieser Arbeit wurde Tangible Interaction als eine einfache, intuitive und schnell erlernbare Form der Interaktion mit Computern vorgestellt. Die bisherigen Untersuchungen zeigen, dass Tangible Interaction Artefakte das Potential haben, diese Versprechen umsetzen zu können. Inwieweit sich diese Beobachtungen im Einsatz außerhalb der Forschungslabore bewahrheitet, werden die nächsten Untersuchungen zeigen müssen. Ob es doch zu einer Überforderung der Benutzer kommt oder ob z.B. über Vierzigjährige mit TI als Interaktionsform schlechter zu recht kommen als sog. „digital natives“¹, sind spannende Fragen in diesem Zusammenhang.

¹siehe (Palfrey und Gasser, 2008)

Literaturverzeichnis

- [802.15 Working Group] 802.15 WORKING GROUP, IEEE: *IEEE Std 802.15.4-2003*.
– URL <http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf>. – letztes Zugriffsdatum: 06.08.2010
- [Aarts 2004] AARTS, E.: Ambient intelligence: a multimedia perspective. In: *Multimedia, IEEE* 11 (2004), jan., Nr. 1, S. 12–19
- [Aarts und Marzano 2003] AARTS, Emile ; MARZANO, Stefano: *The new everyday - Views on Ambient Intelligence*. 010 Uitgeverij, 2003
- [ACM] ACM: *ACM SIGCHI Curricula for Human-Computer Interaction*. – URL <http://sigchi.org/cdg/cdg2.html>. – (letztes Zugriffsdatum: 13.02.2009)
- [Apple] APPLE: *Apple Inc.- Internetseite*. – URL <http://www.apple.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Arduino] ARDUINO: *Arduino - Homepage*. – URL <http://arduino.cc>. – (letztes Zugriffsdatum: 30.01.2011)
- [Atmel] ATMEL: *Atmega328P Datenblatt*. – URL http://atmel.com:80/dyn/resources/prod_documents/doc8271.pdf. – (letztes Zugriffsdatum: 06.09.2010)
- [ATMEL] ATMEL: *Atmel Corporation - Homepage*. – URL <http://www.atmel.com/>. – (letztes Zugriffsdatum: 22.02.2009)
- [Azuma 1997] AZUMA, Ronald T.: A Survey of Augmented Reality. In: *Presence: Teleoperators and Virtual Environments* 6, 1997, S. 355–385
- [Baragan 2004] BARAGAN, Hernando: *Wiring: Prototyping Physical Interaction Design*, Interaction Design Institute Ivrea, Mastersthesis, 2004
- [Beaudouin-Lafon und Mackay 2008] BEAUDOUIN-LAFON, Michel ; MACKAY, Wendy E.: PROTOTYPING TOOLS AND TECHNIQUES. In: *The Human-Computer Interaction Handbook*, 2008, S. 1017–1039

- [Blackler u. a. 2002] BLACKLER, A. ; POPOVIC, V. ; MAHAR, D. P.: *Intuitive use of products*. Paper presented at the Common Ground Design Research Society International Conference London. 2002
- [Blackler u. a. 2003] BLACKLER, A. ; POPOVIC, V. ; MAHAR, D. P.: *Designing for Intuitive Use of Products: An Investigation*. Paper presented at the 6th Asian Design International Conference, Tsukuba, Japan. November 2003
- [Blackler 2006] BLACKLER, Alethea: *Intuitive Interaction with Complex Artefacts*, School of Design Queensland University of Technology, Dissertation, 2006
- [Buur u. a. 2004] BUUR, Jacob ; JENSEN, Mads V. ; DJAJADININGRAT, Tom: Hands-only scenarios and video action walls: novel methods for tangible user interaction design. In: *DIS '04: Proceedings of the 5th conference on Designing interactive systems*. New York, NY, USA : ACM, 2004, S. 185–192
- [Carroll und Reitman Olsen 1987] CARROLL, John M. ; REITMAN OLSEN, Judith: *Mental Models in Human Computer Interaction*. National Academic Press, 1987
- [Couture u. a. 2008] COUTURE, Nadine ; RIVIRE, Guillaume ; REUTER, Patrick: GeoTUI: A Tangible User Interface for Geoscience. In: *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*, 2008, S. 89–96
- [Crockford] CROCKFORD, D.: *JSON - RFC4627*. – URL <http://www.ietf.org/rfc/rfc4627.txt?number=4627>. – letztes Zugriffsdatum: 06.08.2010
- [Djajadiningrat u. a. 2002] DJAJADININGRAT, Tom ; OVERBEEKE, Kees ; WENSVEEN, Stepha: But how, Donald, tell us how ? In: *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques*, ACM, 2002, S. 291
- [Dourish 2001] DOURISH, Paul: *Where the Action Is - The Foundations of Embodied Interaction*. MIT Press, 2001
- [Edge und Blackwell 2009] EDGE, Darren ; BLACKWELL, Alan F.: Peripheral tangible interaction by analytic design. In: *TEI '09: Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*. New York, NY, USA : ACM, 2009, S. 69–76
- [Elektronics] ELEKTRONICS, ST: *ST Electronics - Internetseite*. – URL <http://www.st.com>. – (letztes Zugriffsdatum: 06.08.2010)
- [Erickson 1995] ERICKSON, Thomas D.: Working with interface metaphors. (1995), S. 147–151

- [Fernaesus u. a. 2008] FERNAEUS, Ylva ; THOLANDER, Jakob ; JONSSON, Martin: Towards a new set of ideals: consequences of the practice turn in tangible interaction. In: *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*. New York, NY, USA : ACM, 2008, S. 223–230
- [Ferscha u. a. 2007] FERSCHA, Alois ; VOGL, Simon ; EMSENHUBER, Bernadette ; WALLY, Bernhard: Physical shortcuts for media remote controls. In: *INTETAIN '08: Proceedings of the 2nd international conference on INtelligent TEchnologies for interactive enterTAINment*. ICST, Brussels, Belgium, Belgium : ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, S. 1–8
- [Fishkin 2004] FISHKIN, Kenneth P.: A taxonomy for and analysis of tangible interfaces. In: *Personal Ubiquitous Comput.* 8 (2004), Nr. 5, S. 347–358
- [Fishkin u. a. 2000] FISHKIN, Kenneth P. ; GUJAR, Anuj ; HARRISON, Beverly L. ; MORAN, Thomas P. ; WANT, Roy: Embodied user interfaces for really direct manipulation. In: *Commun. ACM* 43 (2000), Nr. 9, S. 74–80
- [Fitzmaurice 1996] FITZMAURICE, George W.: *Intuitive Interaction with Complex Artefacts*, University of Toronto, Dissertation, 1996
- [Fitzmaurice u. a. 1995] FITZMAURICE, George W. ; ISHII, Hiroshi ; BUXTON, William A. S.: Bricks: laying the foundations for graspable user interfaces. In: *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1995, S. 442–449
- [Freescale] FREESCALE: *Freescale - Internetseite*. – URL <http://www.freescale.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Frens 2006] FRENS, Joep: *Designing for Rich Interaction: Integrating Form, Interaction, and Function*, Eindhoven University of Technology, Dissertation, 2006
- [Frøkjær und Hornbæk 2002] FRØKJÆR, Erik ; HORNBÆK, Kasper: Metaphors of Human Thinking in HCI: Habit, Stream of Thought, Awareness, Utterance, and Knowing. In: *In Proceedings of HF2002/OzCHI 2002*, Basic Books, 2002, S. 964–971
- [FTDI] FTDI: *FTDI FT232R Produktseite*. – URL <http://www.ftdichip.com/Products/ICs/FT232R.htm>. – (letztes Zugriffsdatum: 06.08.2010)
- [Gehn 2010] GEHN, Stefan: *Interpretation mehrdeutiger Eingabedaten auf Infrarot-basierten Multitouchdisplays*, Hochschule für Angewandte Wissenschaften Hamburg, Masterthesis, 2010
- [Gibson 1977] GIBSON, JJ: The theory of affordances. In: *Perceiving, acting, and knowing: Toward an ecological* (1977)

- [Gregor 2009] GREGOR, Sebastian: *Physical Interaction Design*, Hochschule für Angewandte Wissenschaften Hamburg, Seminararbeit, 2009
- [Gregor u. a. 2009] GREGOR, Sebastian ; RAHIMI, Ali M. ; VOGT, Matthias ; SCHULZ, Thomas ; LUCK, Kai von: Tangible Computing revisited: Anfassbare Computer in Intelligenten Umgebungen. In: *4. Kongress Multimediatechnik Wismar*, 2009
- [Harrison u. a. 1998] HARRISON, Beverly L. ; FISHKIN, Kenneth P. ; GUJAR, Anuj ; MOCHON, Carlos ; WANT, Roy: Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In: *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1998, S. 17–24
- [Hinckley 2003a] HINCKLEY, Ken: Distributed and local sensing techniques for face-to-face collaboration. In: *ICMI '03: Proceedings of the 5th international conference on Multimodal interfaces*. New York, NY, USA : ACM, 2003, S. 81–84
- [Hinckley 2003b] HINCKLEY, Ken: Synchronous gestures for multiple persons and computers. In: *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*. New York, NY, USA : ACM, 2003, S. 149–158
- [Hindus 1999] HINDUS, Debby: The Importance of Homes in Technology Research. In: *CoBuild '99: Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*. London, UK : Springer-Verlag, 1999, S. 199–207
- [Hollatz 2010] HOLLATZ, Dennis: *Entwicklung einer nachrichtenbasierten Architektur für Smart Homes*, Hochschule für Angewandte Wissenschaften Hamburg, Masterarbeit, 2010
- [Hornecker 2004] HORNECKER, Eva: *Tangible User Interfaces als kooperationsunterstützendes Medium*, Universität Bremen, Dissertation, 2004
- [Hornecker 2005] HORNECKER, Eva: A Design Theme for Tangible Interaction: Embodied Facilitation. In: *Proceedings of the 9th European Conference on Computer Supported Cooperative Work*, 2005
- [Hornecker 2008] HORNECKER, Eva: Die Rückkehr des Sensorischen: Tangible Interfaces und Tangible Interaction. In: *Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung*, 2008, S. 235–258
- [Hornecker und Buur 2006] HORNECKER, Eva ; BUUR, Jacob: Getting a grip on tangible interaction: a framework on physical space and social interaction. In: *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*. New York, NY, USA : ACM, 2006, S. 437–446

- [Hurtienne und Israel 2007] HURTIENNE, Jörn ; ISRAEL, Johann H.: Image schemas and their metaphorical extensions. In: *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07* (2007), S. 127
- [Ishii 2008a] ISHII, Hiroshi: Tangible bits: beyond pixels. In: *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*. New York, NY, USA : ACM, 2008, S. xv–xxv
- [Ishii 2008b] ISHII, Hiroshi: Tangible User Interfaces. In: *The Human-Computer Interaktion Handbook*, 2008, S. 469–488
- [Ishii und Ullmer 1997] ISHII, Hiroshi ; ULLMER, Brygg: Tangible bits: towards seamless interfaces between people, bits and atoms. In: *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 1997, S. 234–241
- [Johnson-Laird 1980] JOHNSON-LAIRD, P. N.: Mental models in cognitive science. In: *Cognitive Science 4*, 1980, S. 41–115
- [Kirk u. a. 2009] KIRK, David ; SELLEN, Abigail ; TAYLOR, Stuart ; VILLAR, Nicolas ; IZADI, Shahram: Putting the physical into the digital: issues in designing hybrid interactive surfaces. In: *BCS HCI '09: Proceedings of the 2009 British Computer Society Conference on Human-Computer Interaction*. Swinton, UK, UK : British Computer Society, 2009, S. 35–44
- [Kirsh 1995] KIRSH, David: Complementary Strategies: Why we use our hands when we think. In: *Proceedings of 7th Annual Conference of the Cognitive Science Society* (1995)
- [Kirstgen 2010] KIRSTGEN, Benjamin: *CSCW - Grundlagen und Herausforderungen*, Hochschule für Angewandte Wissenschaften Hamburg, Seminarbericht, 2010
- [Klemmer u. a. 2006] KLEMMER, Scott R. ; HARTMANN, Björn ; TAKAYAMA, Leila: How bodies matter: five themes for interaction design. In: *DIS '06: Proceedings of the 6th conference on Designing Interactive systems*. New York, NY, USA : ACM, 2006, S. 140–149
- [Larssen u. a. 2007] LARSSSEN, Astrid T. ; ROBERTSON, Toni ; EDWARDS, Jenny: The feel dimension of technology interaction: exploring tangibles through movement and touch. In: *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*. New York, NY, USA : ACM, 2007, S. 271–278
- [Liquidware] LIQUIDWARE: *Liquidware - Internetseite*. – URL <http://www.liquidware.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Living Place Hamburg Team] LIVING PLACE HAMBURG TEAM: *Living Place Hamburg - Projektseite*. – URL <http://www.livingplace.org/>. – (letztes Zugriffsdatum: 04.08.2010)

- [Löwgren] LÖWGREN, Jonas: *Interaction Design*. – URL http://www.interaction-design.org/encyclopedia/interaction_design.html. – (letztes Zugriffsdatum: 21.02.2009)
- [Mattern 2001] MATTERN, Friedemann: *Pervasive Computing / Ubiquitous Computing*. 2001
- [Maxim] MAXIM: *Maxim Integrated Products - Internetseite*. – URL <http://www.maxim-ic.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Merrill u. a. 2007] MERRILL, David ; KALANITHI, Jeevan ; MAES, Pattie: Siftables: towards sensor network user interfaces. In: *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*. New York, NY, USA : ACM, 2007, S. 75–78
- [Meyer und Rakotonirainy 2003] MEYER, Sven ; RAKOTONIRAINY, Andry: A survey of research on context-aware homes. In: *ACSW Frontiers '03: Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*. Darlinghurst, Australia, Australia : Australian Computer Society, Inc., 2003, S. 159–168
- [Milgram und Kishino 1994] MILGRAM, P. ; KISHINO, F.: A Taxonomy of Mixed Reality Visual Display. In: *IEICE Transactions on Information Systems*, 1994, S. 1321–1329
- [MuP/RP 2010] MUP/RP, HAW: Rapid Prototyping Anlage der HAW Hamburg. URL <http://www.mp.haw-hamburg.de/fachberg/prod/rp/seite2.pdf>, 2010. – Forschungsbericht
- [Nintendo] NINTENDO: *Nintendo of Europe GmbH - Wii Produktseite*. – URL <http://wii.nintendo.de/>. – (letztes Zugriffsdatum: 19.07.2010)
- [Norman 1988] NORMAN, D.A.: *The Psychology of Everyday Things*. Basic Books, 1988
- [Norman 1999] NORMAN, D.A.: Affordance, conventions, and design. In: *interactions* 6 (1999), Nr. 3, S. 38–43
- [Palfrey und Gasser 2008] PALFREY, John ; GASSER, Urs: *Born Digital: Understanding the First Generation of Digital Natives*. 387 Park Avenue South, New York, NY : Basic Books, 2008. – ISBN 978-0-465-00515-4
- [Rahimi und Vogt 2010] RAHIMI, Mohammad A. ; VOGT, Matthias: *Aufbau des Living Place Hamburg*, Hochschule für Angewandte Wissenschaften Hamburg, Projektbericht, 2010
- [Rahimi und Vogt 2011] RAHIMI, Mohammad A. ; VOGT, Matthias: *Seamless Interaction - Natürliche Interaktion in Smart Living Umgebungen*, Hochschule für Angewandte Wissenschaften Hamburg, Masterthesis, 2011

- [Reas und Fry 2005] REAS, Casey ; FRY, Benjamin: Processing.org: a networked context for learning computer programming. In: *SIGGRAPH '05: ACM SIGGRAPH 2005 Web program*, 2005, S. 14
- [Rogers und Sharp 2002] ROGERS, Yvonne ; SHARP, Helen: *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, Inc., 2002
- [Saffer 2005] SAFFER, Dan: *The Role of Metaphor in Interaction Design*, Carnegie Mellon University, Masterthesis, 2005
- [Shaer und Hornecker 2009] SHAER, Orit ; HORNECKER, Eva: Tangible User Interfaces: Past, Present, and Future Directions. In: *Foundations and Trends in Human-Computer Interaction* 3 (2009), Nr. 1-2, S. 1–137
- [Shaer und Jacob 2009] SHAER, Orit ; JACOB, Robert J.: A specification paradigm for the design and implementation of tangible user interfaces. In: *ACM Trans. Comput.-Hum. Interact.* 16 (2009), Nr. 4, S. 1–39
- [Sparkfun] SPARKFUN: *MU 6DOF Razor Datenblatt*. – URL http://www.sparkfun.com/commerce/product_info.php?products_id=10010
- [Sparkfun] SPARKFUN: *Sparkfun Inc.- Internetseite*. – URL <http://www.sparkfun.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Staggers und Norcio 1993] STAGGERS, Nancy ; NORCIO, A. F.: Mental models: concepts for human-computer interaction research. In: *Int. J. Man-Mach. Stud.* 38 (1993), Nr. 4, S. 587–605
- [Steen u. a. 2007] STEEN, Marc ; KUIJT-EVERS, Lottie ; KLOK, Jente: Early user involvement in research and design projects - A review of methods and practices. In: *23rd EGOS Colloquium*, 2007
- [Ullmer und Ishii 2000] ULLMER, B. ; ISHII, H.: Emerging frameworks for tangible user interfaces. In: *IBM Syst. J.* 39 (2000), Nr. 3-4, S. 915–931
- [Vishay] VISHAY: *Vishay Semiconductors - Internetseite*. – URL <http://www.vishay.com/>. – (letztes Zugriffsdatum: 06.08.2010)
- [Weiser 1991] WEISER, Mark: *The Computer for the 21st Century*. 1991. – URL <http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html>. – (letztes Zugriffsdatum: 19.07.2010)
- [Wellner u. a. 1993] WELLNER, Pierre ; MACKAY, Wendy ; GOLD, Rich: Back to the real world. In: *Commun. ACM* 36 (1993), Nr. 7, S. 24–26
- [Wensveen 2005] WENSVEEN, Stephan: *A Tangible Approach to Affective Interaction*, Delft University of Technology, Dissertation, 2005

[Winograd 1997] WINOGRAD, Terry: The design of interaction. (1997), S. 149–161

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §24(5) ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 2. Februar 2011

Ort, Datum

Unterschrift