



Diplomarbeit

# Tangible Augmented Reality für Computerspiele

ausgeführt am

Institut für Softwaretechnik und Interaktive Systeme  
der Technischen Universität Wien

unter der Anleitung von

**Ao. Univ. Prof. Dipl.-Ing. Dr. techn. Dieter Schmalstieg**

durch

**Christiane Ulbricht**

Matrikelnummer 9625056

Friedensgasse 35

2201 Gerasdorf bei Wien

Wien, 9. September 2002

**Kurzfassung**

Diese Diplomarbeit beschreibt die Entwicklung eines Tangible User Interface für ein bestehendes Augmented Reality System und untersucht weiters, inwieweit sich dieses Interface für den Einsatz im Computerspiele-Bereich eignet. Durch die Verwendung eines Augmented Reality Systems sehen die Spieler eine dreidimensionale Darstellung der Grafik. Das spezielle Interface ermöglicht es ihnen außerdem, ihre Spielfiguren direkt und intuitiv – anstatt mit Hilfe von Maus oder Keyboard – zu steuern. Um beurteilen zu können, für welche Spieltypen und welche Steuerungsformen dieses Interface genutzt werden kann, wurden verschiedene Versionen eines Computerspiels implementiert, die entsprechende Untersuchungen ermöglichen. Es hat sich herausgestellt, daß es sowohl für Actionspiele als auch für Strategiespiele gut geeignet ist. Ein Vergleich verschiedener Steuerungsformen hat jedoch ergeben, daß Steuerungsformen besser gehandhabt werden können, je realitätsnäher sie sind.

**Abstract**

This master thesis describes the development of a Tangible User Interface for an existing Augmented Reality System and investigates its suitability for computer games. The use of an Augmented Reality System allows players to see a three-dimensional presentation of the graphics. Moreover, the specific interface makes it possible to manipulate the pieces in a direct and intuitive way – instead of using a mouse or a keyboard. In order to examine for which types of games and which types of manipulation this interface can be used, several versions of a computer game have been implemented. The Tangible Augmented Reality System turned out to be well suited to both action games and strategy games. However, the comparison of different types of manipulation showed that they can be handled more easily the closer they are to real-world forms of manipulation.

***Danksagung***

*Ich danke ao. Univ. Prof. Dipl.-Ing. Dr. techn. Dieter Schmalstieg für sein Engagement und seine ausgezeichnete Betreuung während der Arbeit an dieser Diplomarbeit.*

*Dipl.-Ing. Gerhard Reitmayr danke ich für seine wertvollen Tips.*

*Weiters möchte ich meiner Familie danken, die mir dieses Studium ermöglicht hat und mich immer unterstützt hat.*

*Wien, 9. September 2002*

*Christiane Ulbricht*

## Inhaltsverzeichnis

<b>1 . EINLEITUNG .....</b>	<b>5</b>
<b>2 . STAND DER FORSCHUNG .....</b>	<b>6</b>
2.1 VIRTUAL REALITY .....	6
2.1.1 Non-immersive Virtual Reality Systems.....	7
2.1.2 Immersive Virtual Reality Systems .....	8
2.1.3 Augmented Reality Systems .....	9
2.2 AUFBAU EINES VIRTUAL REALITY SYSTEMS .....	9
2.3 TRACKING .....	10
2.3.1 Elektromechanisches Tracking .....	12
2.3.2 Elektromagnetisches Tracking .....	14
2.3.3 Akustisches Tracking.....	16
2.3.4 Inertiales Tracking .....	17
2.3.5 Optisches Tracking .....	18
2.3.5.1 <i>Markerbasiertes optisches Tracking</i> .....	18
2.3.5.2 <i>Markerloses optisches Tracking</i> .....	21
2.4 EINGABEGERÄTE .....	26
2.4.1 Datenhandschuh.....	26
2.4.2 Pinchglove .....	28
2.4.3 3D-Maus .....	29
2.4.4 Pen & Tablet.....	30
2.4.5 Alternative Eingabegeräte.....	32
2.5 AUSGABEGERÄTE .....	33
2.5.1 Shutter Glasses .....	34
2.5.2 Polarisations-Brillen .....	34
2.5.3 Head Mounted Display .....	35
2.5.4 See Through Head Mounted Display .....	37
2.5.5 Arm Mounted Display .....	37
2.5.6 Virtual Retinal Display .....	38
2.5.7 Projektionssysteme .....	39
2.5.8 Virtual Table.....	41

2.5.9 Haptisches Feedback .....	41
2.5.10 Auditives Feedback .....	42
2.6 3D-INTERAKTIONSFORMEN.....	42
2.6.1 Navigation .....	42
2.6.1.1 Gaze-directed Steering.....	42
2.6.1.2 Pointing.....	43
2.6.1.3 Map-based Travel .....	43
2.6.1.4 Grabbing the Air .....	44
2.6.2 Manipulation.....	44
2.6.2.1 Simple Virtual Hand.....	44
2.6.2.2 Ray Casting.....	45
2.6.2.3 Image Plane Interaction.....	45
2.6.2.4 Go-Go Interaction Technique.....	47
2.6.2.5 HOMER.....	48
2.6.2.6 Scaled World Grab.....	49
2.6.2.7 World in Miniature.....	49
2.6.3 Systemsteuerung .....	51
2.6.3.1 Virtual Widgets.....	51
2.6.3.2 Pen & Tablet .....	52
2.6.3.3 Speech Commands.....	53
2.6.3.4 Gesten .....	53
2.7 TANGIBLE USER INTERFACE .....	54
2.8 TANGIBLE AUGMENTED REALITY .....	55
2.8.1 Tiles.....	56
2.8.2 Shared Space .....	56
<b>3 . ZIELE DES PROJEKTS .....</b>	<b>57</b>
3.1 NEUE HARDWARE.....	58
3.2 TABLETOP GAME.....	59
3.2.1 Spielidee .....	59
3.2.2 Spieltypen.....	59
3.2.3 Spielfiguren .....	60
3.2.4 Steuerung.....	60

---

3.2.5 Zusatzfunktionen .....	61
<b>4 . BESTEHENDES SYSTEM.....</b>	<b>62</b>
4.1 STUDIERSTUBE .....	62
4.1.1 Tracking .....	66
4.1.2 Eingabegeräte .....	69
4.1.2.1 <i>Personal Interaction Panel</i> .....	69
4.1.2.2 <i>Pinchglove</i> .....	70
4.1.2.3 <i>Keyboard</i> .....	71
4.1.3 Ausgabegeräte .....	71
4.1.3.1 <i>See Through HMD</i> .....	71
4.1.3.2 <i>Projektionswand</i> .....	72
4.2 ARTOOLKIT.....	72
4.3 OPENTRACKER .....	74
4.3.1 Spezifikation der Tracker.....	75
4.3.2 Distributed Tracking.....	77
<b>5 . IMPLEMENTIERUNG .....</b>	<b>79</b>
5.1 HARDWARE .....	79
5.2 3D-MODELLE .....	81
5.2.1 Das Katapult .....	81
5.2.2 Die Windmühle.....	83
5.2.3 Der Ballon .....	84
5.2.4 Die Schatztruhe .....	85
5.3 SPIELTYPEN .....	85
5.3.1 Actionspiel.....	85
5.3.2 Strategiespiel .....	87
5.4 STEUERUNG.....	92
5.4.1 Default .....	92
5.4.2 FollowUp.....	92
5.5 OPTIONEN.....	92
5.5.1 Anzahl der Spielfiguren .....	92
5.5.2 Neue Ballons .....	92
5.5.3 Terrain .....	93

---

5.5.4 Strömungsvisualisierung.....	94
5.5.5 Schatztruhen .....	95
5.6 SPEZIFIKATION DER TRACKER.....	96
<b>6 . ERGEBNISSE .....</b>	<b>98</b>
<b>7 . ZUSAMMENFASSUNG .....</b>	<b>104</b>
<b>8 . REFERENZEN.....</b>	<b>106</b>
<b>9 . ABBILDUNGSVERZEICHNIS .....</b>	<b>112</b>

## **1. Einleitung**

Ein User Interface besteht üblicherweise aus einem Bildschirm, einem Keyboard und einer Maus. Diese Form der Interaktion hat sich in den letzten Jahrzehnten entwickelt und gehört mittlerweile zur Standardausstattung eines PCs. Die weite Verbreitung dieser Schnittstelle läßt vermuten, daß es sich dabei um die optimale Kommunikationsform zwischen Mensch und Computer handelt. Dies ist jedoch nicht der Fall, da dies im Gegenteil lediglich eine sehr abstrakte Interaktionsmöglichkeit darstellt. Vergleicht man dieses Interface mit der natürlichen Umgebung eines Menschen, so lassen sich zwei wesentliche Nachteile erkennen:

- Objekte können nicht direkt, sondern nur indirekt mit Hilfe von Maus oder Keyboard manipuliert werden.
- Die Darstellung erfolgt nicht dreidimensional, sondern lediglich zweidimensional.

Die *Tangible Augmented Reality* versucht, die Diskrepanz zwischen der Interaktion mit der natürlichen Umgebung und der Interaktion mit einem Computersystem aufzuheben. Kapitel 2 bietet einen detaillierten Überblick über den aktuellen Stand der Forschung auf diesem Gebiet.

Computerspiele profitieren besonders von einer dreidimensionalen Darstellung und einer direkten Interaktionsform, da dadurch der Realismus erhöht und die Steuerung des Spiels vereinfacht wird. Aus diesem Grund beschäftigt sich diese Diplomarbeit mit der Entwicklung eines *Tangible Augmented Reality* Systems für Computerspiele, dessen Aufbau in Kapitel 3 beschrieben wird. Dabei wird ein bereits bestehendes Augmented Reality System (siehe Kapitel 4) um ein entsprechendes Interface erweitert. Um zu untersuchen, für welche Spieltypen sich dieses Interface besonders gut eignet und welche Möglichkeiten zur Steuerung der Spielfiguren bestehen, wird zusätzlich ein Spiel entwickelt, das eine Analyse dieser Aspekte ermöglicht. Dazu werden verschiedene Versionen dieses Spiels generiert, die sich vor allem durch den Typ des Spiels und die Art der Steuerung unterscheiden. Die Implementierungen der Schnittstelle und des Spiels werden in Kapitel 5 beschrieben. Kapitel 6 enthält die Ergebnisse und Schlußfolgerungen, die sich aus den Tests der verschiedenen Spielversionen ergeben haben.

## **2. Stand der Forschung**

### **2.1 Virtual Reality**

*Virtual Reality* (VR) ist eine Computersimulation verschiedener Eigenschaften unserer Realität. In den meisten Fällen beinhalten diese Eigenschaften visuelle Eindrücke und interaktive Reaktionen. Im Gegensatz zu konventionellen Visualisierungssystemen wird der Benutzer zu einem aktiven Teil der Simulation. Ziel der Virtual Reality ist es, diese Simulation so überzeugend zu gestalten, daß der Benutzer diese als Teil der Wirklichkeit akzeptieren kann. Jede Simulation der Realität kann daher als Virtual Reality bezeichnet werden. Spezielle Hardware ist dazu nicht unbedingt notwendig. Es kann zwischen drei verschiedenen Arten von Virtual Reality Systemen unterschieden werden:

- Non-immersive Virtual Reality Systems
- Immersive Virtual Reality Systems
- Augmented Reality Systems

*Non-immersive Virtual Reality Systems*, auf die in Kapitel 2.1.1 näher eingegangen wird, kommen oft ohne Zusatzhardware aus. Die virtuelle Welt wird z.B. durch einen konventionellen Monitor betrachtet. Lediglich die Grafik muß in Echtzeit dargestellt werden können, um den visuellen Eindruck realistisch zu gestalten. Für spezielle Anwendungen ist diese Art der Simulation aber nicht realistisch genug.

Systeme, die die Umgebung des Benutzers durch eine künstliche ersetzen, werden als *immersive Virtual Reality Systems* (siehe Kapitel 2.1.2) bezeichnet, da dem Benutzer das Gefühl vermittelt werden soll, in die künstliche Realität einzutauchen. In diesem Fall wird versucht, die Simulation durch Verwendung von speziellen Hardwarekomponenten (siehe Kapitel 2.4 und 2.5) überzeugender zu gestalten. Es ist dem Benutzer dann jedoch kaum noch möglich, auf Geschehnisse in seiner realen Umgebung entsprechend zu reagieren.

Dieses Problem kann durch eine Kombination der künstlichen und der realen Welt beseitigt werden. Diese kombinierten Systeme werden als *Augmented Reality Systems* (siehe Kapitel 2.1.3) bezeichnet. Die reale Umgebung des Benutzers wird dabei durch computergenerierte

Objekte erweitert. Dadurch kann der Benutzer sowohl mit der virtuellen, als auch mit der realen Welt interagieren.

### **2.1.1 Non-immersive Virtual Reality Systems**

In einem *Non-immersive Virtual Reality System* ist die Grenze zwischen der realen und der virtuellen Welt klar erkennbar. Die virtuelle Welt wird wie durch ein Fenster betrachtet, daher werden diese Systeme auch als *Window on World Systems* (WoW) bezeichnet. Viele dieser Systeme benutzen einen konventionellen Computermonitor für das visuelle Feedback. Auch *Desktop VR* ist eine gängige Bezeichnung für diese Systeme, da ein herkömmlicher PC ausreichend ist.

Um die Darstellung der virtuellen Welt realistischer zu gestalten, können LCD Shutter Glasses (siehe Kapitel 2.5.1) verwendet werden. Dazu muß der Monitor die Grafik stereoskopisch ausgeben. Der Benutzer sieht nun zwar ein dreidimensionales Bild, doch er betrachtet die virtuellen Objekte immer noch aus einem fixen Blickwinkel. Wird ein mechanischer Tracker an den Shutter Glasses befestigt, kann das vom Computer generierte Bild an den Blickwinkel des Betrachters angepaßt werden. Diese Form der Virtual Reality wird auch als *Fish Tank Virtual Reality* [Ware, 1993] bezeichnet.

Die Verwendung herkömmlicher Eingabegeräte wie Tastatur und Maus eignet sich jedoch nicht sehr gut für die Simulation der Realität, da diese Geräte im Alltag nur zur Interaktion mit Computern benutzt werden, nicht aber um z.B. reale Objekte zu erfassen. Es wird daher versucht, die Interaktion mit virtuellen Objekten intuitiver zu gestalten (siehe Kapitel 2.6). Durch die Verwendung von speziellen Eingabegeräten (siehe Kapitel 2.4) wird dem Benutzer eine Interaktionsmöglichkeit abseits von Tastatur und Maus ermöglicht. Eine Form der Interaktion ist z.B. die Kommunikation mittels Gesten (siehe Kapitel 2.6.3.4). Ein Beispiel einer Interaktion mittels Gesten ist das *Video Mapping* [Isdale, 1993]. Dabei werden die Bewegungen des Benutzers von einer Kamera aufgenommen. Aus jedem Einzelbild wird mittels Bildbearbeitungs-Methoden die Silhouette des Benutzers extrahiert und die Position der Arme analysiert (siehe Kapitel 2.3.5.2). Der Benutzer kann am Monitor die Interaktion seines Körpers mit der virtuellen Welt beobachten.

### 2.1.2 Immersive Virtual Reality Systems

Ein *Immersive Virtual Reality System* vermittelt dem Benutzer den Eindruck, sich in einer künstlichen Umgebung zu befinden. Um diesen Eindruck realistisch zu gestalten, muß der Benutzer so weit wie möglich in die virtuelle Welt integriert werden. Tabelle 1 vergleicht jene Eigenschaften eines VR-Systems, die den Immersionseindruck erhöhen bzw. erniedrigen.

Hoher Immersionseindruck	Geringer Immersionseindruck
Stereoskopische Darstellung	Monoskopische Darstellung
Interaktivität	Keine Interaktivität
Hohe Bildkomplexität	Geringe Bildkomplexität
Head Mounted Display	Computermonitor
Positions- und Orientierungs-Tracking	Kein Tracking
Hohe Bildauflösung	Geringe Bildauflösung
Großes <i>field of view</i> (FOV)	Kleines <i>field of view</i> (FOV)
3D-Ton	2D-Ton oder kein Ton
Hohe Bildwiederholungsrate	Geringe Bildwiederholungsrate

**Tabelle 1: Vergleich der Eigenschaften, die den Immersionseindruck erhöhen bzw. erniedrigen.**

Ein Beispiel für ein *immersive Virtual Reality System* ist der Flugsimulator. Pilot und Copilot sitzen in einem Modell eines Cockpits, in dessen Fenstern eine computergenerierte Landschaft dargestellt wird. Dies erzeugt eine realistische Simulation eines realen Cockpits. Mit herkömmlichen Ein- und Ausgabegeräten könnte dieser Eindruck nicht hervorgerufen werden.

Viele *immersive Virtual Reality Systems* verwenden sogenannte *Head Mounted Displays* (HMD) (siehe auch Kapitel 2.5.3). Diese Datenbrillen ersetzen die Umgebung des Benutzers durch eine dreidimensionale Version der virtuellen Welt. Die *Telepresence* stellt eine Anwendung dieser Technologie dar. Dabei wird die virtuelle Welt entsprechend den Meßwerten von Sensoren verändert, die sich z.B. an einem Roboter in einer Gefahrenzone befinden. Dem Benutzer wird der Eindruck vermittelt, daß er selbst an diesem Ort präsent

sei. Er kann nun auf der Basis dieser Meßdaten den Roboter steuern, ohne sich selbst in Gefahr zu begeben. Diese Technik wird auch in der Medizin angewandt. Chirurgen führen Operationen mit sehr kleinen Instrumenten durch, die über Kabel in den Körper geführt werden. Dadurch wird der Patient weniger stark beeinträchtigt als durch eine herkömmliche Operation. Bei einer *Telepresence*-Operation muß der Chirurg nicht unbedingt vor Ort sein. Auf diese Weise kann z.B. die Hilfe eines Spezialisten in Anspruch genommen werden, der sich in einem anderen Erdteil befindet.

### **2.1.3 Augmented Reality Systems**

Ein *Augmented Reality System* ermöglicht dem Benutzer einen vollständigen Blick auf die reale Welt, der zusätzlich mit virtuellen Objekten überlagert wird. Dies wird z.B. durch durchsichtige HMDs ermöglicht (siehe Kapitel 2.5.4), in denen zusätzliche Daten eingeblendet werden. Ein Beispiel für ein *Augmented Reality System* ist die *Studierstube*, die in Kapitel 4.1 detailliert beschrieben wird.

In Verbindung mit der *Telepresence* (siehe Kapitel 2.1.2) werden *Augmented Reality Systems* z.B. von Piloten verwendet [Vince, 1995]. Der Pilot sieht gleichzeitig seine unmittelbare Umgebung und z.B. topografische Daten der Landschaft.

*Projection Systems* stellen eine weitere Möglichkeit des visuellen Outputs in einem *Augmented Reality System* dar. Dabei werden die computergenerierten Bilder um den Benutzer herum projiziert. Ein Beispiel dafür ist der sogenannte CAVE (siehe Kapitel 2.5.7).

## **2.2 Aufbau eines Virtual Reality Systems**

Ein Virtual Reality System läßt sich im wesentlichen in drei Gerätegruppen aufspalten:

- Eingabegeräte (siehe Kapitel 2.4)
- Ausgabegeräte (siehe Kapitel 2.5)
- Steuer-/Grafikcomputer

Um Position und Orientierung der Ein- und Ausgabegeräte bestimmen zu können, sind Positions-Tracker (siehe Kapitel 2.3) notwendig. Darüber hinaus werden spezielle Softwaresysteme für den Entwurf der virtuellen Welt und die Steuerung des VR-Systems benötigt. Abbildung 2.1 zeigt den schematischen Aufbau eines VR-Systems.

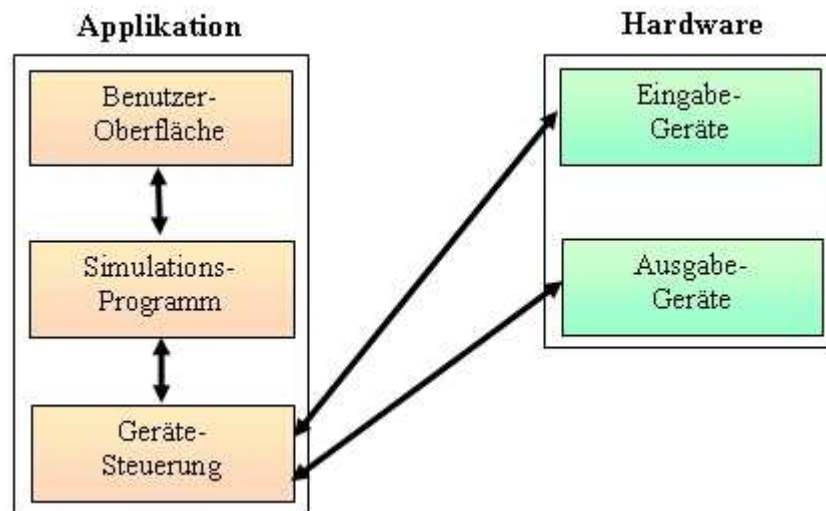


Abbildung 2.1: Schematischer Aufbau eines Virtual Reality Systems.

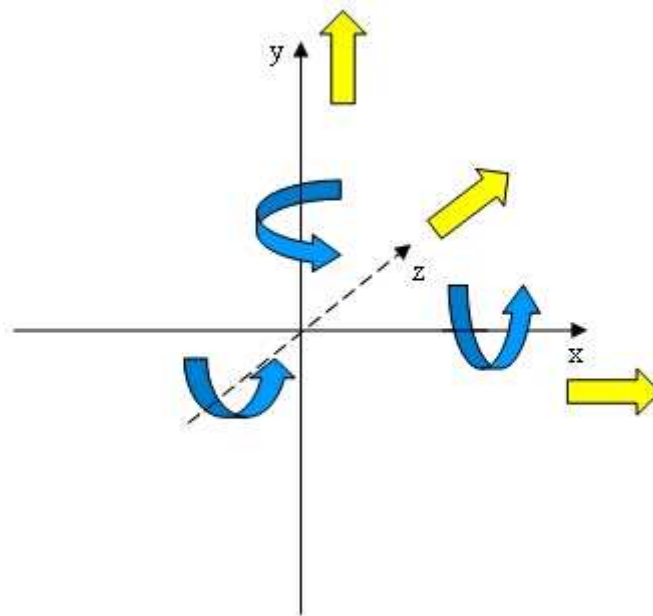
### 2.3 Tracking

Ein Virtual Reality System zeichnet sich vor allem dadurch aus, daß der Benutzer mit der virtuellen Welt interagieren kann. Um diese Interaktion zu ermöglichen, müssen dem System Position, Orientierung und Haltung der Person bekannt sein. Darüber hinaus müssen auch Informationen über Position und Orientierung spezieller Objekte, wie z.B. Ein- und Ausgabegeräte (siehe Kapitel 2.4 und 2.5), vorhanden sein.

Es gibt verschiedene Methoden, diese Bewegungen zu messen und zu analysieren. Beispiele dafür sind elektromechanische, elektromagnetische, akustische, inertielle und optische Methoden. Die Methoden unterscheiden sich nicht nur in der Art der Messung der Bewegungen, sondern auch dadurch, ob die Daten in Echtzeit gewonnen werden. Im Allgemeinen sind Echtzeitanalysen weniger genau, da aufgrund der Tatsache, daß die Ergebnisse der Berechnungen sehr schnell vorliegen müssen, nur eine begrenzte Anzahl an

Daten verarbeitet werden kann. Für ein Virtual Reality System kommen jedoch nur Echtzeitmessungen in Frage.

Um Position und Orientierung eines getrackten Körperteils oder Objekts bestimmen zu können, werden Informationen über die Bewegungen entlang von drei translatorischen und drei rotatorischen Bewegungsachsen benötigt. Die drei translatorischen Bewegungsachsen entsprechen den Bewegungen entlang der x-, y-, und z-Achsen. Sie werden in Abbildung 2.2 durch gelbe Pfeile symbolisiert. Die blauen Pfeile stellen die drei rotatorischen Bewegungsachsen dar. Diese beziehen sich auf die Rotationen um die x-, y-, und z-Achsen. Jede Bewegungsachse wird als Freiheitsgrad bezeichnet. Man benötigt also ein System mit sechs Freiheitsgraden, um Position und Orientierung tracken zu können.



**Abbildung 2.2: Es werden Informationen über die Bewegungen entlang von drei translatorischen und drei rotatorischen Bewegungsachsen benötigt.**

Elektromagnetische Systeme, die in Kapitel 2.3.2 genauer beschrieben werden, sind heutzutage sehr weit verbreitet. Elektromechanische (siehe Kapitel 2.3.1), akustische (siehe Kapitel 2.3.3) und inertielle (siehe Kapitel 2.3.4) Systeme konnten diese weite Verbreitung aus verschiedenen Gründen nicht erreichen. Optische Systeme (siehe Kapitel 2.3.5) stellen jedoch eine interessante Alternative zu elektromagnetischen Systemen dar. Optische

Methoden, die heutzutage schon erfolgreich eingesetzt werden, werden in Kapitel 2.3.5.1 beschrieben. In Kapitel 2.3.5.2 werden Forschungsarbeiten vorgestellt, die einen Blick in die Zukunft des optischen Tracking erlauben.

### **2.3.1 Elektromechanisches Tracking**

Diese Art des *Tracking* bedient sich elektromechanischer Methoden [Menache, 2000; Kolozs, 1996; Furniss, 1999], um Bewegungen zu messen. Hierbei wird einer Person ein Exoskelett, also eine mechanische Vorrichtung, die einem Skelett ähnelt, angelegt. Mit Hilfe dieser Vorrichtung wird die Bewegung als Änderung der Winkel zwischen den Gliedmaßen der Person interpretiert. Diese Änderungen werden mit Potentiometern oder ähnlichen Geräten zur Messung von Winkeln gemessen.

Der größte Nachteil dieser Systeme besteht darin, daß globale Bewegungen durch den Raum und die globale Orientierung der Person nicht gemessen werden können. Bei manchen Systemen ist das Exoskelett sogar am Boden befestigt, was die Bewegungsmöglichkeiten der Person sehr stark einschränkt. Aber auch bei Systemen, die nicht fix verankert sind, können Bewegungen durch den Raum nur mit Hilfe zusätzlicher Meßeinrichtungen gemessen werden. Eine Möglichkeit besteht darin, elektromagnetische Sensoren zur Bestimmung der Position im Raum zu verwenden. Allerdings bringt die Verwendung elektromagnetischer Sensoren auch einige Nachteile mit sich, auf die in Kapitel 2.3.2 näher eingegangen wird.

Abbildung 2.3 zeigt ein elektromechanisches Exoskelett.



**Abbildung 2.3: Ein elektromechanisches Exoskelett.**

Vorteile elektromechanischer Tracking-Methoden:

- Die Lichtverhältnisse in der Meßzone haben keine Auswirkungen auf das Ergebnis der Messung. Probleme, die bei optischen Methoden z.B. durch unzureichende Beleuchtung verursacht werden können, treten hier nicht auf.
- Solange keine elektromagnetischen Sensoren verwendet werden, wird die Messung nicht durch äußere Einflüsse, wie z.B. Magnetfelder, beeinträchtigt.
- Die verwendete Technologie muß nicht neu entwickelt werden, sondern ist schon seit längerer Zeit in Verwendung. Dadurch können die benötigten Teile preiswert hergestellt werden, wodurch elektromechanische Methoden ein sehr kostengünstiges Tracking-Verfahren darstellen. Die Kosten sind geringer als die elektromagnetischer und optischer Verfahren.
- Sofern das Exoskelett nicht am Boden befestigt ist, ist man nicht an einen bestimmten Aufnahmebereich gebunden.
- Echtzeit-Tracking ist möglich.
- Die Bewegungen mehrerer Personen können gleichzeitig gemessen werden. Dies ist z.B. bei kollaborativer *Augmented Reality* sehr hilfreich.

Nachteile elektromechanischer Tracking-Methoden:

- Das Exoskelett beeinträchtigt die Person in ihren Bewegungen. Um genaue Messungen zu erhalten, muß es sehr eng sitzen. Je enger es jedoch sitzt, desto eher fühlt sich die Person in ihren Bewegungen eingeengt.
- Es muß möglich sein, das Exoskelett an Größe und Figur verschiedener Personen anzupassen.
- Ist das Exoskelett allzu schwer und sperrig, beeinträchtigt es die Person in ihren Bewegungen.
- Vor jedem Tracking-Vorgang ist es notwendig, das System zu kalibrieren.
- Globale Bewegungen durch den Raum und die globale Orientierung der Person können nicht ohne zusätzliche Sensoren gemessen werden.
- Der Abstand zwischen einem Gelenk des Exoskeletts und dem entsprechenden Gelenk der Person stellt ein weiteres Problem dar. Da die Messungen nicht direkt am Gelenk vorgenommen werden können, kann es zu Meßungenauigkeiten kommen.
- Wenn das Exoskelett verrutscht, können ebenfalls Meßfehler auftreten.

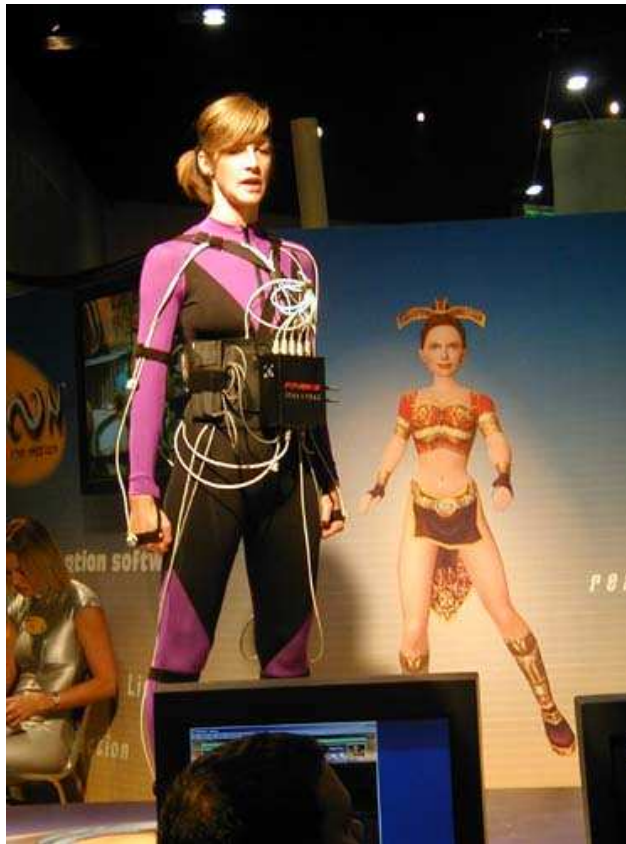
### **2.3.2 Elektromagnetisches Tracking**

Elektromagnetische Tracking-Systeme [Menache, 2000, Kolozs, 1996; Dyer, 1995; Furniss, 1999] sind sehr weit verbreitet, da sie trotz relativ geringer Kosten gute Ergebnisse liefern. Während des Tracking-Vorganges wird ein Emitter, der ein niederfrequentes elektromagnetisches Feld erzeugt, in der Nähe des Meßbereiches plaziert. Sensoren, die Änderungen im Magnetfeld erkennen können, werden an der Person befestigt. Von jedem Sensor führt ein Kabel zu einer elektronischen Kontrolleinheit, deren Aufgabe es ist, die empfangenen Daten zu filtern und zu verstärken. Im Anschluß daran berechnet ein Computer mit Hilfe dieser Daten die Position der Sensoren im Raum. Werden diese Berechnungen schnell genug durchgeführt, können elektromagnetische Systeme auch für Echtzeit-Tracking verwendet werden.

Metalle, die sich in der Nähe des Meßbereiches befinden, können das Meßergebnis beeinflussen, weil durch Induktion neue Magnetfelder entstehen können, die das vom Emitter erzeugte elektromagnetische Feld stören. Welche Metalle Störungen hervorrufen

können, hängt jedoch davon ab, ob das elektromagnetische Feld mittels Gleichstrom oder Wechselstrom erzeugt wird. Verwendet man Gleichstrom, so verfälschen vor allem eisenhaltige Metalle, wie z.B. Eisen und Stahl, das Ergebnis, nicht jedoch z.B. Aluminium oder Kupfer. Im Gegensatz dazu reagieren Magnetfelder, die mit Hilfe von Wechselstrom erzeugt wurden, sehr stark auf Aluminium, Kupfer und Kohlenstoffstahl, nicht so sehr jedoch auf eisenhaltige Metalle. Einige elektromagnetische Tracking-Systeme versuchen dieses Problem zu umgehen, indem sie Informationen über Metallvorkommen in der Nähe des Meßbereiches in ihre Berechnungen miteinbeziehen. Bestehende Magnetfelder werden vor Beginn des Tracking-Vorganges eingemessen. Die Abweichung zwischen der realen Position eines Objektes und der vom System gemessenen Position dieses Objektes wird verwendet, um die während des Tracking-Vorganges ermittelten Daten zu entzerren. Diese Methode funktioniert allerdings nur bei geringen Metallvorkommen.

Abbildung 2.4 zeigt eine Person, die eine elektromagnetische Tracking-Vorrichtung trägt.



**Abbildung 2.4: Eine elektromagnetische Tracking-Vorrichtung.**

Vorteile elektromagnetischer Tracking Methoden:

- Elektromagnetische Systeme sind sehr kostengünstig.
- Probleme, die bei optischen Systemen z.B. durch Verdeckung von Körperteilen entstehen können, treten hier nicht auf.
- Es ist möglich, die Bewegungen mehrerer Personen gleichzeitig zu messen.
- Die Sensoren sind sehr klein und leicht.

Nachteile elektromagnetischer Tracking Methoden:

- Die Kabel, die von den Sensoren zur Kontrolleinheit führen, stören den natürlichen Bewegungsablauf der Person. Komplizierte Bewegungsmuster sind eventuell nicht oder nur schwer durchführbar.
- Metall beeinflusst das Meßergebnis.
- Der Meßbereich ist auf den Bereich des vom Emitter erzeugten elektromagnetischen Feldes beschränkt.
- Mit zunehmender Distanz zum Emitter wird das Meßergebnis ungenauer.
- Sehr schnelle Bewegungen können mit diesen Systemen nicht genau gemessen werden.

### **2.3.3 Akustisches Tracking**

Akustische Tracking-Verfahren [Kolozs, 1996] nutzen die Eigenschaften von Ultraschallwellen, um Position und Bewegung einer Person zu messen. Benötigt werden dazu einerseits Sender, wie z.B. Hochfrequenzlautsprecher, und andererseits Empfänger, wie z.B. spezielle Mikrofone. Man befestigt entweder die Sender oder die Empfänger an der Person. Das jeweilige Gegenstück wird am Rand der Tracking-Zone positioniert. Verwendet man nur einen Sender und einen Empfänger, so kann man nur die Entfernung zu der Person messen. Möchte man die Position messen, verwendet man entweder drei Sender und einen Empfänger oder drei Empfänger und einen Sender. Um auch die Orientierung messen zu können, benötigt man drei Sender und drei Empfänger.

Es gibt zwei verschiedene Möglichkeiten, um Bewegungen mit Hilfe von akustischen Methoden zu messen. Entweder mißt man die Zeit, die ein Signal vom Sender zum

Empfänger benötigt, und schließt dann auf den Abstand zwischen Sender und Empfänger, oder man mißt die Phasendifferenz der Signale.

Vorteile akustischer Tracking-Methoden:

- Akustische Verfahren sind kostengünstiger als optische und magnetische Systeme.
- Die verwendeten Mikrofone sind sehr klein und leicht, sie beeinträchtigen die Person daher kaum in ihren Bewegungen.

Nachteile akustischer Tracking-Methoden:

- Hintergrundgeräusche können die Meßergebnisse beeinflussen.
- Ultraschallwellen, die an festen Oberflächen reflektiert werden, verfälschen das Meßergebnis.
- Um genaue Messungen zu erhalten, werden sehr viele Sender und Empfänger benötigt.

### **2.3.4 Inertiales Tracking**

Inertiale Tracking-Systeme [Kolozs, 1996] nutzen die Massenträgheit, um die Bewegungen einer Person zu messen. Man verwendet Beschleunigungsmesser, um die Geschwindigkeit zu messen, mit der ein Körperteil bewegt wird. Um Informationen über die Richtung der Bewegung zu erhalten, werden Gyroskope verwendet.

Vorteile inertialer Tracking-Methoden:

- Es werden keine zusätzlichen Geräte, wie z.B. Sender oder Kameras benötigt.
- Die Messungen werden nicht durch äußere Einflüsse, wie z.B. Magnetfelder, Metalle oder schlechte Lichtverhältnisse beeinflusst.
- Die verwendeten Meßgeräte sind sehr klein und behindern daher die Person nicht.

Nachteile inertialer Tracking-Methoden:

- Die Systeme müssen von Zeit zu Zeit recalibriert werden, da sich durch diese Meßmethode Meßfehler aufsummieren.
- Die verwendete Technologie ist noch nicht ausgereift.

### 2.3.5 Optisches Tracking

Optische Tracking-Systeme analysieren Aufnahmen von Bewegungen, die mit Hilfe von Kameras hergestellt werden. Dabei wird versucht, bestimmte Objekte in den Einzelbildern wiederzufinden, um dann deren dreidimensionale Koordinaten zu berechnen. Je nachdem, nach welchen Objekten in den Bildern gesucht wird, unterscheidet man

- markerbasierte (siehe Kapitel 2.3.5.1) und
- markerlose optische Systeme (siehe Kapitel 2.3.5.2).

Optische Verfahren sind im Vergleich zu anderen Systemen verhältnismäßig teuer, die Meßergebnisse sind aber im allgemeinen genauer.

Bevor mit den Aufnahmen begonnen werden kann, muß sichergestellt werden, daß die verschiedenen Kameras synchron aufnehmen, das heißt, daß alle Kameras die Aufnahmen zur selben Zeit machen. Das ist notwendig, da auch kleine Abweichungen das Meßergebnis beeinflussen können. Um auch sehr schnelle Bewegungen messen zu können, muß die *frame rate*, das ist die Anzahl der Bilder pro Sekunde, sehr hoch sein. Sie liegt üblicherweise zwischen 30 und 1000 Bildern pro Sekunde.

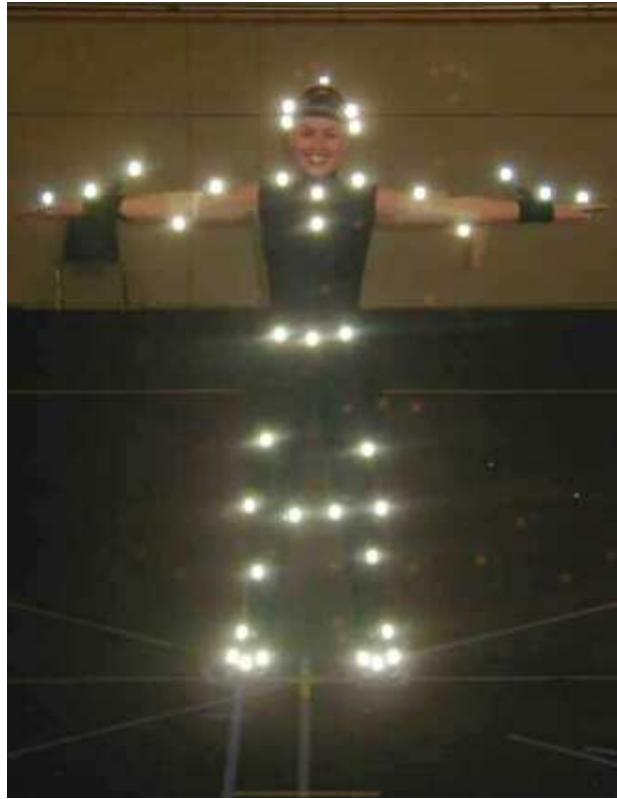
#### 2.3.5.1 Markerbasiertes optisches Tracking

Bei diesem Verfahren [Menache, 2000; Kolozs, 1996; Dyer, 1995; Furniss, 1999] werden Marker – das sind kreisförmige Objekte mit einem Durchmesser von einigen Millimetern bis hin zu mehreren Zentimetern – an der Person angebracht. Man unterscheidet aktive Marker, die infrarotes Licht ausstrahlen, und passive Marker, die meist mit einem reflektierenden Material beschichtet sind. Um die Position eines Markers im dreidimensionalen Raum rekonstruieren zu können, sind zumindest zwei Kameras notwendig. Je mehr Kameras verwendet werden, desto geringer ist die Wahrscheinlichkeit, daß Marker z.B. durch Körperteile verdeckt werden. Jede zusätzliche Kamera erhöht jedoch den Rechenaufwand.

Um die genaue Position der Kameras in Bezug zum Meßbereich festzustellen, müssen optische Systeme kalibriert werden. Dabei werden die Bewegungen eines Objektes

gemessen, dessen Ausmaße bekannt sind, wie z.B. ein mit Markern gekennzeichneter Würfel. Mit Hilfe dieser Meßergebnisse kann auf die Position der einzelnen Kameras im Raum rückgeschlossen werden. Verändert sich jedoch die Position einer Kamera, z.B. durch Vibrationen, die von vorbeigehenden Personen ausgelöst werden, können die Meßergebnisse verfälscht werden. Daher erweist es sich als günstig, die Kameras in bestimmten Zeitabständen, z.B. nach einigen Minuten, zu recalibrieren.

Abbildung 2.5 zeigt eine Person, die Marker an für die Messung günstigen Stellen trägt. Im Durchschnitt werden 20 bis 30 Marker verwendet. Eine typische Markerverteilung ist z.B. drei Marker am Kopf und je ein Marker im Genick, auf der Wirbelsäule, den Schultern, den Ellenbogen, den Handgelenken, den Händen, den Hüften, den Knien, den Knöcheln und den Füßen. Je höher die Anzahl der Marker, desto genauer ist zwar das Ergebnis, aber der Rechenaufwand wird durch jeden Marker erhöht. Aus diesem Grund wird auch im allgemeinen bei Messungen der Bewegungen des menschlichen Körpers darauf verzichtet, Marker im Gesicht der Person anzubringen. Um die Veränderungen der Mimik einer Person genau zu messen, müßte eine große Anzahl an relativ kleinen Markern verwendet werden. Diese kleinen Marker verursachen jedoch höhere Meßungenauigkeiten, da sie auf den Einzelbildern schlechter zu erkennen sind, als die etwas größeren Marker, die am übrigen Körper eingesetzt werden. Daher werden Messungen der Mimik separat durchgeführt.



**Abbildung 2.5: Günstige Markerpositionen.**

Im Anschluß an die Messungen wird versucht, die Positionen der Marker in den Einzelbildern festzustellen. Dazu werden Techniken aus der Bildverarbeitung verwendet. Reflektionen, die während des Meßvorganges ungewollt auftreten, erschweren die Rekonstruktion der Markerpositionen. Ein weiteres Problem ist die Verdeckung von Markern durch Körperteile der Person. Wurden alle Marker lokalisiert, kann von der Lage jedes Markers auf den Einzelbildern auf seine Position im dreidimensionalen Raum rückgeschlossen werden.

*ARToolKit* (siehe Kapitel 4.2) ist eine kostengünstige Variante eines optischen Tracking-Verfahrens. Es eignet sich jedoch aufgrund der Größe der Marker nur eingeschränkt, um die Bewegungen einer Person zu messen. Für die Messung von Position und Orientierung von weniger komplexen Objekten, wie z.B. *Pen & Tablet* (siehe Kapitel 2.4.4), ist es jedoch sehr gut einsetzbar.

Vorteile markerbasierter optischer Tracking-Methoden:

- Optische Systeme liefern überdurchschnittlich genaue Meßergebnisse.
- Die Person wird nicht durch Kabel oder ein Exoskelett in ihren Bewegungen beeinträchtigt.
- Ist die frame rate hoch genug, können auch sehr schnelle Bewegungen gemessen werden.

Nachteile markerbasierter optischer Tracking-Methoden:

- Sind Marker über längere Zeit hinweg verdeckt, ist es eventuell nicht mehr möglich, die Bewegungen korrekt zu rekonstruieren.
- Die korrekte Beleuchtung des Aufnahmebereiches ist von großer Bedeutung. Ungewollte Reflektionen müssen so weit wie möglich vermieden werden.
- Häufige Rekalibrierungen des Systems sind notwendig.
- Die Rekalibrierungen sind verhältnismäßig aufwendig.

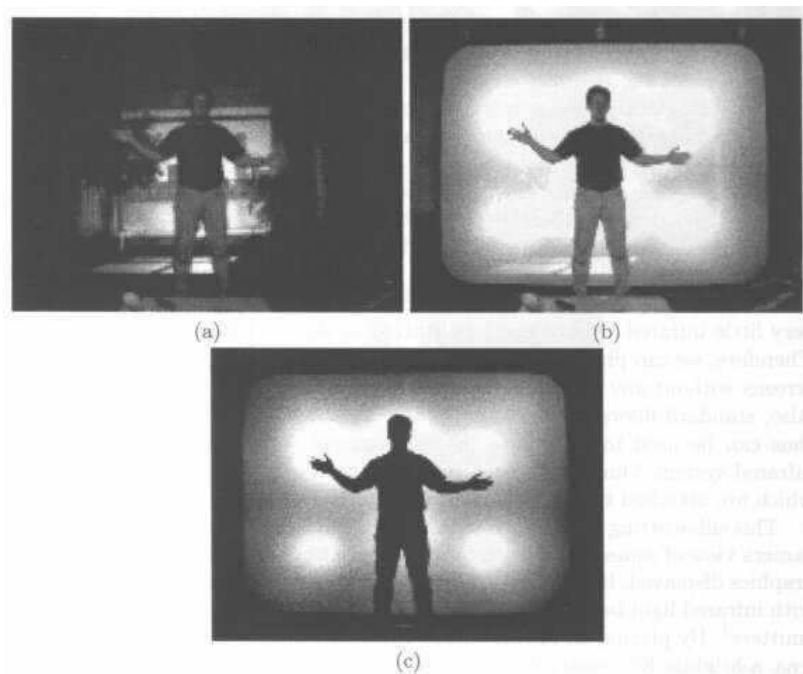
#### 2.3.5.2 Markerloses optisches Tracking

Markerlose optische Systeme versuchen die Position des menschlichen Körpers in einem Einzelbild, ohne Zuhilfenahme von Markern, zu erkennen. Es gibt verschiedene Forschungsansätze, um dies durchzuführen. Ein stabiles Verfahren, das kommerziell genutzt werden könnte, gibt es heutzutage allerdings noch nicht. Im folgenden Abschnitt werden zwei Forschungsarbeiten vorgestellt, die Prototypen markerloser optischer Systeme beschreiben. Die Anwendungsmöglichkeiten dieser Systeme sind jedoch sehr begrenzt.

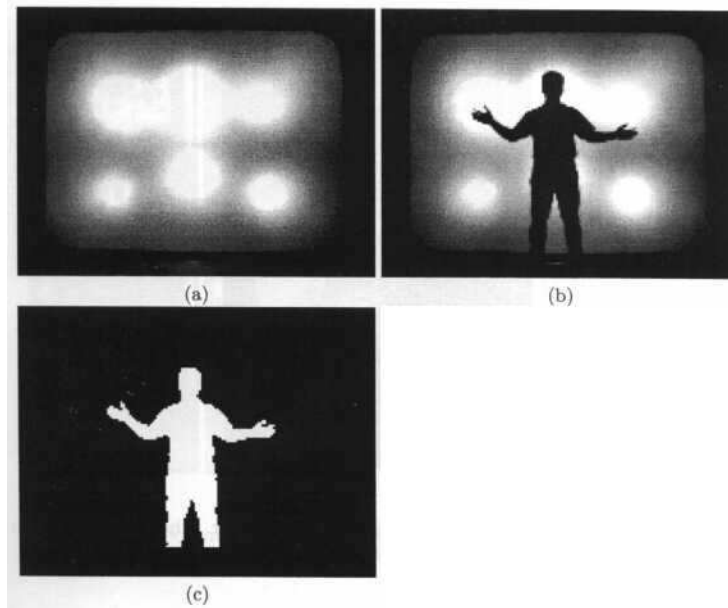
##### **Erkennen einer Silhouette mit Hilfe von Infrarot-Filtern**

Bei diesem Verfahren [Magnenat-Thalmann, 1998] bewegt sich die Person während der Aufnahme vor einer Leinwand. Hinter dieser Leinwand befinden sich einige Scheinwerfer, die infrarotes Licht in Richtung Kamera ausstrahlen. Die verwendete Kamera muß infrarotes Licht aufnehmen können. Befindet sich nun eine Person vor der Leinwand, so blockiert sie das infrarote Licht. Wird auf die entstandenen Aufnahmen ein Infrarot-Filter angewandt, so entsteht ein Bild mit einer deutlichen schwarzen Silhouette der Person. Abbildung 2.6a zeigt die Person, die vor einer Leinwand steht, auf die ein beliebiges Video projiziert wird. Diese Projektion stört den Tracking-Vorgang nicht. Abbildung 2.6b zeigt

die Aufnahme der Infrarot-Kamera. Auf diesem Bild sind noch Teile des Videos zu erkennen. Nach der Anwendung des Infrarot-Filters sind diese Teile jedoch nicht mehr zu sehen (siehe Abbildung 2.6c). Nach dem Filtervorgang wird die Silhouette mit Hilfe von Bildverarbeitungstechniken in den Einzelbildern lokalisiert. Dazu wird vor Beginn des Tracking-Vorganges eine Aufnahme der Leinwand ohne Person im Vordergrund erstellt. Alle Bilder, auf denen eine Person zu sehen ist, werden dann Pixel für Pixel mit diesem Referenzbild verglichen. Dadurch kann festgestellt werden, welche Pixel zur Silhouette gehören. Abbildung 2.7a zeigt das Referenzbild, Abbildung 2.7b ein Bild mit Person und Abbildung 2.7c die daraus rekonstruierte Silhouette.



**Abbildung 2.6: (a) Ein Foto des Aufnahmebereiches, (b) eine Aufnahme der Infrarot-Kamera, (c) die Aufnahme nach dem Filtervorgang.**



**Abbildung 2.7:** (a) Ein Referenzbild, (b) ein gefiltertes Bild, (c) die rekonstruierte Silhouette.

Zu diesem Verfahren existiert eine Beispielanwendung: ein virtueller Aerobic-Trainer. Der Benutzer kann sich ein individuelles Trainingsprogramm zusammenstellen, das automatisch startet, sobald er vor die Leinwand tritt. Mit Hilfe der Silhouetten-Erkennung kann festgestellt werden, ob sich eine Person vor der Leinwand befindet und auch wie sie die Aerobic-Übungen durchführt. Das System kommentiert in Folge die Durchführung der Übungen. Wurde eine Übung erfolgreich absolviert, fällt dieser Kommentar positiv aus, im anderen Fall wird die Person auf die Fehler in ihren Bewegungen aufmerksam gemacht.

Abbildung 2.8 zeigt ein Foto dieses virtuellen Aerobic-Trainingssystems.



**Abbildung 2.8: Ein virtuelles Aerobic-Trainingssystem.**

Vorteile der Silhouetten-Erkennung mit Hilfe von Infrarot-Filtern:

- Es müssen keine Vorbereitungen an der Person direkt vorgenommen werden.
- Es ist nicht notwendig, daß die Person eine spezielle Kleidung trägt.
- Die Analyse der Bilder erfolgt in Echtzeit.
- Es wird nur eine Kamera benötigt.

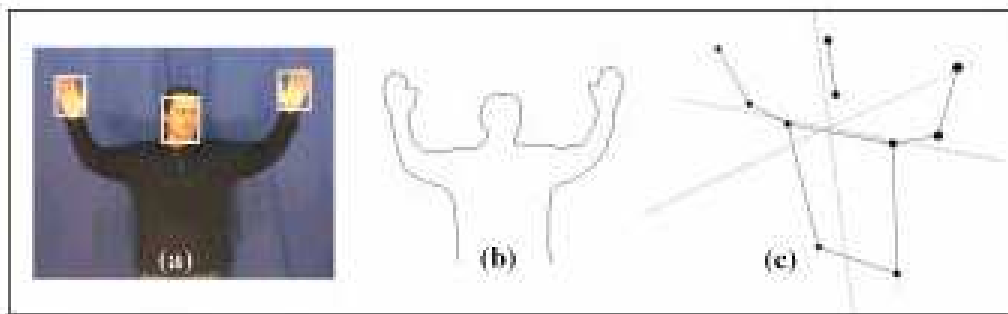
Nachteile der Silhouetten-Erkennung mit Hilfe von Infrarot-Filtern:

- Der Aufnahmebereich ist auf den Raum vor der Leinwand beschränkt.
- Das Aufnehmen mehrerer Personen kann zu Problemen durch Überschneidung der Silhouetten führen.
- Es werden nur Bewegungen im zweidimensionalen Raum analysiert.

### **Erkennen einer Silhouette mit Hilfe von anatomischen Daten**

Dieses Verfahren [Bowden, 1998; Bowden, 2000] beruht auf der Kombination von Informationen über den menschlichen Körper und der Lokalisierung von Kopf und Händen

einer Person in einem Bild. Ein mathematisches Modell beschreibt die möglichen Positionen des menschlichen Körpers und Informationen über dessen anatomische Struktur. Bildverarbeitungstechniken ermöglichen das Auffinden markanter Körperteile – Kopf und Hände einer Person – in den Einzelbildern. Die Person steht vor einem einfarbigen Hintergrund. Dadurch kann der Umriß der Person vom Hintergrund unterschieden werden. Innerhalb dieser Silhouette wird nun nach hautfarbenen Flächen gesucht. Zu diesem Zweck wird vor Beginn des Tracking-Vorganges eine Aufnahme der Haut der Person erstellt. Mit Hilfe des mathematischen Modells kann anschließend die Position der Person im dreidimensionalen Raum rekonstruiert werden. Abbildung 2.9a zeigt ein Einzelbild der Person, Abbildung 2.9b die Silhouette der Person und Abbildung 2.9c das daraus generierte dreidimensionale Modell.



**Abbildung 2.9: (a) Position von Kopf und Händen, (b) die Silhouette, (c) das generierte dreidimensionale Modell.**

Vorteile der Silhouetten-Erkennung mit Hilfe von anatomischen Daten:

- Nur eine einzige Kamera ist notwendig.
- Es müssen keine Vorbereitungen an der Person getroffen werden.

Nachteile der Silhouetten-Erkennung mit Hilfe von anatomischen Daten:

- Der Aufnahmebereich ist auf den Raum vor dem einfarbigen Hintergrund beschränkt.
- Die Kleidung der Person muß langärmlig sein und eine andere Farbe als der Hintergrund haben.
- Wechselt die Person, so muß das System erst auf die neue Hautfarbe eingestellt werden.

- Das Aufnehmen mehrerer Personen kann zu Problemen bei der Rekonstruktion des dreidimensionalen Modells führen.
- Die Ergebnisse sind weniger genau als bei markerbasierten Systemen.

## 2.4 Eingabegeräte

Die klassischen Eingabegeräte des Computers, wie Tastatur, Joystick oder Maus, sind für Virtual Reality Systeme nicht praktikabel. Da sich der Benutzer eines solchen Systems in einem dreidimensionalen Raum bewegt, muß auch das Eingabegerät eine dreidimensionale Auflösung anbieten. Aus diesem Grund wurden spezielle Eingabegeräte entwickelt, die die natürliche Bewegung der Hand bzw. des Körpers in diskrete Daten umwandeln. Zusätzlich erlauben Positions-Tracker (siehe Kapitel 2.3) ein räumliches Orten des Benutzers bzw. der Ein- und Ausgabegeräte. In den folgenden Kapiteln werden zunächst das Funktionsprinzip und anschließend die wesentlichen Vor- und Nachteile typischer Eingabegeräte erläutert.

### 2.4.1 Datenhandschuh

Der Datenhandschuh gestattet es dem Benutzer, mit einem grafischen Abbild seiner Hand, das simultan den Bewegungen der natürlichen Hand folgt, virtuelle Gegenstände im Cyberspace zu greifen oder auf andere Weise zu manipulieren. Darüber hinaus kann der Datenhandschuh zur Erkennung von Gesten benutzt werden (siehe Kapitel 2.6.3.4). Um die Bewegung der Hand zu digitalisieren, werden folgende Informationen benötigt:

- Wo befindet sich die Hand im Raum?
- Wie wird die Hand gehalten?
- Wie stehen die Finger zueinander?

Die absolute Lage und Ausrichtung der Hand im Raum kann mit einem Positions-Tracker (siehe Kapitel 2.3) ermittelt werden, der in der Regel auf dem Handschuh Rücken befestigt ist. Um die Stellung der Fingergelenke zu messen, wurden verschiedene Verfahren entwickelt. Beim *DataGlove* [Bormann, 1994] verlaufen auf dem Handrücken und auf der Oberseite der einzelnen Finger Glasfaserschlaufen, die zur Messung der Fingerkrümmung dienen. Pro Finger werden die Krümmung des ersten Fingergliedgelenks und die des

Fingerknöchels gemessen. An einem Ende der Glasfaser sendet eine Leuchtdiode Licht in die Glasfaser, die schlaufenförmig über dem zu messenden Gelenk verläuft. Am anderen Ende der Glasfaser mißt eine Fotodiode die Stärke des ausgesandten Lichts. Je stärker das Gelenk gekrümmt ist, desto weniger ausgesandtes Licht trifft auf die Fotodiode.

Abbildung 2.10 zeigt einen Datenhandschuh, bei dem die Glasfaserschlaufen in den Handschuh eingearbeitet wurden.



**Abbildung 2.10: Ein Datenhandschuh, bei dem die Glasfaserschlaufen in den Handschuh eingearbeitet wurden.**

Auf der Basis eines mechanischen Tracking Systems (siehe Kapitel 2.3.1) kombiniert mit magnetischen Trackern (siehe Kapitel 2.3.2) wurde der sogenannte „Dextrous Hand Master“ entwickelt. Dabei handelt es sich um ein Exoskelett, das vorwiegend aus Aluminium besteht. Das Gerät wird an jedem Fingergelenk mit Hilfe eines Bandes befestigt. Die einzelnen Gelenke sind durch bewegliche Scharniere miteinander verbunden, in denen sich je ein kleiner Magnet und ein magnetischer Sensor befinden. Je nach Krümmung der Finger bewegt sich der Magnet zum Sensor hin oder von ihm weg. Da der Sensor seine Spannung proportional zur Stärke des ihn umgebenden magnetischen Feldes verändert, kann die Stellung der Gelenke ermittelt werden.

Vorteil von Datenhandschuhen:

- Datenhandschuhe erlauben eine sehr intuitive Form der Interaktion mit virtuellen Objekten.

Nachteile von Datenhandschuhen:

- Um genaue Messungen zu gewährleisten, müssen Datenhandschuhe an jeden Benutzer angepaßt werden, da die Hände und Finger unterschiedlich groß sind.
- Bei der Verwendung eines Exoskeletts ist das Anlegen des Gerätes sehr zeitaufwendig. Außerdem kann der Benutzer bei angelegtem Exoskelett kaum noch andere Tätigkeiten mit seiner Hand ausführen.
- Die verwendeten Gesten und deren Bedeutung für das System müssen vom Benutzer erlernt werden.
- Die Software muß in der Lage sein, die Gesten der Hand zu erkennen und zu deuten.

#### **2.4.2 Pinchglove**

Die große Zahl an möglichen Gesten, die mit einem Datenhandschuh (siehe Kapitel 2.4.1) interpretiert werden können, ist für viele Anwendungen nicht notwendig. In diesem Fall empfiehlt es sich, *Pinchgloves* [LaViola, 1999; Bowman, 2001a] zu verwenden. Dabei handelt es sich um leichte Handschuhe, die nicht die Krümmung, sondern das Aneinanderdrücken von Fingern messen. An den Fingerkuppen befinden sich elektrische Kontakte (siehe Abbildung 2.11). Zusätzlich verfügen sie über Tracker zur Bestimmung von Position und Orientierung der Hand. Der Benutzer kann dadurch z.B. ein virtuelles Objekt mit Daumen und Zeigefinger ergreifen.



**Abbildung 2.11: Pinchgloves verfügen über elektrische Kontakte an den Fingerkuppen.**

Vorteile von Pinchgloves:

- Sie sind leichter und angenehmer zu tragen als Datenhandschuhe.
- Die Software ist weniger komplex, da sie nicht in der Lage sein muß, Gesten zu erkennen.

Nachteil von Pinchgloves:

- Wenn Pinchgloves von mehreren Personen benutzt werden sollen, kann es zu Problemen mit unterschiedlichen Handgrößen kommen.

### **2.4.3 3D-Maus**

Die Funktionsweise der 3D-Maus [Vince, 1995] ist der der 2D-Maus sehr ähnlich. Sie ermöglicht jedoch, im Gegensatz zur 2D-Maus, Bewegungen entlang von sechs Freiheitsgraden. Position und Orientierung der 3D-Maus werden durch Tracker (siehe Kapitel 2.3) ermittelt. Zusätzlich verfügt sie über Buttons, die für verschiedene Funktionen, wie z.B. zur Fortbewegung in der virtuellen Welt oder zum Selektieren von Objekten, verwendet werden können.

Vorteile der 3D-Maus:

- Das Konzept der Maus als Eingabegerät ist aus konventionellen Computersystemen bekannt.
- 3D-Mäuse sind meist sehr klein und leicht.

Nachteil der 3D-Maus:

- Die Funktionsweise der Buttons muß vom Benutzer erlernt werden.

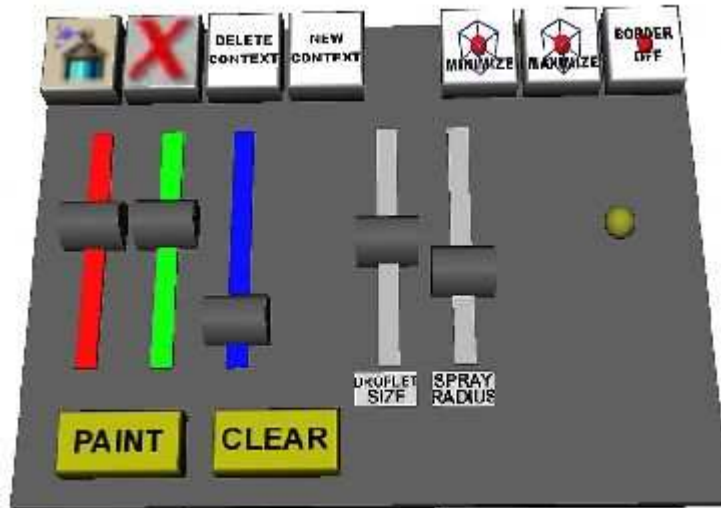
#### **2.4.4 Pen & Tablet**

3D-Eingabegeräte wie der Datenhandschuh (siehe Kapitel 2.4.1) und die 3D-Maus (siehe Kapitel 2.4.3) erlauben zwar Bewegungen entlang von sechs Freiheitsgraden, sie ermöglichen jedoch kein haptisches Feedback (siehe Kapitel 2.5.9). Ein Benutzer kann also z.B. ein virtuelles Objekt manipulieren, er hat dabei aber nicht das Gefühl, dieses Objekt auch wirklich angreifen zu können. Unerfahrene Benutzer finden es oft schwierig, mit Objekten zu interagieren, die ausschließlich in der virtuellen Welt existieren. *Pen & Tablet* (siehe Abbildung 2.12) wurden entwickelt, um die Interaktion mit virtuellen Objekten zu vereinfachen. Dabei handelt es sich um ein Tablett und einen Stift, die in jeweils einer Hand gehalten werden.



**Abbildung 2.12: Bei *Pen & Tablet* handelt es sich um ein Tablett und einen Stift, die in jeweils einer Hand gehalten werden.**

Sobald das Tablett im Blickfeld des Benutzers erscheint, werden darauf Interaktionselemente, wie z.B. Buttons oder Sliders angezeigt (siehe Abbildung 2.13). Der Benutzer kann diese mit Hilfe des Stifts betätigen.



**Abbildung 2.13:** Sobald das Tablett im Blickfeld des Benutzers erscheint, werden darauf Interaktionselemente, wie z.B. Buttons oder Sliders angezeigt.

Vorteile von Pen & Tablet:

- Durch das physische Vorhandensein von Tablett und Stift liefert dieses Eingabegerät haptisches Feedback und ermöglicht daher eine *tangible Interaction*, d.h., die virtuellen Interaktionselemente werden greifbar bzw. fühlbar.
- Buttons, Sliders u.ä. sind aus dem Umgang mit konventionellen Computersystemen bekannt.

Nachteil von Pen & Tablet:

- Der Benutzer muß das Tablett und den Stift in je einer Hand halten, dadurch kann er keine anderen Tätigkeiten mit den Händen ausführen. Es ist allerdings möglich, Tablett und Stift abzulegen.

### 2.4.5 Alternative Eingabegeräte

Neben den bereits beschriebenen Eingabegeräten gibt es noch eine Reihe weiterer Konstruktionen, die in diesem Kapitel kurz vorgestellt werden.

Der Datenanzug (DataSuit) stellt eine Weiterentwicklung des Datenhandschuhs (siehe Kapitel 2.4.1) dar. An einem elastischen Ganzkörperanzug werden Sensoren angebracht, mit deren Hilfe Position und Orientierung des gesamten Körpers ermittelt werden. Je nach Anzahl der Sensoren können mehr als fünfzig Freiheitsgrade zwischen Hals und Fußknöchel gemessen werden. Diese große Menge an Sensoren macht den Datenanzug jedoch fehleranfällig. Der Anzug verrutscht leicht auf dem Körper, sodaß in Folge die Sensoren nicht mehr präzise über dem zu messenden Gelenk liegen. Außerdem müssen die Bedeutungen der einzelnen Körperhaltungen definiert, vom Benutzer erlernt und vom System interpretiert werden. Da für die Interaktion mit virtuellen Objekten oft nicht Position und Orientierung des gesamten Körpers benötigt werden, wird der Datenanzug meist nur für spezielle Anwendungen, die dies auch tatsächlich erfordern, eingesetzt.

So wie die 3D-Maus eine Erweiterung der 2D-Maus darstellt, ist der *Forceball* eine Erweiterung des 2D-Joysticks. Er besteht aus einem Ball, der auf einer Halterung montiert ist. Dieser Ball kann gekippt, gedreht und auf- und abbewegt werden und ermöglicht so eine Interaktion entlang von sechs Freiheitsgraden.

Der *Wand* ist eine Variation der 3D-Maus. Er verfügt über Buttons und Thumbwheels, mit denen entsprechende Funktionen gesteuert werden können.

Beim *Poolball* handelt es sich um eine ausgehöhlte Billardkugel, in deren Inneren sich ein Positionssensor befindet. Auf der Kugeloberfläche befinden sich Buttons für Klickoperationen.

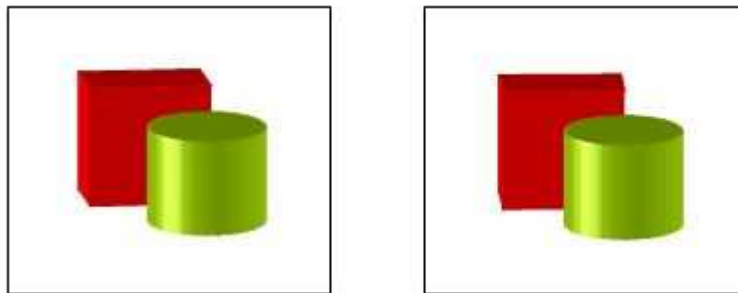
Da VR-Anwendungen den Benutzer durch die visuelle Informationsfülle sowie durch die manuelle Eingabe sehr fordern, ist die Sprache als zusätzliche Eingabemöglichkeit nützlich und entlastend. Unerfahrene Benutzer stellen jedoch möglicherweise unvollständige Anfragen an das System oder möchten nicht-implementierte Funktionen aufrufen, da sie

einem natürlichsprachlichen System eine höhere Intelligenz zuordnen, als dieses tatsächlich besitzt.

## 2.5 Ausgabegeräte

Zu den VR-Ausgabegeräten zählen alle Apparate, die auf einen der menschlichen Sinne Einfluß nehmen können. Da das Sehen der wichtigste Sinn des Menschen ist und daher eine entsprechend große Rolle bei der Wahrnehmung der Umwelt spielt, stellen visuelle Ausgabegeräte einen Hauptbestandteil eines VR-Systems dar. Auditives und haptisches Feedback können bis zu einem gewissen Grad in ein VR-System integriert werden, olfaktorische und gustatorische Ausgabegeräte werden bis jetzt allerdings noch nicht verwendet.

Da herkömmliche zweidimensionale Ausgabegeräte, wie z.B. ein Computermonitor, dem Benutzer nicht das Gefühl vermitteln können, sich innerhalb einer virtuellen Welt zu befinden, wurden spezielle Ausgabegeräte entwickelt. Um dem Benutzer einen räumlichen Eindruck zu vermitteln, erzeugen diese Geräte stereoskopische Bilder [Ware, 1996]. Dabei werden den Augen zwei perspektivisch verschobene Ansichten eines Bildes präsentiert (siehe Abbildung 2.14). Die Verschiebung der Bilder entspricht dem Abstand der Augen. Die verschiedenen Verfahren der stereoskopischen Darstellung und deren Vor- und Nachteile werden in den folgenden Kapiteln erläutert.



**Abbildung 2.14: Zwei perspektivisch verschobene Ansichten derselben Szene.**

### 2.5.1 Shutter Glasses

Verschlußgesteuerte Brillen (*Shutter Glasses* [Jacobson, 1994]) werden in Kombination mit einem herkömmlichen Monitor verwendet. Auf dem Monitor wird abwechselnd das Bild für das linke und für das rechte Auge dargestellt. Wenn das Bild für das linke Auge dargestellt wird, wird die Sicht des rechten Auges auf den Bildschirm blockiert und umgekehrt, so daß jedes Auge immer nur das ihm zugeordnete Bild sehen kann. Durch die Schnelligkeit der Wechsel ergibt sich für den Betrachter ein stereoskopischer Effekt. Um den Blickwinkel der Darstellung an den des Benutzers anzupassen, können Position und Orientierung der Shutter glasses mit Hilfe von Sensoren bestimmt werden.

Vorteile von Shutter Glasses:

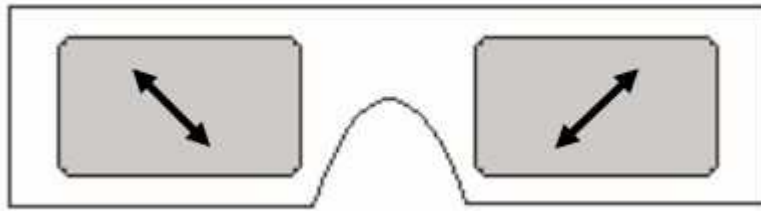
- Shutter Glasses sind im Vergleich zu anderen VR-Ausgabegeräten sehr kostengünstig.
- Die Bildqualität ist sehr hoch.

Nachteil von Shutter Glasses:

- Der Benutzer betrachtet die virtuelle Welt wie durch ein Fenster (siehe Kapitel 2.1.1).

### 2.5.2 Polarisations-Brillen

Dieses Ausgabegerät verwendet polarisiertes Licht, um zwischen den verschiedenen Ansichten der stereoskopischen Darstellung zu unterscheiden. Dabei werden das linke und das rechte Bild orthogonal zueinander polarisiert. Die Polarisations Ebenen können z.B. bei +45 und -45° von der vertikalen Achse liegen (siehe Abbildung 2.15). Spezielle Polarisations-Brillen [Woods, 2001] filtern dieses polarisierte Licht, bevor es das Auge erreicht.



**Abbildung 2.15:** Die Polarisierungsebenen liegen z.B. bei  $+45^\circ$  und  $-45^\circ$  der vertikalen Achse.

Vorteil von Polarisations-Brillen:

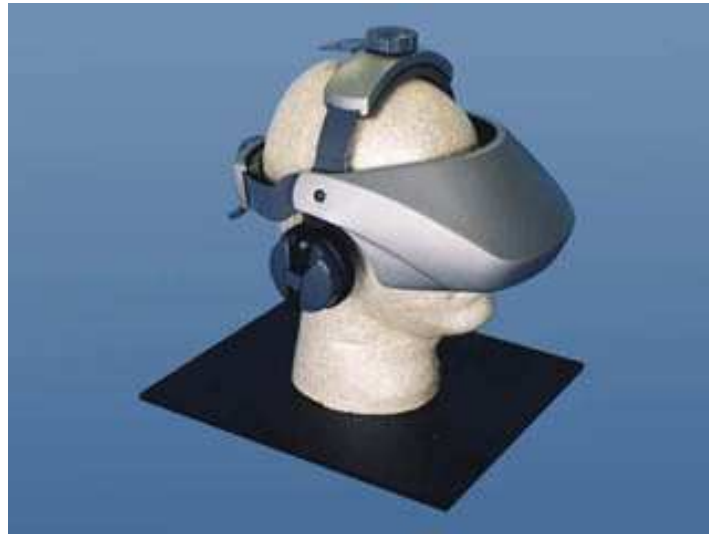
- Polarisations-Brillen sind sehr leicht und behindern den Benutzer daher kaum.
- Sie sind im Vergleich zu anderen Ausgabesystemen sehr kostengünstig.

Nachteil von Polarisations-Brillen:

- Es ist ein spezielles polarisierendes, stereoskopisches Projektionssystem notwendig.

### 2.5.3 Head Mounted Display

Ein *Head Mounted Display* (HMD) (siehe Abbildung 2.16) besteht aus zwei kleinen Bildschirmen, die vor je einem Auge positioniert sind. Als Bildschirme kommen sowohl herkömmliche Kathodenstrahlröhren (*cathod ray tube* – CRT) als auch Flüssigkristall-Bildschirme (*liquid crystal displays* – LCD) in Betracht. Der Nachteil an Kathodenstrahlröhren ist ihre Größe und das damit verbundene große Gewicht. Die Bildqualität ist jedoch im allgemeinen besser als bei Flüssigkristall-Bildschirmen.



**Abbildung 2.16: Ein Head Mounted Display.**

Da sich die Bildschirme nur vor, aber nicht seitlich der Augen befinden, wird das Blickfeld des Benutzers nicht vollständig ausgefüllt. Die Größe des Blickfelds wird durch das *field of view* (FOV) spezifiziert. Das FOV entspricht dem Winkel, der für das Auge sichtbar ist. Das menschliche Auge hat ein horizontales FOV von ungefähr  $180^\circ$  und ein vertikales FOV von ungefähr  $150^\circ$ . Das horizontale FOV von HMDs liegt bei  $45$  bis  $110^\circ$ , das vertikale bei  $25$  bis  $65^\circ$ . Je geringer das FOV, desto eher entsteht der Eindruck, durch ein Fernglas oder eine Röhre zu blicken.

Durch Positions- und Orientierungs-Tracker, die am HMD befestigt sind, wird das dargestellte Bild an den Blickwinkel des Benutzers angepaßt.

Vorteile von HMDs:

- HMDs vermitteln dem Benutzer das Gefühl, ein Teil der virtuellen Welt zu sein.
- Sie ermöglichen gutes stereoskopisches Sehen.

Nachteile von HMDs:

- Da HMDs die reale Umgebung des Benutzers durch eine virtuelle ersetzen, kann er nicht mehr angemessen auf Ereignisse in seinem Umfeld reagieren.

- Das Blickfeld des Betrachters ist eingeschränkt, da HMDs nicht das gesamte FOV abdecken.

#### **2.5.4 See Through Head Mounted Display**

Das *See Through Head Mounted Display* entspricht im Aufbau dem HMD (siehe Kapitel 2.5.3). Es werden jedoch spezielle durchsichtige Bildschirme verwendet, die es ermöglichen, das reale Bild mit Computergrafik zu überlagern.

Vorteil von See Through HMDs:

- Der Benutzer wird nicht von der Außenwelt abgeschirmt und kann daher auch mit dieser interagieren.

Nachteil von See Through HMDs:

- Das Gefühl des Eintauchens in die virtuelle Realität ist nicht so stark wie bei immersiven HMDs.

#### **2.5.5 Arm Mounted Display**

Ein *Arm Mounted Display* wird im Vergleich zum *Head Mounted Display* (siehe Kapitel 2.5.3) nicht am Kopf des Benutzers, sondern an einem Gelenkarm befestigt. Dadurch können die im Vergleich zu LCDs schwereren CRTs verwendet werden, die ein besseres Bild mit höherer Auflösung liefern. Der Benutzer wird dabei nicht durch das Gewicht der CRTs belastet. Ein Beispiel dafür ist der *Binocular Omni-Oriented Monitor* (BOOM) [Bryson, 1991], der in Abbildung 2.17 zu sehen ist. Dabei wird ein HMD an einem Gelenkarm mit einem Gegengewicht montiert. Der Benutzer kann den Gelenkarm mit der Hand bewegen. Position und Orientierung des Kopfes lassen sich durch Sensoren, die die Stellung des Gelenkarms ermitteln, sehr genau messen.



**Abbildung 2.17: Beim BOOM wird ein HMD an einem Gelenkarm mit einem Gegengewicht montiert.**

Vorteile des Arm Mounted Displays:

- Es besteht die Möglichkeit, daß der Benutzer eine bestimmte Sicht der virtuellen Welt einem anderen Benutzer zeigen kann, da der Gelenkarm die Bildschirme fixiert.
- Die Auflösung ist höher als bei LCDs.

Nachteil des Arm Mounted Displays:

- Die Bewegungsfreiheit des Benutzers ist durch den Gelenkarm stark eingeschränkt.

### **2.5.6 Virtual Retinal Display**

Das *Virtual Retinal Display* (VRD) [Pryor, 1998] wurde entwickelt, um die Einschränkungen herkömmlicher visueller Ausgabegeräte zu umgehen. Dabei wird moduliertes Laserlicht direkt auf die Retina des Benutzers gestrahlt. Die Auflösung dieses Ausgabegeräts ist sehr hoch, außerdem ist das Bild flimmerfrei.

Vorteil des VRD:

- Das Bild wird direkt auf das Auge projiziert, daher wird das Blickfeld des Benutzers nicht eingeschränkt.

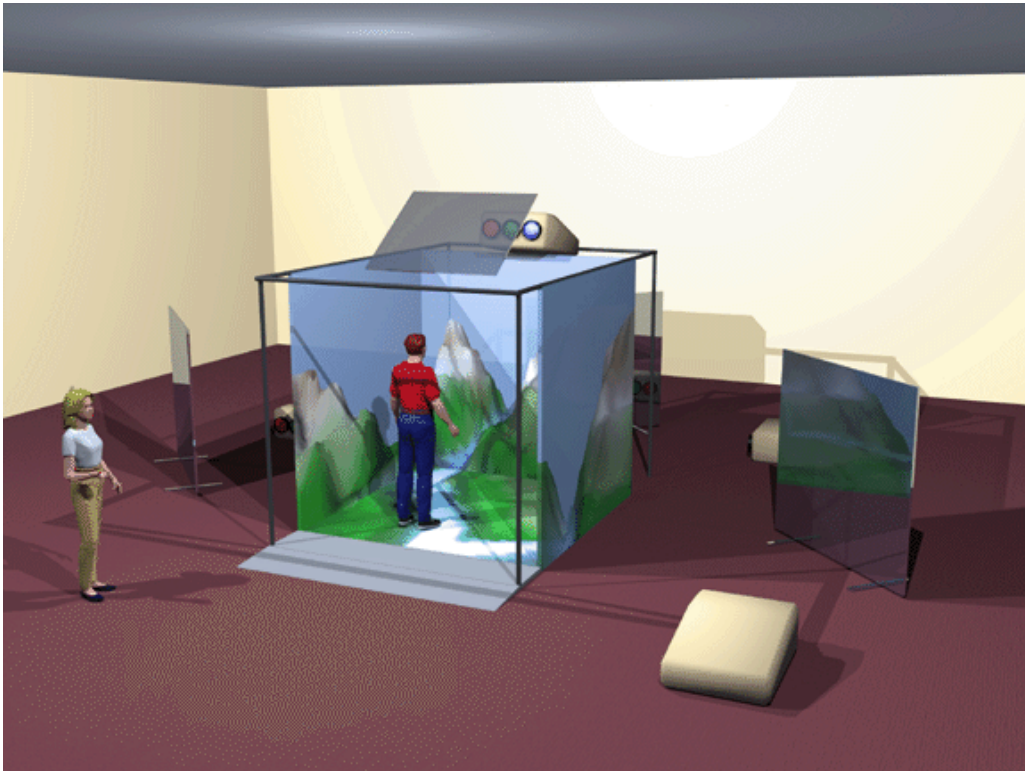
Nachteil des VRD:

- Das VRD befindet sich noch im Entwicklungsstadium.

### **2.5.7 Projektionssysteme**

Die Verwendung einer Projektionswand erlaubt es, eine virtuelle Welt mehreren Personen gleichzeitig zugänglich zu machen. Um einen räumlichen Eindruck zu erzielen, werden Shutter Glasses (siehe Kapitel 2.5.1) oder Polarisations-Brillen (siehe Kapitel 2.5.2) verwendet [Roehl, 1994]. Für eine Präsentation vor einem großen Publikum kann eine Rückprojektionswand verwendet werden [Fuhrmann, 2001]. Der Präsentator steht dabei links oder rechts von der Projektionswand. Die Zuseher sehen ein dreidimensionales Bild der Präsentation. Der Blickpunkt für die Darstellung liegt dabei in der Mitte des Zuschauerraumes. Dadurch ist der Stereo-Effekt für alle Zuseher ungefähr gleich gut.

Es können auch mehrere Projektionswände verwendet werden, um den Benutzer stärker in die virtuelle Welt einzubinden. Ein Beispiel dafür ist der *Cave Automatic Virtual Environment* (CAVE) [Cruz-Neira, 1993; Pape, 1996], der in Abbildung 2.18 zu sehen ist. Dieser wurde im *Electronic Visualization Laboratory* (EVL) der University of Illinois, Chicago entwickelt. Der CAVE ist ein würfelförmiger Raum, in dem an drei bis sechs Wänden stereoskopische Grafik projiziert wird. Für die seitlichen Wände werden Rückprojektionswände verwendet. Die Grafik für den Boden wird von oberhalb des CAVEs projiziert.



**Abbildung 2.18: Der Cave.**

Die Personen im CAVE tragen Shutter Glasses (siehe Kapitel 2.5.1), um den 3D-Effekt wahrnehmen zu können. Da es sich dabei um nicht-immersive Ausgabegeräte handelt, können die Personen sich selbst, die übrigen Personen und auch alle realen Objekte, die sich im CAVE befinden, sehen. Die virtuellen Objekte befinden sich aufgrund der stereoskopischen Darstellung innerhalb des CAVEs. Die Personen können sich im CAVE bewegen, um die virtuellen Objekte genauer zu betrachten. Dabei kann es jedoch zu Problemen mit der stereoskopischen Darstellung kommen, wenn die Sicht auf eine Projektionswand durch eine andere Person behindert wird, oder wenn ein Objekt zwischen zwei Personen projiziert werden sollte.

Die Darstellung der Szene kann nur an den Blickwinkel eines einzigen Benutzers angepaßt werden. Die Shutter Glasses dieses Benutzers sind mit einem Positions- und Orientierungs-Tracker versehen. Die übrigen Personen agieren somit als Zuseher. Die Interaktion mit dem System erfolgt durch einen *Wand* (siehe Kapitel 2.4.5) und virtuelle Interaktionselemente.

Vorteil von Projektionssystemen:

- Projektionssysteme erlauben es einer Gruppe von Personen, sich frei in der virtuellen Welt zu bewegen.

Nachteil von Projektionssystemen:

- Sie benötigen sehr viel Rechenleistung und sind im Vergleich zu anderen Ausgabegeräten sehr teuer.

### **2.5.8 Virtual Table**

Der *Virtual Table* (VT) ist ein Monitor in der Größe eines Tisches. Der stereoskopische Effekt wird durch die Verwendung von Shutter Glasses (siehe Kapitel 2.5.1) erzeugt.

Vorteil des Virtual Tables:

- Durch die Größe der Darstellung können mehrere Personen gleichzeitig die virtuelle Welt betrachten.

Nachteile des Virtual Tables:

- Wird die Darstellung an den Blickwinkel eines bestimmten Benutzers angepaßt, kommt es für die übrigen Benutzer zu perspektivischen Verzerrungen.
- Die Bewegungsmöglichkeit der Benutzer ist eingeschränkt, da der VT fix am Boden montiert ist.

### **2.5.9 Haptisches Feedback**

Der Tastsinn ist der zweitwichtigste Sinn des Menschen, daher ist es naheliegend, VR-Systeme durch haptisches Feedback zu erweitern. Ein optimales haptisches Rückkopplungssystem müßte den gesamten Körper bedecken, da die taktilen Rezeptoren in der Haut lokalisiert sind [Bormann, 1994]. Die Entwicklung haptischer Ausgabegeräte konzentriert sich derzeit jedoch hauptsächlich auf die Haptik der Hand. Dabei wird z.B. ein Datenhandschuh (siehe Kapitel 2.4.1) durch aufblasbare Luftpolster erweitert. Durch ein entsprechendes Aufblasen der Luftpolster können unterschiedliche haptische Reize erzeugt werden.

### 2.5.10 Auditives Feedback

Das Gehör ist nach Sehen und Tasten der drittichtigste Sinn des Menschen. Daher kann auditives Feedback sehr dazu beitragen, die Wirklichkeitstreue eines VR-Systems zu erhöhen. Insbesondere die Orientierung in der virtuellen Welt kann durch einen Raumton unterstützt werden. Dabei wird z.B. der Ton lauter, je näher man der Tonquelle kommt. Geräusche können aber z.B. auch auf verborgene Objekte hinweisen oder beim Berühren von Objekten entstehen.

## 2.6 3D-Interaktionsformen

Die Interaktion mit einer virtuellen Szene gliedert sich im Wesentlichen in drei Interaktionsformen: Navigation (siehe Kapitel 2.6.1), Manipulation (siehe Kapitel 2.6.2) und Systemsteuerung bzw. Applikationskontrolle (siehe Kapitel 2.6.3). Da sich die Interaktion mit einem dreidimensionalen System von der mit einem herkömmlichen zweidimensionalen System unterscheidet, werden in den folgenden Kapiteln spezielle 3D-Interaktionsformen beschrieben.

### 2.6.1 Navigation

Es gibt unterschiedliche Formen der Navigation in einer virtuellen Welt. *Travel* bezeichnet die Fortbewegung von einem Ort zu einem anderen. Erfolgt die Erkundung der Szene zielgerichtet, wird sie als *Search* bezeichnet. Das Durchschreiten einer Szene ohne ein bestimmtes Ziel wird *Exploration* genannt. Die folgenden Kapitel erläutern einige gebräuchliche Navigationsmetaphern.

#### 2.6.1.1 Gaze-directed Steering

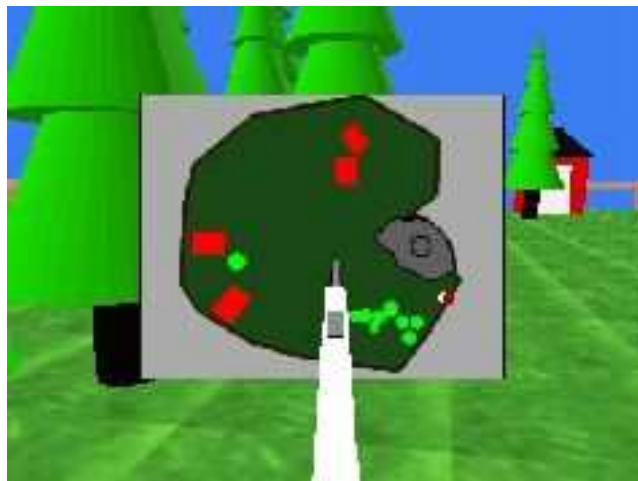
Beim *gaze-directed steering* [Mine, 1995] erfolgt die Navigation entlang der Blickrichtung des Benutzers. Im Allgemeinen wird dabei jedoch nicht die Stellung der Augen, sondern die Stellung des Kopfes interpretiert, die mit Hilfe von Sensoren ermittelt wird. Beim *gaze-directed steering* handelt es sich um eine sehr einfache Form der Navigation. Der Benutzer hat jedoch während der Fortbewegung nicht die Möglichkeit, auf die Seite zu blicken.

### 2.6.1.2 Pointing

*Pointing* ähnelt der blickgesteuerten Fortbewegung (siehe Kapitel 2.6.1.1). Die Bewegungsrichtung wird dabei aber nicht durch die Stellung des Kopfes, sondern durch die Orientierung der Hand angezeigt. Dieses Verfahren wird daher auch als manuelle Fortbewegung bezeichnet. Für den ungeübten Benutzer ist diese Form der Navigation ungewohnt, dafür kann er aber während der Fortbewegung den Kopf frei bewegen.

### 2.6.1.3 Map-based Travel

Die kartenbasierte Fortbewegung (*map-based travel* [Bowman, 1998]) ist eine zielgerichtete Form der Navigation. Der Benutzer sieht zusätzlich zu seiner virtuellen Umgebung eine Karte, die die Topologie dieser Umgebung repräsentiert (siehe Abbildung 2.19).



**Abbildung 2.19: Karte der virtuellen Umgebung**

Die Position des Benutzers wird durch ein Symbol angezeigt. Der Benutzer kann dieses Symbol auf der Karte an eine andere Stelle setzen. Anschließend wird die Bewegung von der alten zur neuen Position des Symbols durch das System animiert dargestellt. Dieses Verfahren vereinfacht vor allem die Navigation in sehr großen virtuellen Welten, da der Benutzer jederzeit seine genaue Position innerhalb dieser Welt kennt. Für eine exakte Fortbewegung ist es jedoch weniger geeignet.

#### 2.6.1.4 Grabbing the Air

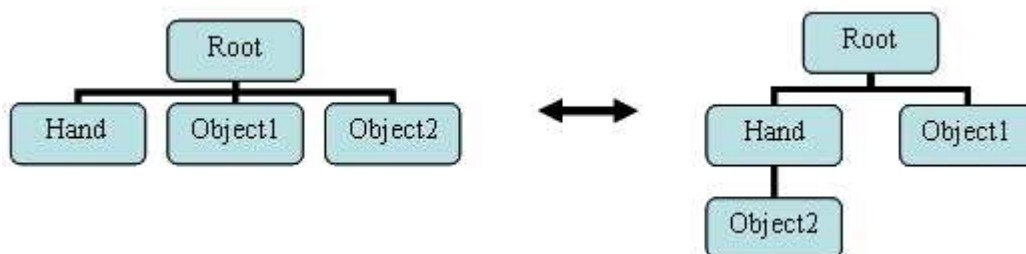
Bei der *Grabbing the Air Technique* [Bowman, 2001b] zieht sich der Benutzer entlang eines imaginären Seils durch die Szene. Er ergreift einen Punkt seiner virtuellen Umgebung und zieht diesen zu sich. Diese Methode kann ein- und zweihändig implementiert werden. Für das Tracking der Hände werden meist Pinchgloves (siehe Kapitel 2.4.2) verwendet.

### 2.6.2 Manipulation

Es gibt drei grundlegende Formen der Manipulation von virtuellen Objekten. Dabei handelt es sich um Selektion, Positionierung und Orientierung. Dieses Kapitel beschreibt einige der gängigen Manipulationstechniken. Bei der Entwicklung solcher Techniken steht vor allem die Benutzerfreundlichkeit im Vordergrund. Weiters muß festgelegt werden, wann ein Objekt als selektiert gilt und welches Feedback der Benutzer in Folge erhält.

#### 2.6.2.1 Simple Virtual Hand

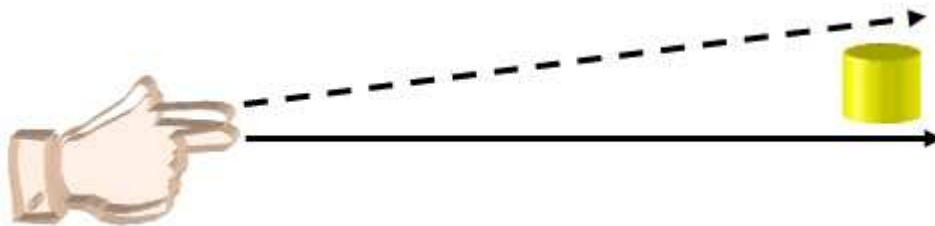
Bei der *Simple Virtual Hand* Methode wird an der Position der Hand des Benutzers in der virtuellen Welt eine virtuelle Hand angezeigt. Ein Objekt gilt als selektiert, wenn es einen Schnittpunkt mit der virtuellen Hand besitzt. Das selektierte Objekt folgt nun den Bewegungen der virtuellen Hand, d.h. es übernimmt deren Position und Orientierung. Das wird erreicht, indem das selektierte Objekt im Szenengraph der virtuellen Hand zugeordnet wird (siehe Abbildung 2.20). Dadurch erhält der Benutzer das Gefühl, die Objekte berühren zu können, wenn auch ohne haptisches Feedback (siehe Kapitel 2.5.9). *Simple Virtual Hand* ist eine leicht verständliche, intuitive Form der Objektmanipulation. Es kann jedoch nur mit Objekten interagiert werden, die sich in Reichweite des Benutzers befinden.



**Abbildung 2.20:** Das selektierte Objekt *Object2* wird im Szenengraph der virtuellen Hand zugeordnet.

### 2.6.2.2 Ray Casting

Um, im Gegensatz zur *Simple Virtual Hand Methode* (siehe Kapitel 2.6.2.1), auch Objekte manipulieren zu können, die außerhalb der Reichweite des Benutzers positioniert sind, wird die *Ray Casting Methode* eingesetzt. Diese Methode bedient sich der Metapher eines Laserpointers. Dabei wird die Hand durch einen Strahl verlängert. Das erste Objekt, das von diesem Strahl getroffen wird, wird selektiert. Das Auswählen von kleinen, sehr weit entfernten Objekten kann sich jedoch als schwierig erweisen, da auch sehr kleine Handbewegungen den Strahl in der Entfernung stark beeinflussen. Abbildung 2.21 verdeutlicht dieses Problem.



**Abbildung 2.21: Die Selektion kleiner, weit entfernter Objekte kann sich als schwierig erweisen.**

Bei der *Flash Light Technique* wird versucht, dieses Problem zu umgehen, indem statt eines Strahls ein Kegel zur Selektion von Objekten verwendet wird. Es kann jedoch zu Unklarheiten kommen, wenn sich mehr als ein Objekt innerhalb des Kegels befindet.

Die *Aperture Based Selection Technique* [Forsberg, 1996] unterscheidet sich dadurch von der *Flash Light Technique*, daß der Kegel seinen Ursprung nicht in der Hand, sondern bei den Augen des Benutzers hat. Außerdem kann der Radius des Kegels verändert werden, indem die Hand vom Körper weg oder zum Körper hin bewegt wird.

### 2.6.2.3 Image Plane Interaction

Die Selektion eines Objekts erfolgt bei der *Image Plane Interaction Technique* [Pierce, 1997] dadurch, daß die Stellung der Hände in einer zweidimensionalen Abbildung der Szene interpretiert wird. Diese Abbildung entspricht einer Fotografie der Szene aus der Sicht des Benutzers.

Es gibt verschiedene Interpretationsmöglichkeiten für die Stellung der Hände. Bei der *Head Crusher Technique* umschließt der Benutzer das gewünschte Objekt mit Daumen und Zeigefinger. Diese Methode eignet sich jedoch nicht, um sehr große Objekte zu selektieren.

Die *Sticky Finger Technique* löst dieses Problem. Dabei wird jenes Objekt selektiert, das von der virtuellen Hand verdeckt wird. Aus der Sicht des Benutzers befindet sich seine Hand also direkt vor dem zu selektierenden Objekt (siehe Abbildung 2.22). In dem Fall, daß mehrere Objekte gleichzeitig verdeckt werden, wird jenes Objekt ausgewählt, das als erstes von der Verlängerung der imaginären Linie zwischen den Augen des Benutzers und seiner Hand geschnitten wird.



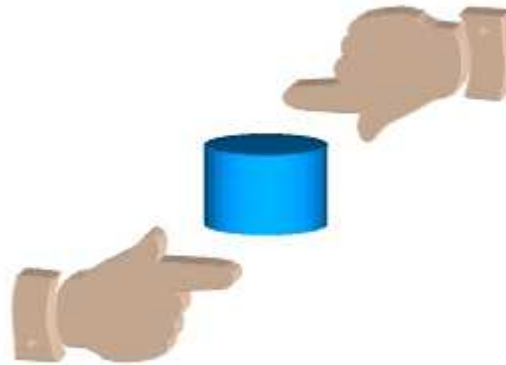
**Abbildung 2.22: Die *Sticky Finger Technique* aus der Sicht des Benutzers.**

Bei der *Lifting Palm Technique* selektiert der Benutzer ein Objekt, indem er seine flache Hand so positioniert, daß das Objekt über dieser zu schweben scheint (siehe Abbildung 2.23).

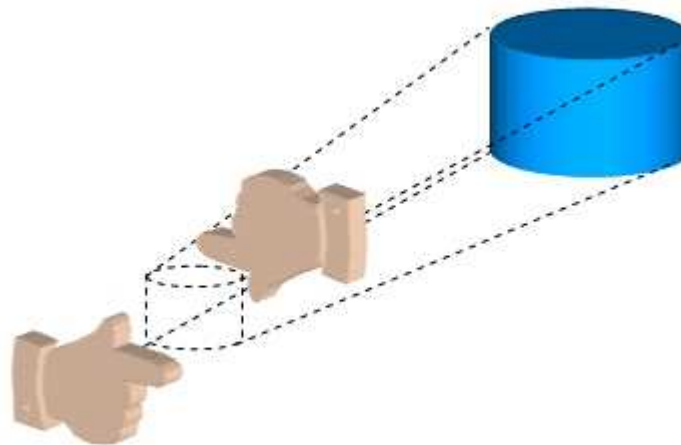


**Abbildung 2.23: Die *Lifting Palm Technique* aus Benutzersicht.**

Die *Framing Hands Technique* interpretiert die Stellung beider Hände. Der Benutzer rahmt dabei das zu selektierende Objekt mit Daumen und Zeigefinger beider Hände ein (siehe Abbildung 2.24). Abbildung 2.25 zeigt dieselbe Szene aus der Sicht eines Beobachters. Das Objekt wird entlang der gestrichelten Linien in die Bildebene des Benutzers projiziert.



**Abbildung 2.24:** Die Stellung der Hände bei der *Framing Hands Technique*.



**Abbildung 2.25:** Die Szene aus Abbildung 2.24 aus der Sicht eines Beobachters.

Die *Image Plane Interaction Technique* eignet sich sehr gut zur Selektion, jedoch weniger gut zu Positionierung und Orientierung von virtuellen Objekten.

#### 2.6.2.4 Go-Go Interaction Technique

Die *Go-Go Interaction Technique* [Poupyrev, 1996] stellt eine Erweiterung der *Simple Virtual Hand* Methode (siehe Kapitel 2.6.2.1) dar. Dabei wird die Reichweite des Benutzers verlängert, indem er die Länge seines virtuellen Armes verändern kann. Der Raum um den Benutzer wird in zwei Bereiche geteilt. Der erste Bereich erstreckt sich vom Benutzer bis zu ungefähr zwei Drittel der Länge seines Armes. In diesem Bereich erfolgt das Mapping

zwischen der realen und der virtuellen Hand linear. Position und Orientierung von virtueller und realer Hand stimmen überein. Streckt der Benutzer seine Hand jedoch weiter aus, kommt sie in den zweiten Bereich, der sich bis zur gesamten Armlänge erstreckt. In diesem Bereich erfolgt das Mapping nicht-linear, d.h., die Entfernung zwischen virtueller Hand und Benutzer wächst schneller als die tatsächliche Entfernung von realer Hand und Benutzer (siehe Abbildung 2.26). Bringt der Benutzer seine Hand erneut in den ersten Bereich, findet wieder lineares Mapping statt. Diese Methode ist sehr intuitiv, da das Strecken des Armes die natürliche Vorgehensweise ist, wenn eine Person einen entfernten Gegenstand ergreifen möchte.



**Abbildung 2.26: Reale und virtuelle Hand bei der *Go-Go Interaction Technique*.**

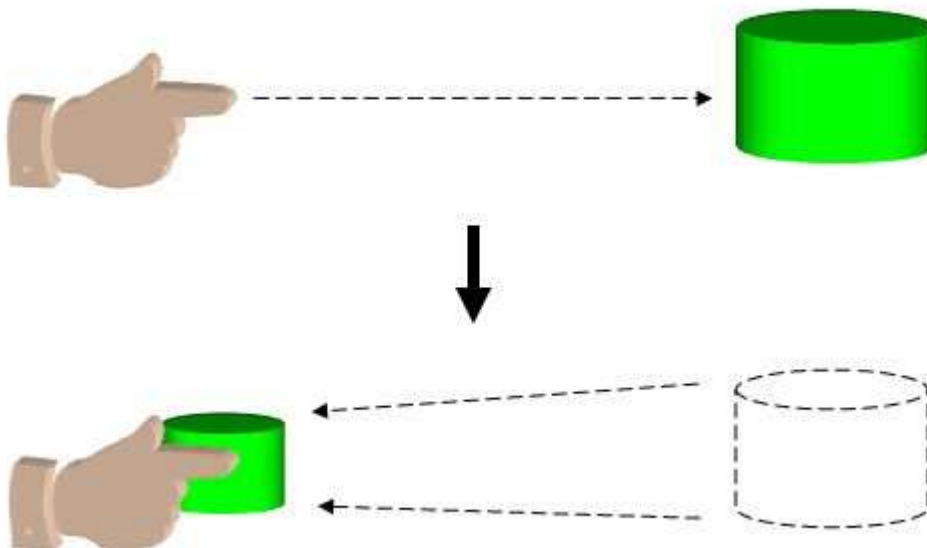
#### 2.6.2.5 HOMER

Die *Hand-centered Object Manipulation Extending Ray-casting technique* (HOMER) [Bowman, 1997] kombiniert die Techniken des *Ray-casting* (siehe Kapitel 2.6.2.2) und der *Simple Virtual Hand* (siehe Kapitel 2.6.2.1). Die Selektion eines Objekts erfolgt durch einen virtuellen Strahl. Anschließend bewegt sich die virtuelle Hand zu der Position des Objekts. Das Objekt wird nicht dem Strahl, sondern der Hand zugeordnet. Der Benutzer kann das Objekt daher intuitiv mit Hilfe der Hand manipulieren, auch wenn sich dieses außerhalb seiner Reichweite befindet. Wird das Objekt losgelassen, kehrt die virtuelle Hand wieder an die Position der realen Hand zurück.

Das Mapping zwischen realer und virtueller Hand erfolgt bei der HOMER-Methode relativ zum Benutzer. Die Position des Objekts wird durch seine Entfernung vom Benutzer und durch den Vektor vom Benutzer zu seiner Hand bestimmt.

### 2.6.2.6 Scaled World Grab

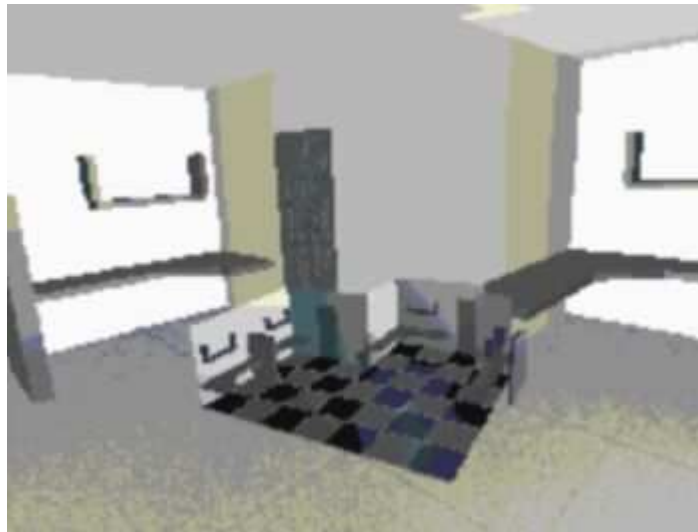
Die *Scaled World Grab technique* [Mine, 1997; Pierce, 1997] eignet sich vor allem für die Manipulation von großen, weit entfernten Objekten. Dabei wird die ganze Szene so verkleinert, daß sich das selektierte Objekt in Reichweite des Benutzers befindet (siehe Abbildung 2.27). Die Selektion kann dabei z.B. mit Hilfe einer *Image Plane Interaction technique* (siehe Kapitel 2.6.2.3) durchgeführt werden. Das Objekt wird anschließend wie bei der *Simple Virtual Hand* Methode (siehe Kapitel 2.6.2.1) manipuliert. Sobald das Objekt deselektiert wird, wird die ursprüngliche Größe der Szene wiederhergestellt. Dadurch können bei der *Scaled World Grab technique* nahe und entfernte Objekte auf die gleiche Art und Weise bearbeitet werden.



**Abbildung 2.27:** Die Szene wird so verkleinert, daß sich das Objekt in Reichweite des Benutzers befindet.

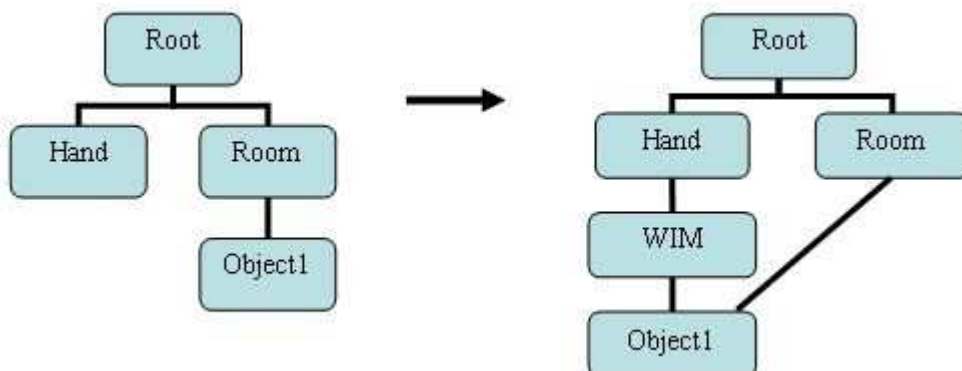
### 2.6.2.7 World in Miniature

Bei der *World in Miniature technique* (WIM) [Stoakley, 1995; Pausch, 1995] werden nicht die virtuellen Objekte selbst, sondern verkleinerte Abbilder dieser Objekte manipuliert. Dabei wird zusätzlich zur virtuellen Umgebung eine stark verkleinerte Version der Szene dargestellt. Diese verkleinerte Darstellung wird als WIM bezeichnet (siehe Abbildung 2.28).



**Abbildung 2.28:** Zusätzlich zur virtuellen Umgebung wird eine verkleinerte Version der Szene dargestellt.

Die Objekte der WIM können mit Hilfe der *Simple Virtual Hand technique* (siehe Kapitel 2.6.2.1) bearbeitet werden. Die vorgenommenen Veränderungen wirken sich in gleicher Weise auf die Objekte der Szene aus. Dies wird dadurch erreicht, daß der virtuellen Hand im Szenengraph eine skalierte Version der Szene zugeordnet wird.



**Abbildung 2.29:** Der virtuellen Hand wird eine skalierte Version der Szene zugeordnet.

Diese Technik eignet sich vor allem für großflächige Manipulationen, wie z.B. das Verschieben eines Objekts in einen anderen Raum der virtuellen Welt. Die Interaktion mit

kleinen Objekten kann sich jedoch als schwierig erweisen, da diese innerhalb der WIM zusätzlich verkleinert werden.

### 2.6.3 Systemsteuerung

Da in den meisten VR-Systemen auf den Einsatz von Tastatur und 2D-Maus verzichtet wird, sind andere Interaktionsformen zur Systemsteuerung notwendig. Im Bereich konventioneller Anwendungen stehen Entwicklern umfangreiche Bibliotheken mit Interface-Elementen zur Verfügung. In VR-Applikationen hat sich jedoch noch kein einheitlicher Standard durchgesetzt. Die grundlegenden Richtlinien für das User Interface Design konventioneller Anwendungen gelten jedoch ebenso für VR-Systeme. Demzufolge sollte ein User Interface daher

- leicht erlernbar sein,
- schnell und einfach zu bedienen sein,
- gut verständliches visuelles Feedback bieten,
- den Benutzer nicht zu sehr von der eigentlichen Anwendung ablenken.

In den folgenden Kapiteln werden einige Methoden zur Systemsteuerung in VR-Systemen vorgestellt.

#### 2.6.3.1 Virtual Widgets

Eine Möglichkeit zur Systemsteuerung ist die Verwendung von Widgets [Conner, 1992], das sind Interaktionselemente wie z.B. Menus, Buttons und Sliders. Diese sind aus dem Umgang mit konventionellen Computersystemen bekannt. Daher entsteht für den Benutzer kein zusätzlicher Lernaufwand. Die Bedienung der Interaktionselemente erfolgt mit Hilfe einer Manipulationsmethode (siehe Kapitel 2.6.2). Abbildung 2.30 zeigt mehrere Widgets, die durch das Ray Casting Verfahren (siehe Kapitel 2.6.2.2) selektiert und manipuliert werden können. Wird jedoch eine große Anzahl an Widgets benötigt, nehmen diese entsprechend viel Platz in der virtuellen Welt ein. Die eingeschränkte Sicht vermindert beim Benutzer den Eindruck der Immersion.

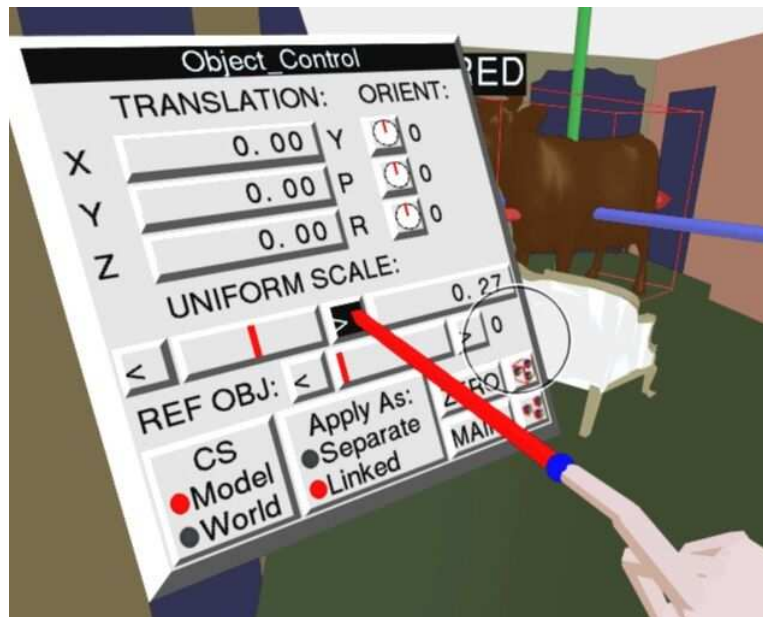
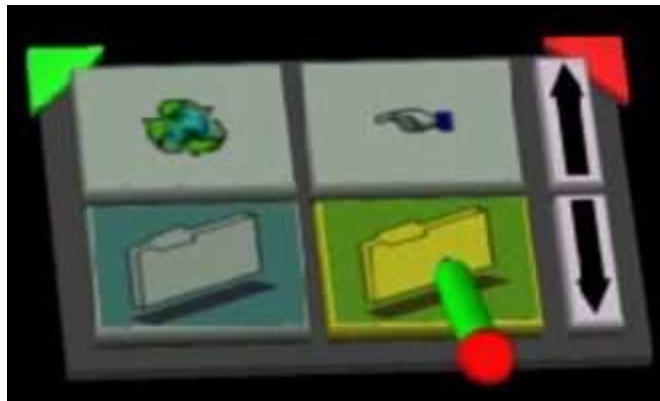


Abbildung 2.30: Die Widgets werden mit Hilfe des Ray Casting Verfahrens bedient.

### 2.6.3.2 Pen & Tablet

Das in Kapitel 2.4.4 beschriebene Eingabegerät *Pen & Tablet* eignet sich sehr gut zur Steuerung eines VR-Systems. Dabei werden die Widgets (siehe Kapitel 2.6.3.1) nicht im Raum, sondern auf dem Tablet platziert. Die Bedienung erfolgt mit Hilfe des Pens. Da der Benutzer das Tablet bewegen kann, versperrt es ihm nicht die Sicht auf die virtuelle Szene. Außerdem erleichtert das haptische Feedback (siehe Kapitel 2.5.9), das vom Tablet ausgeht, dem Benutzer die Interaktion mit dem System. Abbildung 2.31 zeigt eine virtuelle Darstellung von Pen & Tablet.



**Abbildung 2.31: Eine virtuelle Darstellung von Pen & Tablet.**

#### 2.6.3.3 Speech Commands

Die Technik der Spracherkennung kommt in VR-Systemen eher selten zum Einsatz, da sie im allgemeinen sehr fehleranfällig ist. Um die Fehlerrate gering zu halten, müssen die meisten Spracherkennungssysteme auf einen bestimmten Benutzer eingestellt und trainiert werden. Mittlerweile gibt es jedoch auch schon Systeme, die ohne Training auskommen und von mehreren Personen benutzt werden können.

Um die natürliche Sprache zur Systemsteuerung einsetzen zu können, muß der Benutzer jedoch darüber bescheid wissen, welche Wörter vom System verarbeitet werden können. Entweder muß er diese vor der ersten Benutzung des Systems erlernen, oder das System teilt ihm die möglichen Kommandos zur Laufzeit mit. Diese Art der Interaktion ist jedoch im Vergleich zu anderen Systemsteuerungsmethoden eher langsam und zeitaufwendig. Der Vorteil der Spracherkennung liegt jedoch darin, daß keine speziellen Eingabegeräte und Tracker notwendig sind und der Benutzer sich daher auf die virtuelle Szene konzentrieren kann.

#### 2.6.3.4 Gesten

Da die Bedeutung der menschlichen Gesten nicht immer eindeutig und auch von Kultur zu Kultur unterschiedlich ist, kommt diese Form der Interaktion in VR-Systemen eher selten zum Einsatz. Die möglichen Gesten und deren Bedeutung müssen genau definiert werden.

Zur Interpretation der menschlichen Bewegungen werden z.B. optisches Tracking (siehe Kapitel 2.3.5) oder Datenhandschuhe eingesetzt. Gesten können z.B. dazu benutzt werden, um den Ablauf einer Präsentation (siehe Kapitel 2.5.7) zu steuern [Baudel, 1993]. Es ist jedoch eine gewisse Einarbeitungszeit für den Benutzer notwendig.

## 2.7 Tangible User Interface

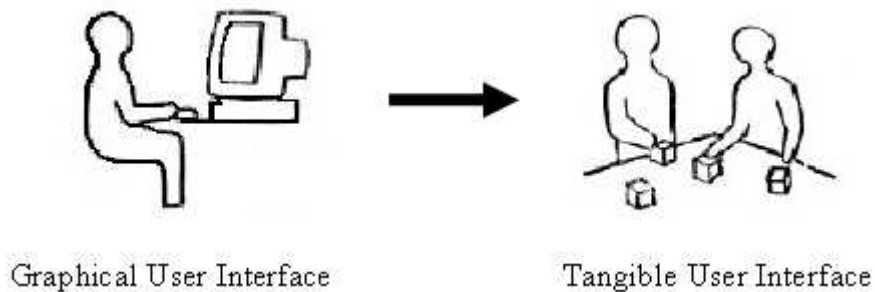
Die Kommunikation zwischen Mensch und Computer erfolgt üblicherweise über ein *Graphical User Interface* (GUI). Dieses wurde 1981 entwickelt und löste den Kommandozeilenmodus ab, bei dem der Benutzer alle vorhandenen Befehle im Gedächtnis behalten mußte. Ein GUI bedient sich der Desktop-Metapher, d.h. der Bildschirm simuliert eine Schreibtischoberfläche. Icons und Windows symbolisieren die einzelnen Anwendungsbereiche. Die Auswahl der gewünschten Anwendung erfolgt mit Hilfe der Maus. Ein GUI hat gegenüber dem Kommandozeilenmodus den entscheidenden Vorteil, daß der Benutzer sich nicht an die einzelnen Befehle erinnern muß. Er sieht anhand der Icons, welche Anwendungen auf dem jeweiligen Computersystem vorhanden sind. Weiters wird er durch Interaktionselemente, wie z.B. Buttons und Menüs, auf die möglichen Aktionen aufmerksam gemacht. Trotzdem fällt vielen Menschen der Umgang mit dem Computer schwer. Neueinsteiger haben oft Probleme mit der Bedienung der Maus und mit der Interpretation der verwendeten Symbole. Sie sind gewohnt, ein Objekt, das sie manipulieren wollen, direkt anzugreifen, anstatt es mit Hilfe eines Zeigegeräts auszuwählen. Um diese direkte Interaktion zwischen Mensch und Computer zu ermöglichen, wurden *Tangible<sup>1</sup> User Interfaces* (TUIs) [Ishii, 1997] entwickelt.

Ein TUI verwendet Objekte aus dem alltäglichen Umfeld des Menschen als Schnittstelle zum Computer. Der Computer selbst tritt dabei in den Hintergrund. Ziel ist es, die natürliche Umgebung des Menschen als User Interface einzusetzen und dadurch die reale mit der virtuellen Welt zu verbinden. Diese intuitive Form der Interaktion erleichtert vor allem ungeübten Benutzern die Kommunikation mit dem Computer. Da ein TUI nicht auf einen Bildschirm und eine Maus beschränkt ist, können auch mehrere Benutzer gleichzeitig

---

<sup>1</sup> greifbar, fühlbar

mit dem System interagieren. Abbildung 2.32 verdeutlicht den Unterschied zwischen GUI und TUI.



**Abbildung 2.32: Ein TUI verwendet Objekte aus dem alltäglichen Umfeld des Menschen als Schnittstelle zum Computer**

## 2.8 Tangible Augmented Reality

*Tangible Augmented Reality* (TAR) [Billingham, 2001] bezeichnet eine Kombination aus einem Augmented Reality System (siehe Kapitel 2.1.3) und einem Tangible User Interface (siehe Kapitel 2.7). Dabei interagiert der Benutzer mit den virtuellen Objekten, indem er reale Objekte manipuliert. Das hat gegenüber den in Kapitel 2.6 beschriebenen Interaktionsformen den Vorteil, daß der Benutzer keine spezielle Technik erlernen muß, um virtuelle Objekte zu manipulieren. Die Interaktion mit diesen virtuellen Objekten erfolgt mit denselben Methoden, wie die Interaktion mit realen Objekten. Ein TAR-System bietet daher einerseits eine dreidimensionale Darstellung der virtuellen Objekte innerhalb der realen Welt und andererseits eine besonders intuitive Form der Interaktion mit diesen Objekten. Folgende Eigenschaften sind für ein TAR-System wesentlich:

- Die Interaktion mit virtuellen Objekten erfolgt mit Hilfe von realen Objekten.
- Der Benutzer kann beide Hände gleichzeitig zur Interaktion verwenden.
- Mehrere Objekte können gleichzeitig manipuliert werden.
- Eine Zusammenarbeit von mehreren Benutzern ist möglich.

In den folgenden Kapiteln werden zwei bereits bestehende TAR-Systeme vorgestellt.

### 2.8.1 Tiles

Bei *Tiles* [Billinghurst, 2001] handelt es sich um ein TAR-System, das zur schnellen Herstellung von Prototypen für Flugzeug-Instrumententafeln entwickelt wurde. Das TUI besteht aus einer Metalltafel, einem Buch und mehreren Karten. Die Metalltafel fungiert als Arbeitsbereich, auf dem die mit Magneten versehenen Karten angeordnet werden können. Das Buch entspricht einem Menü. Auf den einzelnen Seiten sind Marker angebracht, die mit virtuellen Modellen von Flugzeug-Instrumenten überlagert werden. Bei den Markern handelt es sich um ARToolKit-Marker, die in Kapitel 4.2 genauer beschrieben werden. Die Karten sind ebenfalls mit Markern versehen. Sie werden in Systemsteuerungs-Karten und Daten-Karten eingeteilt. Die Systemsteuerungs-Karten dienen zur Durchführung von allgemeinen Aktionen, wie z.B. Kopieren, Löschen oder Hilfe. Sie werden durch entsprechende dreidimensionale Symbole überlagert. Einer Daten-Karte kann mit Hilfe des Buchs und der Kopieren-Funktion ein Instrumenten-Modell zugewiesen werden.

Das System ermöglicht die Zusammenarbeit mehrerer Benutzer. Sie sitzen vor der Metalltafel und können die Daten-Karten mit den Instrumenten-Modellen beliebig anordnen. Die virtuellen Grafiken werden mit durchsichtigen HMDs betrachtet.

### 2.8.2 Shared Space

*Shared Space* [Billinghurst, 2000] ist ein einfaches TAR-Spiel, das speziell für Computerteilnehmer entwickelt wurde. Die Benutzer stehen dabei um einen Tisch herum, auf dem Karten liegen, die mit ARToolKit-Markern (siehe Kapitel 4.2) versehen wurden. Sie tragen durchsichtige HMDs, auf denen Videokameras montiert sind, und können daher die zu den Markern gehörende Grafik sehen. Je zwei dieser virtuellen Modelle stehen miteinander in Beziehung. Ziel des Spieles ist es, den Zusammenhang zwischen den Modellen zu erkennen und die entsprechenden Karten nebeneinander zu plazieren. Wenn zwei passende Karten nebeneinander liegen, werden die einzelnen Modelle durch eine Animation beider Objekte ersetzt.

### **3. Ziele des Projekts**

Im Rahmen des *Tabletop Game* Projekts, das in dieser Diplomarbeit beschrieben wird, soll ein Tangible Augmented Reality System (siehe Kapitel 2.8) für Computerspiele entwickelt werden. Das bestehende Augmented Reality System *Studierstube*, das in Kapitel 4.1 näher beschrieben wird, soll um ein Tangible User Interface (siehe Kapitel 2.7) erweitert werden, das speziell für die Steuerung eines Computerspiels konzipiert ist. Dabei wird auf dem Konzept von *Shared Space*, das in Kapitel 2.8.2 vorgestellt wird, aufgebaut. Das Shared Space Projekt beschäftigt sich jedoch hauptsächlich damit, ob sich TAR-Systeme für die Zusammenarbeit mehrerer Benutzer besser eignen als herkömmliche GUIs. Bei dem in dieser Diplomarbeit beschriebenen Projekt steht jedoch das Computerspiel im Vordergrund. Es soll vor allem untersucht werden, für welche Spieltypen ein TAR-System sich besonders gut eignet und welche Möglichkeiten zur Steuerung der Spielfiguren bestehen. Zu diesem Zweck soll, neben dem Interface selbst, ein ansprechendes Spiel (siehe Kapitel 3.2) entwickelt werden. Anhand dieses Spiels sollen die oben genannten Aspekte analysiert und evaluiert werden. Dazu werden verschiedene Versionen dieses Spiels generiert, die sich durch den Typ des Spiels und die Art der Steuerung unterscheiden. Diese verschiedenen Implementierungen werden auf deren Tauglichkeit im Einsatz für Computerspiele getestet.

Anstatt Marker für das Tracking eines Eingabegeräts zu verwenden, können sie auch selbst als Eingabegerät fungieren. Position und Orientierung des Markers bestimmen dann direkt die Lage eines virtuellen Objekts. Dadurch wird das virtuelle Objekt für den Benutzer greifbar. Auf dieser Eigenschaft basiert das TUI des *Tabletop Game* Projekts. Die Spielfiguren werden mit Hilfe von Markern gesteuert. Dabei wird je ein Marker mit einer Spielfigur assoziiert. Bei markerbasiertem Tracking (siehe Kapitel 2.3.5.1) besteht jedoch die Möglichkeit, daß ein Marker durch Gegenstände oder den Benutzer selbst zeitweise verdeckt ist. Die Lage des Markers kann in diesem Fall nicht mehr festgestellt werden. In Folge stimmen Position und Orientierung des entsprechenden virtuellen Objekts nicht mit denen des Markers überein. Erst wenn der Marker erneut sichtbar wird, kann das virtuelle Objekt wieder korrekt gesteuert werden. Da bei einem Computerspiel das Timing eine große Rolle spielt, kann der Ausfall einer Spielfigur wegen fehlender Tracking-Informationen nicht toleriert werden. Diese Ungenauigkeit könnte über Sieg und

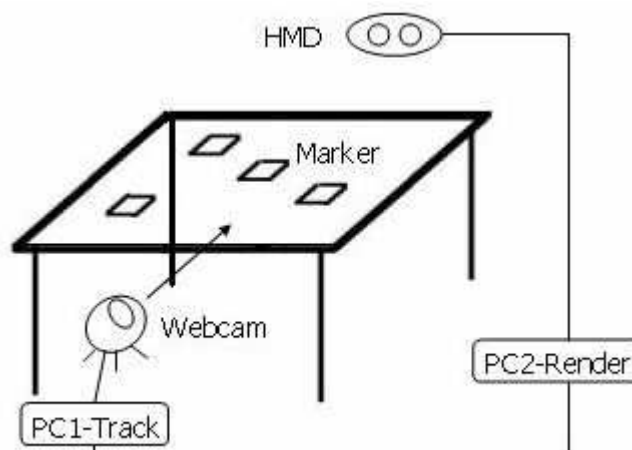
Niederlage entscheiden. Um das Problem der Verdeckung von Markern zu lösen, wird daher eine neue Hardware benötigt, die im folgenden Kapitel näher beschrieben wird.

### 3.1 Neue Hardware

Die neue Hardware muß vor allem folgende Eigenschaften erfüllen:

- Die Steuerung der virtuellen Objekte erfolgt durch Marker.
- Die Marker sollen für die Kamera jederzeit sichtbar sein.
- Der Benutzer soll durch die neue Hardware nicht in seiner Bewegungsfreiheit eingeschränkt sein.

Um die Verdeckung von Markern zu verhindern, muß sichergestellt werden, daß sich während des Spiels keine Personen oder Gegenstände zwischen der Kamera und den Markern befinden. Dies kann durch die Verwendung eines Glastisches erreicht werden. Die Marker liegen dabei so auf dem Tisch, daß sie von einer Kamera, die sich unterhalb des Tisches befindet, verfolgt werden können. Die virtuellen Objekte werden auf der Oberseite des Tisches an der Stelle der entsprechenden Marker dargestellt. Der Benutzer sieht diese durch ein durchsichtiges HMD. Um eine gute Performance zu gewährleisten, werden das Tracking und das Rendering auf verschiedenen Computern durchgeführt. Abbildung 3.1 zeigt eine Skizze der neuen Hardware und deren Verbindung zum AR-System.



**Abbildung 3.1: Eine Skizze der Hardware.**

## 3.2 Tabletop Game

### 3.2.1 Spielidee

Beim Tabletop Game handelt es sich um ein Spiel für zwei Spieler, die um den Glastisch (siehe Kapitel 3.1) herum stehen oder sitzen. Jeder Spieler verfügt über ein Katapult und über eine festgelegte Anzahl an Ballons. Ziel des Spieles ist es, die gegnerischen Ballons mit Hilfe des Katapults abzuschießen. Sieger ist, wer zuerst alle gegnerischen Ballons abschießen konnte.

Ein Spieler kann zwar sein Katapult, nicht aber seine Ballons steuern. Um die eigenen Ballons aus der Schußlinie des gegnerischen Katapults bewegen zu können, stehen dem Spieler Windmühlen zur Verfügung. Die Ballons fliegen automatisch mit dem Wind, der von diesen Windmühlen erzeugt wird.

Im Rahmen des Projekts werden der Spielablauf, die Komplexität des Spiels und die Interaktionsmöglichkeiten mit dem System variiert, um zu ermitteln, welche Spielform sich am ehesten für eine intuitive Interaktion mit einem AR-System eignet.

### 3.2.2 Spieltypen

Der Benutzer kann zwischen folgenden Spieltypen wählen:

- Actionspiel
- Strategiespiel

Beim Actionspiel können beide Spieler gleichzeitig ihre Katapulte und Windmühlen bewegen. Die Auswirkungen ihrer Spielzüge werden sofort sichtbar. Das Strategiespiel dagegen ist rundenbasiert, d.h. jeder Spieler hat die Möglichkeit, verschiedene Spielzüge auszuprobieren, bevor er sich festlegt.

### 3.2.3 Spielfiguren

Der Spieler kann mit folgenden Spielfiguren interagieren:

- Katapulte
- Ballons
- Windmühlen

Im Actionspiel-Modus schießen die Katapulte beider Spieler kontinuierlich. Nach einem Schuß erfolgt automatisch der nächste. Im Strategiespiel-Modus jedoch stehen die Katapulte während eines Spielzuges still. Nach Abschluß dieses Spielzuges schießt nur das Katapult des aktuellen Spielers.

Die Position der Ballons kann nur indirekt verändert werden, indem die Position der Windmühlen verändert wird. Im Actionspiel-Modus bewegen sich die Ballons kontinuierlich von den Windmühlen fort. Mit zunehmender Entfernung von den Windmühlen nimmt die Geschwindigkeit der Ballons ab. Im Strategiespiel-Modus bewegen sich die Ballons nur nach Ende eines Spielzuges von den Windmühlen weg.

### 3.2.4 Steuerung

Die Steuerung der Katapulte und Windmühlen erfolgt durch Marker. Für jede dieser Spielfiguren ist ein separater Marker vorhanden. Dadurch unterscheidet sich dieses System von einem konventionellen Computersystem, bei dem mehrere Spielfiguren mit demselben Eingabegerät gesteuert werden und der Spieler dem System bekannt geben muß, welche Figur er bewegen möchte. Diese Auswahl erfolgt beim Tabletop Game intuitiv durch die Wahl des Markers.

Es ist jedoch nicht immer wünschenswert, die Bewegungen eines Markers direkt auf die entsprechende Spielfigur zu übertragen. Das Tabletop Game sieht daher zwei verschiedene Arten vor, um Position und Richtung einer Spielfigur zu verändern:

- Das Verschieben eines Markers bewirkt eine sofortige Positions- und Richtungsänderung der Spielfigur.

- Die Spielfigur bewegt sich mit konstanter Geschwindigkeit in die Richtung, in die der Marker geschoben wurde.

Eine sofortige Änderung der Position der Spielfigur ist für den Benutzer einfach nachvollziehbar, da sich das Verschieben eines Markers direkt auf die Spielfigur auswirkt. Wird aber z.B. ein Marker sehr schnell bewegt, folgt die Spielfigur dieser Bewegung. Ein Katapult, das sich mit hoher Geschwindigkeit bewegt, wirkt jedoch unrealistisch. Aus diesem Grund wird eine zweite Form der Steuerung implementiert, bei der die Spielfigur dem Marker mit konstanter Geschwindigkeit folgt.

### **3.2.5 Zusatzfunktionen**

Neben den verschiedenen Spieltypen (siehe Kapitel 3.2.2) und den unterschiedlichen Formen der Figurensteuerung (siehe Kapitel 3.2.4) sind zusätzlich folgende Funktionen vorgesehen:

- Die Anzahl der Ballons, Katapulte und Windmühlen kann variiert werden.
- Es kann festgelegt werden, daß für jeden abgeschossenen Ballon nach kurzer Zeit ein neuer Ballon ins Spiel kommt, um die Anzahl der Zielobjekte ungefähr konstant zu halten.
- Ein Terrain kann ein- oder ausgeblendet werden.
- Die Luftströmungen, die durch die Windmühlen erzeugt werden, können mittels Strömungsvisualisierung dargestellt werden.
- In der Landschaft sind Extras in Form von Schatztruhen verteilt, die man mit einem Katapult aufsammeln kann. Das Katapult kann daraufhin weiter schießen.

## **4. Bestehendes System**

### **4.1 Studierstube**

Im folgenden Kapitel wird die *Studierstube* [Schmalstieg, 2000; Szalavári, 1998a], das ist das bereits bestehende Augmented Reality System, das im Rahmen dieser Diplomarbeit erweitert werden soll, beschrieben. Das standardmäßige Eingabegerät ist das *Personal Interaction Panel* (PIP), das in Kapitel 4.1.2.1 näher beschrieben wird. Dabei handelt es sich um eine Implementierung der *Pen & Tablet* Methode (siehe Kapitel 2.4.4). Darüber hinaus können jedoch auch die in Kapitel 4.1.2 beschriebenen Eingabegeräte verwendet werden. Als Ausgabegerät dient im allgemeinen ein See Through HMD (siehe Kapitel 2.5.4). Für spezielle Anwendungen wurden jedoch auch andere Ausgabegeräte in das System integriert, auf die in Kapitel 4.1.3 näher eingegangen wird.

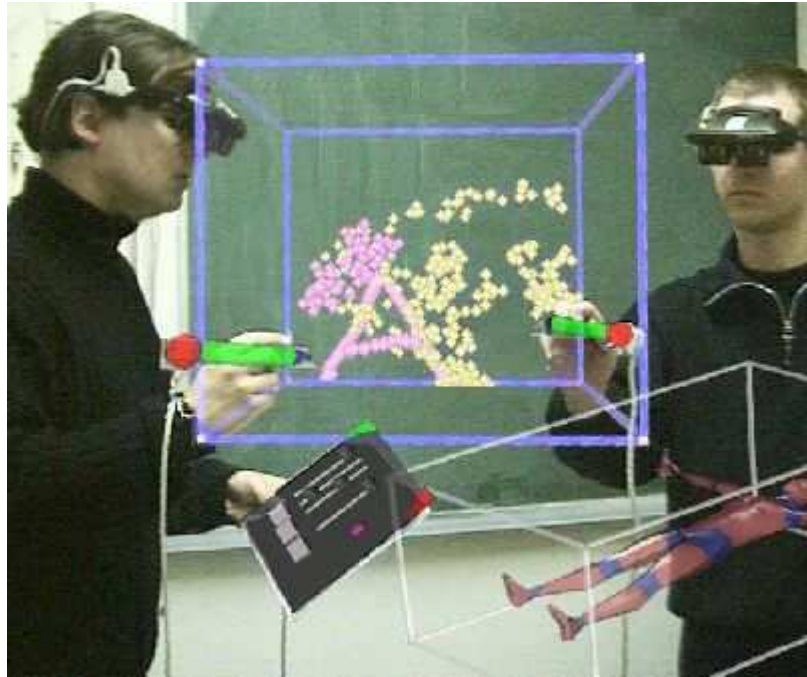
Die Studierstube unterstützt, neben den Eigenschaften eines AR-Systems (siehe Kapitel 2.1.3), auch folgende Funktionen:

- mehrere Personen können gleichzeitig mit dem System interagieren
- es besteht die Möglichkeit zur Zusammenarbeit mehrerer Benutzer
- die Darstellung der Szene kann für jeden Benutzer separat definiert werden
- es können mehrere Anwendungen zur gleichen Zeit ausgeführt werden
- die virtuelle Umgebung ist nicht auf einen bestimmten Ort beschränkt

Die Studierstube ermöglicht es mehreren Benutzern gleichzeitig, die virtuelle Welt zu betrachten und zu manipulieren. Dabei wird jedoch nicht zwischen aktiven Benutzern und Zusehern unterschieden, wie das beim CAVE (siehe Kapitel 2.5.7) der Fall ist. In der Studierstube kann jeder Benutzer seinen Blickpunkt frei wählen und auch Einfluß auf die dargestellte Geometrie nehmen.

Da ein AR-System den Benutzer nicht von seiner Umwelt abschneidet, ist es ihm möglich, mit anderen Benutzern zu interagieren. Dadurch können mehrere Benutzer zugleich mit derselben Anwendung arbeiten und in Folge das Ergebnis besprechen und gegebenenfalls

neue Lösungswege testen. Diese Form der Zusammenarbeit wird auch als *Collaborative Augmented Reality* [Fuhrmann, 1999] bezeichnet (siehe Abbildung 4.1).



**Abbildung 4.1: Die *Collaborative Augmented Reality* ermöglicht die Zusammenarbeit mehrerer Benutzer.**

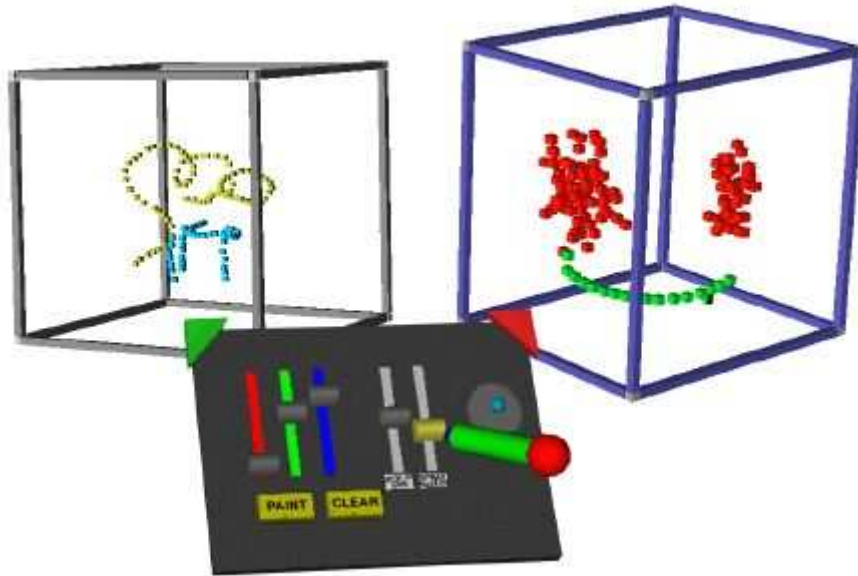
Bei der Zusammenarbeit mehrerer Benutzer ist es auch möglich, die Ansicht der Szene für jede Person individuell zu gestalten. Soll das AR-System z.B. im Unterricht verwendet werden [Kaufmann, 2002], so kann die Darstellung für den Schüler und den Lehrer unterschiedlich erfolgen. Der Lehrer sieht z.B. die Lösung der gestellten Aufgabe, die dem Schüler jedoch vorenthalten wird. Abbildung 4.2 zeigt den möglichen Einsatz der Studierstube im Geometrie- und Mathematikunterricht.



**Abbildung 4.2:** Die Darstellung der Szene kann für jeden Benutzer separat definiert werden.

Die Studierstube kann, wie auch konventionelle Computersysteme, mehrere Anwendungen gleichzeitig ausführen. In Analogie zum 2D-Window werden diese Anwendungen innerhalb eines Quaders – einem 3D-Window – ausgeführt. Dieses 3D-Window kann beliebig im dreidimensionalen Raum positioniert werden. Auch die Größe des 3D-Window kann nach Belieben verändert werden. Abbildung 4.3 zeigt zwei 3D-Window und das PIP.

Um festzulegen, welches 3D-Window von den Interaktionen des Benutzers beeinflusst wird, wird zwischen aktiven und passiven 3D-Window unterschieden. Pro Benutzer kann es höchstens ein aktives, jedoch beliebig viele passive Window geben. Das aktive Window wird durch einen blauen Rahmen gekennzeichnet. Der Benutzer kann zu einem anderen Window wechseln, indem er dessen Rahmen selektiert.



**Abbildung 4.3: Die Studierstube kann mehrere Anwendungen gleichzeitig ausführen.**

Die Studierstube ist nicht auf ein Labor oder einen ähnlichen Ort beschränkt. Durch die Verwendung eines mobilen Setups [Reitmayr, 2001a] kann sich der Benutzer an einen beliebigen Ort begeben. Das mobile Setup verwendet anstatt eines stationären Computers einen Laptop, der vom Benutzer in einem Rucksack getragen wird. Als Eingabegerät wird das PIP verwendet, das mit Hilfe von ARToolKit (siehe Kapitel 4.2), einem optischen Tracking-Verfahren, getrackt wird. Das Koordinatensystem der dargestellten Geometrie kann entweder relativ zum realen Raum oder relativ zum Benutzer positioniert werden. Abbildung 4.4 zeigt einen Benutzer, der das mobile Setup verwendet.



**Abbildung 4.4:** Ein Benutzer verwendet das mobile Setup der Studierstube.

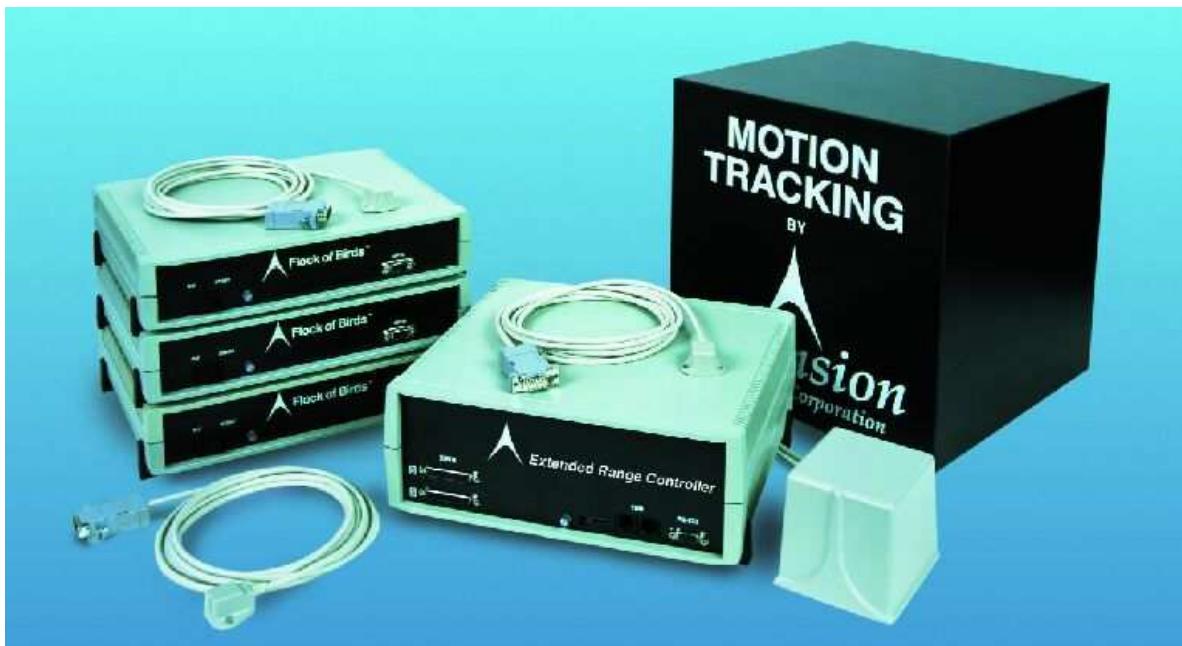
Ein Benutzer, der ein mobiles Setup verwendet, kann – ebenso wie ein Benutzer, der ein stationäres Setup verwendet – mit anderen Benutzern zusammenarbeiten. Die Verwaltung der Benutzer erfolgt mit Hilfe eines *distributed shared scene graph* [Schmalstieg, 2002]. Dabei besitzt jeder Computer eine Kopie des Szenengraphs, die automatisch mit den übrigen Kopien synchronisiert wird. Um die Trennung zwischen einer Applikation und ihrer grafischen Darstellung aufzuheben, werden Applikations-Daten als Knoten in den Szenengraph eingefügt. Auf diese Weise werden neben der grafischen Darstellung auch die Applikationen über das Netzwerk verteilt.

#### **4.1.1 Tracking**

Folgende Arten von Trackern stehen derzeit in der Studierstube zur Verfügung:

- Magnetischer Tracker
- Inertialer Tracker
- Marker

Als magnetischer Tracker wird das „Flock of Birds“-System der Firma Ascension verwendet, das in Abbildung 4.5 zu sehen ist. Es kommt bei der stationären Version der Studierstube zum Einsatz. Der Transmitter erzeugt ein kugelförmiges Magnetfeld mit einem Radius von zweieinhalb Metern. Da er aus Platzgründen oberhalb des Trackingbereichs angebracht ist, wird nur die untere Hälfte des Magnetfelds verwendet. Das magnetische Tracking wird vor allem zur Bestimmung von Position und Orientierung der durchsichtigen HMDs und des PIPs benutzt.



**Abbildung 4.5:** Das „Flock of Birds“-System der Firma Ascension wird als magnetischer Tracker eingesetzt.

Der inertielle Tracker kommt beim mobilen Setup der Studierstube zum Einsatz, um die Orientierung des durchsichtigen HMDs zu messen. Abbildung 4.6 zeigt das Modell „Intertrax2“ der Firma Intersense, das in der Studierstube verwendet wird.



**Abbildung 4.6: Für das inertielle Tracking wird das Modell „Intertrax2“ der Firma Intersense verwendet.**

Die Position und die Orientierung des PIPs werden beim mobilen Setup mit Hilfe von markerbasiertem optischem Tracking (siehe Kapitel 2.3.5.1) ermittelt. Dafür wird ARToolKit, ein kostengünstiges System für optisches Tracking, verwendet, das in Kapitel 4.2 detailliert beschrieben wird. Es benötigt lediglich eine Webcam und Marker aus Papier. Die Webcam wird oberhalb des durchsichtigen HMDs montiert. Anschließend werden die Marker am PIP befestigt. Abbildung 4.7 zeigt die einzelnen Komponenten des mobilen Setups. Links sind die zwei Teile des PIPs, das Tablett und der Stift, zu sehen, die mit Markern gekennzeichnet wurden. Rechts befindet sich ein Helm, auf den das durchsichtige HMD und die Webcam montiert wurden. Auf dem Laptop im Vordergrund laufen die Studierstube-Applikation und die für das Tracking benötigte Software.



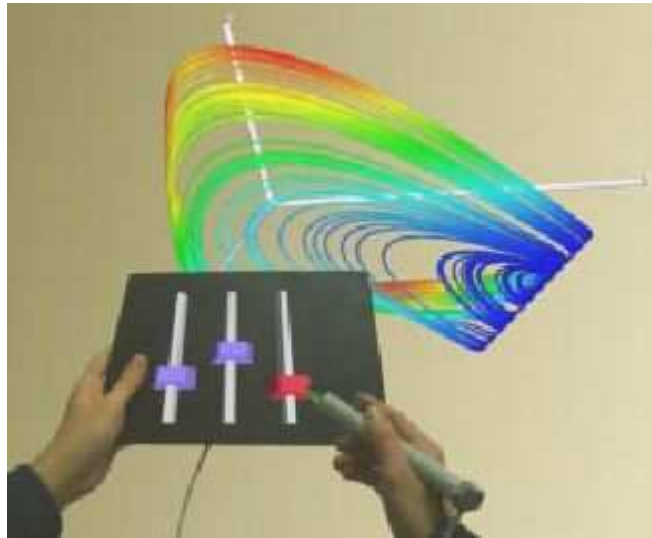
**Abbildung 4.7: Die Komponenten des mobilen Setups.**

Um während der Entwicklungsphase einer Anwendung für die Studierstube nicht auf die oben genannten Tracking-Systeme angewiesen zu sein, besteht auch die Möglichkeit, die virtuellen Objekte mit Hilfe des Keyboards zu manipulieren (siehe Kapitel 4.1.2.3).

## **4.1.2 Eingabegeräte**

### **4.1.2.1 Personal Interaction Panel**

Das *Personal Interaction Panel* (PIP) [Szalavári, 1998b] stellt eine Implementierung der *Pen & Tablet* Methode (siehe Kapitel 2.4.4) dar. Es besteht aus einem Tablett und einem Stift, deren Position und Orientierung durch magnetisches oder optisches Tracking bestimmt werden. Abbildung 4.8 zeigt das PIP und die virtuellen Interaktionselemente aus der Sicht des Benutzers.



**Abbildung 4.8: Der Benutzer sieht sowohl das PIP, als auch die darauf angezeigten Interaktionselemente.**

#### 4.1.2.2 Pinchglove

Neben dem PIP können auch Pinchgloves (siehe Kapitel 2.4.2) zur Interaktion mit der Studierstube verwendet werden. Dabei handelt es sich um handelsübliche Handschuhe, die an den Fingerkuppen mit drucksensitiven Metallstreifen versehen wurden. Die Bestimmung von Position und Orientierung der Handschuhe erfolgt mit Hilfe von optischem Tracking. Abbildung 4.9 zeigt einen Benutzer, der die Pinchgloves zur Manipulation der virtuellen Szene verwendet.



**Abbildung 4.9: Zur Interaktion mit der Studierstube können auch Pinchgloves verwendet werden.**

#### 4.1.2.3 Keyboard

Es besteht auch die Möglichkeit, ein Keyboard als Eingabegerät zu verwenden. Dabei lösen einzelne Tasten Translationen oder Rotationen entlang der drei Hauptachsen aus. Da diese Form der Interaktion jedoch keinen Immersionseindruck erzeugt, wird das Keyboard hauptsächlich während der Entwicklungsphase einer Anwendung für die Studierstube benutzt, um nicht auf die in Kapitel 4.1.1 beschriebenen Tracking-Systeme angewiesen zu sein. Studierstube-Anwendungen können daher auf herkömmlichen PCs entwickelt werden. Nach der Transferierung auf das Echtssystem muß lediglich die Spezifikation der Tracker angepaßt werden.

### 4.1.3 Ausgabegeräte

#### 4.1.3.1 See Through HMD

In der Studierstube werden hauptsächlich See Through HMDs (siehe Kapitel 2.5.4) zur Ausgabe der Grafik verwendet. Das Tracking erfolgt mit Hilfe von magnetischen oder inertialen Methoden.

#### 4.1.3.2 Projektionswand

Um die virtuelle Szene auch für Außenstehende sichtbar zu machen, kann sie, wie in Abbildung 4.10 zu sehen, auf einer Projektionswand angezeigt werden. In diesem Fall erfolgt die Darstellung jedoch nur zweidimensional. Es besteht darüber hinaus aber auch die Möglichkeit, durch stereoskopische Darstellung und die Verwendung von Polarisations-Brillen (siehe Kapitel 2.5.2) beim Betrachter einen dreidimensionalen Eindruck hervorzurufen.



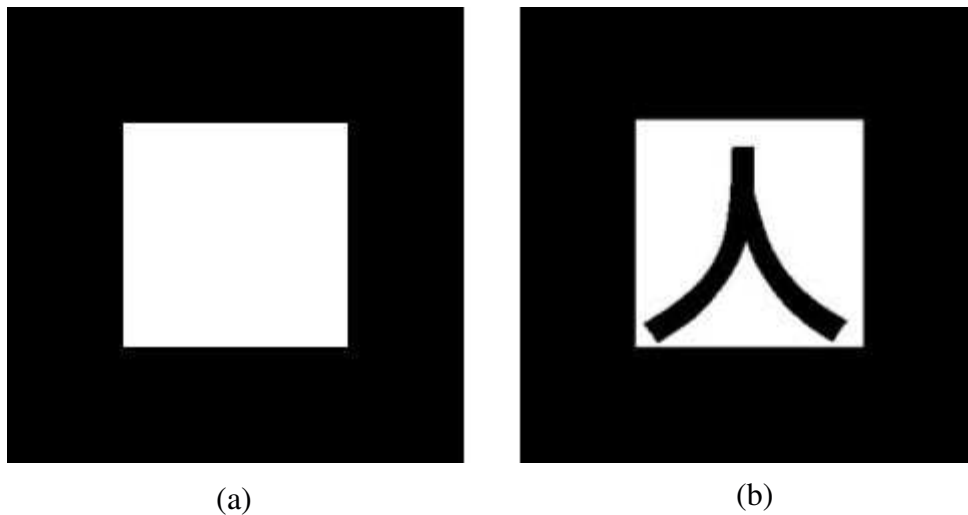
**Abbildung 4.10:** Die virtuelle Szene kann auf einer Projektionswand angezeigt werden.

## 4.2 ARToolKit

ARToolKit [Kato, 2000] ist ein kostengünstiges, markerbasiertes optisches Tracking-System. Es wurde entwickelt, um Programmierern von AR-Anwendungen ein einfaches optisches Tracking-System zur Verfügung zu stellen, das ohne großen Aufwand in ein bestehendes AR-System integriert werden kann. Es handelt sich dabei um eine C-Library, die in die vorhandene Software eingebunden werden kann. An Hardware ist lediglich eine Videokamera bzw. eine Webcam notwendig. Die Marker können mit Hilfe eines Druckers selbst hergestellt werden.

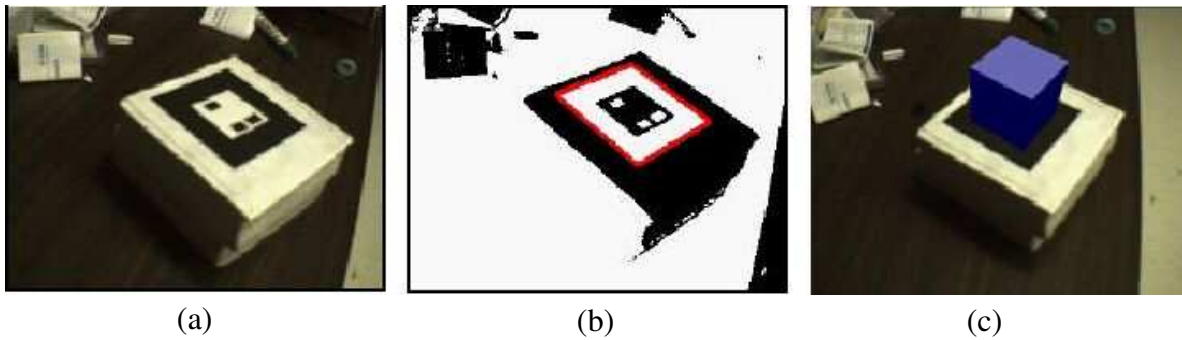
Ein Marker ist ein quadratisches Stück Papier, auf dem ein eindeutiges Muster innerhalb eines schwarzen Rahmens abgebildet ist. Abbildung 4.11a zeigt diesen Rahmen, in den mit

Hilfe eines Bildbearbeitungsprogramms ein Muster eingefügt wird. Ein Beispiel für einen vollständigen Marker ist in Abbildung 4.11b zu sehen. Die Seitenlänge eines Markers kann beliebig gewählt werden. Sie liegt im allgemeinen jedoch zwischen fünf und zwanzig Zentimetern. Um zu verhindern, daß der Marker während des Tracking-Vorganges umgebogen wird, ist es günstig, ihn auf ein Stück Karton oder ähnliches zu kleben.



**Abbildung 4.11: (a) der Rahmen eines Markers, (b) ein Beispiel für einen vollständigen Marker.**

Um zwischen den einzelnen Marker unterscheiden zu können, müssen der Software die Muster bekannt sein. Daher wird für jeden neuen Marker ein File erzeugt, das ein Videobild dieses Markers enthält. Bevor ein neuer Marker für das Tracking eingesetzt werden kann, müssen der Software der Ort dieses Files und die Größe des Markers bekannt sein. Während des Trackings versucht die Software, die einzelnen Marker anhand dieser Files im Bild der Videokamera zu finden. Dazu wird dieses Bild, welches in Abbildung 4.12a zu sehen ist, in ein Schwarz-Weiß-Bild umgewandelt. Dabei werden bis zu einem bestimmten Schwellwert alle hellen Pixel auf schwarz und alle dunklen auf weiß abgebildet. Danach wird in dem daraus resultierenden Bild nach quadratischen Flächen gesucht. Abbildung 4.12b zeigt das Schwarz-Weiß-Bild und eine rot markierte quadratische Fläche. Das Muster innerhalb dieser Flächen wird mit den zuvor erzeugten Files verglichen. Wenn eine Übereinstimmung gefunden wird, können Position und Orientierung des Markers festgestellt werden. Diese Daten werden verwendet, um den Marker mit einer Grafik zu überlagern (siehe Abbildung 4.12c).



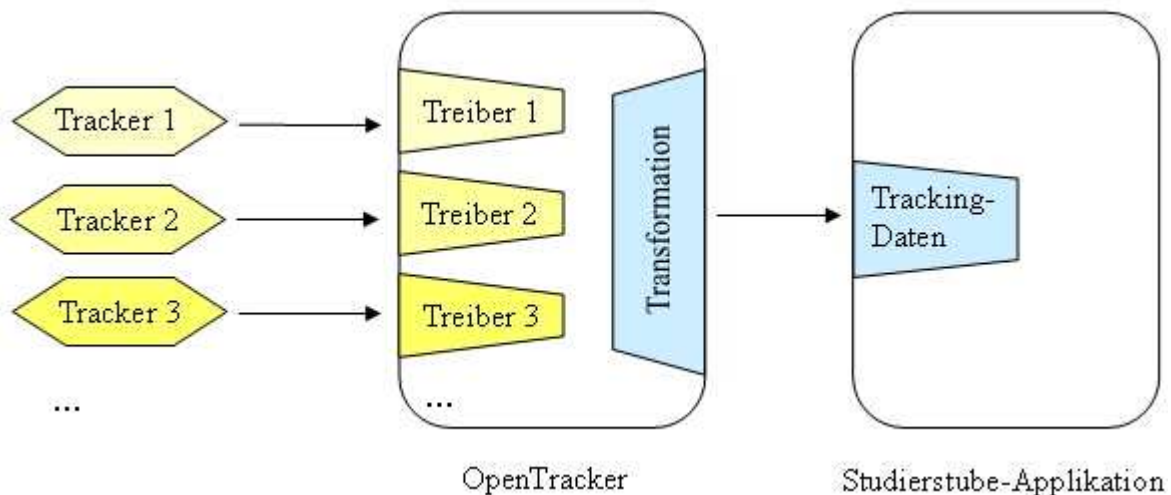
**Abbildung 4.12: (a) das Bild der Videokamera, (b) das Schwarz-Weiß-Bild mit einer rot markierten quadratischen Fläche, (c) der Marker wird mit einer Grafik überlagert.**

Während des Tracking-Vorganges muß jederzeit der gesamte Marker sichtbar sein, um ein fehlerfreies Tracking zu ermöglichen. Wird der Marker jedoch z.B. vom Benutzer in der Hand gehalten, wird er oft durch den Daumen verdeckt. Darüber hinaus können sich auch Personen oder Gegenstände zwischen Marker und Kamera befinden. Im Rahmen dieser Diplomarbeit wurde ein System zur Lösung dieses Problems entwickelt.

### 4.3 OpenTracker

Für das Tracking ist nicht die Studierstube-Applikation selbst, sondern OpenTracker [Reitmayr, 2001b], eine separate Applikation zur Verwaltung der Tracking-Daten, zuständig. OpenTracker empfängt die Daten aller verwendeten Tracker, transformiert sie entsprechend und leitet sie an die Studierstube-Applikation weiter.

Abbildung 4.13 stellt die Beziehung zwischen OpenTracker und der Studierstube-Applikation dar.



**Abbildung 4.13: Die Beziehung zwischen OpenTracker und der Studierstube.**

### 4.3.1 Spezifikation der Tracker

Die Spezifikation der Tracker und der notwendigen Transformationen erfolgt in einem XML-File. Die Elemente werden in Form einer Baumstruktur angeordnet, d.h. jedes Element entspricht einem Knoten oder einem Blatt. Es gibt folgende drei Knotenarten:

- Source Nodes
- Sink Nodes
- Filter Nodes

Source Nodes empfangen Tracking-Daten und verarbeiten sie je nach Art der Daten weiter. Sie enthalten z.B. Informationen über die Treiber für die zusätzlichen Geräte, die beim magnetischen und inertialen Tracking benötigt werden. Andere Source Nodes stellen Verbindungen zur ARToolKit Library, zum Keyboard oder zum Netzwerk dar. Soll also z.B. das Tracking mittels Keyboard durch optisches Tracking ersetzt werden, muß lediglich der Typ des Source Nodes geändert werden. Außerdem kann das Tracking, wie in Kapitel 4.3.2 beschrieben, auf einen anderen Computer im Netzwerk ausgelagert werden. Es besteht weiters die Möglichkeit, konstante Werte für die Tracking-Information in das XML-File einzutragen. Dies ist vor allem während der Testphase einer Anwendung sehr nützlich.

Im Gegensatz zu Source Nodes versenden Sink Nodes Tracking-Daten. Sie leiten die Werte ihrer Kindknoten z.B. an ein Netzwerk oder einen Shared Memory weiter.

Filter Nodes erhalten von ihren Kindknoten – im allgemeinen Source Nodes oder weitere Filter Nodes – Werte und transformieren diese. Die Art der Transformation richtet sich nach dem Typ des Filter Nodes. Die folgende Liste enthält die wichtigsten Filter Nodes und eine kurze Beschreibung:

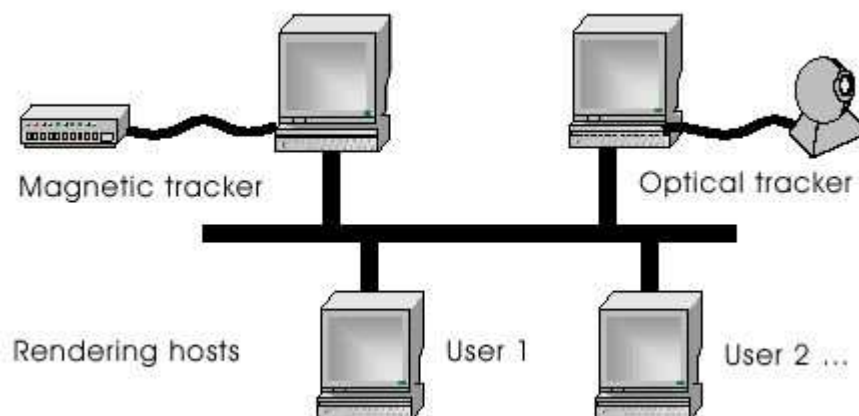
- Transformation Filter Nodes führen geometrische Transformationen – wie z.B. Translation, Rotation oder Skalierung – durch. Auf diese Weise kann z.B. die Transformation der Daten von dem Koordinatensystem des Trackers in das Koordinatensystem der virtuellen Szene einfach vorgenommen werden.
- Noise Filter Nodes korrigieren bis zu einem gewissen Grad fehlerhafte Messungen. Wird das Tracking-System durch äußere Einflüsse gestört, werden möglicherweise fehlerhafte Daten übermittelt. Ist z.B. die Entfernung zwischen der letzten Position und der aktuellen auffallend groß, handelt es sich mit großer Wahrscheinlichkeit um einen Meßfehler. Um diesen Meßfehler nicht an die Studierstube-Applikation weiterzugeben, kann z.B. anstatt der Tracking-Daten selbst ein gewichtetes Mittel der vorangegangenen Meßwerte verwendet werden.
- Merge Filter Nodes kombinieren die Werte ihrer Kindknoten. So können z.B. die Position eines getrackten Objekts durch einen optischen Tracker, seine Orientierung aber durch einen inertialen Tracker ermittelt werden.
- Conversion Filter Nodes konvertieren die Werte ihrer Kindknoten in einen anderen Datentyp. Dadurch können z.B. 2D-Daten von einer Maus in 3D-Daten umgewandelt werden, indem für die dritte Koordinate ein konstanter Wert hinzugefügt wird.
- Confidence Filter Nodes leiten jenen Tracking-Wert ihrer Kindknoten weiter, der die höchste Genauigkeit hat. Ein Tracker übermittelt zusätzlich zu den Tracking-Daten meist auch einen sogenannten Confidence-Wert. Dieser gibt an, mit welcher Wahrscheinlichkeit der gemessene Wert der tatsächlichen Position des getrackten Objekts entspricht. Um ein möglichst gutes Tracking-Ergebnis zu erhalten, können auch zwei oder mehr Tracker zum Tracken desselben Objekts verwendet werden.

Ein Confidence Filter wählt aus diesen Tracking-Daten jene mit der höchsten Wahrscheinlichkeit aus.

- Prediction Filter Nodes versuchen die kommenden Tracking-Daten vorherzusagen, um Störungen des Trackers zu kompensieren. Ist also z.B. ein Marker für kurze Zeit verdeckt, analysiert der Prediction Filter die vorangegangenen Tracking-Werte und berechnet die wahrscheinlichste Position des Markers.
- Store-and-forward Filter Nodes senden bei Ausfall eines Trackers den zuletzt gemessenen Wert solange, bis der Tracker wieder neue Daten generiert.
- Undistortion Filter Nodes werden verwendet, um Störungen im Magnetfeld eines magnetischen Trackers zu neutralisieren.

### 4.3.2 Distributed Tracking

OpenTracker ermöglicht das Senden und Empfangen von Tracker-Daten über ein Netzwerk (siehe Abbildung 4.14). Dadurch können das Tracking und die Studierstube-Applikation auf unterschiedlichen Computern ausgeführt werden. Die Performance einer Studierstube-Anwendung kann auf diese Art deutlich verbessert werden. Außerdem können dadurch auch Multiprozessor-Systeme und unterschiedliche Betriebssysteme in die Studierstube integriert werden.



**Abbildung 4.14: OpenTracker ermöglicht das Senden und Empfangen von Tracker-Daten über ein Netzwerk.**

Die Kommunikation zwischen den einzelnen Computern des Netzwerks erfolgt mittels Multicast [Boger, 1999]. Dabei wird jede Nachricht vom Sender an alle Mitglieder einer Multicast-Gruppe geschickt. Alle Teilnehmer einer solchen Multicast-Gruppe müssen sich zunächst bei der Gruppe anmelden. Anschließend kann an alle Gruppenmitglieder gesendet werden oder von anderen empfangen werden. Um die Mitgliedschaft in der Gruppe zu beenden, muß eine Abmeldung erfolgen. Zur Multicast-Kommunikation werden bestimmte IP-Adressen verwendet, die nicht an Hostrechner gebunden sind, sondern ausschließlich für die Multicast-Kommunikation reserviert sind. Diese Adressen liegen in einem Bereich zwischen 224.0.0.1 und 239.255.255.255, wobei die Adressen bis 224.0.0.255 für die Übertragung von Multicast-Routinginformationen reserviert sind.

## **5. Implementierung**

### **5.1 Hardware**

Der Tisch wurde aus einer Plexiglasplatte, einem Holzrahmen und zwei Böcken gefertigt. Die Plexiglasplatte wurde zur Stabilisierung auf den Holzrahmen geschraubt. Der Holzrahmen wurde nicht fix auf die Böcke montiert, um den Tisch bei Bedarf abbauen zu können. Es wurde jedoch eine Halterung angebracht, um die Tischplatte auf den Böcken fixieren zu können. Die Ausmaße der Tischplatte betragen 1,5 x 1 Meter. Die Höhe der Böcke kann verstellt werden, sodaß die Spieler entweder sitzen oder stehen können.

Abbildung 5.1 zeigt den fertigen Tisch.



**Abbildung 5.1: Der Glastisch.**

Es wurden außerdem ARToolKit-Marker mit einer Seitenlänge von 10 Zentimetern angefertigt. Diese Größe wurde gewählt, um auch dann ein gutes Tracking-Ergebnis zu erhalten, wenn die Entfernung zwischen Kamera und Marker sehr groß ist. Das ist dann der Fall, wenn sich der Marker auf der, von der Kamera aus gesehen, gegenüberliegenden Seite des Tisches befindet oder wenn die Böcke auf die höchste Stufe eingestellt wurden. Die Marker wurden auf einen fünf Millimeter dicken Karton geklebt, um zu verhindern, daß sie verbogen werden. Außerdem lassen sie sich dadurch leichter ergreifen. Als Symbole wurden Zahlen verwendet, wie in Abbildung 5.2 zu sehen ist, die einige der Marker zeigt.



**Abbildung 5.2: Einige der ARToolKit-Marker.**

Die Halterung für die Kamera wurde an einem der Böcke befestigt. Sie wurde aus fünf Millimeter dickem Draht zurechtgebogen. Abbildung 5.3 zeigt die Kamerahalterung.



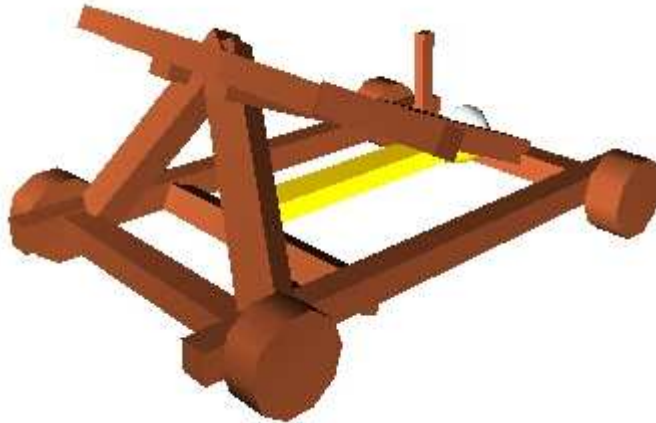
**Abbildung 5.3: Die Kamerahalterung wurde an einem der Böcke befestigt.**

## **5.2 3D-Modelle**

Für die im Spiel vorkommenden Objekte wurden 3D-Modelle angefertigt. Die folgenden Kapitel beschreiben die Modelle und die dazugehörigen Animationen. Um die Spielfiguren der beiden Spieler unterscheiden zu können, wird jedem Spieler eine Farbe zugewiesen. Spieler 1 spielt mit gelb, während Spieler 2 mit blau spielt. Die Katapulte und die Windmühlen sind mit Zahlen in den entsprechenden Farben gekennzeichnet. Diese Zahlen geben an, um die wievielte Spielfigur derselben Art es sich handelt. Wird also z.B. mit zwei Katapulten pro Spieler gespielt, so sind diese mit den Ziffern 1 und 2 in gelb und blau gekennzeichnet.

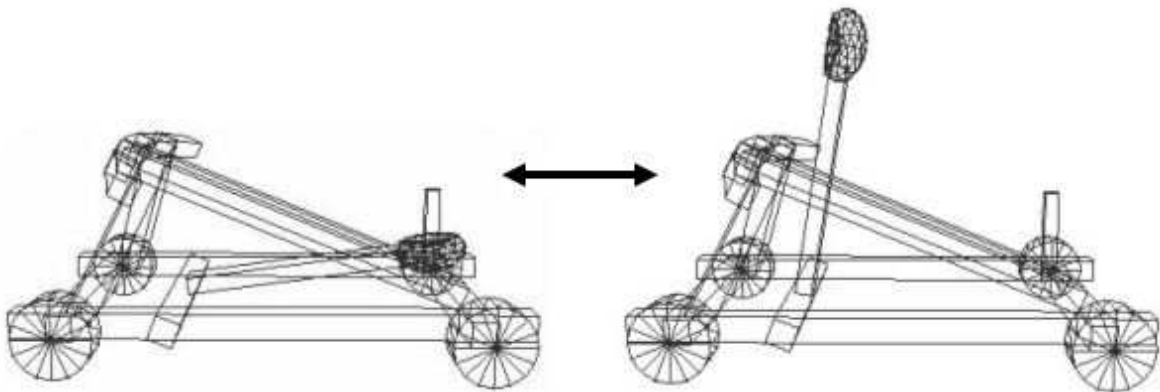
### **5.2.1 Das Katapult**

Das Katapult (siehe Abbildung 5.4) dient zum Abschießen der gegnerischen Ballons. Ein Teil des Katapults ist in der Farbe des jeweiligen Spielers eingefärbt.



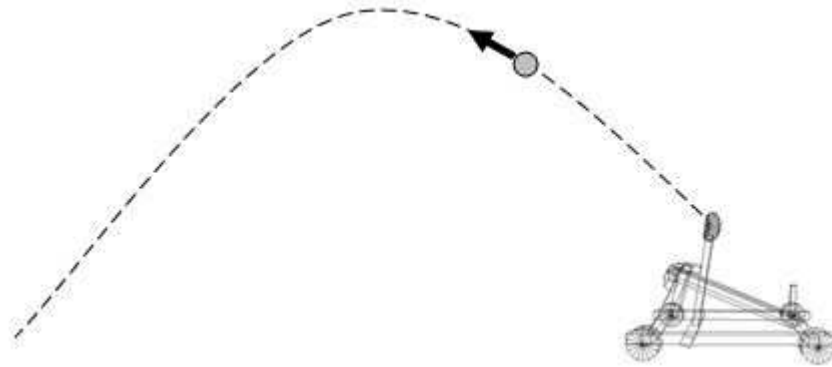
**Abbildung 5.4: Das Katapult des Spielers mit der Farbe gelb.**

Es besitzt einen beweglichen Katapultarm, auf dem eine Kugel liegt. Der Katapultarm bewegt sich von der Ausgangsstellung aus nach oben und anschließend wieder zurück (siehe Abbildung 5.5).



**Abbildung 5.5: Der Katapultarm bewegt sich von der Ausgangsstellung aus nach oben und anschließend wieder zurück.**

Am höchsten Punkt löst sich die Kugel vom Katapultarm und fliegt auf einer parabolischen Flugbahn. Abbildung 5.6 zeigt eine Skizze der Flugbahn .



**Abbildung 5.6: Die Flugbahn der Kugel hat die Form einer Parabel.**

### 5.2.2 Die Windmühle

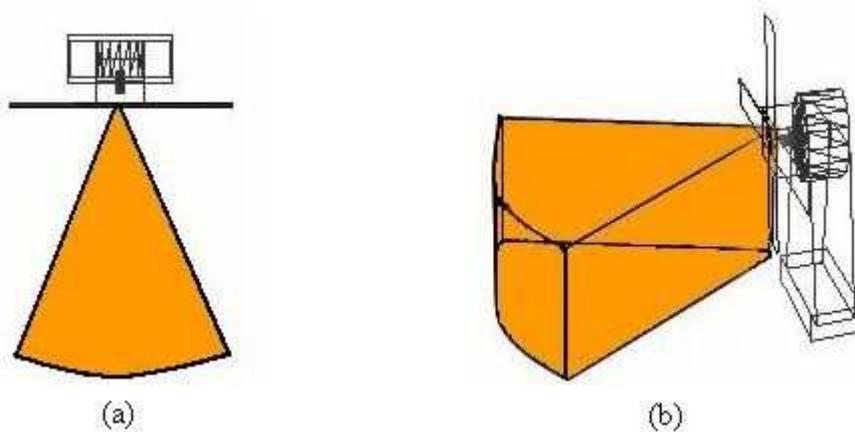
Mit Hilfe der Windmühle kann der Spieler seine Ballons aus der Flugbahn der Kugel eines gegnerischen Katapults befördern. Im Actionspiel-Modus und am Ende eines Zuges im Strategiespiel-Modus drehen sich die Flügel der Windmühle. Abbildung 5.7 zeigt die Windmühle des Spielers mit der Farbe blau.



**Abbildung 5.7: Die Windmühle des Spielers mit der Farbe blau.**

Die Form des Raums, der von dem Wind, der von der Windmühle ausgeht, beeinflusst wird, ist in Abbildung 5.8 orange eingefärbt. Abbildung 5.8a zeigt diesen Raum von oben, während er in Abbildung 5.8b von der Seite zu sehen ist. Die Windstärke nimmt mit der

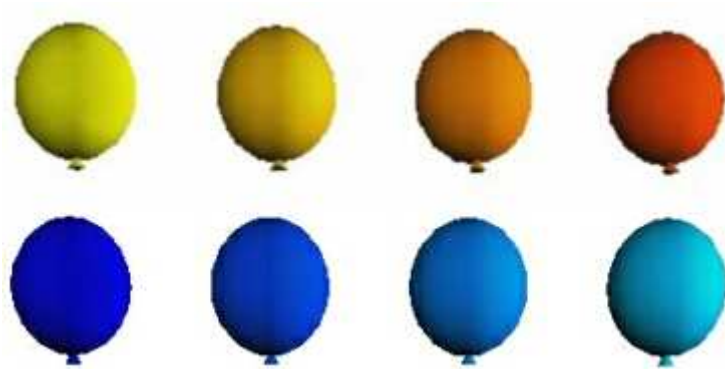
Entfernung von der Windmühle ab. Die Höhe wird hierbei jedoch nicht berücksichtigt, da die Höhe der Windmühle nicht verändert werden kann. Dadurch wird der Fall vermieden, daß ein Ballon so hoch steigt, daß er von dem Wind der Windmühle nicht mehr erreicht werden kann.



**Abbildung 5.8: Der vom Wind beeinflusste Raum von oben (a) und von der Seite (b).**

### 5.2.3 Der Ballon

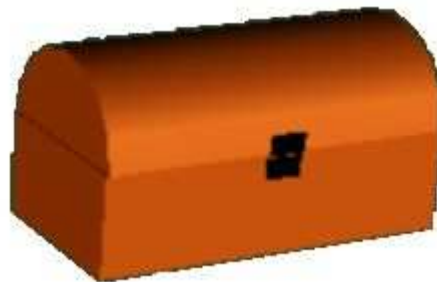
Zu Beginn des Spieles schwebt jeder Ballon auf seiner jeweiligen Ausgangsposition. Die Höhe des Ballons wird dabei zufällig bestimmt, sie liegt jedoch innerhalb eines festgelegten Intervalls. Während des Spiels bewegen sich die Ballons leicht auf und ab, um die Szene ansprechender zu gestalten. Aus demselben Grund variieren die Farben der Ballons. Für den Spieler, der mit der Farbe gelb spielt, folgen die Farben einem Farbverlauf von gelb nach rot. Die Farben der Ballons des anderen Spielers liegen zwischen blau und türkis (siehe Abbildung 5.9). Wird ein Ballon von der Kugel eines Katapults getroffen, so verliert er an Luft und verschwindet anschließend.



**Abbildung 5.9: Die Farben der Ballons.**

### **5.2.4 Die Schatztruhe**

Mehrere Schatztruhen (siehe Abbildung 5.10) stehen zufällig auf dem Spielfeld verteilt. Bewegt ein Spieler ein Katapult auf eine dieser Schatztruhen, so verschwindet sie und das Katapult kann anschließend weiter schießen.

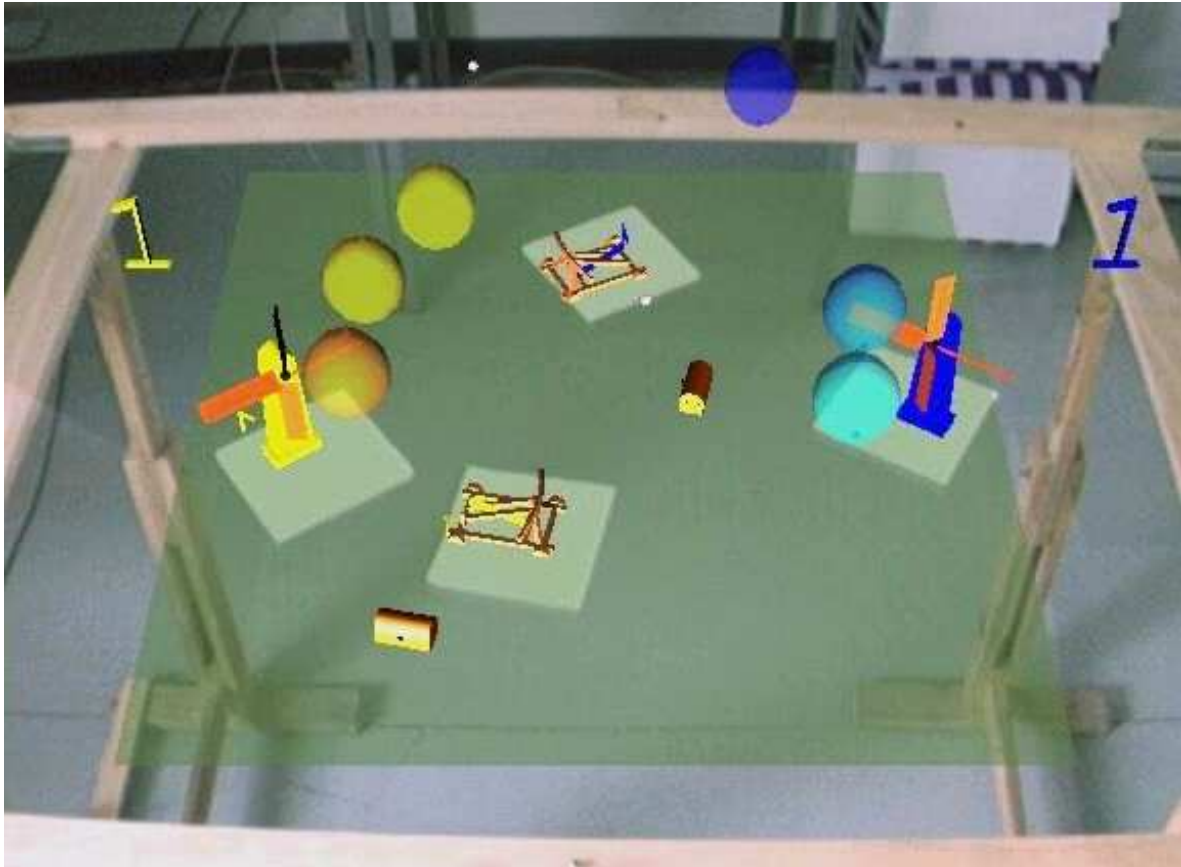


**Abbildung 5.10: Die Schatztruhe.**

## **5.3 Spieltypen**

### **5.3.1 Actionspiel**

Im Actionspiel-Modus können beide Spieler ihre Figuren gleichzeitig bewegen. Es geht bei diesem Modus vor allem um Schnelligkeit, gutes Zielvermögen und eine gute dreidimensionale Vorstellungskraft. Abbildung 5.11 zeigt eine Szene des Actionspiels.



**Abbildung 5.11: Der Actionspiel-Modus.**

Der aktuelle Punktestand wird oberhalb des Spielfeldes in der Farbe des jeweiligen Spielers angezeigt. Jeder gegnerische Ballon, der getroffen wird, zählt einen Punkt. Wird jedoch versehentlich ein eigener Ballon abgeschossen, so wird ein Punkt vom Punktestand abgezogen.

Die unterschiedliche Höhe der Ballons und ein eventuell vorhandenes Terrain erschweren den Zielvorgang. Jener Spieler, der zuerst alle gegnerischen Ballons getroffen hat, gilt als Sieger. Am Ende des Spiels wird der Sieger durch einen animierten Schriftzug bekanntgegeben, wie in Abbildung 5.12 zu sehen ist.



**Abbildung 5.12: Der Spieler, der mit der Farbe gelb spielt, hat gewonnen.**

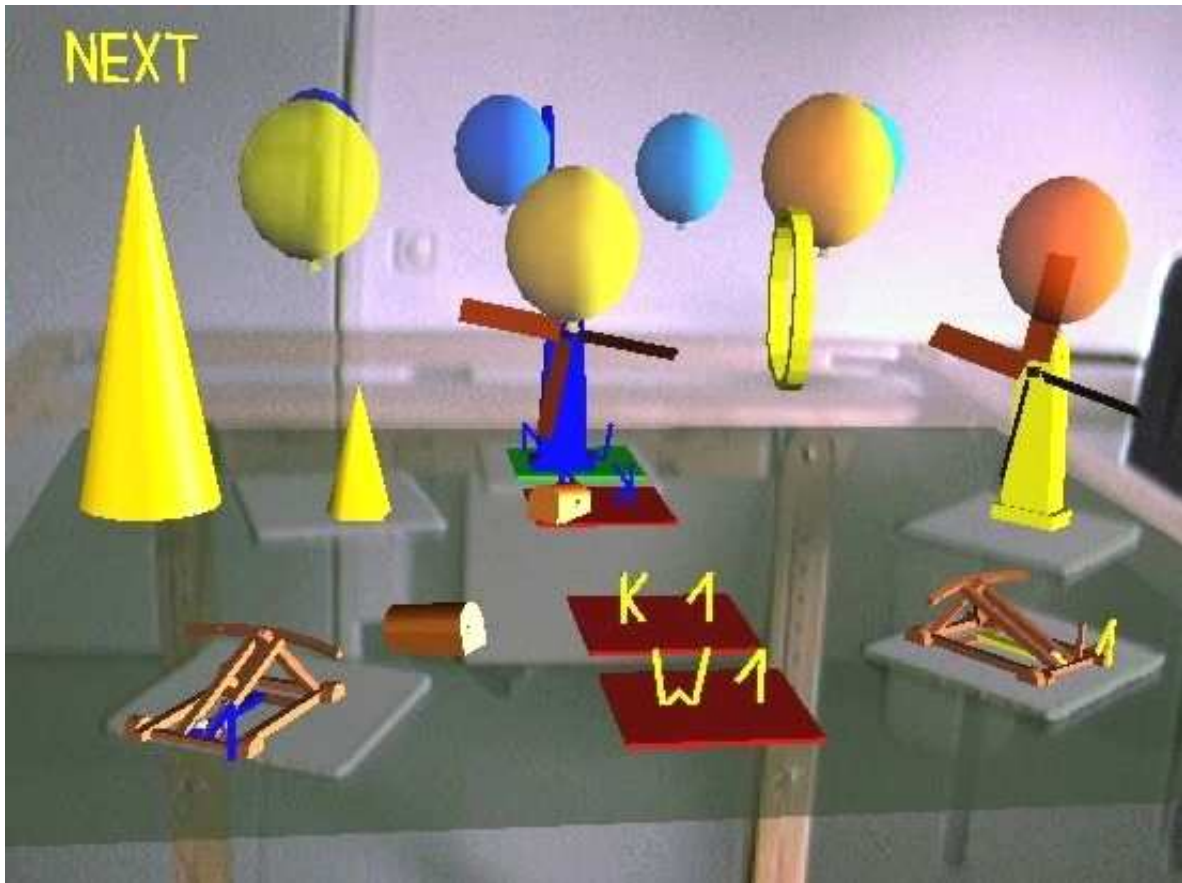
### 5.3.2 Strategiespiel

Der Ablauf des Strategiespiels lässt sich in mehrere Phasen einteilen:

- Die Marker werden auf ihre jeweiligen Ausgangspositionen gelegt.
- Die Spieler führen abwechselnd ihre Spielzüge durch.
- Zwischen diesen Spielzügen werden die Spielfiguren des aktiven Spielers animiert.
- Sobald alle Ballons eines Spielers getroffen wurden, gilt das Spiel als beendet.

Um beiden Spielern gleiche Chancen zu garantieren, werden die Marker zu Beginn des Spieles auf vordefinierte Positionen gelegt. Diese Positionen sind durch rote Quadrate markiert. Sobald ein Marker korrekt plaziert ist, wird das entsprechende Quadrat grün. Um zwischen den Feldern der einzelnen Spielfiguren unterscheiden zu können, sind diese durch Farbe und Text gekennzeichnet. Der Schriftzug „K 1“ in gelber Farbe bezeichnet das Feld

des ersten Katapults des gelben Spielers. „W 2“ in blau entspricht der zweiten Windmühle des blauen Spielers, etc. In Abbildung 5.13 ist diese Startphase des Strategiespiel-Modus zu sehen. Die Windmühle des blauen Spielers befindet sich bereits auf dem korrekten Feld, was an der grünen Farbe zu erkennen ist. Die übrigen Spielfiguren müssen noch entsprechend positioniert werden.

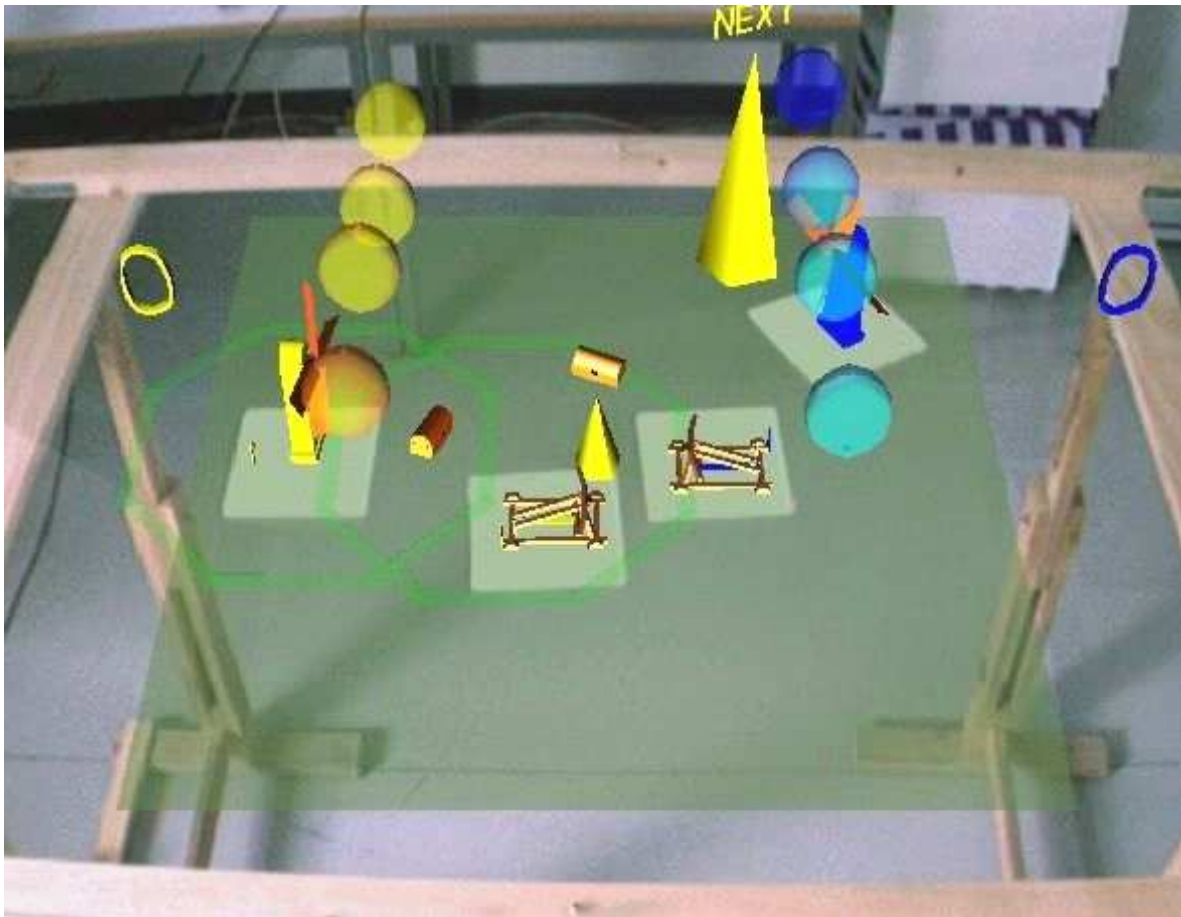


**Abbildung 5.13: Die Startphase des Strategiespiel-Modus.**

Befinden sich alle Spielfiguren auf ihren Ausgangspositionen, verschwinden die Positionsmarkierungen. Jener Spieler, der mit der Farbe gelb spielt, ist an der Reihe und kann seine Spielfiguren nun bewegen.

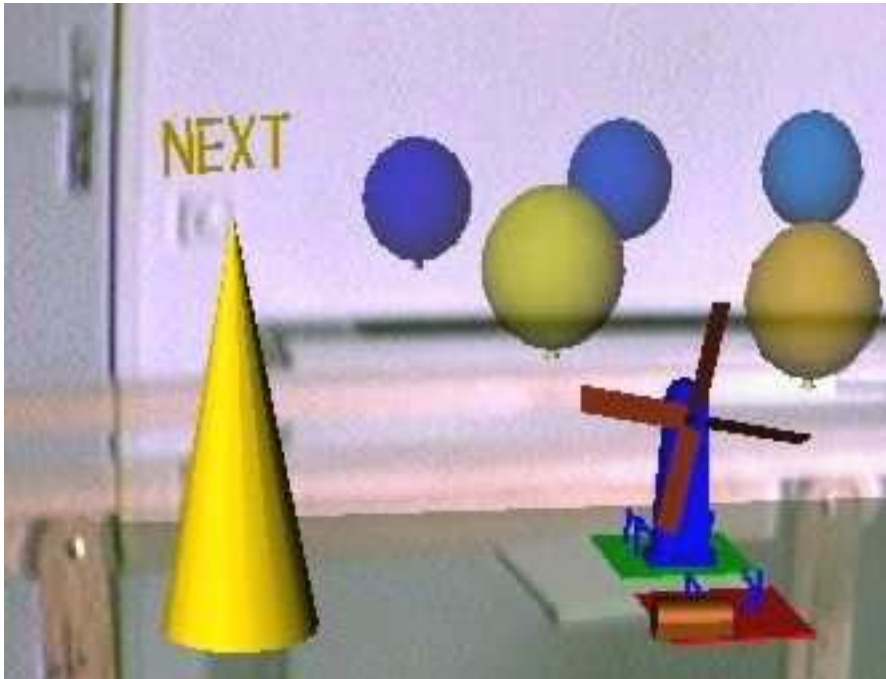
Das Tabletop Game verfügt jedoch nicht, wie die meisten Brettspiele, über vordefinierte Spielfelder. Bei Brettspielen dient dies zur Beschränkung der möglichen Spielzüge. Eine Spielfigur darf z.B. nur um eine bestimmte Anzahl an Feldern bewegt werden oder die Richtung der Bewegung ist vorgegeben. Diese Art der Bewegungseinschränkung wurde

beim Tabletop Game bewußt vermieden. Die Spieler sollen nicht durch diese Form der Beschränkung der Spielzüge vom eigentlichen Spiel abgelenkt werden. Daher gibt es beim Tabletop Game auch kein Würfeln oder eine andere Form der Spielzugbestimmung. Die Bewegungsmöglichkeiten der Spielfiguren werden jedoch auf eine andere Weise eingeschränkt. Die Einschränkung besteht darin, daß die Entfernung vom Ausgangspunkt einen Maximalwert nicht überschreiten darf. Der Bereich, innerhalb dessen die Spielfigur bewegt werden darf, ist durch ein grünes Achteck gekennzeichnet (siehe Abbildung 5.14). Wird diese maximale Entfernung überschritten, so färbt sich das Achteck rot. Das zu dem jeweiligen Marker gehörende virtuelle Modell bleibt am Rand des Spielbereichs stehen. Erst wenn der Marker wieder in den Spielbereich hineinbewegt wird, folgt die Grafik dem Marker erneut.



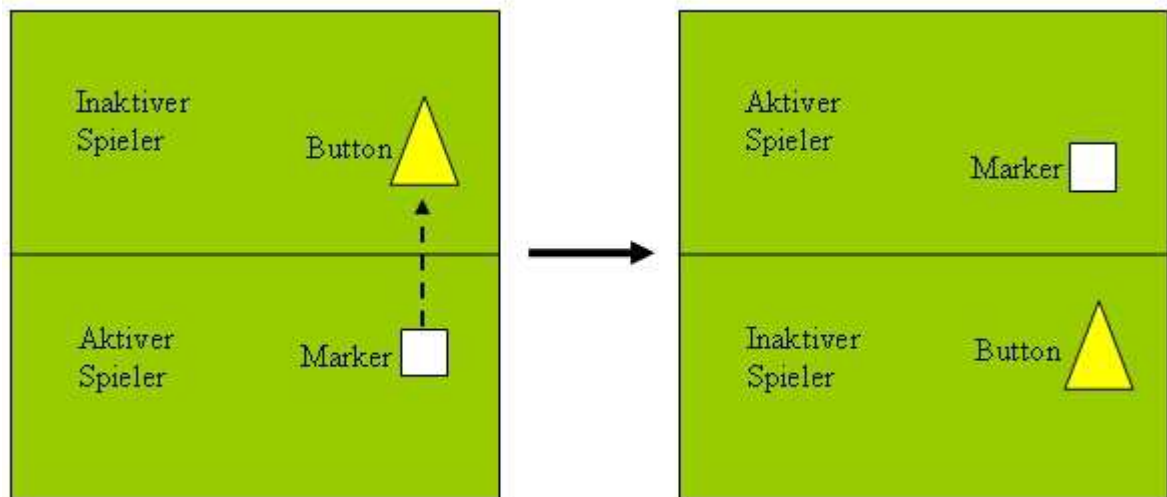
**Abbildung 5.14: Die Spielfiguren können nur innerhalb des jeweiligen Achtecks bewegt werden.**

Ein Spieler darf während seines Spielzuges jede seiner Spielfiguren beliebig innerhalb der Achtecke bewegen. Ist er mit der Position seiner Spielfiguren zufrieden, zeigt er dies dem System mit Hilfe eines Buttons an, der sich am Spielfeldrand befindet. Dieser Button hat die Form eines Kegels, damit er auch bei eingebledetem Terrain gut sichtbar ist, und trägt den Schriftzug „Next“.



**Abbildung 5.15: Der Button hat die Form eines Kegels.**

Ausgelöst wird der Button durch einen speziellen Marker, der zusätzlich zu den Spielfigur-Markern vorhanden ist. Der Spielerwechsel erfolgt, indem der aktive Spieler den Marker auf den Button, der sich in der Spielhälfte des inaktiven Spielers befindet, schiebt. Nachdem der Button ausgelöst wurde, wechselt er seine Position und befindet sich anschließend in der Spielhälfte des anderen Spielers. Dadurch wird der Marker zwischen den Spielern hin- und hergeschoben. Der Marker liegt daher immer beim aktiven Spieler. Abbildung 5.16 zeigt den Wechsel des aktiven Spielers durch Verschieben des Markers und die neue Position des Buttons.



**Abbildung 5.16: Der Spielerwechsel erfolgt mit Hilfe eines speziellen Markers.**

Zwischen dem Auslösen des Buttons und dem Wechsel seiner Position werden die Spielfiguren des aktuellen Spielers animiert, d.h., die Katapulte schießen je eine Kugel ab und die Ballons, die sich im Einflußbereich der Windmühlen befinden, bewegen sich entsprechend. Getroffene Ballons verlieren an Größe und verschwinden anschließend. Erst nach Ablauf dieser Animationen wechselt der Button auf die andere Seite des Spielfelds und der inaktive Spieler wird zum aktiven Spieler.

Es kann jedoch vorkommen, daß unabsichtlich auch ein Marker des inaktiven Spielers bewegt wird. Die Position der Spielfigur stimmt in diesem Fall nicht mehr mit jener Position überein, die sie am Ende des letzten Spielzuges eingenommen hat. Aus diesem Grund werden die Positionen der Marker am Ende eines Spielzuges gespeichert. Diese Daten werden zu Beginn des nächsten Spielzuges mit den aktuellen Positionen der Marker verglichen. Sollte sich die Position eines oder mehrerer Marker verändert haben, so erscheinen anstatt der Achtecke die quadratischen Markierungen aus der Anfangsphase des Spiels an den gespeicherten Positionen des vorigen Spielzuges. Bei jenen Markern, die an den korrekten Positionen liegen, ist die Farbe des Quadrats grün. Im anderen Fall ist das Quadrat rot eingefärbt. Sobald der Spieler diese Marker jedoch wieder an ihre Position zurückgelegt hat, verschwinden die Quadrate. Statt dessen werden die Achtecke angezeigt und der Spieler kann seinen Spielzug wie gewohnt ausführen.

Der Strategiespiel-Modus unterscheidet sich vom Actionspiel-Modus (siehe Kapitel 5.3.1) zwar durch die Regeln, nicht aber durch das Ziel des Spieles. Gewonnen hat ebenfalls, wer zuerst alle gegnerischen Ballons getroffen hat. Auch die Ermittlung des Punktestandes und die Animation am Ende des Spiels entsprechen jenen des Actionspiel-Modus.

## **5.4 Steuerung**

### **5.4.1 Default**

Bei dieser Form der Steuerung wird das virtuelle Objekt immer an der Position des Markers angezeigt. Da es sich hierbei um die einfachste Form der Steuerung handelt, wird sie im Tabletop Game als „Default“-Steuerung bezeichnet. Die von der Tracking-Software gelieferten Positionsdaten werden unverändert auf die Spielfigur übertragen. Wird jedoch im Strategiespiel-Modus ein Marker aus dem ihn umgebenden Achteck herausbewegt, so wird diese Übertragung so lange unterbrochen, bis sich der Marker wieder innerhalb des Achtecks befindet.

### **5.4.2 FollowUp**

Im FollowUp-Modus werden die Daten der Tracking-Software nicht direkt auf das virtuelle Objekt übertragen. Statt dessen folgt das virtuelle Objekt dem Marker mit gleichbleibender Geschwindigkeit. Sobald es den Marker erreicht hat, bleibt es stehen.

## **5.5 Optionen**

### **5.5.1 Anzahl der Spielfiguren**

Die Anzahl der Spielfiguren pro Spieler kann variiert werden. So kann also z.B. mit zwei Katapulten oder drei Windmühlen gespielt werden. Auch die Anzahl der Ballons und der Schatztruhen kann vor Beginn des Spieles festgelegt werden.

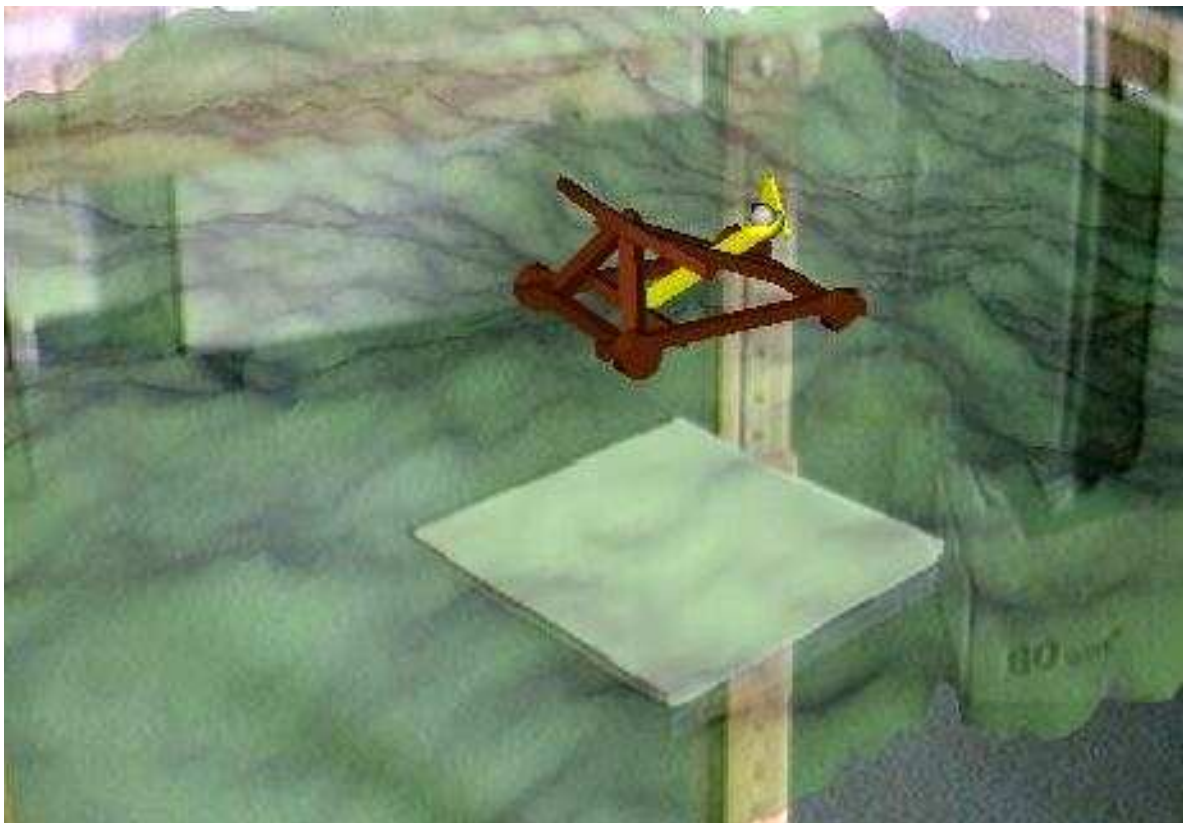
### **5.5.2 Neue Ballons**

Um immer die gleiche Anzahl an Ballons am Spielfeld zu haben, gibt es die Option, für jeden abgeschossenen Ballon einen neuen Ballon in das Spiel einzubringen. Die Anzahl der

Zielobjekte wird dadurch konstant gehalten. In diesem Fall endet das Spiel dann, wenn eine zuvor festgelegte Anzahl an gegnerischen Ballons getroffen wurde.

### 5.5.3 Terrain

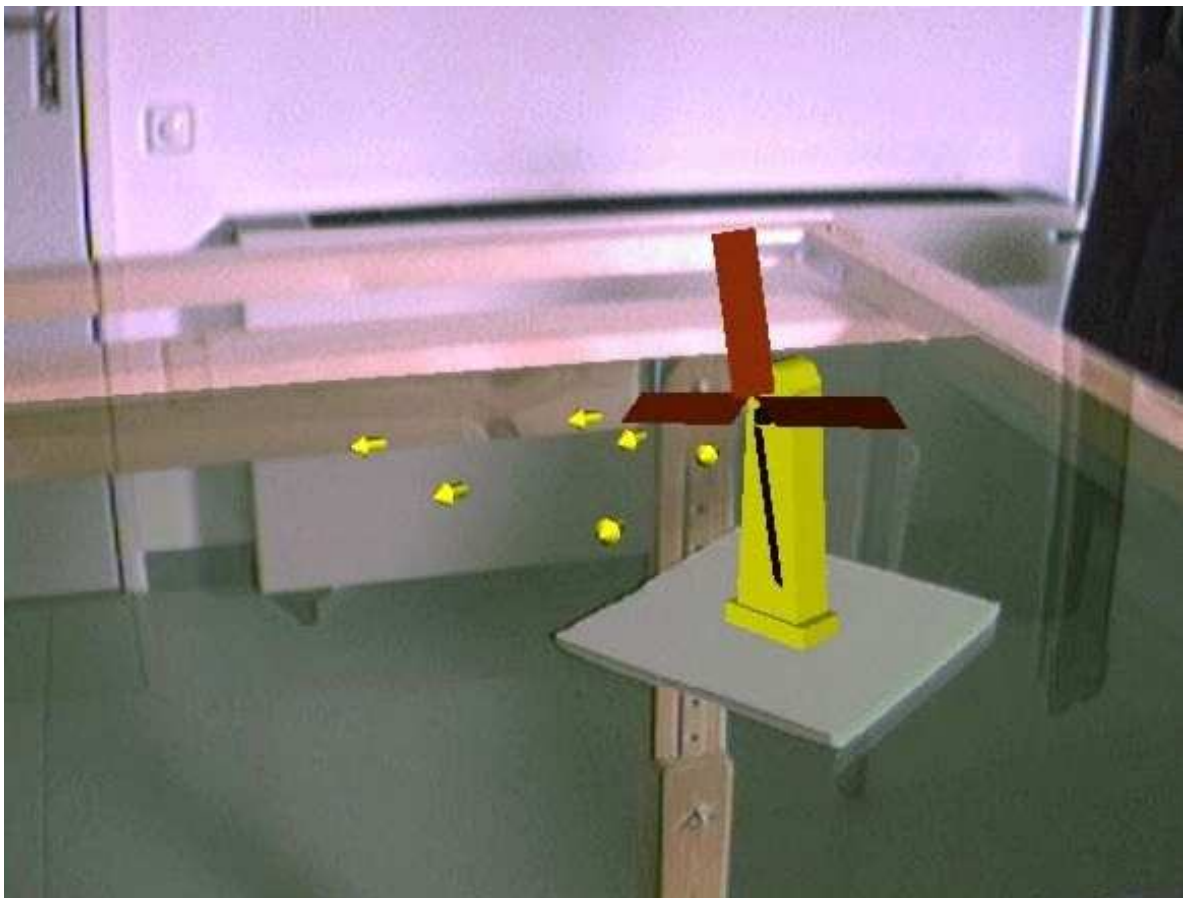
Um die Szene anspruchsvoller zu gestalten und das Zielen auf die Ballons zu erschweren, kann ein Terrain eingeblendet werden. Dieses Terrain wird mit Hilfe des Diamond Square Algorithmus [Martz, 1996] beim Start des Spiels neu erzeugt. Während des Spiels wird die Höhenangabe, die die Tracking-Software für die Marker liefert, durch die jeweilige Höhe des Terrains ersetzt. Die 3D-Modelle bewegen sich daher auf dem Terrain, wie in Abbildung 5.17 zu sehen ist.



**Abbildung 5.17: Die 3D-Modelle bewegen sich auf dem Terrain.**

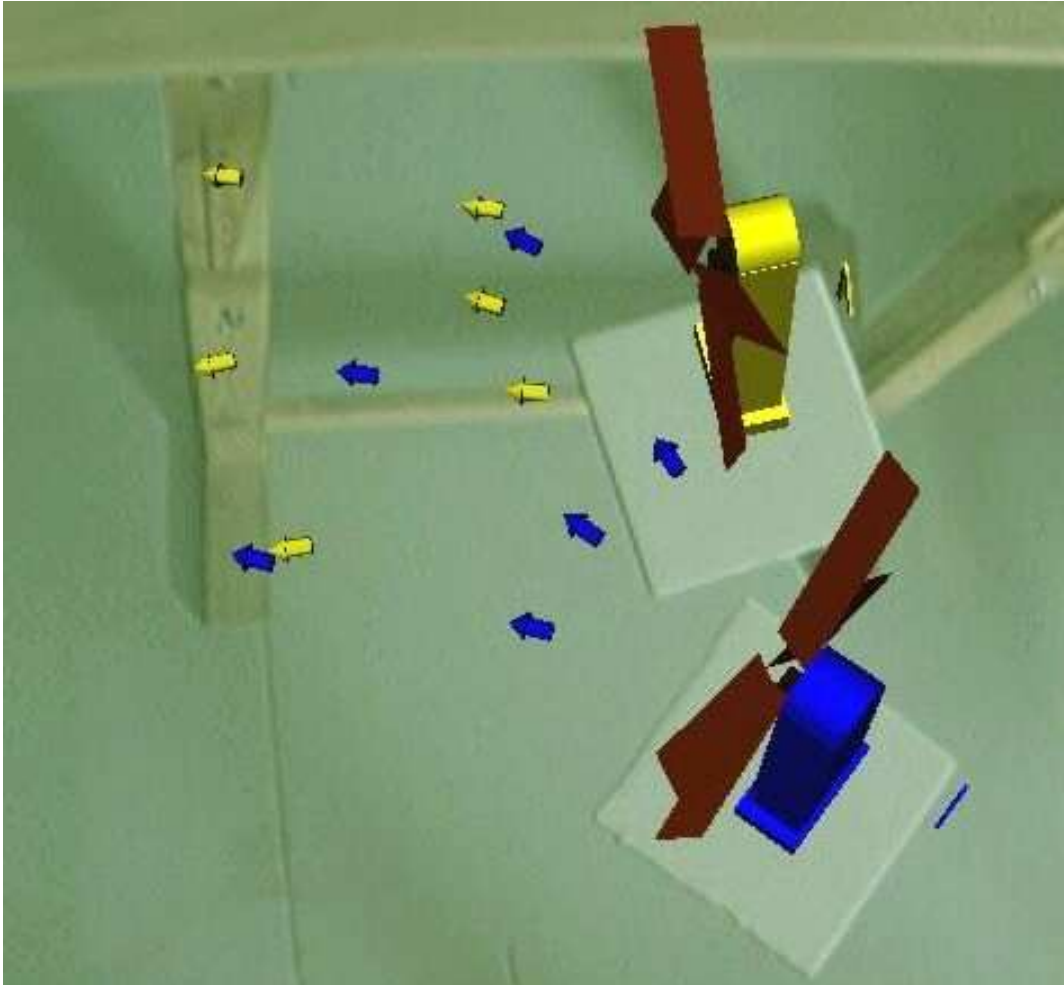
#### 5.5.4 Strömungsvisualisierung

Die Windströmungen, die von den Windmühlen ausgehen, können angezeigt werden, um dem Spieler die Positionierung seiner Windmühlen zu erleichtern. Die Darstellung dieser Strömungen erfolgt durch Pfeile, die sowohl den von der Strömung beeinflussten Raum, als auch die Strömungsrichtung anzeigen. Je nach Größe dieses Raums werden mehr oder weniger Reihen von Pfeilen dargestellt. Eine Reihe besteht aus je drei Pfeilen, die sich jeweils an den Rändern und in der Mitte der Reihe befinden (siehe Abbildung 5.18).



**Abbildung 5.18: Eine Reihe besteht aus je drei Pfeilen.**

Die Pfeile werden nur in einer Ebene angezeigt, da die Höhe der Luftströmung keinen Einfluß auf die Bewegungen der Ballons hat (siehe Kapitel 5.2.2). Sollten sich zwei solcher Einflußbereiche überschneiden, so ändern sich, wie in Abbildung 5.19 zu sehen, an dieser Stelle die Richtungen der Strömungen.



**Abbildung 5.19: Überschneiden sich zwei Strömungen, so ändern sich die Strömungsrichtungen.**

### 5.5.5 Schatztruhen

Bewegt ein Spieler ein Katapult auf eine Schatztruhe, so hat das Auswirkungen auf die Flugbahn der Kugeln dieses Katapults. Die Parameter der Parabel, entlang der sich die Kugeln bewegen, werden verändert. Die Kugeln fliegen in Folge höher und weiter. Die Ballons können nun also auch aus größerer Entfernung getroffen werden. Die anderen Katapulte werden davon nicht beeinflusst.

## 5.6 Spezifikation der Tracker

Die Spezifikation der Tracker erfolgt bei OpenTracker, wie in 4.3.1 erläutert, in XML-Files. Da das Tracking und das Rendering beim Tabletop Game auf verschiedenen Rechnern durchgeführt werden, müssen die Tracking-Daten über ein Netzwerk zum Render-PC übertragen werden. Der Tracking-PC fungiert dabei als Server, der Render-PC als Client. Es muß sowohl für den Server als auch für den Client ein XML-File spezifiziert werden.

Im XML-File des Servers wird für jeden Marker ein ARToolKit Source Node angelegt. Anschließend werden die Daten mit Hilfe von Transformation Filter Nodes vom Koordinatensystem der Kamera in das des Tabletop Games transformiert. Durch einen Network Sink Node werden die transformierten Daten an das Netzwerk weitergeleitet. Als Multicast-Adresse wurde 224.100.200.200 gewählt. Es könnte jedoch auch eine beliebige andere Adresse verwendet werden, wie in Kapitel 4.3.2 beschrieben wird. Abbildung 5.20 zeigt einen Ausschnitt aus dem XML-File des Servers, der die für einen einzelnen Marker notwendigen Spezifikationen enthält.

```
<NetworkSink name="Camera" number="1" multicast-address="224.100.200.200" port="12345">
  <ConsoleSink comment="Catapult1">
    <EventVirtualOrientationTransform rotation="1 0 0 -1.570796" rotationtype="axisangle">
      <EventTransform rotationtype="axisangle" rotation="1 0 0 0.7171">
        <EventTransform scale="0.001 0.001 0.001" rotation="1 0 0 0" translation="0 -0.1 1.5">
          <ARToolKitSource center="0 0" size="100 100" tag-file="../src/apps/tabletop/data/zero.patt"/>
        </EventTransform>
      </EventTransform>
    </EventVirtualOrientationTransform>
  </ConsoleSink>
</NetworkSink>
```

**Abbildung 5.20: Ein Ausschnitt aus dem XML-File des Servers.**

Im XML-File des Clients werden die Daten mit Hilfe eines Network Source Nodes empfangen und durch einen Stb<sup>1</sup> Sink Node an die Studierstube weitergeleitet. In Abbildung 5.21 ist der clientseitige Teil der Spezifikation eines Markers zu sehen.

```
<StbSink station="0" event="on">  
  <NetworkSource number="1" multicast-address="224.100.200.200" port="12345"/>  
</StbSink>
```

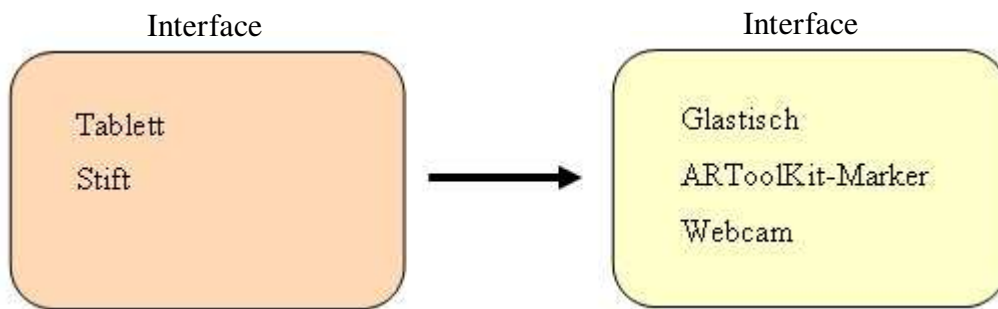
**Abbildung 5.21: Ein Ausschnitt aus dem XML-File des Clients.**

---

<sup>1</sup> Abkürzung für Studierstube

## **6. Ergebnisse**

Die Studierstube wurde im Rahmen dieses Projekts erfolgreich um ein Tangible User Interface erweitert. Anstatt Tablett und Stift werden hierbei Marker zur Steuerung einer Applikation und zur Interaktion mit den 3D-Modellen verwendet. Das neue Interface setzt sich aus einem Glastisch, ARToolKit-Markern und einer Webcam zusammen (siehe Abbildung 6.1).



**Abbildung 6.1: Das neue Interface setzt sich aus einem Glastisch, ARToolKit-Markern und einer Webcam zusammen.**

Dadurch ergeben sich folgende wesentliche Vorteile:

- Die virtuellen Objekte werden greifbar.
- Die Interaktion erfolgt kabellos.
- Es steht eine größere Anzahl an Eingabegeräten zur Verfügung.

Durch die Verwendung eines Tangible User Interface entfällt die Implementierung einer der in Kapitel 2.6.2 beschriebenen Methoden zur Manipulation von virtuellen Objekten. Bei all diesen Methoden müssen dem System Position und Orientierung der Hand des Benutzers bekannt sein. Das Tracking der Hand kann z.B. mit Hilfe eines Datenhandschuhs (siehe Kapitel 2.4.1) oder eines Pinchgloves (siehe Kapitel 2.4.2) durchgeführt werden. Dabei muß der Benutzer jedoch einen Handschuh tragen, was bei längerer Anwendung als unangenehm empfunden werden kann. Bei Verwendung einer 3D-Maus (siehe Kapitel 2.4.3) oder einer Kombination aus Pen & Tablet (siehe Kapitel 2.4.4) sind eine oder sogar beide Hände des Benutzers durch das Eingabegerät blockiert. Es besteht zwar die

Möglichkeit, das Eingabegerät abzulegen, sofern es nicht gebraucht wird, der Benutzer wird durch diesen Vorgang jedoch von der Applikation abgelenkt. Erfolgt die Interaktion mit dem System durch Spracheingabe, bleiben die Hände des Benutzers unbeeinflusst. Das Steuern eines Objekts mit Hilfe der Stimme erscheint jedoch unnatürlich, da dies der Realität widerspricht. Beim Tabletop Game muß die Interaktion mit der virtuellen Szene nicht vom Benutzer erlernt werden, da es sich um dieselbe Form der Interaktion handelt, die aus dem Alltag bekannt ist. Die Auswahl der virtuellen Objekte erfolgt direkt über die ARToolKit-Marker.

Ein weiterer Vorteil des Tabletop Games ist das Fehlen von Kabeln. Herkömmliche VR-Eingabegeräte sind meist durch Kabel mit dem System verbunden. Verwendet man z.B. Pen & Tablet, führen zusätzlich zu den Kabeln des Ausgabegeräts noch zwei weitere Kabel vom Benutzer zum System. Die Bewegungsfreiheit des Benutzers wird dadurch stark eingeschränkt. Beim Tabletop Game entfallen diese zusätzlichen Kabel durch die Verwendung von markerbasiertem optischen Tracking.

Die Anzahl der möglichen Eingabegeräte des Tabletop Games ist weder durch das Betriebssystem, noch durch die Anzahl der Anschlüsse des Computersystems begrenzt. Lediglich die Anzahl an Markern, die von ARToolKit getrackt werden kann, stellt eine obere Schranke dar. Um das Tabletop Game z.B. von zwei auf vier Spieler zu erweitern, müssen lediglich zusätzliche Marker hergestellt und die Software angepaßt werden. Der Kauf teurer Eingabegeräte und die anschließende Installation entfallen jedoch.

Die Verwendung eines Glastisches löst das Problem der Verdeckung von Markern. Dadurch können sich nicht unbeabsichtigt Personen oder Gegenstände zwischen der Kamera und den Markern befinden. Es sind daher immer alle virtuellen Objekte sichtbar. Um die Objekte aus verschiedenen Richtungen zu betrachten, kann sich der Benutzer nach Belieben um den Tisch herum bewegen oder die einzelnen Marker selbst drehen. Es ist sogar möglich, die Marker hochzuheben, solange sie dabei von der Kamera erfaßt werden können.

Die Lichtverhältnisse am Standort haben einen gewissen Einfluß auf die Qualität des Trackings. ARToolKit ist relativ unempfindlich gegenüber Änderungen der Helligkeit des

Umgebungslichts. Helle Deckenlampen können das Meßergebnis jedoch negativ beeinflussen, was bei der Wahl des Standorts berücksichtigt werden sollte.

Da die Bildwiederholungsrate einen großen Einfluß auf den Immersionseindruck des Benutzers hat, ist es wichtig, diese möglichst hoch zu halten. Dies kann durch eine Optimierung des Szenengraphs unterstützt werden. Die Anzahl der für ein 3D-Objekt verwendeten Dreiecke sollte minimiert werden, um die notwendigen Transformationen zu beschleunigen. Vor allem kleine Objekte mit runden Oberflächen werden oft mit unnötig vielen Dreiecken dargestellt.

Wird das Tabletop Game mit Keyboard Tracking ausgeführt, so wird annähernd die maximale Bildwiederholungsrate des Ausgabegeräts erreicht. Wird hingegen ARToolKit für das Tracking verwendet und werden Tracking und Rendering auf demselben PC durchgeführt, so sinkt die Bildwiederholungsrate auf vierzig Prozent des Maximalwerts. Aus diesem Grund werden beim Tabletop Game Interface Tracking und Rendering auf verschiedenen Computern ausgeführt. Die Netzwerkkommunikation beeinträchtigt den Rendering-PC im Vergleich zum optischen Tracking nur gering. Dabei wird eine Bildwiederholungsrate von 92 Prozent des Maximalwerts erreicht.

Das Tabletop Game Interface ist für die beiden Spieltypen Actionspiel und Strategiespiel gleichermaßen gut geeignet. Der Benutzer hat durch die spezielle Form der Interaktion die Möglichkeit, die Spielfiguren eigenhändig zu steuern. Bei konventionellen Computerspielen geschieht dies statt dessen meist nicht direkt, sondern indirekt durch Maus oder Joystick. Bei realen Spielen erfolgt das Bewegen der Spielfiguren jedoch immer direkt. Daher stellt das Tabletop Game Interface eine intuitivere und natürlichere Art der Interaktion dar als herkömmliche User Interfaces. Diese Eigenschaft unterstreicht im Actionspiel-Modus die Schnelligkeit des Spiels. Im Strategiespiel-Modus liegt der Schwerpunkt jedoch eher darauf, einen guten Überblick über die Positionen der Spielfiguren zu haben, um die nächsten Spielzüge planen zu können. Dieser Überblick ist beim Tabletop Game von vorn herein vorhanden, da der Spieler das gesamte Spielfeld sehen kann. Er muß nicht, wie bei herkömmlichen Computerspielen, den Blickpunkt der Kamera mit Hilfe eines Eingabegeräts festlegen.

Heutige Computerspiele werben oft mit dem Schlagwort „3D-Grafik“. Es handelt sich dabei jedoch genaugenommen nicht um 3D-Grafik, da die Darstellung dieser Grafik auf einem zweidimensionalen Ausgabegerät – einem Bildschirm – erfolgt. Man spricht in diesem Zusammenhang von „2 ½ D-Grafik“, da die dritte Dimension auf einer zweidimensionalen Ebene simuliert wird. Es ist jedoch anzunehmen, daß in Zukunft auch echte 3D-Grafik den Computerspielermarkt bereichern wird. Es gibt bereits seit einiger Zeit Versuche in diese Richtung. Bis jetzt konnte sich jedoch keines dieser Systeme am Markt behaupten. Die Hauptgründe liegen dabei vor allem bei den hohen Kosten und der im Vergleich zu Bildschirmdarstellungen schlechten Qualität der Produkte. Dabei handelt es sich jedoch um technische Probleme, die gelöst werden können.

Ziel des Tabletop Game Interface ist es, beim Spieler den Eindruck zu erwecken, mit einem realen Spiel zu spielen. Dabei werden die Vorteile des Computerspiels mit dem wesentlichen Vorteil des Brettspiels vereint. Computerspiele beeindrucken durch ansprechende Animationen, die bei einem herkömmlichen Brettspiel nicht vorhanden sind. Brettspiele haben jedoch gegenüber Computerspielen den großen Vorteil, daß die Figuren vom Spieler direkt bewegt werden können. Das Tabletop Game Interface bietet Animationen und direkte Steuerung der Spielfiguren. Darüber hinaus hat es gegenüber Brettspielen den entscheidenden Vorteil, daß durch die Augmented Reality auch die dritte Dimension in das Spielgeschehen miteinbezogen werden kann. So können beim Tabletop Game Ballons, die sich oberhalb des Spielfelds befinden, verwendet werden.

Für die Art dieser direkten Steuerung stehen beim Tabletop Game Interface jedoch – im Gegensatz zu einem realen Spiel – mehrere Möglichkeiten zur Verfügung. Die virtuellen Objekte können entweder auf der Position des jeweiligen Markers dargestellt werden oder dem Marker folgen (siehe Kapitel 3.2.4). Die zweite Variante erhöht den Realismus der Szene, da sie verhindert, daß sich langsame Objekte schnell bewegen. Sie widerspricht jedoch den Eigenschaften eines realen Spiels, da Spielfiguren im allgemeinen klein und leicht sind und daher auch schnell bewegt werden können. Spielfiguren sind oft stark verkleinerte Abbildungen realer Objekte. Auch in dem Fall, daß die realen Objekte übermäßig groß und schwer sind, können die Spielfiguren ohne große Anstrengungen bewegt werden. Bezieht man diesen Aspekt in die Beurteilung der verschiedenen Steuerungsformen mit ein, so ist die erste Art der Steuerung für die Simulation eines

Brettspiels zu bevorzugen. Möchte man jedoch das Tabletop Game Interface für einen anderen Zweck, z.B. die Simulation realer Bedingungen, einsetzen, so kann, je nach Bedarf, auch die zweite Steuerungsform gewählt werden.

Die Anzahl der Spielfiguren kann beim Tabletop Game beliebig variiert werden. Es hat sich jedoch gezeigt, daß eine große Anzahl an Spielfiguren eher hinderlich ist, da sich die Marker dann gegenseitig im Weg liegen. Auf der Tischplatte haben theoretisch sehr viele Marker platz, sie können dann jedoch nicht mehr beliebig bewegt werden. Um eine sinnvolle Obergrenze für die Anzahl der Marker zu finden, müssen die beiden Spieltypen getrennt betrachtet werden. Beim Actionspiel-Modus sind die Bewegungsmöglichkeiten der einzelnen Spielfiguren nicht eingeschränkt, d.h. ein Spieler kann z.B. sein Katapult an jede beliebige Stelle bewegen. Aus diesem Grund verbessert eine größere Anzahl an Katapulten nicht die Größe der Fläche, auf die der Spieler Einfluß nehmen kann. Außerdem ist es für das Actionspiel charakteristisch, daß alle Spielfiguren gleichzeitig gesteuert werden können. Ein Spieler kann jedoch höchstens zwei Spielfiguren mit je einer Hand gleichzeitig bewegen. Jede weitere Spielfigur würde ungenutzt bleiben. Daher sind zwei Marker pro Spieler im Actionspiel-Modus absolut ausreichend.

Im Strategiespiel-Modus können die Spielfiguren jedoch nicht beliebig bewegt werden (siehe Kapitel 5.3.2). Es macht daher durchaus Sinn, mehrere Katapulte über das Spielfeld zu verteilen, um bei einem Spielzug möglichst viele Ballons zu treffen. Da die Spielfiguren nicht gleichzeitig bewegt werden müssen, können während eines Spielzugs auch mehr als zwei Marker positioniert werden. Die Zahl der Marker ist in diesem Modus also nur durch die Größe des Tisches beschränkt. Um eine uneingeschränkte Bewegungsmöglichkeit zu garantieren, sollten jedoch höchstens drei bis vier Marker pro Spieler verwendet werden. Dazu kommt noch der Marker, der zum Wechseln des Spielers verwendet wird. Dieser befindet sich jedoch am Rand des Spielfelds und stört die Bewegungen der anderen Marker daher kaum.

Um den Schwierigkeitsgrad des Spiels zu erhöhen, kann ein Terrain eingeblendet werden. Der Zielvorgang wird erschwert, da nun zusätzlich zu der unterschiedlichen Höhe der Ballons auch die Höhe des Katapults nicht mehr konstant ist. Das Terrain wurde aber nicht nur aus spieltechnischen Gründen implementiert. Es dient vor allem der Demonstration der

Möglichkeiten eines TAR-Systems. Bei einem realen Spiel ist die Einbeziehung der dritten Dimension in das Spielgeschehen nur bedingt möglich. Im Gegensatz dazu stellt das bei einem Computerspiel kein Problem dar. Der Spieler sieht aber nur ein zweidimensionales Abbild der Szene auf dem Bildschirm. Ein TAR-System bietet dem Spieler jedoch echte 3D-Grafik und eine intuitive Interaktionsform.

Die Visualisierung der Windströme hilft dem Spieler bei der Positionierung der Windmühlen. Es werden sowohl die Windrichtung als auch die Größe des beeinflussten Raumes gekennzeichnet. Es können auch andere Visualisierungs-Methoden eingesetzt werden, um dem Benutzer den Umgang mit dem System zu erleichtern. Im Fall des Tabletop Games könnten z.B. die Flugbahnen der Kugeln angezeigt werden. Diese Funktion wurde nicht implementiert, da dadurch das eigentliche Ziel des Spiels verloren ginge. Es wäre jedoch denkbar, einen Trainings-Level anzubieten. In diesem Trainings-Level könnte die Flugbahn angezeigt werden, um dem Spieler am Anfang des Spiels das Zielen zu erleichtern.

## **7. Zusammenfassung**

Im Rahmen dieser Diplomarbeit wurde ein Tangible User Interface für ein bestehendes Augmented Reality System entwickelt und seine Eignung für Computerspiele untersucht. Zu diesem Zweck wurde zusätzlich ein Spiel – das Tabletop Game – entwickelt. Die Steuerung der Spielfiguren erfolgt durch Marker, deren Position und Orientierung mit Hilfe einer Webcam und einer speziellen Tracking-Software ermittelt werden. Um zu vermeiden, daß sich Personen oder Gegenstände zwischen den Markern und der Webcam befinden und dadurch den Tracking-Vorgang behindern, wurde ein Glastisch konstruiert. Die Marker liegen dabei so auf dem Tisch, daß sie von der Webcam, die unterhalb des Tisches montiert wurde, ungehindert verfolgt werden können. Die virtuellen Objekte werden auf der Oberseite des Tisches an der Stelle der entsprechenden Marker dargestellt. Der Benutzer sieht diese durch ein durchsichtiges HMD.

Beim Tabletop Game handelt es sich um ein Spiel für zwei Spieler, die um den Glastisch herum stehen oder sitzen. Jeder Spieler verfügt über ein Katapult, eine Windmühle und mehrere Ballons. Ziel des Spieles ist es, die gegnerischen Ballons mit Hilfe des Katapults abzuschießen. Die eigenen Ballons können durch die Windmühlen aus der Schußlinie des gegnerischen Katapults bewegt werden. Die Ballons fliegen automatisch mit dem Wind, der von diesen Windmühlen erzeugt wird. Sieger ist, wer zuerst alle gegnerischen Ballons abschießen konnte.

Um zu ermitteln, welche Spielform sich am ehesten für eine intuitive Interaktion mit einem AR-System eignet, wurden verschiedenen Versionen des Spiels entwickelt. Diese unterscheiden sich vor allem durch den Typ des Spiels und die Steuerungsmöglichkeiten der Spielfiguren. Das Tangible User Interface ist für jede der getesteten Spielformen – Actionspiel und Strategiespiel – gleichermaßen gut geeignet. Es ermöglicht einerseits eine schnelle Interaktion mit den virtuellen Objekten, was besonders beim Actionspiel von Vorteil ist. Andererseits bietet es durch die dreidimensionale Darstellung einen guten Überblick über das Spielfeld, was das Zurechtlegen einer Strategie begünstigt.

Es wurden weiters zwei verschiedene Arten der Steuerung implementiert. Entweder bewirkt das Verschieben eines Markers eine sofortige Positions- und Richtungsänderung der

Spielfigur, oder sie bewegt sich mit konstanter Geschwindigkeit in die Richtung, in die der Marker geschoben wurde. Es hat sich gezeigt, daß die erste Variante für die Entwicklung eines Computerspiels zu bevorzugen ist, da Spielfiguren realer Spiele im allgemeinen klein und leicht sind und daher auch schnell bewegt werden können.

Die Verwendung von Tangible Augmented Reality für Computerspiele vereinigt die Vorteile von Brettspielen mit den Vorteilen von Computerspielen, denn einerseits können die Spielfiguren direkt und intuitiv gesteuert werden und andererseits können – wie bei Computerspielen üblich – ansprechende Animationen und Grafiken in das Spiel eingebracht werden.

**8. Referenzen**

[Baudel, 1993] T. Baudel, M. Beaudouin-Lafon, „CHARADE: Remote Control of Objects using Free-Hand Gestures“, 1993

[Billinghurst, 2000] M. Billinghurst, I. Poupyrev, H. Kato, R. May, „Mixing Realities in Shared Space: An Augmented Reality Interface for Collaborative Computing“, [www.hitl.washington.edu/magicbook/papers/icme2000.pdf](http://www.hitl.washington.edu/magicbook/papers/icme2000.pdf), 2000

[Billinghurst, 2001] M. Billinghurst, H. Kato, I. Poupyrev, „Collaboration with Tangible Augmented Reality Interfaces“, <http://www.hitl.washington.edu/publications/r-2002-61/r-2002-61.pdf>, 2001

[Boger, 1999] M. Boger, „Java in verteilten Systemen – Nebenläufigkeit, Verteilung, Persistenz“, dpunkt.verlag, 1999

[Bormann, 1994] S. Bormann, „Virtuelle Realität – Genese und Evaluation“, Addison-Wesley, 1994

[Bowden, 1998] R. Bowden, T. A. Mitchell, M. Sarhadi, „Reconstructing 3D Pose and Motion from a Single Camera View“, <http://www.ee.surrey.ac.uk/Personal/R.Bowden/publications/bmvc98/p114.pdf>, 1998

[Bowden, 2000] R. Bowden, T. A. Mitchell, M. Sarhadi, „Non-linear statistical models for the 3D reconstruction of human pose and motion from monocular image sequences“, <http://www.ee.surrey.ac.uk/Personal/R.Bowden/publications/ivc/bowdenivc2000.pdf>, 2000

[Bowman, 1997] D. Bowman, L. Hodges, „An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments“, Proceedings of the Symposium on Interactive 3D Graphics 1997, pp. 35-38, 1997

[Bowman, 1998] D. Bowman, J. Wineman, L. Hodges, D. Allison, „Designing Animal Habitats Within an Immersive VE”, IEEE Computer Graphics & Applications, September/October 1998, vol. 18, no. 5, pp. 9-13, 1998

[Bowman, 2001a] D. A. Bowman, V. Q. Ly, J. M. Campbell, „Pinch Keyboard: Natural Text Input for Immersive Virtual Environments”,  
[http://www.cs.vt.edu/TR/pinch\\_keyboard.pdf](http://www.cs.vt.edu/TR/pinch_keyboard.pdf), 2001

[Bowman, 2001b] D. A. Bowman, „Advanced Topics in 3D User Interface Design”, SIGGRAPH, 2001

[Bryson, 1991], S. Bryson, C. Levit, „The Virtual Windtunnel: An Environment for the Exploration of Three-Dimensional Unsteady Flows”, Proceedings of Visualization 1991, pp. 17-24, 1991

[Conner, 1992] D. B. Conner, S. S. Snibbe, K. P. Herndon, D. C. Robbins, R. C. Zeleznik, A. van Dam, „Three-Dimensional Widgets”, in Proceedings of Symposium on Interactive 3D Graphics 1992, pp 183 – 188, 1992

[Cruz-Neira, 1993] C. Cruz-Neira, D. Sandin, T. DeFanti, „Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE”, in Proceedings of SIGGRAPH'93, pp 135-142, 1993

[Dyer, 1995] S. Dyer, J. Martin, J. Zulauf, „Motion Capture White Paper”,  
[http://mirrors.siliconbunny.com/reality.sgi.com/jam\\_sb/mocap/MoCapWP\\_v2.0.html](http://mirrors.siliconbunny.com/reality.sgi.com/jam_sb/mocap/MoCapWP_v2.0.html), 1995

[Forsberg, 1996] A. Forsberg, K. Herndon, R. Zeleznik, „Aperture Based Selection for Immersive Virtual Environments”, Proceedings of the ACM Symposium on User Interface Software and Technology, 1996

[Fuhrmann, 1999] A. Fuhrmann, D. Schmalstieg, „Concept and Implementation of a Collaborative Workspace for Augmented Reality”, <ftp://ftp.cg.tuwien.ac.at/pub/TR/99/TR-186-2-99-04Paper.pdf>, 1999

---

[Fuhrmann, 2001] A. L. Fuhrmann, J. Přikryl, R. F. Tobler, „Interactive Content for Presentations in Virtual Reality”, [http://www.vrvis.at/br1/vrssi/presentation\\_paper.pdf](http://www.vrvis.at/br1/vrssi/presentation_paper.pdf), 2001

[Furniss, 1999] M. Furniss, „Motion Capture”, presented at the Media in Transition Conference at MIT on October 8, 1999

[Isdale, 1993] J. Isdale, „What Is Virtual Reality? A Homebrew Introduction and Information Resource List“, <http://www.cse.dmu.ac.uk/~cph/VR/whatisvr-orig.html>, Version 2.1, 8. Oktober 1993

[Ishii, 1997] H. Ishii, B. Ullmer, „Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms”, in Proceedings of Conference on Human Factors in Computing Systems (CHI '97), ACM Press, pp. 234-241, 1997

[Jacobson, 1994] L. Jacobson, „Garage Virtual Reality – The Affordable Way to Explore Virtual Worlds”, Sams Publishing, 1994

[Kato, 2000] H. Kato, M. Billinghurst, I. Poupyrev, „ARToolKit”, [http://www.hitl.washington.edu/research/shared\\_space/download/](http://www.hitl.washington.edu/research/shared_space/download/), Version 2.33, November 2000

[Kaufmann, 2002] H. Kaufmann, D. Schmalstieg, „Mathematics and Geometry Education with Collaborative Augmented Reality”, [http://www.ims.tuwien.ac.at/media/documents/publications/Construct3D\\_SIGGRAPH\\_Final.pdf](http://www.ims.tuwien.ac.at/media/documents/publications/Construct3D_SIGGRAPH_Final.pdf), 2002

[Kolozs, 1996] J. Kolozs, „Position Trackers for Virtual Reality”, <http://www.cs.utah.edu/classes/cs6360/Kolozs/trackers.html>, 1996

[LaViola, 1999], J. LaViola, R. Zeleznik, „Flex and Pinch: A Case Study of Whole Hand Input Design for Virtual Environment Interaction“, <http://ftp.cs.brown.edu/people/jjl/pubs.html>, October 1999

---

[Magenat-Thalmann, 1998] N. Magenat-Thalmann, D. Thalmann „Modelling and Motion Capture Techniques for Virtual Environments“, Springer Verlag, 1998

[Martz, 1996] P. Martz, „Generating Random Fractal Terrain“, <http://gameprogrammer.com/fractal.html>, 1996

[Menache, 2000] A. Menache, "Understanding Motion Capture for Computer Animation and VideoGames", Morgan Kaufmann, 2000

[Mine, 1995] M. Mine, „Virtual Environment Interaction Techniques“ [http://www.cs.unc.edu/~mine/mine\\_publications.html](http://www.cs.unc.edu/~mine/mine_publications.html), 1995

[Mine, 1997] M. Mine, F. P. Brooks Jr., C. Sequin, „Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction“, Proceedings of SIGGRAPH 97, vol. 31, pp. 19-26, 1997

[Pape, 1996] D. Pape, C. Cruz-Neira, M. Czernuszenko, „CAVE User's Guide“, <http://evlweb.eecs.uic.edu/pape/CAVE/prog/CAVEGuide.2.5.6.html>, Electronic Visualization Laboratory, University of Illinois at Chicago, 23. Juni 1996

[Pausch, 1995] R. Pausch, T. Burnette, D. Brockway, M. Weiblen, „Navigation and Locomotion in Virtual Worlds via Flight into Hand-Held Miniatures“, Proceedings of ACM SIGGRAPH, pp. 399-400, 1995

[Pierce, 1997] J. Pierce, A. Forsberg, M. Conway, S. Hong, R. Zeleznik, M. Mine, „Image Plane Interaction Techniques in 3D Immersive Environments“, Proceedings of the ACM Symposium on Interactive 3D Graphics, pp. 39-44, 1997

[Poupyrev, 1996] I. Poupyrev, M. Billinghurst, S. Weghorst, T. Ichikawa, „Go-Go interaction technique: Non-linear mapping for direct manipulation in VR“, Proceedings of UIST'96, pp. 79-80, 1996

---

[Pryor, 1998] H. L. Pryor, T. A. Furness III, E. Viirre „The Virtual Retinal Display: A New Display Technology Using Scanned Laser Light”, <http://www.hitl.washington.edu/research/vrd/papers.html>, 1998

[Reitmayr, 2001a] G. Reitmayr, D. Schmalstieg, „Mobile Collaborative Augmented Reality”, <http://www.ims.tuwien.ac.at/media/documents/publications/mobileISAR2001.pdf>, 2001

[Reitmayr, 2001b] G. Reitmayr, D. Schmalstieg, „An Open Software Architecture for Virtual Reality Interaction“, <http://www.ims.tuwien.ac.at/media/documents/publications/opentrackervrst2001final.pdf>, 2001

[Roehl, 1994] B. Roehl, „Playing God – Creating Virtual Worlds With REND386”, Waite Group Press, 1994

[Schmalstieg, 2000] D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavári, L. M. Encarnação, M. Gervautz, W. Purgathofer, „The Studierstube Augmented Reality Project”, <ftp://ftp.cg.tuwien.ac.at/pub/TR/00/TR-186-2-00-22Paper.pdf>, 2000

[Schmalstieg, 2002] D. Schmalstieg, G. Reitmayr, G. Hesina, „Distributed Applications for Collaborative Three-Dimensional Workspaces”, [http://www.ims.tuwien.ac.at/media/documents/publications/distributed\\_applications.pdf](http://www.ims.tuwien.ac.at/media/documents/publications/distributed_applications.pdf), 2002

[Stoakley, 1995] R. Stoakley, M. Conway, R. Pausch, „Virtual Reality on a WIM: Interactive Worlds in Miniature”, Proceedings of CHI: Human Factors in Computing Systems, pp. 265-272, 1995

[Szalavári, 1998a] Z. Szalavári, D. Schmalstieg, A. Fuhrmann, M. Gervautz, „Studierstube – An Environment for Collaboration in Augmented Reality“, <http://www.cg.tuwien.ac.at/research/vr/studierstube/jvrs-paper.pdf>, 1998

[Szalavári, 1998b] Z. Szalavári, D. Schmalstieg, „The Personal Interaction Panel – a Two-Handed Interface for Augmented Reality”, <http://www.cg.tuwien.ac.at/research/vr/pip/>, 1998

[Vince, 1995] J. Vince, „Virtual Reality Systems“, Addison-Wesley, 1995

[Ware, 1993] C. Ware, K. Arthur, K. S. Booth., „Fish Tank Virtual Reality”, <http://iihm.imag.fr/fberard/ens/ensimag/ensi3srvra/download/biblio/ware93.pdf>, April 1993

[Ware, 1996] C. Ware, „Multimedia Output Devices and Techniques”, ACM Comp. Surv. 28, 1 (1996), pp. 181-183, 1996

[Woods, 2001] A. J. Woods, „Optimal Usage of LCD Projectors for Polarised Stereoscopic Projection”, <http://www.curtin.edu.au/curtin/centre/cmst/publicat/2001-01.pdf>, 2001

**9. Abbildungsverzeichnis**

Abbildung 2.3, Seite 13, <http://www.metamotion.com/gypsy/gypsy-motion-capture-system-mocap.htm>

Abbildung 2.4, Seite 15, <http://www.big-i.com/siggraph2001/images/mocapgirl.jpg>

Abbildung 2.5, Seite 20, <http://www.gopostal.com/mocap/mocap3.htm>

Abbildung 2.6, Seite 22, [Magenat-Thalmann, 1998]

Abbildung 2.7, Seite 23, [Magenat-Thalmann, 1998]

Abbildung 2.8, Seite 24, [Magenat-Thalmann, 1998]

Abbildung 2.9, Seite 25, [Bowden, 1998]

Abbildung 2.10, Seite 27, <http://www.vrealities.com/5dtglove.html>

Abbildung 2.11, Seite 29, <http://www.fakespacelabs.com/products/pinch.html>

Abbildung 2.12, Seite 30, [Fuhrmann, 2001]

Abbildung 2.13, Seite 31, [Fuhrmann, 2001]

Abbildung 2.16, Seite 36 <http://www.5dt.com/products/phmd.html>

Abbildung 2.17, Seite 38, <http://www.fakespacelabs.com/products/boom3c.html>

Abbildung 2.18, Seite 40, <http://www.indiana.edu/~rcapub/v21n2/p28.html>

Abbildung 2.19, Seite 43, [Bowman, 2001b]

Abbildung 2.28, Seite 50, [Stoakley, 1995]

Abbildung 2.30, Seite 52, <http://www.cs.unc.edu/~mine/chimp.html>

Abbildung 2.31, Seite 53, [Fuhrmann, 1999]

Abbildung 2.32, Seite 55, „Tangible Interfaces for Remote Collaboration and Communication”, S. Brave, H. Ishii, A. Dahley

Abbildung 4.1, Seite 63, [Fuhrmann, 1999]

Abbildung 4.2, Seite 64, [Kaufmann, 2002]

Abbildung 4.3, Seite 65, [Schmalstieg, 2000]

Abbildung 4.4, Seite 66, [Reitmayr, 2001a]

Abbildung 4.5, Seite 67, <http://www.ascension-tech.com/products/flockofbirds/>

Abbildung 4.6, Seite 68, <http://www.isense.com/products/pro/>

Abbildung 4.7, Seite 69, [Reitmayr, 2001a]

Abbildung 4.8, Seite 70, „Occlusion in Collaborative Augmented Environments“, A. Fuhrmann, G. Hesina, F. Faure, M. Gervautz

Abbildung 4.9, Seite 71, „Dynamische Geometrie in Virtual Reality - Construct3D für den Geometrie- und Mathematikunterricht“, H. Kaufmann

Abbildung 4.10, Seite 72, [Schmalstieg, 2000]

Abbildung 4.12, Seite 74, [Kato, 2000]

Abbildung 4.14, Seite 77, [Reitmayr, 2001b]