

Westfälische Wilhelms-Universität Münster  
Institut für Informatik  
Einsteinstraße 62  
48149 Münster

WS 2004/05

Arbeit zum Projektseminar  
„Interaktive 3D-Stadtplanung“

# **3D-Interaktion**

Johannes Schwall

04. Oktober 2004

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung: Interaktion in drei Dimensionen</b>	<b>2</b>
<b>2</b>	<b>Interaktionsmetaphern</b>	<b>8</b>
2.1	Exozentrische Metaphern . . . . .	9
2.2	Egozentrische Metaphern . . . . .	10
2.3	3D-Metaphern . . . . .	18
2.4	Weitere Metaphern . . . . .	18
<b>3</b>	<b>Interaktionstechniken</b>	<b>20</b>
3.1	Selektion . . . . .	20
3.2	Manipulation . . . . .	22
3.3	Menü-Interaktion und System-Kontrolle . . . . .	27
3.4	Objekt-Interaktion . . . . .	30
3.5	Navigation . . . . .	31
<b>4</b>	<b>Interaktionsgeräte</b>	<b>33</b>
4.1	Eingabegeräte . . . . .	34
4.2	Ausgabegeräte . . . . .	37
<b>5</b>	<b>3D-Stadtplanung</b>	<b>41</b>
	<b>Literaturverzeichnis</b>	<b>44</b>

# 1 Einführung: Interaktion in drei Dimensionen

„Where did you put it?“  
„Put what?“  
„You know?“  
„Where do you think?“  
„Oh.“

---

(Nicholas Negroponte,  
Director of the MIT Media Lab,  
stating his ideal model of  
human-computer interaction)

## Was ist Interaktion?

Interaktion ist „aufeinander bezogenes Handeln zweier oder mehrerer Personen“ bzw. die „Wechselbeziehung zwischen Handlungspartnern“ und darüber hinaus ist „sprachliche Kommunikation [...] die wichtigste Form menschlicher Interaktion“<sup>1</sup>. Es geht also um eine Wechselbeziehung zwischen zwei Kommunikationspartnern, im vorliegenden Fall zwischen dem Benutzer eines Computersystems und dem Computer selbst, wobei unterschiedliche Wege zur Interaktion bestehen. Diese Kommunikation wird als *Human-Computer Interaction (HCI)* bezeichnet und ist zunächst einmal als universell aufzufassen, umfasst somit jegliche Kommunikation zwischen Mensch und Maschine, die über das sogenannte Interface – die aus Software und Hardware bestehende Schicht zwischen den von Tufte als „zwei starke Prozessoren“<sup>2</sup> bezeichneten Kommunikationspartnern – abgewickelt wird. Dieses Interface ist dabei von vornherein schmalbandig und eingeschränkt; „Flaschenhals“ sind hier auf der einen Seite die menschlichen Sinne, auf der anderen Seite die immer noch begrenzte Leistungsfähigkeit aktueller Computer. Ein grundsätzliches Ziel der HCI-Forschung ist daher die Verbesserung der Kommunikation über dieses Interface. Interaktionstechniken bieten dabei einen nützlichen Schwerpunkt, da sie zwar spezifisch, aber nicht an eine einzelne Anwendung gebunden sind. Als Interaktionstechnik wird dabei die Benutzung eines physikali-

---

<sup>1</sup>Deutsches Universalwörterbuch, (Mannheim, Wien, Zürich: Dudenverlag, 1989), S. 773, Stichwort „Interaktion“.

<sup>2</sup>E. R. Tufte. *Visual Design of the User Interface*, (IBM Corporation, Armonk, N. Y., 1989).

schen Ein- oder Ausgabegerätes bezeichnet, mit der eine generelle Aufgabe im Mensch-Computer-Dialog ausgeführt werden kann.<sup>3</sup>

Musste Konrad Zuse noch mühsam die Register seiner Rechenmaschinen bestücken, kamen die ersten Homecomputer in den 80-er Jahren bereits mit Tastaturen in die Geschäfte. Bald folgten grafische Benutzeroberflächen und die Bedienung über Mäuse, Joysticks und weitere Eingabegeräte. Der derzeit erfolgende Schritt in die dritte Dimension bedingt die Umnutzung der bestehenden und die Entwicklung neuer Eingabegeräte (*Human-Machine Interfaces, HMI*) um die erweiterten und neuen Möglichkeiten nutzen zu können. Darüber hinaus soll der Bedienkomfort gesteigert werden, was bereits früh zur Entwicklung von angepassten Metaphern und Interaktionstechniken führte. Bewegt sich der Benutzer in einer sogenannten *virtuellen Umgebung (Virtual Environment, VE)*, erwartet er vor allem Performanz (Effizienz, Präzision, Produktivität), Benutzerfreundlichkeit (Leichtigkeit der Benutzung und des Lernens, Benutzerkomfort) und Nützlichkeit (Interaktion hilft, die Design-Ziele des Systems zu realisieren; eine relative Transparenz des Interfaces lässt den Benutzer sich auf die vor ihm liegenden Aufgaben konzentrieren), die somit als die besonderen Ziele des Interaktionsdesigns anzusehen sind.

HMI

Virtual Environment

Das Idealbild eines VE wird definiert als künstliche Echtzeit-Umgebung und sie soll dabei das Gefühl vollständiger oder teilweiser Immersion<sup>4</sup> bieten, den Benutzer im Raum also vollständig zu umgeben scheinen bzw. Teil der Umgebung zu sein (z. B. Desktop-VRWorkbench, siehe Abb. 5.1, S. 43).

### Warum 3D-Interaktion?

Bereits ohne technisches Hintergrundwissen fallen dem Interessierten viele Beispiele ein, in denen 3D-Technik bereits erfolgreich eingesetzt wird. Darunter finden sich so komplexe Anwendungsgebiete wie z. B. Architektur, Ausbildung/Schulung und Medizin. Der Hauptantrieb für 3D-Entwicklung kommt neben diesen vor allem aber aus der Unterhaltungsindustrie, die Virtualität und 3-Dimensionalität seit langem erfolgreich in Computerspielen einsetzt und somit die Entwicklung neuer Soft- und Hardwaretechniken voran treibt. Weitere Anwendungsgebiete finden sich in der Simulation (z. B. Fahrertraining), der Visualisierung von Informationen, dem Design und Prototyping sowie dem Bereich Kollaboration/Kommunikation.

<sup>3</sup>Robert J. K. Jacob. 'New Human-Computer Interaction Techniques', *Human-Machine Communication for Educational Systems Design*, (1994), S. 131–138.

<sup>4</sup>spätlat. immersio = Eintauchung; Immersion bezeichnet das möglichst vollständige Eintauchen des Benutzers in die virtuelle Welt. Um dies zu erreichen, werden immer neue Techniken entwickelt, die auf die fünf menschlichen Sinne (sehen, hören, riechen, schmecken, fühlen) gerichtet sind und versuchen, diesen möglichst umfassend die künstliche Welt vorzutauschen. Vollständige Immersion wird erreicht mit Systemen, die den Benutzer vollkommen von der Außenwelt abschließen (z. B. CAVE, HMD, siehe S. 39), teilweise Immersion bei Systemen, die den Benutzer nicht vollkommen von der Umgebung abschirmen.

Das Design von 3D-Anwendungen ist dabei stark auf ihre Nützlichkeit ausgerichtet. Durch Immersion und die zusätzlichen Möglichkeiten der Visualisierung setzen sich 3D- von herkömmlichen 2D-Anwendungen ab. Bei der Implementierung ist dabei besonders darauf zu achten, die Anwendung auf die natürlichen Fähigkeiten des Menschen abzustimmen. Das Problem bisheriger Ansätze ist zumeist, dass sie über kaum komplexe Interaktionsmöglichkeiten verfügen oder, sofern der Schwerpunkt entsprechend gesetzt ist, ernsthafte Probleme mit der Benutzbarkeit haben. So zeigt sich erneut: Technology alone ist not the solution!

So entwickelte sich im Interaktionsdesign die Wahl zwischen zwei Wegen. Auf der einen Seite steht der wissenschaftliche Ansatz, dessen Designentscheidungen sich auf die formale Charakterisierung von Benutzern, Aufgaben und Umgebung, quantitative Auswertungsergebnisse und Performanz-Ansprüche stützen. Dem gegenüber findet sich der eher künstlerische Ansatz, der sich auf die Intuition von Benutzern, Aufgaben und Umgebungen stützend, Heuristiken, Metaphern und „gesunden Menschenverstandes“ bedient und somit eher auf die Ästhetik bedacht ist. Oft wird dabei die Adaption bzw. Inversion vorhandener Interfaces (zumeist aus 2-dimensionalen Anwendungen) genutzt.

Ebenso gibt es Grabenkämpfe zwischen denjenigen, die auf ein einziges Interaktionsgerät als Standard zur Benutzung in allen virtuellen Umgebungen setzen, gegenüber denen, die für jede Anwendung und jede Umgebung ein neues Gerät zum Einsatz bringen wollen. Wie so oft, ist die Realität vermutlich zwischen diesen beiden Extremen zu suchen. Doch ergeben sich aus dieser Diskussion Fragen, die in dieser Arbeit zumindest angesprochen, wenn auch nicht endgültig geklärt werden sollen: Welche Faktoren und Interaktionstechniken bieten sich besonders an? Welche Interaktionsgeräte lassen sich in einer bestimmten Umgebung für eine bestimmte Aufgabe besonders gut einsetzen? Ebenso soll auf einige Schwierigkeiten der 3D-Interaktion eingegangen werden: räumliche Eingabe, (fehlende) Beschränkungen, fehlende Standards, Werkzeuge und Genauigkeit, Benutzermüdigkeit und Wahrnehmung.

## 2D- vs. 3D-Interaktion in VEs

Beim Übergang von zwei zu drei Dimensionen kommen mehrere neue Regeln ins Spiel, die es zunächst zu analysieren gilt und die im Folgenden dargestellt kurz werden sollen. Ein grundlegender Begriff ist dabei der der *Freiheitsgrade* (*Degrees of Freedom, DoF*). Freiheitsgrade beschreiben die Möglichkeiten der Bewegung im Raum und arbeiten dabei mit einem auf Achsen basierenden Koordinatensystem. Im echt zweidimensionalen Raum ist die Bewegung auf lediglich zwei Achsen (x und y) möglich, wohin gegen der echt dreidimensionale Raum eine weitere (z-Achse) hinzufügt, die im zweidimensionalen Raum allerdings auch visualisiert werden kann. In der 3D-Welt stehen dabei sechs mögliche Freiheitsgrade zur Verfügung, die sich aus drei Trans-

Degrees of  
Freedom

lationsrichtungen und jeweiliger Drehung um diese Achsen erklären (siehe Abb. 1.1).

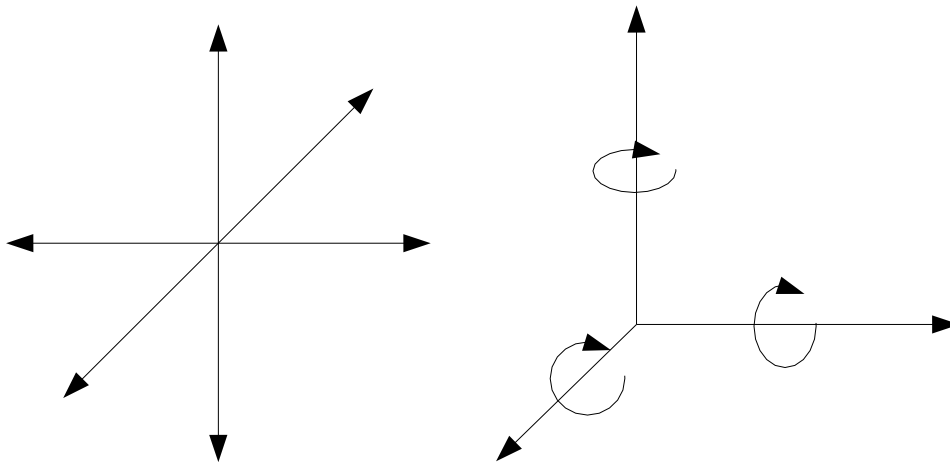


Abb. 1.1: Sechs Freiheitsgrade: drei Translationsrichtungen (links) und drei Rotationen um die Translationsachsen (rechts)

Interaktion kann sowohl *direkt* als auch *indirekt* sein. Der wesentliche Unterschied liegt darin, ob der Benutzer direkt mit dem Objekt interagiert, oder ob eine Zwischenschicht eingeführt wird. Demnach handelt sich das Drehen eines Objektes, das mit einem Datenhandschuh „gefasst“ und mit der Hand gedreht wird, um direkte Interaktion. Findet diese Interaktion über ein Menü oder einen Slider statt, wird von indirekter Interaktion gesprochen. Dies gilt unabhängig von der Anzahl der Dimensionen.

direkt vs.  
indirekt

Ebenfalls unabhängig von der Dimension ist die Unterscheidung zwischen *diskreter* und *kontinuierlicher* Interaktion. Die Frage, ob ein Objekt jede Position in der virtuellen Umgebung einnehmen kann, ist gerade bei der Stadtplanung als äußerst wichtig zu betrachten. Muss ein Benutzer ein Haus-Objekt erst (kontinuierlich) über die Straße schieben um es auf der anderen Seite zu positionieren? Oder kommt ihm das System entgegen und „weiß“, dass sich dieses Objekt nur auf bestimmten Flächen befinden darf, setzt es also bei einer Bewegung in die entsprechende Richtung in einem (diskreten) Schritt auf die nächstmögliche Position? Letzterer Fall wird als „snap-to-grid“ (Abb. 1.2) bezeichnet, weil die Sprungpositionen üblicherweise auf einem unsichtbaren Gitter angeordnet sind.

diskret vs.  
kontinuierlich

Eine Besonderheit bei der Interaktion mit drei Dimensionen ist die Entfernung zu einem Objekt. Bei der Arbeit mit zwei Dimensionen bleibt das Werkzeug (z. B. der Mauszeiger) immer in der gleichen Ebene mit allen Objekten und kann daher leicht zur Auswahl des jeweils angesteuerten Objektes verwendet werden. Bei Einführung einer dritten Dimension, und arbeiten beispielsweise mit einem Datenhandschuh, stellt sich die Frage, wie (*entfernte*) Objekte, die hinter der Projektionsfläche liegen, angesteuert werden können. Hier ist eine

lokal vs.  
remote

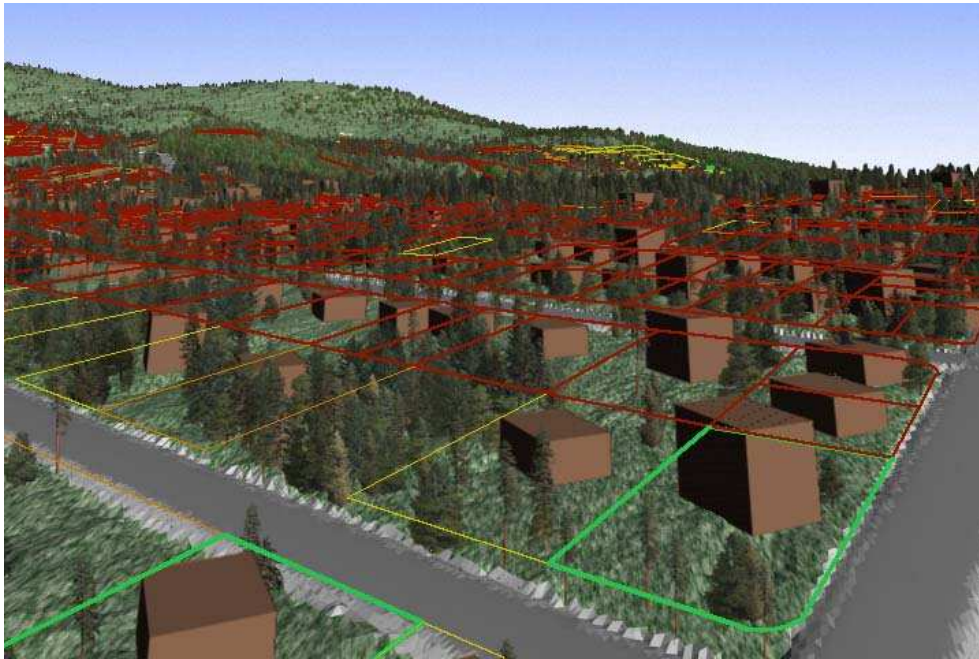


Abb. 1.2: Gitter über einer virtuellen Landschaft helfen beim Positionieren von Objekten, Quelle: [www.jamesreserve.edu](http://www.jamesreserve.edu)

andere Metapher einzusetzen als das möglicherweise für (*lokale*) Objekte vor dem Bildschirm verwendete virtuelle „Greifen“.<sup>5</sup>

Der auch als User Interaction Flow, User Flow oder Process Flow bezeichnete *Interaction Flow* verdeutlicht, wie ein Benutzer einen bestimmten Prozess innerhalb des Systems bedienen oder das gesamte System benutzen könnte. Ein Interaction Flow ist somit vergleichbar einem Use Case; er visualisiert die Reihenfolge von Interaktionen zwischen Benutzer und System. Die Darstellung kann in unterschiedlichen Formen erfolgen, z. B. als Flussdiagramm oder Ablaufplan, und beinhaltet nicht solche Details wie die Bezeichnung von Knöpfen, Farben, Positionierungen usw.

Interaction  
Flow

Virtuelle Umgebungen können heute realistische, immersive graphische Welten entstehen lassen und visualisieren. Dennoch ist die Interaktion mit diesen Welten noch immer nicht selbstverständlich und oft umständlich. In vielen Fällen fehlt das haptische und akustische *Feedback*, das dem Benutzer die richtigen Reaktionen auf seine Aktionen vermittelt. Haptisches Feedback wird zum Beispiel über die vor allem aus Computerspielen bekannten Force-Feedback-Joysticks oder -Lenkräder wiedergegeben. Es wird aber auch versucht, den Mangel durch diverse Techniken zu kompensieren. Wie oben beschrieben, wird versucht, möglichst viele Sinne des Benutzers anzusprechen. Ein Feedback, das für den einen Sinn bestimmt ist, kann also durch ein Feedback für einen anderen Sinn ersetzt werden oder dieses unterstützen. Kommen bei einer solchen Interaktion mehrere Interaktionsgeräte und -techniken

Feedback

<sup>5</sup>Weitere Informationen über Metaphern in den Kapiteln 2 und 3.

gleichzeitig zum Einsatz, wird von *multimodaler*, andernfalls von *unimodaler* Interaktion gesprochen. Ein häufig in Verbindung mit anderen zur multimodalen Interaktion eingesetztes Verfahren ist die Spracherkennung bzw. -ausgabe, da diese nicht mit anderen Verfahren um bestimmte Ein- oder Ausgabegeräte konkurriert, sondern einen anderen als den haptischen oder visuellen Sinn des Benutzers anspricht.

unimodale vs.  
multimodale  
Interaktion

## Affordances und Constraints

Trotz aller Möglichkeiten, die der Umgang mit drei Dimensionen im virtuellen Raum mit sich bringt, bestehen auf Seiten des Benutzers Einschränkungen, die durch die menschlichen Kommunikationswege, -organe bzw. -fähigkeiten bedingt sind. Anders herum beschränkt sich die Interaktion des Computers mit dem Menschen auf die verfügbaren Ein- bzw. Ausgabegeräte und die Methoden zur Interaktion, die in der jeweiligen Anwendung eingesetzt werden. Solange Neuro-, Bio- und Informationstechnologie nicht noch stärker miteinander verschmelzen, ergibt sich daraus das Erfordernis, weitere Geräte und Verfahren für den Dialog zwischen Mensch und Computer zu finden, vorhandene besser anzupassen und auf der anderen Seite die für die Kommunikation relevanten Charakteristiken des Menschen besser auszunutzen. *Affordances* und *Constraints* kommen dabei den Interaktionsverfahren zu Hilfe, indem sie den Benutzer „leiten“. Möchte ein Benutzer in einer virtuellen Umgebung einen Gegenstand auf einem anderen platzieren (z. B. ein Haus auf einer Ebene), ist es für ihn sehr schwierig, dieses Haus genau so zu positionieren, dass es sich nicht teilweise unterhalb der Ebene befindet (es also keine Schnittmenge gibt), es aber auch nicht darüber schwebt. Constraints in Form von künstlicher Schwerkraft (die das Schweben verhindert) und einer virtuellen Grenze (die die Festigkeit der Ebene repräsentiert) würden diese Probleme für den Benutzer lösen.

Auf der anderen Seite kommen Affordances zum Einsatz, wenn virtuelle Elemente mit Eigenschaften auszustatten sind. Möchte der Benutzer ein virtuelles Blatt Papier mit einer virtuellen Schere zerteilen, würde ein Constraint zunächst das Zerteilen des Blattes verhindern. In Verbindung mit der Schere wäre das Zerschneiden jedoch aufgrund der Affordance erlaubt. Die Einsatzmöglichkeiten für Affordances sind, ebenso wie für Constraints, nahezu unbeschränkt: ein Deckel einer Mülltonne ließe sich öffnen, ein Knopf an einer Ampel drücken, usw. Während Constraints die Umgebung einschränken – und damit vor allem Vereinfachen –, erweitern Affordances das Virtual Environment durch das Anbieten von speziellen Möglichkeiten.



## 2 Interaktionsmetaphern

Zunächst ist es wichtig, den Begriff der *Interaktionsmetapher* deutlich zu machen. Das Wort *Metapher* kommt aus dem Griechischen und bedeutet „Übertragung“; als sprachliches Ausdrucksmittel wird dabei das eigentlich gemeinte durch ein anderes Wort ersetzt, „das eine sachliche oder gedankliche Ähnlichkeit oder dieselbe Bildstruktur aufweist“.<sup>6</sup> Metaphern werden verwendet, um abstrakte Teilaspekte in einem komplexen Zusammenhang zu erklären und so eine Grundlage für das menschliche Verstehen darzustellen. Demnach wird eine Interaktionsmetapher genutzt, um eine Interaktionsform, die nur schwer dem menschlichen Verstehen nahe zu bringen ist, eindeutiger und leichter verständlich zu machen, indem fremde Gegenstände – im vorliegenden Fall 3-dimensionale, virtuelle Objekte und deren Handling – durch alltägliche, dem Benutzer bekannte Objekte und Interaktionsformen ersetzt werden: das Greifen eines Objektes im virtuellen Raum wird beispielsweise durch eine virtuelle Hand simuliert.

Bei der Diskussion von Benutzerinteraktion in und mit einer virtuellen Welt, gilt es darüber hinaus, besonders die Abgrenzung von Interaktionsmetaphern und Interaktionstechniken klar zu definieren. Während, wie beschrieben, die Interaktionsmetaphern zunächst allgemein gültig sind und ein beliebiges Objekt oder eine beliebige Tätigkeit des Benutzers visualisieren und verdeutlichen sollen, handelt es sich bei Interaktionstechniken um die Anwendung von bestimmten Verfahren auf virtuelle Objekte. Dabei kann sowohl das virtuelle Objekt, wie auch die Interaktion als Tätigkeit durch eine Metapher vertreten sein. Eine virtuelle Hand stellt somit eine Interaktionsmetapher dar, das Verändern eines Objektes im virtuellen Raum mithilfe dieser Hand ist jedoch eine Interaktionstechnik.

Abgrenzung zu  
Interaktions-  
techniken

Der näheren Erläuterung von ausgewählten Interaktionsmetaphern im Folgenden liegt zunächst die Unterscheidung von *exozentrischen*<sup>7</sup> und *egozentrischen*<sup>8</sup> Metaphern zugrunde, die auf unterschiedliche Darstellungsmöglichkeiten einer virtuellen Umgebung zurück zu führen ist: Je nach Darstellungsform, findet sich der Benutzer entweder innerhalb oder außerhalb der virtuel-

<sup>6</sup>Brockhaus. *Die Enzyklopädie*, (Leipzig, Mannheim: F. A. Brockhaus, 1998), S. 554 f., Stichwort „Metapher“.

<sup>7</sup>exozentrisch: „sprachliche Konstruktion, deren Teile zu anderen Formklassen gehören als die Konstruktion selbst“, siehe: *Wahrig Fremdwörterlexikon*, (München: dtv, 1999)

<sup>8</sup>egozentrisch: „das eigene Ich in den Mittelpunkt stellend“, siehe: *Wahrig Fremdwörterlexikon*, (München: dtv, 1999)

len Welt. Wenn er die gesamte virtuelle Welt im Blickfeld hat und diese möglicherweise von mehreren Seiten betrachten kann, befindet er sich außerhalb dieser Welt und es kommen exozentrische Metaphern zum Einsatz. Von egozentrischen Metaphern wird dagegen gesprochen, wenn der Anwender sich während der Benutzung innerhalb der Welt befindet und die Dinge um sich herum betrachtet und verändert.

## 2.1 Exozentrische Metaphern

Die wohl bekannteste exozentrische Metapher ist die sogenannte *World-in-miniature* (WIM)<sup>9</sup>. Dabei arbeitet der Benutzer mit einer verkleinerten Darstellung der virtuellen Welt, die ikonische Repräsentationen aller Objekte enthält und vor allem den Vorteil bietet, viel einfacher manipulierbar zu sein als eine riesige Welt, solange es um die Manipulation von größeren Objekten in dieser Welt handelt. Alle Objekte in dieser Miniaturwelt können in sechs Freiheitsgraden auf jede beliebige Distanz verändert werden. Der Benutzer kann z. B. ein Haus auswählen und dieses dort manipulieren. Da sich die WIM immer in der Nähe des Benutzers befindet, kann leicht die intuitiv verwendbare virtuelle Hand (siehe S. 10) für Selektion und Manipulation verwendet werden. Die Benutzung der WIM wird darüber hinaus auch als Navigationstechnik eingesetzt, indem der Benutzer eine verkleinerte Version von sich selbst in der Welt positioniert; so haben kleine Bewegungen in der realen Welt große Auswirkungen in der virtuellen Welt. Zum Einsatz dieser Technik ist das Tracken der Handposition und -ausrichtung nötig. Darüber hinaus werden diskrete Eingabeknöpfe zum Ein- und Ausschalten der virtuellen Welt benötigt und Knöpfe zum Beginnen und Beenden der Selektion bzw. Manipulation. Die Implementierung erfolgt in der Regel über eine Kopie des Szenengraphen, der bei entsprechend gesetzter Kameraorientierung und -perspektive die gesamte Funktionalität der großen Umgebung besitzt.

World-in-miniature (WIM)

Wie der Name schon vermuten lässt, wird der *Scaled-world grab* in erster Linie zur Skalierung der Welt um den Benutzer herum eingesetzt, je nach Implementierung kann das Verfahren aber auch so funktionieren, dass statt der Welt der Benutzer selbst skaliert wird. Ziel ist es, dass der Benutzer mit der virtuellen Hand das ausgewählte Objekt berühren kann. Das Skalieren hat den Effekt, dass der Benutzer, solange er sich nicht bewegt, die Skalierung nicht wahrnimmt. Dreht er sich jedoch oder bewegt sich von seiner bisherigen Position weg, stellt er fest, dass er zum Riesen oder zum Zwerg geworden ist. Die Implementierung ist verhältnismäßig einfach, lediglich ist die Welt (bzw. der Benutzer) nach einem bestimmten Faktor<sup>10</sup> zu skalieren. Dazu werden wie-

Scaled-world grab

<sup>9</sup>R. Stoakley, M. Conway und R. Pausch. 'Virtual Reality on a WIM: Interactive Worlds in Miniature', *Proceedings of CHI*, (1995), S. 265–272.

<sup>10</sup>Auge-Hand-Entfernung/Auge-Objekt-Entfernung



Abb. 2.1: World-in-miniature, Quelle: Virtual Cities, Universität Bonn

derum die Handposition und -ausrichtung sowie die Kopfposition getrackt und es wird ein Knopf zum Kennzeichnen von Beginn bzw. Ende der Manipulation benötigt.

## 2.2 Egozentrische Metaphern

Die Gruppe der egozentrischen Metaphern lässt sich weiter untergliedern nach der Nutzung der Metaphern für die lokale oder die Distanz-Interaktion. Virtual-Hand-Metaphern (z. B. „klassische“ virtuelle Hand, Go-Go) sind für lokale Interaktion geeignet, die Distanz-Interaktion ist teilweise unkomfortabel. Für solche Fälle werden in der Regel Virtual-Pointer-Metaphern (z. B. Raycasting) eingesetzt, bzw. mit Virtual-Hand-Metaphern kombiniert. Auch für die egozentrischen Metaphern gilt aber, dass diese nicht für alle möglichen Interaktionstechniken gleich gut geeignet sind.

### Virtual-Hand-Metaphern

Größter Vorteil und erstes Argument für die Einführung der *virtuellen Hand* war die Natürlichkeit, die diese Metapher mit sich bringt: Die Bewegungen

klassische  
virtuelle Hand

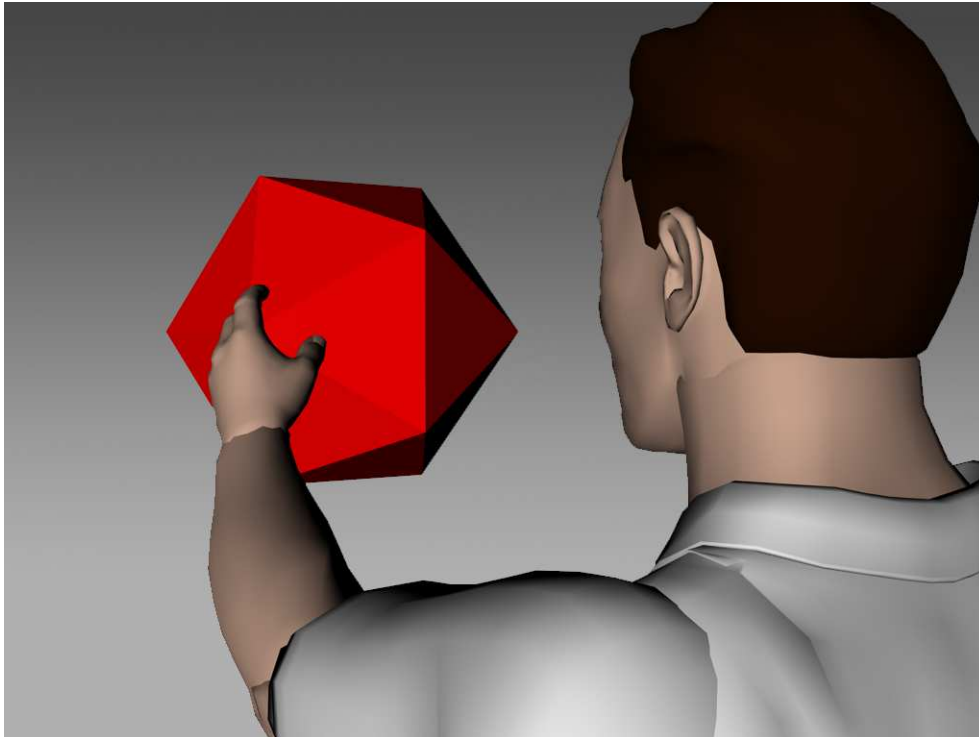


Abb. 2.2: Virtual-Hand-Metapher

der realen Hand des Benutzers werden eins-zu-eins auf die 3-dimensionale, virtuelle Hand übertragen und lassen das Selektieren und Manipulieren von Objekten besonders leicht nachvollziehen. Nahe virtuelle Objekte können einfach gegriffen und manipuliert werden. In Kombination mit einem Datenhandschuh kann so eine besonders starke Annäherung an Interaktion in der wirklichen Welt erzielt werden, doch auch der Einsatz anderer Eingabegeräte lässt sich auf die virtuelle Hand übertragen. Wichtig ist dabei in erster Linie das Tracking der Handposition und -orientierung sowie das Überwachen eines Eingabeknopfes, der die Benutzung einer Interaktionstechnik (z. B. Selektion) signalisiert. Selektion erfolgt dabei innerhalb des Volumens des Objektes und kann somit über die Erkennung einer Kollision von Hand und Objekt realisiert werden. Für die Manipulation wird das zu bearbeitende Objekt im Szenengraph mit der Benutzerhand verknüpft und erbt somit die Transformationen der Hand, bis es anschließend wieder von dieser gelöst wird. Weniger gut lässt sich die virtuelle Hand für die Selektion und Manipulation entfernter Objekte anwenden. Somit ist ihre Reichweite als beschränkt anzusehen.

Eine Erweiterung erfuhr das Konzept der virtuellen Hand durch die Einführung der *Go-Go arm-extension*<sup>11</sup>, die später zu *stretch Go-Go* (unendliche Reichweite) und *fast Go-Go* weiterentwickelt wurde. Dabei wird der virtuelle Arm des Benutzers verlängert und ermöglicht so, auch weiter entfernte

Go-Go  
arm-extension

<sup>11</sup>I. Poupyrev et al., 'The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR', *Proceedings of the ACM Symposium on User Interface Software and Technology* (1996).

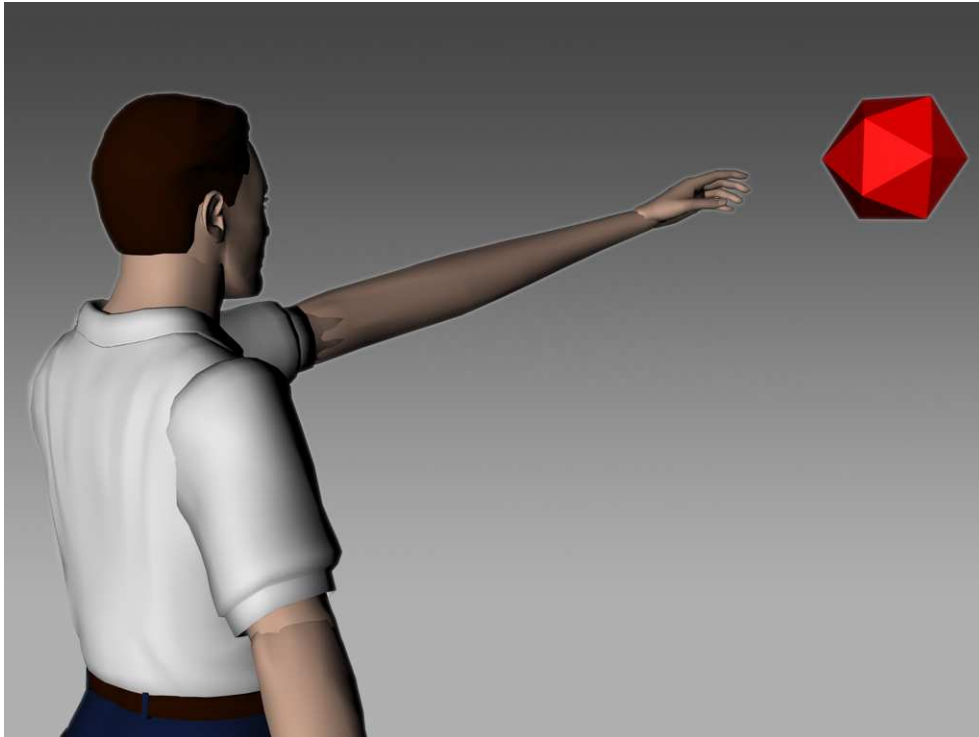


Abb. 2.3: Go-Go arm extension

Objekte zu selektieren und zu manipulieren. Auch eine solche automatische Armverlängerung bietet eine stark intuitive Interaktion, die auch das Überschreiten von großen Distanzen ermöglicht. Dabei wird die tatsächliche Streckung des Armes um einen bestimmten Faktor vergrößert und auf den virtuellen Arm übertragen. Dies ermöglicht das Greifen von Objekten in endlicher Entfernung, macht aber aufgrund der nicht-linearen Verlängerungsrate, das genaue Positionieren der Hand schwierig.

Eine virtuelle Hand kann darüber hinaus ebenso als Metapher für Travelling (siehe S. 16) verwendet werden. Wird *grabbing the air* angewendet, streckt der Benutzer seinen Arm aus und greift in den virtuellen (leeren) Raum. Jede darauf folgende Geste bewegt dann die Welt um den Benutzer herum, ähnlich dem Ziehen an einem Seil, bis dieser die Hand wieder öffnet. Die Translation der Welt wird dabei nach einem Vektor, ausgehend vom ursprünglichen Griffpunkt zur aktuellen Handposition vollzogen, bis der Griff gelöst wird. Aufgrund der starken Verknüpfung dieser Metapher mit dem Greifen, lässt sie sich am besten mit einem entsprechend ausgestatteten Handschuh umsetzen, der zum einen getrackt wird und zum anderen ein solches Greifen ermöglicht; machbar wäre es allerdings auch, das Greifen über einen Knopf zu realisieren.

grabbing  
the air

### Virtual-Pointer-Metaphern

Die *Pointer*-Metapher wird u. a. als Navigationstechnik eingesetzt. Der Be-

Pointing

nutzer spezifiziert dabei die Richtung des Reise-Vektors mit einem getrackten Gerät, das üblicherweise an die Hand gekoppelt ist, beispielsweise ein Stab oder Handschuh. Zusätzlich wird lediglich ein Eingabeknopf zum Starten bzw. Beenden der Interaktion benötigt; alternativ kann bei manchen Anwendungen eine kontinuierliche Bewegung erwünscht sein. Die Implementierung erfolgt in der Regel über die Translation des Standpunktes.<sup>12</sup>

Das zum Reisen in der virtuellen Umgebung eingesetzte *Gaze-directed steering* verwendet ein Tracking des Kopfes zum Ermitteln des Richtungsvektors. Dabei wird nicht etwa die Blickrichtung des Benutzers verwendet, sondern die Ausrichtung des Kopfes. Dies ermöglicht dem Benutzer, zumindest durch Bewegung der Augen die Umgebung auch während der Reise zu betrachten und die ggf. ebenfalls getrackte Hand für andere Anwendungen zu verwenden, ist aber stärker eingeschränkt als beim Pointing.

Gaze-directed  
steering



Abb. 2.4: Ray-casting

Die klassische Lichtstrahl-Technik ist unter dem Namen *Ray-casting* bekannt geworden und ebenfalls den Virtual-Pointer-Metaphern zuzurechnen. Sie wird vornehmlich zur Selektion von virtuellen Objekten eingesetzt und benutzt dabei die Metapher eines virtuellen Lichtstrahls als Zeiger, der aus der virtuellen Hand (oder einem anderen Zeigegerät) hervorgeht. Schneidet dieser Auswahlstrahl ein Objekt, das theoretisch beliebig weit entfernt sein kann, so kann dieses durch Betätigen eines Tasters ausgewählt werden. Der Umgang mit den Lichtstrahltechniken ist sehr natürlich und erfordert daher wenig Lernzeit. Jedoch ist es oft schwierig, kleine und weit entfernte Objekte genau zu

Ray-casting

<sup>12</sup>Hier wird in der Regel der Faktor aus Richtungsvektor und Geschwindigkeit gebildet.

treffen. Darüber hinaus ist der Einsatz von Ray-casting als Manipulationstechnik für entfernte Elemente sehr schwierig, da das Objekt nicht, wie bei den arm-extension-Techniken an der Hand zentriert wird. Statt dessen tritt ein Hebelarm-Problem auf, bei dem das Objekt an das Ende eines sehr langen Hebels geknüpft scheint und eine Entfernungskontrolle fast, andere Manipulationsarten sogar vollkommen unmöglich werden: Beispielsweise ist eine intuitive Rotation des Objektes nur um die Achse des Lichtstrahls möglich und somit ist das Verfahren auf nur einen unabhängigen Freiheitsgrad beschränkt; soll das Objekt vertikal bewegt werden, erfährt es gleichzeitig eine Rotation und eine Translation.<sup>13</sup> Ein weiterer Nachteil dieser Technik ist es, dass Objekte, die hinter anderen Objekten liegen, nicht oder nur sehr umständlich ausgewählt werden können.

Die *HOMER*<sup>14</sup>-Technik ist eigentlich eine hybride Technik, die eine Kombination von Ray-casting- und Virtual-Hand-Technik darstellt: Sie ermöglicht es dem Benutzer, ein Objekt in der Szene mit Hilfe eines Lichtstrahls auszuwählen. Ist das Objekt gewählt, wird es an der Hand zentriert (oder, je nach Implementierung, die Hand am Objekt positioniert) und kann wie bei den virtuelle-Hand-Techniken manipuliert werden. Dadurch erfolgt eine Rotation des gewählten Objekts um das Objektzentrum und nicht, wie bei anderen Ray-casting-Techniken, um die virtuelle Hand.

HOMER

Basierend auf den Ray-casting-Techniken entstanden Techniken, die einen Lichtkegel zur Selektion einsetzen (*Cone-Metaphern*, siehe Abb. 2.5). Die Basisform wurde unter dem Titel *Flash light*<sup>15</sup> bekannt. Der benutzte Lichtkegel erlaubt eine leichtere Auswahl auch kleinerer Objekte über größere Distanzen, macht allerdings eine Disambiguierung<sup>16</sup> nötig, wenn mehr als ein Objekt im Spotlight sind. Die Umsetzung des Konzeptes unter dem Titel *aperture* (Abb. 2.6) benutzt einen Hand- oder Augen-kontrollierten Kegel, der eine interaktive und intuitive Objekt-Disambiguierung durch Verändern der Kegelgröße ermöglicht und die Auswahl vollständig auf zwei Dimensionen reduziert. Die Manipulation von Objekten gestaltet sich allerdings als schwierig. Die Variante *Liang's Cone* arbeitet mit einem mit einem Richtungsvektor ausgestatteten Hand-kontrollierten Kegel, in dem das jeweils dem Benutzer nächste Objekt gewählt wird. Ursprünglich entwickelt um leicht sehr kleine Objekte auszuwählen, bestehen bei diesem Verfahren besonders Schwierigkeiten bei der Auswahl von teilweise verdeckten oder auch weiter hinten liegenden Objekten. Die Implementierung erfolgt ähnlich wie beim Ray-casting,

Flashlight

aperture

Liang's Cone

<sup>13</sup>Doug A. Bowman und Larry F. Hodges. *An Evaluation of Techniques for Grabbing and Manipulation Remote Objects in Immersive Virtual Environments*, <http://people.cs.vt.edu/~bowman/papers/grab.pdf>, 08.09.2004.

<sup>14</sup>Hand-centered Object Manipulation Extending Ray-casting, siehe auch Doug A. Bowman und Larry F. Hodges. *An Evaluation of Techniques for Grabbing and Manipulation Remote Objects in Immersive Virtual Environments*, <http://people.cs.vt.edu/~bowman/papers/grab.pdf>, 08.09.2004

<sup>15</sup>auch: *spot light*

<sup>16</sup>Vermeiden von Mehrdeutigkeit

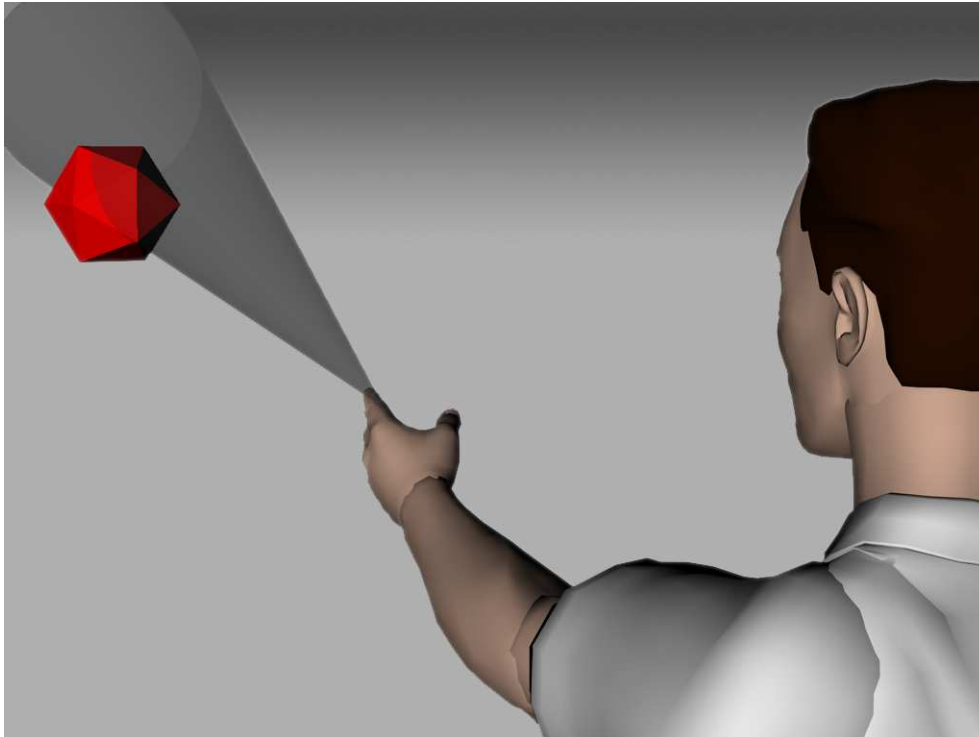


Abb. 2.5: Cone-Metapher

jedoch müssen alle Objekte der Szene daraufhin überprüft werden, ob sie innerhalb des Kegels liegen, der durch Position und Richtung des benutzten Zeigergerätes festgelegt ist. In einer weiteren Variante, bekannt als *Steed's Cone*, werden beim Drücken des Auswahlknopfes alle Objekte innerhalb des Kegels hervor gehoben und bilden ein potenzielles Set von auswählbaren Objekten. Anschließend bewegt der Benutzer den Kegel und alle Objekte, die dabei den Kegel verlassen, werden aus dem Set entfernt. Befindet sich nur noch ein einziges Objekt im Set, kann dieses durch Loslassen des Knopfes ausgewählt werden. Vorteil dieses Verfahrens ist es, dass auch möglicherweise verdeckte Objekte ausgewählt werden können.

Steed's Cone

Metaphern, die auf der Image-Plane-Technik (Abb. 2.7) basieren, projizieren den virtuellen 3-dimensionalen Raum auf eine 2-dimensionale Ebene (*image plane*) und vereinfachen so die Selektion von Objekten, bei der dadurch auch die Entfernung keine Rolle spielt. Die Auswahl erfolgt durch einen Strahl, der definiert wird durch die Position des dominanten Auges des Benutzers und des Eingabegerätes. In Verbindung mit dieser Projektion werden zur Auswahl unterschiedliche Verfahren eingesetzt. Bei der als *sticky finger* bekannten Methode wird die Hand über dem Objekt auf der Bildebene platziert und somit das Objekt ausgewählt, bei der *Framing-hand*-Methode wird das Objekt von zwei Händen eingeschlossen und dabei die Hand- sowie die Kopf- bzw. Augenposition getrackt, um die Auswahl festzustellen. Diesen und weiteren Methoden mit sehr ähnlichen Verfahren, wie *lifting palm* und *head crusher*, ist

image plane

sticky finger

framing hand

lifting palm,  
head crusher



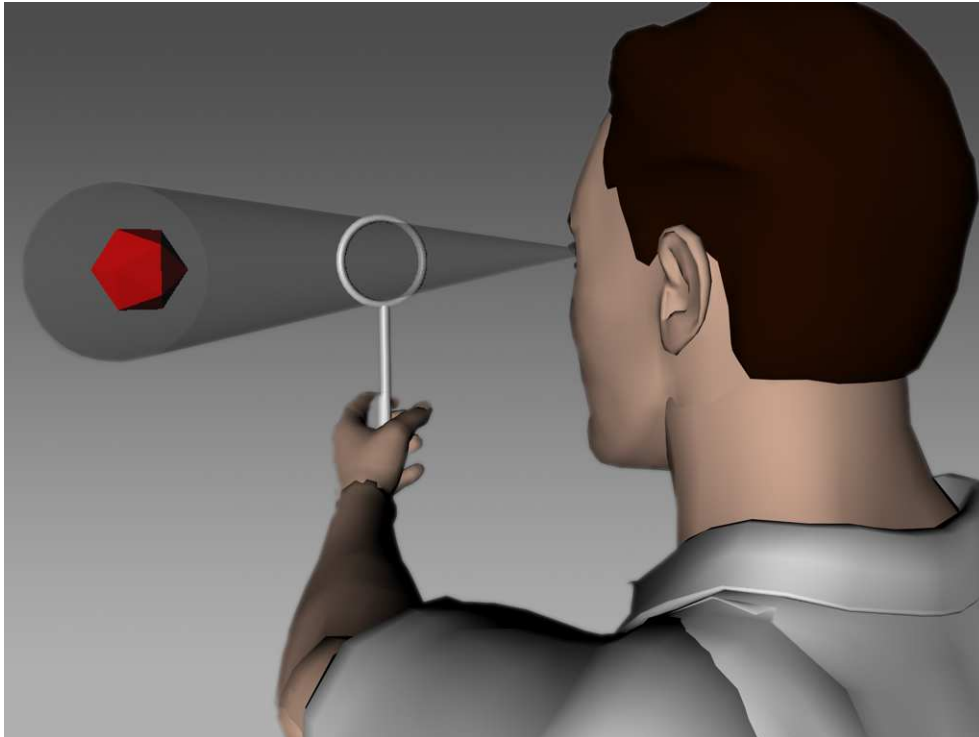


Abb. 2.6: Aperture-Metapher

gemeinsam, dass die Manipulation von Objekten recht schwierig ist. Daher werden sie in der Regel mit weiteren Metaphern verknüpft.

### Travel-Metaphern

Travelling ist eine der am häufigsten verwendeten Anwendungen in virtuellen Umgebungen. Je nach Anwendungsfall kommen dabei alle auch in der Realität verwendeten Reisearten zum Einsatz. Ganz allgemein kann Travelling beschrieben werden als die Bewegung zwischen zwei Orten. Die aus der Realität bekannten Bewegungsmöglichkeiten werden dabei zunächst eins-zu-eins auf die virtuelle Umgebung übertragen: *walking*, *driving* und *flying* bilden die Grundlage der Bewegung. Hinzu kommen *diving/swimming* und die *Teleportation*. Die Umsetzung der Metaphern erfolgt auf unterschiedlichen Wegen: am leichtesten lässt sich das Fahren übertragen, dem Benutzer wird lediglich ein physikalisches oder virtuelles Lenkrad und ggf. Pedale gegeben. Ähnlich kann beim Fliegen verfahren werden, wenn dies mit einem Flugzeug geschehen soll. Gehen oder Fliegen ohne Hilfsmittel muss jedoch anders umgesetzt werden. Steht dem Benutzer ein (eingeschränkter) physikalischer Raum zum Bewegen zur Verfügung, wie dies beim Einsatz eines Laufbandes oder eines Fahrrades der Fall sein kann, kann dieses getrackt und in das System übertragen werden. Die Übertragung kann dabei entweder eins-zu-eins erfolgen oder um einen bestimmten Faktor skaliert werden. Steht diese Möglichkeit nicht zur Verfügung, muss die Bewegung anders gesteuert werden. Dazu stehen

walking,  
driving, flying,  
teleportation

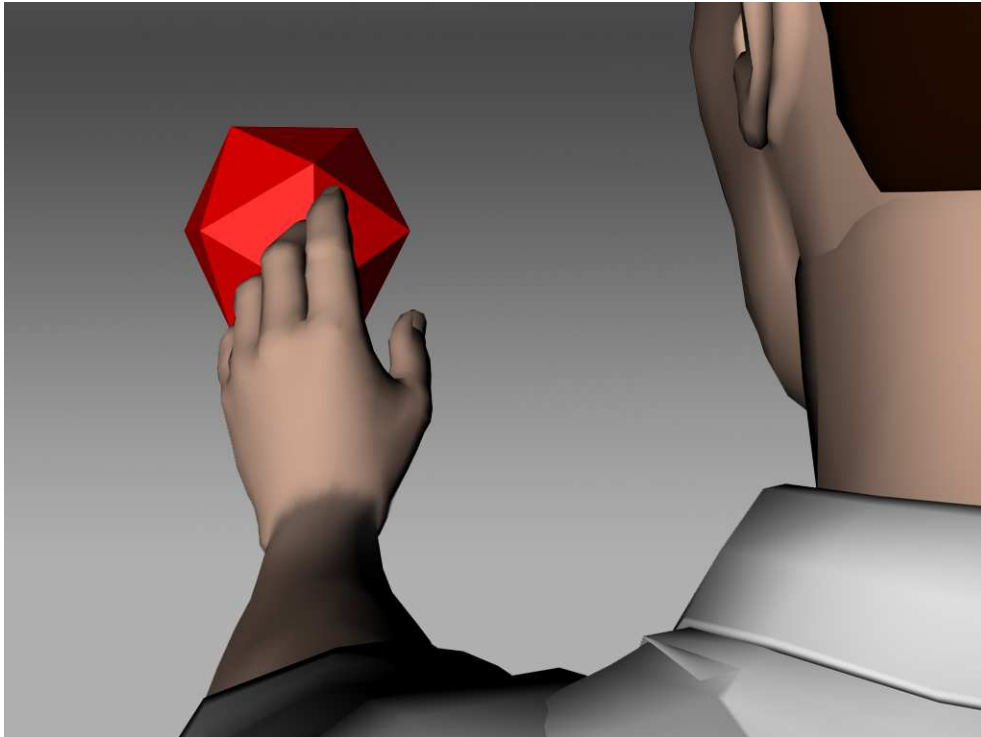


Abb. 2.7: Image-Plane-Metapher

grundsätzlich alle anderen Eingabemöglichkeiten zur Verfügung, die sinnvoll mit der eingesetzten Navigationstechnik (siehe S. 31) kombiniert werden können. Beispielsweise ließe sich die Hand mit einem Tracking-System verfolgen und je nach Position der Hand in drei virtuellen Feldern vor dem Körper die Bewegung beschleunigen (Hand ist entfernt vom Körper), die Geschwindigkeit halten (Hand etwa auf mittlerer Armlänge) oder verringern (Hand nahe am Körper); zusätzlich wäre ein weiteres Eingabegerät nötig, z. B. ein Knopf um die Reisetätigkeit zu aktivieren bzw. zu deaktivieren. Auch für die Bewegungsrichtung ist eine entsprechende Eingabe erforderlich.

Es ist zu beachten, dass bei einigen Travel-Metaphern die Freiheitsgrade eingeschränkt sind, während bei anderen theoretisch absolute Freiheit herrscht: walking und driving sollten so implementiert werden, dass der Benutzer sich in einer Ebene (bzw. einer festen Höhe über dem Boden) befindet und diese nicht verlassen kann. Dies erfolgt in Übereinstimmung mit Constraints und Affordances (siehe S. 7), die dem Benutzer eine starke Vereinfachung der Navigation bieten, indem sie verhindern, dass er in unerwünschte Regionen der virtuellen Welt vordringt, zum Beispiel in den Boden sinkt. Eine Einschränkung wird dabei als Barriere gesetzt, die die Koordinaten des Untergrundes hat.

## 2.3 3D-Metaphern

Bei der Kombination und Anwendung verschiedener Interaktionsmetaphern und -techniken ist es besonders wichtig, dem Benutzer die in der jeweiligen Situation angebotenen Möglichkeiten zu verdeutlichen. Dies wird in der Regel durch den Einsatz von ikonographischen Repräsentationen gemacht, wie es in der 2-dimensionalen Anwendung beispielsweise durch wechselnde Mauszeiger (Sanduhr bei Wartezeit, Pfeil über Menüeinträgen, „I“ über Textfeldern usw.) geschieht. Ähnliche Verfahren finden in virtuellen Umgebungen als 3D-Metaphern Anwendung. So wird die *Virtual Sphere* eingesetzt, um Objekten, die möglicherweise eine komplizierte Form haben, einen leichter greifbaren Körper zu geben. Wie in Abb. 2.8 verdeutlicht, wird eine durchsichtige virtuelle Kugel um das Objekt gelegt, auf die dann eine Rotation angewendet und anschließend auf das zu bearbeitende Objekt übertragen werden können. Selbstverständlich kann eine solche Sphere auch um mehrere Objekte gelegt werden, die vorher durch eine entsprechende Auswahltechnik selektiert wurden. Eine Erweiterung dieser Idee ist die *3D-Widget-Metapher*, die auch schon aus der 2-dimensionalen Bildverarbeitung bekannt ist. Dabei wird wiederum eine Form um das Objekt gelegt, häufiger jedoch ein Quader, dessen Ecken mit Symbolen für die verschiedenen Interaktionen versehen sind (siehe Abb. 3.4, S. 26). So kann in der Mitte einer Kante das Objekt in die entsprechende Richtung gedehnt (bzw. gestaucht) werden, während an den Ecken möglicherweise eine Skalierung über zwei oder sogar drei Dimensionen möglich ist. Andere Punkte auf dem Quader können für weitere Interaktionen präpariert sein, z. B. eine Rotation. Derartige Widgets bieten sich vor allem für die Mausinteraktion an, bei der die Widgets mit dem Mauszeiger erfasst und so die gewünschte Interaktion ausgeführt werden kann. Der Nachteil der Maus, lediglich zwei Bildschirmdimensionen ausnutzen zu können, wird so überwunden und die Maus kann ebenfalls für 3D-Interaktion verwendet werden. Dennoch ist auch hier der Einsatz von geeigneten Constraints bzw. Affordances ratsam, um dem Benutzer so viel Hilfe wie möglich zu geben. So sollte das Greifen einer Ecke des Objekts nur eine Aktion, z. B. das Drehen in einem Freiheitsgrad, ermöglichen, um maximale Genauigkeit bei der Interaktion zu erhalten.

Virtual Sphere

3D-Widgets

## 2.4 Weitere Metaphern

Neben den hier bereits angesprochenen gibt es viele weitere Metaphern, die ihren Weg aus der realen in die virtuelle Welt gefunden haben, so dass diese Vorstellung als eine Auswahl zu verstehen ist. Nicht angesprochen wurden bisher jene Metaphern, die dem Kenner von 2-dimensionalen Computer-Anwendungen mindestens ebenso bekannt sind: Alle in aktueller Soft- und Hardware implementierten Interaktionen greifen auf ein Klicken mit der Maus

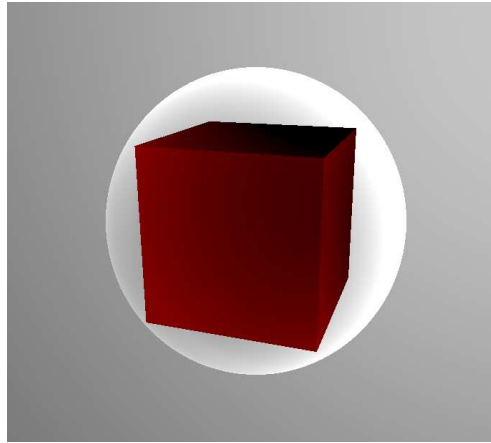


Abb. 2.8: Die Virtual-Sphere-Metapher legt eine durchsichtige Kugel um Objekte und macht diese so einfacher handhabbar.

oder die Eingabe über Tastaturen und andere Geräte zurück. Dabei stellt auch der Klick mit der Maus eine Metapher dar, deren Wichtigkeit aufgrund ihrer großen Verbreitung – und damit ihres enormen Bekanntheitsgrades – kaum zu überschätzen ist. Das Design von 3D-Software sollte daher gerade weit verbreitete Hardware möglichst gut unterstützen. Besondere Bedeutung kommt hier auch der *Desktop-Metapher* zu, die sich an das Bild eines Schreibtisches zur Grundlage macht und durch den Einsatz anklickbarer Elemente, die, wiederum als Metaphern, wie reale Gegenstände gestaltet oder benannt sind (z. B. „Papierkorb“). Der Einsatz solcher Metaphern ist dem Benutzer aus anderer Software bereits bekannt bzw. ihre Bedeutung sofort ersichtlich.

Desktop-  
Metapher

# 3 Interaktionstechniken

Die im Bereich der 2-dimensionalen Interaktion in der bekannten Fenster-Technik eingesetzten Interaktionsmethoden sind im Hinblick auf ihre Verwendung in 3D-Anwendungen neu zu bewerten. Einige Verfahren stellen sich als nicht benutzbar heraus, andere müssen ggf. umstrukturiert bzw. auf andere Art verwendet werden. Beim Einsatz verschiedener Interaktionstechniken ist auch zu überlegen, wie gut diese mit den zur Verfügung stehenden Ein- und Ausgabegeräten korrespondieren und welche Interaktionsmetaphern zum Erreichen der gesteckten Ziele am sinnvollsten eingesetzt werden. Gerade in Büro-Umgebungen ist die Anschaffung von zusätzlicher Hardware oft nicht durchführbar. Jede der Interaktionstechniken ist daher dahin gehend umzusetzen, dass ihre Benutzung auch mit herkömmlicher Hardware (Maus, Tastatur, etc.) sinnvoll möglich ist.

## 3.1 Selektion

Eine sehr grundlegende Interaktionstechnik ist die Selektion, da sie Bedingung für den Einsatz von Manipulation und Objekt-Interaktion ist; durch Selektion wird das Ziel der gewünschten (folgenden) Interaktionstechnik angezeigt. Selektion ist die Auswahl eines oder mehrerer Objekte aus der gesamten Menge der auswählbaren Objekte, z. B. der Menge aller Häuser die in der Szene positioniert sind (Abb. 3.1). Darüber hinaus muss Selektion nicht auf Objekte beschränkt sein: Es kann auch die Möglichkeit geboten werden, Punkte oder Kanten von Objekten auszuwählen; ein Verfahren, das häufig für das Verformen von Objekten verwendet wird. Die Auswahl von Flächen oder Flächenteilen (Teile des digitalen Geländemodells) kann benutzt werden, um die Ebene zu verformen oder Aktionen auszuführen, die das ganze Gelände betreffen, z. B. der Einsatz eines Darstellungsfilters. Voraussetzung für Selektion sind also eine Menge von auswählbaren Objekten, eine Technik zur Identifizierung der auszuwählenden Objekte, sowie ein Mechanismus, mit dem Beginn und Ende der Auswahl gekennzeichnet werden können (zumeist ein Mausklick oder Tastendruck).

Bei der Selektion kommen drei verschiedene Gruppen von Auswahlverfahren zum Einsatz: *diskrete*, *lokale* und *entfernte* Selektion. Die diskrete Selektion

diskret,  
lokal,  
entfernt

umfasst dabei alle Verfahren, bei denen das oder die Objekte indirekt ausgewählt werden, zum Beispiel über eine Liste in einem Menü (siehe S. 27), über Spracherkennung oder über direkte Texteingabe. Die diskrete Selektion, die folglich ohne jeglichen Kontakt mit dem Objekt auskommt, macht es so möglich, auch nicht sichtbare Objekte auszuwählen. Auf der anderen Seite wird es dadurch jedoch auch nötig, jedes Objekt mit einem dem Benutzer bekannten Identifikationsnamen zu versehen um eine eindeutige Auswahl zu ermöglichen. Bei lokaler und entfernter Selektion ist dies nicht nötig, da der Benutzer die Objekte selbst greift. Der Unterschied dieser Techniken liegt dagegen in der Art, wie der Benutzer auf die Objekte zugreifen kann: lokale Objekte liegen (virtuell) im Arbeitsbereich des Benutzers und können somit über Virtual-Hand-Metaphern direkt gegriffen werden; entfernte Objekte liegen nicht in Reichweite des Benutzers und erfordern somit zur Auswahl z. B. Ray-casting- oder Cone-Metaphern.



Abb. 3.1: Selektion mehrerer Objekte gleichzeitig, Quelle: Virtual Cities, Universität Bonn

Über diese Sonderfälle hinaus gelten bei der Selektion alle Grundsätze, die schon aus der 2-dimensionalen Interaktion bekannt sind. So ist es wichtig, sowohl einfache als auch mehrfache Selektion zu ermöglichen. Selektion von mehreren Objekten wird dabei noch unterschieden nach der Art der Auswahl: Kann nur mit einem Rahmen ausgewählt werden oder sind auch frei definierbare Formen verfügbar? Bei diskreter Selektion über eine Liste oder auch über das Anklicken mehrerer Objekte nacheinander muss verdeutlicht werden, wann die Selektion beginnt und wann sie endet; hier ist also ein Eingabeknopf vonnöten. Ebenso überlegenswert ist es, ob der vorliegende Anwendungsfall möglicherweise auch eine Auswahl von Gebieten nötig machen

könnte. Anders als bei der normalen Auswahl würden so nicht nur alle Objekte ausgewählt, die vollständig innerhalb der Auswahl liegen, sondern zusätzlich alle Teile von Objekten, die nur eine Schnittfläche mit der Auswahl bilden. Dies kann beispielsweise für die Anwendung von Texturlinsen von Bedeutung sein.<sup>17</sup>

Für alle Interaktionstechniken gilt der Grundsatz, dass dem Benutzer ausreichend Feedback gegeben werden muss: Im Fall der Selektion muss ständig erkennbar sein, ob bzw. welche Objekte ausgewählt sind, sowie auch die, die potenziell ausgewählt würden (Mouse-Over-Effect). Dies kann beispielsweise durch farbliche Hervorhebung realisiert werden.

## 3.2 Manipulation

Als Manipulation wird die Spezifikation bzw. Neuspezifikation von Objekteigenschaften bezeichnet. Diese Eigenschaften umfassen Position, Orientierung, Skalierung, Form und viele weitere Attribute, wie z. B. die Objektfarbe. Es handelt sich demnach um eine räumlich festgelegte Änderung eines Objektes, die in einer weiter gefassten Definition auch mehrere Manipulationstechniken in einer komplexen Manipulation vereinen kann. Eine Manipulation kann sich auch immer auf mehrere Objekte beziehen, im Folgenden wird zugunsten einer leichteren Lesbarkeit jedoch immer von einem Objekt gesprochen. Diese Betrachtung als ein einziges, komplexeres Objekt wird noch eindeutiger bei der Anwendung von 3D-Metaphern, wie der Virtual Sphere (siehe S. 18), die einen umgebenden Körper um die Auswahl legt, der somit das zu manipulierende Objekt repräsentiert.

Manipulation stellt ein wesentliches Verfahren zur Interaktion mit realen, physikalischen Umgebungen dar und wird als solches auf virtuelle Umgebungen übertragen. Die Qualität von Manipulation beeinflusst direkt die Qualität des gesamten Interfaces und der Benutzererfahrung, daher gilt die Entwicklung von guten Manipulationsinterfaces als wichtig aber auch verhältnismäßig kompliziert.

Wie auch die Selektion und andere Interaktionstechniken lässt sich die Manipulation grundsätzlich mit jeder Interaktionsmetapher kombinieren. Grenzen sind lediglich durch die menschliche Vorstellungskraft oder das zur Verfügung stehende Budget gesetzt. Es liegt jedoch auch hier nahe, die Interaktionsmetaphern auf ihre Leistungsfähigkeit bezüglich der zu bewältigenden Aufgabe zu prüfen. Wie schon bei der Diskussion der Ray-casting-Techniken

---

<sup>17</sup>Jürgen Döllner, Oliver Kersting und Klaus Hinrichs. *Programmierbare, interaktive 3D-Karten zur Kommunikation raumbezogener Information*, ([http://www.hpi.uni-potsdam.de/source/cgs/dhk\\_umwinf00\\_final.pdf](http://www.hpi.uni-potsdam.de/source/cgs/dhk_umwinf00_final.pdf)), 13.09.2004.

(siehe S. 13) angedeutet, lassen sich diese nur schwer für die Manipulation von Objekten einsetzen, die Virtual-Hand-Techniken (S. 10) bieten dagegen einen großen Komfort für die Manipulation, haben jedoch Nachteile bei der Selektion von entfernten Objekten. Da jede Manipulation eines Objektes eine vorhergehende Selektion des Ziels erforderlich macht, ist oft der Einsatz einer gezielten Kombination von Interaktionstechniken sinnvoll, wie sie in der HOMER-Technik (S. 14) angewendet werden. Die Auswahl der Metaphern richtet sich dabei nach den Erfordernissen und der Position des Objektes. Bei entfernten Objekten wird je nach Verfahren zwischen Objekttransfermethode und Handtransfermethode unterschieden: im ersten Fall wird das Objekt auf die Zeigerposition geholt, manipuliert und anschließend zur Ursprungsposition zurück gebracht; im zweiten Fall bewegt sich die virtuelle Hand zum Objekt und führt die Manipulation dort durch.

Wie bereits angedeutet, umfasst Manipulation in einer virtuellen Umgebung alle Veränderungen eines virtuellen Objekts. Hier sollen als Manipulationstechniken zunächst lediglich die Positionsänderung im Raum, Rotation und Skalierung des Objektes behandelt werden. Die Änderung von weiteren Eigenschaften ist sehr stark von der Art des Objektes<sup>18</sup> abhängig und erfolgt in der Regel über Menü-Interaktion (S. 27).

## Translation

Die Änderung der Objektposition ist stark abhängig von der Anwendung: hier kommen viele Einschränkungen (Constraints, siehe S. 7) zum Tragen. Im Falle der Modellierung einer virtuellen Landschaft, ist es vermutlich nicht erwünscht, Häuser unter der „Erde“ oder in der Luft zu positionieren. Die Einschränkungen müssen hier also gewährleisten, dass der Benutzer den Vorgaben entsprechend handelt. Zur Positionierung der Elemente kommt in der Regel ein Koordinatensystem (siehe Abb. 3.2) zum Einsatz, das mit drei Dimensionen (x, y und z) arbeitet. Eine Änderung der Objektposition wird auch als *Translation* bezeichnet (Abb. 3.3). Bei dieser Verschiebung wird die neue Position ausgehend vom Nullpunkt der Szene ermittelt, indem der Translationsvektor zum aktuellen Positionsvektor hinzu addiert wird.<sup>19</sup> Die genaue Positionierung des Objektes kann dabei über verschiedene Wege umgesetzt werden: Beim Einsatz einer Virtual-Hand-Metapher kann diese durch „grab and move“ direkt zur Positionierung verwendet werden; andere Möglichkeiten sind der Einsatz von physikalischen oder virtuellen Kontrollen (z. B. Joystick, Slider), über die entweder die Bewegungsrichtung und –geschwindigkeit von der aktuellen zur neuen Position, oder aber der neue Ort als absolute Ko-

<sup>18</sup>Es sind viele Fälle denkbar, bei denen die Änderung der Objektfarbe in keinem sinnvollen Zusammenhang zur Umgebung steht, in anderen Fällen kann dies aber gerade erwünscht sein, beispielsweise bei der interaktiven Präsentation eines neuen Autos.

<sup>19</sup>Carl von Ossietzky Universität Oldenburg Computer Graphics & Software Ergonomie. *Graffiti: 3D-Transformationen; Translation*, <http://olli.informatik.uni-oldenburg.de/Graffiti3/graffiti/flow7/page3.html>, 14.09.2004.



ordinaten definiert werden. Kommen beim Verschieben in der 3D-Welt 2D-Eingabegeräte, wie z. B. die Maus, zum Einsatz, so ist ein *Übertragen (Mapping)* der Bildschirm-Koordinaten auf die Szenen-Koordinaten nötig, um eine genaue Positionierung zu ermöglichen.

Mapping

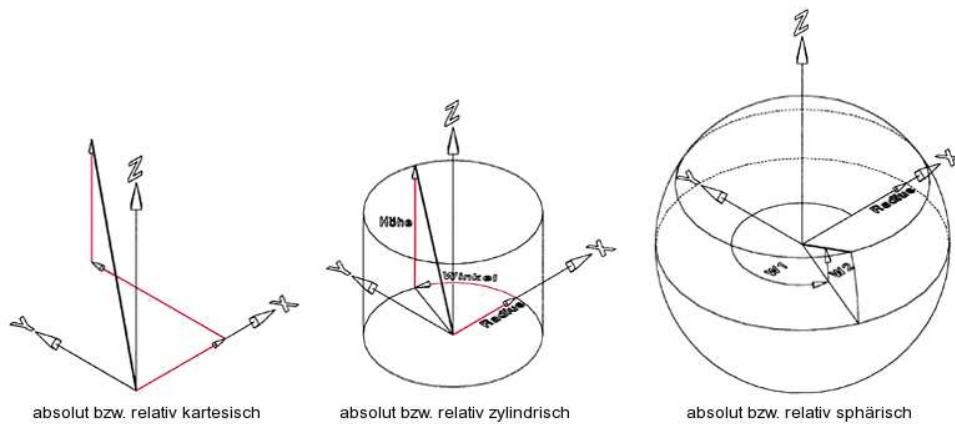


Abb. 3.2: Koordinatensysteme, Quelle: www.glossar.de

Ebenso wichtig ist es, ob ein Objekt innerhalb des erlaubten Raumes wiederum jeden Punkt einnehmen darf, oder ob es nur in bestimmten Schritten versetzt werden darf. Soll ein virtuelles Haus von einem Grundstück auf das andere verschoben werden, macht es in der Regel wenig Sinn, dieses *kontinuierlich* über die Straße zu schieben, so dass der Eindruck vermittelt wird, dies wäre ebenfalls eine zulässige Position für das Objekt. Statt dessen würde dem Benutzer vom System sofort die nächstmögliche Position (in diesem Fall auf der anderen Straßenseite) angeboten werden. Eine weitere Variante wäre es, wenn ein Objekt nur alle fünf Längeneinheiten positioniert werden könnte und beim Verschieben diese Sprünge nachvollziehen würde. In diesem Fall wird von *diskreter* Positionierung gesprochen.

kontinuierlich

diskret

### Rotation

Diskrete und kontinuierliche Varianten kommen auch bei der *Rotation* in Betracht. Dabei handelt es sich um das Drehen eines Objektes um bis zu drei Achsen. Auch für die Rotation gelten dabei möglicherweise Constraints, die das Rotieren auf bestimmte Freiheitsgrade beschränken (z. B. sollte ein Haus nicht auf die Seite gekippt werden können) oder nur eine diskrete Rotation in festen Schritten erlauben. Darüber hinaus ist bei der Rotation das *Rotationszentrum* zu beachten, das der mathematische Mittelpunkt der Rotation ist. Je nach Lage dieses Punktes (innerhalb bzw. außerhalb des Objekts) ergeben sich unterschiedliche Rotationsergebnisse. Der häufigste Fall ist wohl die sogenannte „object-centered rotation“, bei der das Rotationszentrum genau in der Objektmitte liegt und es somit um sich selbst gedreht werden kann. Bei

Rotation um ein Zentrum

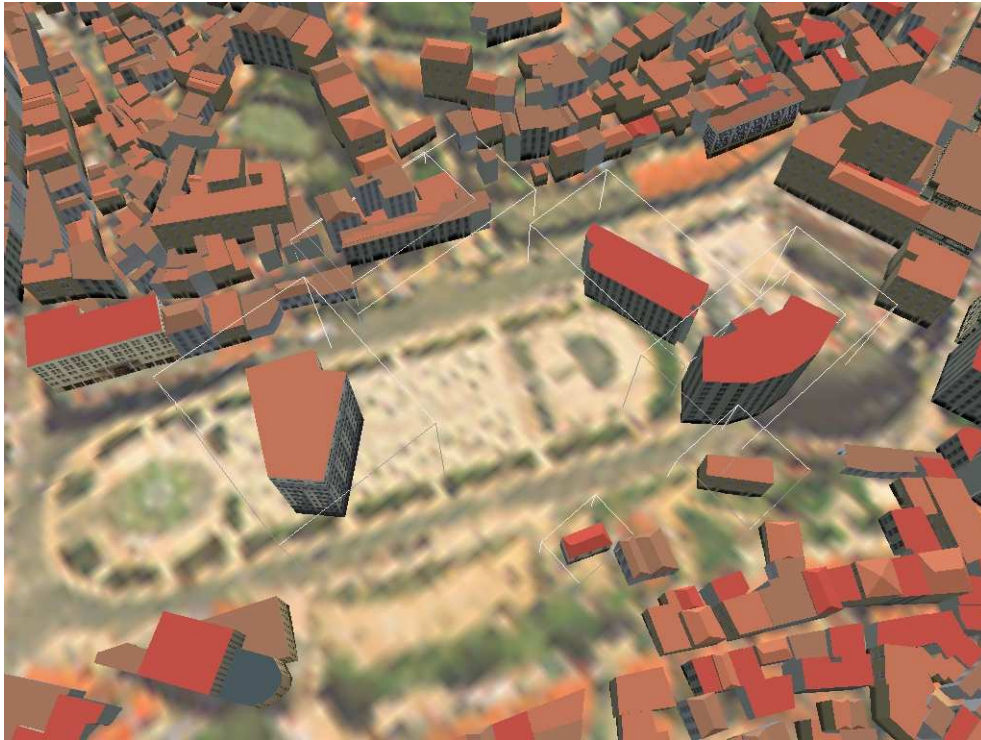


Abb. 3.3: Verschieben mehrerer Objekte gleichzeitig, Quelle: Virtual Cities, Universität Bonn

der Implementierung ist außerdem zu klären, ob das Objekt an seiner aktuellen Position zu drehen ist, oder ob es für diese Manipulation an die Hand des Benutzers geholt wird.

## Skalierung

Ähnlich verhält es sich mit der *Skalierung* von Objekten, bei der es um die Änderung der Objektgröße geht. Das *Skalierungszentrum* ist hier der Punkt, auf den sich alle Objekte zubewegen wenn sie verkleinert werden. Liegt dieser Punkt innerhalb des Objekts, wird das Objekt „normal“ skaliert. Liegt der Punkt außerhalb, wird in der Regel auch der Abstand zwischen dem Skalierungszentrum und Objekt mit skaliert. Dieser Effekt wird auch ausgenutzt, wenn die Skalierung für die Navigation in der virtuellen Umgebung genutzt wird: Der Benutzer verkleinert die Welt, bewegt sich – über den nun verkürzten Weg – zur neuen Position und bringt anschließend die Welt wieder auf ihre „echte“ Größe. Auch bei der Skalierung ist die Verwendung unterschiedlicher Metaphern möglich und sinnvoll, je nach Anwendungsfall: Zu den im Zusammenhang mit Navigation verwendeten Metaphern zählen die World-in-Miniature und der Scaled-world Grab (S. 9), für die direkte Skalierung von Objekten bieten sich in erster Linie die Virtual-Hand-Metaphern (S. 10) an.

Skalierung um ein Zentrum

Neben dem Zentrum spielt bei der Skalierung vor allem der *Skalierungsfaktor*

Skalierungsfaktor

eine große Rolle. Ausgehend von einem Faktor von 1 für die aktuelle Objektgröße kann durch eine Vergrößerung des Faktors das Objekt gedehnt, durch eine Verkleinerung gestaucht werden. Dabei werden alle drei Dimensionen des Objekts um entweder den selben (gleichmäßig/uniform) oder unterschiedliche (ungleichmäßig/non-uniform) Faktoren skaliert. Die Steuerung der Skalierung kann dabei ebenso wie die Rotation über direkte Hand-Interaktion oder über physikalische oder virtuelle Eingabegeräte erfolgen.

## Widgets

Bei der Interaktion in drei Dimensionen ist es oft erforderlich, zusätzliche Informationen über Objekte zu erhalten, oder diese komfortabler zu bearbeiten. *Widgets* sind Hilfsmittel bzw. Steuerelemente, die den Benutzer bei unterschiedlichen Aktionen unterstützen können. Widgets können dabei sowohl physikalische Gegenstände, wie z. B. einen Stift, als Metapher verwenden und idealer Weise dessen Funktionalität anbieten, oder ohne direkte Metapher weitere Interaktionsmöglichkeiten anbieten, wie es bei der *clipping plane* umgesetzt ist.<sup>20</sup> Oft handelt es sich auch um Pfeile, bei deren Aktivierung eine Aktion (z. B. Drehen eines Objektes) in die angedeutete Richtung ausgeführt wird, wie in Abb. 3.4 angedeutet ist.

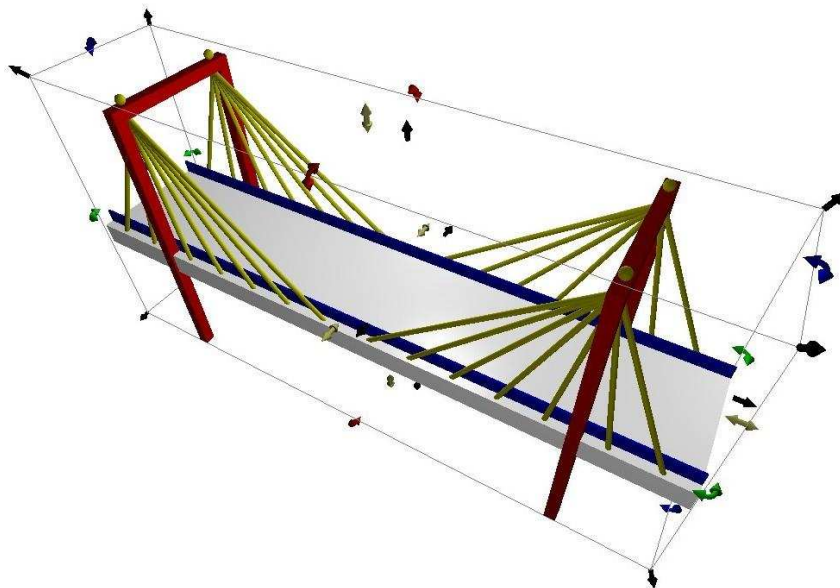


Abb. 3.4: 3D-Widgets als Hilfsmittel in der virtuellen Welt

Werden Widgets aus der 2D-Welt übernommen, so können sie zumeist leicht transformiert werden, indem sie als sogenannte *floating widgets*, im virtuel-

<sup>20</sup>Joe Kniss, Gordon Kindlmann und Charles Hansen. *Interactive Volume Rendering Using Multi-Dimensional Transfer Functions and Direct Manipulation Widgets*, (<http://www.cs.utah.edu/~jmk/papers/vis01/index.html>), 15.09.2004.

len Raum schwebende, 2-dimensionale Elemente, erstellt werden. Die Basis der Gestaltung sind dabei weiterhin traditionelle 2D-Widgets wie Slider und Buttons und sie werden oft in Kombination mit anderen Systemkontroll-Techniken eingesetzt. Um den Widgets einen Rahmen zu geben, werden sie häufig auf virtuellen Steuergeräten zusammen gefasst. Als Metapher kommen dabei virtuelle Hand-helds oder Tablets mit Stiftsteuerung zum Einsatz. Diese können auch als virtuelle Gegenstücke zu realen, getrackten Eingabegeräten eingesetzt werden. Wirkliche 3D-Widgets, die nicht auf 2D-Prinzipien basieren, sind sehr selten.

### 3.3 Menü-Interaktion und System-Kontrolle

Trotz der vielen Möglichkeiten, die Interaktionsmetaphern bieten, lassen sich viele Änderungen von Objekteigenschaften nicht sinnvoll mit Metaphern verknüpfen oder würden die Manipulation nur unnötig verkomplizieren. Dies gilt im Besonderen auch für die Aufgaben, die im Zusammenhang mit dem System stehen. In solchen Fällen werden in der Regel Menüs eingesetzt, die verschiedene Formen annehmen können, zu den bekanntesten zählen Drop-Down-Menüs, Toolbars und Slider. Die Anordnung der Menüs in 2-dimensionalen Anwendungen erfolgt in der Regel in einer Menüleiste, die sich oft am oberen Bildschirmrand befindetet. Zusätzlich kommen sogenannte *Kontextmenüs* zum Einsatz, die an die jeweilige Aktion bzw. das aktivierte Objekt geknüpft sind. Diese Anordnungen können zumeist ohne Schwierigkeiten auf eine virtuelle 3D-Umgebung übertragen werden. Zusätzlich bieten sich hier jedoch andere Möglichkeiten zum Einsatz von Menüs an, wie sie zum Beispiel im Zusammenhang mit *proprioception*, der Wahrnehmung von nicht sichtbaren Elementen durch das Gefühl und Wissen des Benutzers über die Position und Ausrichtung seines eigenen Körpers, diskutiert werden. Mine, Brooks und Sequin schlagen vor, Werkzeuge außerhalb der Sichtweite des Benutzers abzulegen, damit diese nicht das Sichtfeld verdecken. Um die Erreichbarkeit dauerhaft und in jeder Situation zu gewährleisten, setzen sie proprioception ein: „If, however, controls are fixed relative to the user’s body, he can use proprioception to find the controls, as one finds his pen in his pocket, or his pliers in his tool belt.“<sup>21</sup> Ein solches Werkzeug kann z. B. auch ein Hand-held Widget sein, wie es für Menüs in 3-dimensionalen Umgebungen eingesetzt wird.

Kontextmenüs

proprioception

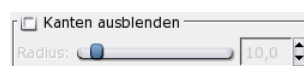


Abb. 3.5: Slider als 1-dimensionales Menü

<sup>21</sup>Mark R. Mine, Frederick P. Brooks Jr. und Carlo H. Sequin. *Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction*, (<http://www.cs.unc.edu/~mine/papers/minecows.pdf>), 16.09.2004.

Bei der Interaktion in drei Dimensionen kann selbstverständlich auch das Menü drei Dimensionen annehmen, doch bedeutet diese Möglichkeit nicht direkt, dass jedes Menü durch die Erweiterung auf drei Dimensionen auch einen Zugewinn an Funktionalität und Bedienkomfort erfährt. Wie bereits beschrieben, ist die Interaktion in drei Dimensionen nicht immer intuitiv und so kann der Einsatz von dem Benutzer bekannten Verfahren von größerem Vorteil sein als der Zugewinn an graphischen Effekten durch die Erweiterung auf drei Dimensionen.

1-dimensionale Menüs bieten sich vor allem an, wenn nur ein Eintrag ausgewählt werden soll oder wenn ein Objekt nur entlang einer Achse rotiert werden soll – das Menü wird dann oft in Form eines Sliders (Abb. 3.5) realisiert. Beim Schritt zum 2-dimensionalen Menü (Abb. 3.6) werden oft die von 2D-Desktop-Anwendungen bekannten Verfahren und Menüs eingesetzt. Menüs über zwei Dimensionen zeichnen sich dadurch aus, dass sie dem Benutzer eine weitere Auswahlebene zur Verfügung stellen.

Dimension von  
Menüs

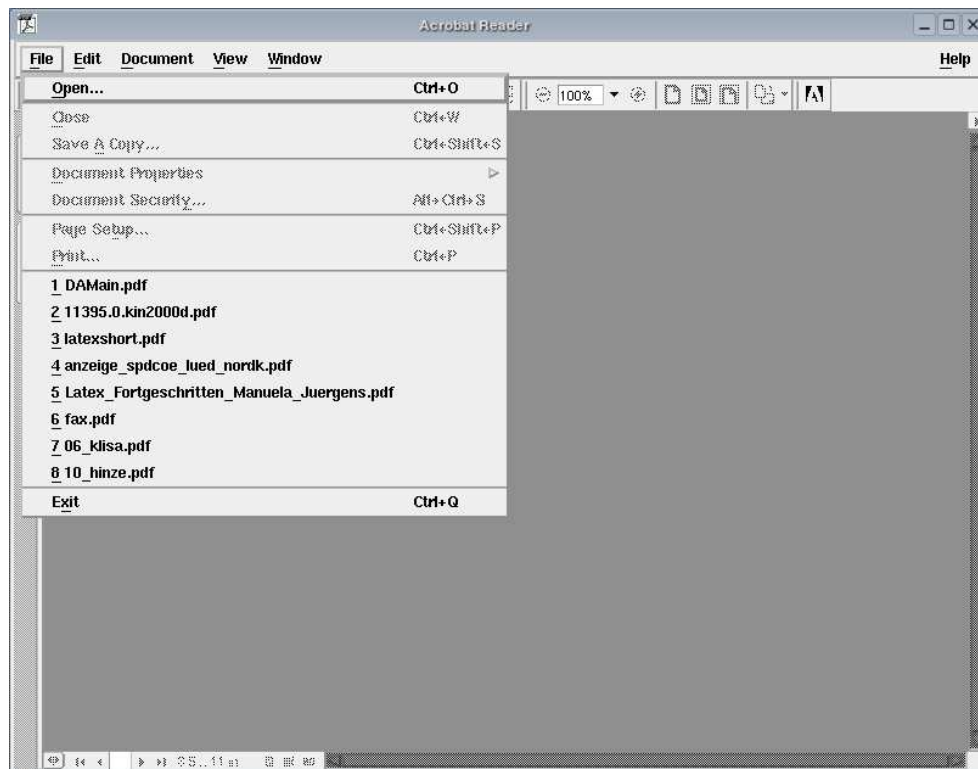


Abb. 3.6: 2-dimensionales Drop-Down-Menü

Eine beliebte 3D-Menüform ist das *Ringmenü* (auch: *sundial menu* oder *rotary tool chooser*), das aus einem durchsichtigen Ring besteht, auf dem die einzelnen Menüpunkte als 3D-Objekte angeordnet sind. Das vorderste Objekt ist dabei in der aktuellen Auswahl, lässt aber den Blick auf alle anderen Menüpunkte zu. Es handelt sich bei solchen Menüs um die 3-dimensionale Repräsentation eines Sliders. Die Hand des Benutzers rotiert das Menü um eine Achse bis das gewünschte Menüobjekt in die Auswahl fällt. Diese Form

des Menüs ist relativ einfach zu benutzen und bringt gegenüber der 1-dimensionalen Variante den Vorteil, dass der Benutzer nicht in eine feste Richtung suchen muss um ein Element zu finden. Eine neuere Entwicklung sind „echte 3D-Desktops“, wie in Abb. 3.7 dargestellt, mit deren Hilfe mehr Platz auf dem Bildschirm geschaffen und der Benutzerkomfort weiter erhöht werden soll.

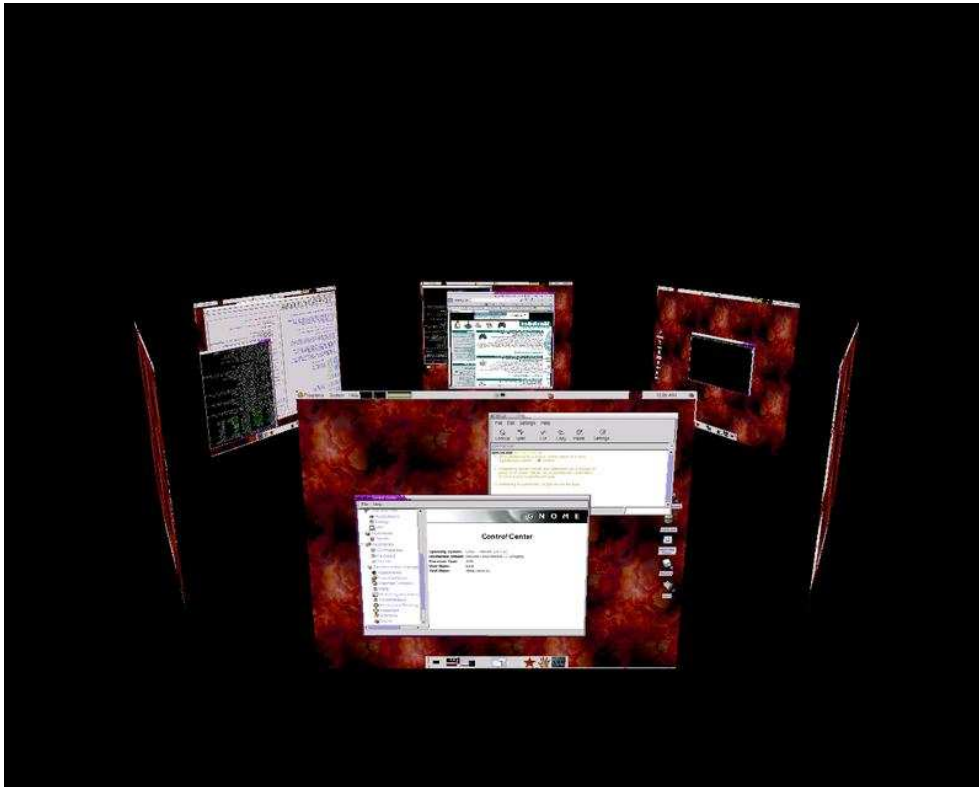


Abb. 3.7: 3D-Desktop, Quelle: desk3d.sourceforge.net

Unabhängig von der Interaktion mit den 3D-Objekten und der virtuellen Umgebung erfordert der Umgang mit jedem Computersystem immer eine Reihe von zusätzlichen Funktionen, die sehr grundlegende Einstellungen betreffen. Dies sind in erster Linie die auch für 2D-Anwendungen geltenden Befehle für das Öffnen, Speichern und das Rückgängig machen von Befehlen. Diese unter dem Begriff *System-Kontrolle* zusammen gefassten Interaktionen können auch als Zusammensetzung mehrerer Befehle zustande kommen und werden darüber hinaus ergänzt durch systemspezifische, auf die direkte Interaktion mit Objekten abzielende Befehle. System-Kontrolle umfasst dabei vor allem auch Wechsel im Modus der Interaktion mit dem System, die wiederum ebenfalls als Befehle angesehen werden können. Da in der Vergangenheit bei der Entwicklung von VE-Systemen der System-Kontrolle oft nur wenig Beachtung geschenkt wurde, ist auf diesem Gebiet noch wenig geforscht worden. Es gelten aber die auch für 2D-Anwendungen gültigen Grundsätze, dass System-Kontrolle schnell erlernbar und auch selbst schnell sein muss. Sie muss derart gestaltet werden, dass sie es dem Benutzer erlaubt, auf die vor ihm liegende

System-  
Kontrolle

Aufgabe fokussiert zu bleiben und ihm gleichzeitig nützliches Feedback zu geben.

### 3.4 Objekt-Interaktion

Alle Anwendungen der Objekt-Interaktion lassen sich grundsätzlich von der 2-dimensionalen in die 3-dimensionale Umgebung übertragen. Verfahren, die dabei in der 2-dimensionalen Welt seit langem eingesetzt werden, können oft eins-zu-eins übernommen werden und bieten dem Benutzer so die gewohnte Bedienung: Kommt auch in der virtuellen Umgebung eine Tastatur zum Einsatz, bietet es sich an, die üblichen Tastaturkombinationen hier mit der gleichen Funktionalität zu belegen, wie es in den 2D-Welten von Windows- und Unix-System der Fall ist.

Bei der Interaktion mit Objekten, die über die in Abschnitt 3.2 vorgestellten Verfahren hinaus geht, handelt es sich um Verfahren, die auf alle Objekte gleich anzuwenden sind: Hinzufügen, Kopieren, Löschen und Ausschneiden. Dies bedeutet allerdings nicht, dass jedes Objekt der virtuellen Umgebung entfernbar sein muss – das kann (und sollte) durch sinnvoll eingesetzte Einschränkungen im Einzelfall geregelt sein; lediglich das grundsätzliche Verfahren dieser Interaktionstechniken ist für alle Objekte gleich, sofern es denn zum Einsatz kommt.

Die vier Techniken sind in ihren Grundlagen hinlänglich bekannt, so dass hier nur auf ein paar Besonderheiten verwiesen werden soll. Beim *Hinzufügen* von Objekten ist zu beachten, an welcher Stelle in der virtuellen Umgebung sie auftauchen. Oft ist es sinnvoll, jedes neu hinzugefügte Objekt entweder an einer festen Stelle einzusetzen oder es dem Benutzer an die Hand zu geben, so dass er die Platzierung selbst vornehmen kann. Beim *Kopieren* ist es wichtig, Abhängigkeiten der Objekte untereinander nach den Erfordernissen sauber zu trennen: Wird ein Objekt kopiert und dann als Kopie neu in die Szene eingefügt, sollte dieses im Normalfall als eigenständiges Objekt behandelt werden. Hier macht sich die Unterscheidung von Zeigern auf Objekte und echten Kopien bemerkbar. Das *Löschen* von Objekten aus der Umgebung bringt dasselbe Problem mit sich: Es muss dem Benutzer klar visualisiert werden, wenn andere Objekte mit dem zu löschenden verknüpft sind und ggf. durch ein Löschen ebenfalls entfernt würden. Für diese Techniken, wie auch für das *Ausschneiden* virtueller Objekte gilt, dass verschiedene Interaktionsmetaphern und –techniken zum Tragen kommen können: Objekt-Interaktion kann sowohl über Menüs wie auch über Widgets stattfinden. Ebenso sind virtuelle Umgebungen denkbar, in denen solche Interaktionstechniken vollständig auf diese Verfahren verzichten und ausschließlich über Gesten, beispielsweise mit der virtuellen Hand, oder Sprache gesteuert werden.

Hinzufügen

Kopieren

Löschen

Ausschneiden

### 3.5 Navigation

Neben den oben besprochenen Interaktionstechniken, bei denen der Benutzer zumeist mit einem Objekt in der Szene oder aber – im Fall der Systemkontrolle – mit dem System selbst kommuniziert, geht es bei den Navigationstechniken darum, sich innerhalb der virtuellen Umgebung zu orientieren und Objekte und Positionen zu verorten. Jeder Punkt und jedes Objekt im Raum kann dabei Ziel, Position oder Hindernis sein und es muss definiert werden, ob ein solcher Punkt gesucht wird, oder ob es vielmehr um den Weg dorthin geht. Dies macht auch deutlich, dass Navigation sehr stark mit *Travelling* verknüpft ist, das, wie in der realen Welt, als Bewegung von einem Ort zu einem anderen definiert wird. Beim Travelling werden unterschiedliche Travel-Metaphern (siehe S. 16) eingesetzt, die in der Regel an die natürlichen Fortbewegungsweisen angelehnt sind.

travelling

Zentraler Punkt der Navigation ist neben dem Travelling als „Antriebskomponente“ das *Way-finding*, die Suche nach einem Ort und dem Weg dorthin, das sozusagen die kognitive Komponente darstellt. Beim Way-finding kommen Elemente der Wegeplanung ins Spiel und es müssen besonders die möglicherweise vorgegebenen Constraints beachtet werden, die dann für das Travelling selbst von Relevanz sind.

Way-finding

Zur Navigation bzw. zum Finden des Weges werden diverse Techniken und Metaphern in unterschiedlichen Kombinationen eingesetzt. Eine der am besten verständlichen und beliebtesten ist die sogenannte *map-based travel*, bei der der Benutzer einen Icon, der ihn selbst darstellt, auf einer Karten-Repräsentation der virtuellen Umgebung positioniert. Anschließend wird im Hintergrund der optimale Weg unter Berücksichtigung der vorgegebenen Constraints ermittelt und in einer Animation die Reise durchgeführt. In ähnlichen Techniken wird auf die Karten-Metapher verzichtet und statt dessen *target selection* eingesetzt, bei dem der Benutzer das Ziel seiner Reise durch Zeigen auf das Objekt oder Auswahl aus einem Menü bestimmt. Diese Verfahren setzen auf eine *Suchtechnik*, bei der das Ziel der Reise bereits am Anfang fest steht und vom Benutzer nicht verändert wird. Anders sieht dies aus, wenn auf *Entdeckung* als Navigationstechnik gesetzt wird: Entdeckung funktioniert ohne spezifisches Ziel und setzt stark auf die Interaktion des Benutzers, der sich (wiederum im Rahmen der gegebenen Einschränkungen) frei durch die virtuelle Umgebung bewegt. Freies Bewegen im Raum wird in vielen Fällen durch Navigationshilfen wie Mini-Karten, Kompass, eine Suchfunktion oder ein Koordinatensystem unterstützt, um dem Benutzer ein Maximum an Orientierung zu bieten.

map-based travel

target selection

suchen

entdecken

Darüber hinaus ist auch das Setzen von *Markern* ein übliches Verfahren zur Navigation, Marker stellen dabei Wegpunkte dar, die auf der Reise anzu- steuern sind. Somit handelt es sich um ein Verfahren, bei dem der Benut-

Marker



zer eingeschränkte Eingriffsmöglichkeiten hat, zumal er an den Wegpunkten mit der Frage konfrontiert werden kann, ob er seine Reiseroute ändern möchte. Erweitert wird dieses Verfahren zur *Viewpoint-Navigation*, die als nicht-immersives Verfahren auf der Basis von definierten Position in der Szene funktioniert. Diese Punkte sind gleichzeitig mit Informationen zur Blickrichtung und der Weite des Sichtfeldes verbunden und werden gezielt der Reihe nach angesteuert. Diesen Verfahren sehr ähnlich ist der Einsatz von *visual bookmarks*, die das direkte Anspringen oder auch das animierte Anfliegen von einzelnen Punkten in der Umgebung erlauben, ähnlich der von Internet-Browsern bekannten Favoriten. Hier wird eine Selektionstechnik zur Navigation verwendet.

Viewpoint-  
Navigation

visual  
bookmarks

Es kann nicht genug betont werden, dass in virtuellen Landschaften in besonderem Maße auf die Bedeutung von Constraints zu achten ist: Das Geländemodell muss konsistent sein, Wände, Zäune und Gebäude dürfen in der Regel also nicht durchschritten werden, können aber bei Einsatz einer Flugmetapher überquert werden. Besondere Beachtung verdienen auch Straßen, die je nach Klassifizierung von einem Fußgänger bzw. Auto genutzt werden dürfen oder eben nicht (Fußweg vs. Autobahn), falls das System diese Details berücksichtigen soll.

Constraints in  
der Navigation

## 4 Interaktionsgeräte

Gemeinsam mit der verwendeten Software stellen Interaktionsgeräte das Interface zwischen Computer und Mensch dar: jedes Kontrollinterface besteht aus Eingabegerät, Transfer-Funktion (Interaktionstechnik) und Ausgabegerät. Während die Software die Schnittstelle aus Sicht des Computers darstellt, sind Interaktionsgeräte die Hilfsmittel mit denen sich der Mensch dem Computer annähert. In vielen Fällen können diese Geräte sogar als Übertragung der menschlichen Gliedmaßen oder Sinne in den virtuellen Raum gesehen werden. Ein Beispiel sind Datenhandschuhe, die die menschliche Hand direkt in die virtuelle Umgebung übertragen. Aber auch Ausgabegeräte, wie z. B. Shutterbrillen verbinden die menschlichen Sinne mit der künstlichen Welt.

Auch bei der Diskussion der Hardware ist eine genaue Begriffsdefinition wichtig: Interaktionsgeräte sind keine Interaktionstechniken. Mithilfe eines einzelnen Interaktionsgerätes können viele verschiedene Interaktionstechniken implementiert werden. Im Folgenden soll eine Auswahl von Geräten vorgestellt werden, die in den bekanntesten 3D-Anwendungen eingesetzt werden. Dabei sollen grundsätzlich Eingabe- und Ausgabegeräte getrennt betrachtet werden, da sie in getrennte Richtungen, also entweder vom Mensch zum Computer oder eben anders herum funktionieren. Doch lässt sich in einigen Fällen das Eingabe- nicht eindeutig vom Ausgabegerät trennen: So vermittelt der Joystick mit ForceFeedback-Technik, der unbestritten ein Eingabegerät ist, dem Benutzer ein Gefühl, das abhängig von der Position oder Situation des gesteuerten Objektes in der Anwendung sein kann, ist somit also gleichzeitig Ausgabegerät.

Mit der sich weiter beschleunigenden Entwicklung von Computerhardware ist darüber hinaus damit zu rechnen, dass bald Geräte zur Verfügung stehen werden, die aufgrund ihrer geringen Größe und Vielseitigkeit sowohl für den Einsatz in virtuellen Räumen wie auch in der realen Welt einsetzbar sind. Ein derartiger Schritt würde voraussichtlich die Akzeptanz von Virtual Reality stark erhöhen und schnell zu einer weiter beschleunigten Entwicklung führen, die wiederum der Qualität virtueller Umgebungen zugute käme. Erste

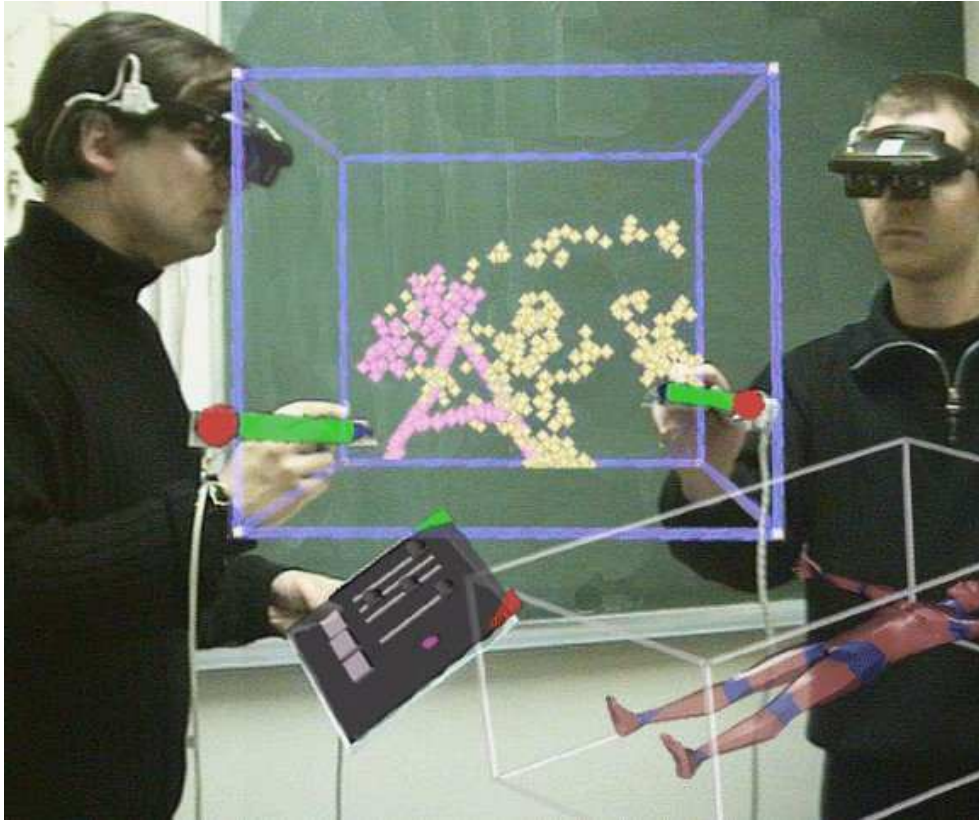


Abb. 4.1: Interaktionsgeräte in Aktion

Ansätze deuten sich an und sind in erster Linie aus dem Bereich der *embedded systems*<sup>22</sup> bzw. des *pervasive computing*<sup>23</sup> zu erwarten.

## 4.1 Eingabegeräte

Wie bereits angedeutet, sind Eingabegeräte Hardware, die es dem Benutzer erlauben, dem System Befehle zu geben, die dort interpretiert und entsprechend ausgeführt werden. Bei der Planung von VE-Anwendungen sind oft schwierige Entscheidungen bezüglich der einzusetzenden Geräte zu treffen.

<sup>22</sup>„A combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function. In some cases, embedded systems are part of a larger system or product, as in the case of an antilock braking system in a car. Contrast with general-purpose computer.“ (Michael Barr. *Michael Barr's Embedded Systems Glossary*, (<http://www.netrino.com/Publications/Glossary/>), 21.09.2004)

<sup>23</sup>„Information and communication technology will be an integrated part of our environments: from toys, milk cartons and desktops to cars, factories and whole city areas – with integrated processors, sensors, and actuators connected via high-speed networks and combined with new visualisation devices ranging from projections directly into the eye to large panorama displays.“ (Center for Pervasive Computing. *Concepts and Technology for the Future*, (<http://www.pervasive.dk/>), 21.09.2004)

Die grundsätzliche Entscheidung für oder gegen ein bestimmtes Eingabegerät wird dabei oft aufgrund des Preises getroffen. Darüber hinaus sind manche Eingabegeräte besser oder schlechter für eine bestimmte Interaktionstechnik geeignet. Mindestens ebenso wichtig bei dieser Entscheidung ist aber die Berücksichtigung des Ausgabegerätes: die Wahl des Eingabegerätes beeinflusst die verwendbaren Ausgabegeräte und umgekehrt. Und nicht zuletzt spielt auch die Kreativität eine wichtige Rolle. Viele Benutzer sind nicht bereit, sich auf neue Techniken einzulassen, wenn die alten aus ihrer Sicht hinreichend funktionieren.

So stehen am Anfang einer Betrachtung unterschiedlicher Eingabegeräte zunächst die herkömmlichen, auch aus 2D-Welten bekannten Geräte, die in die 3D-Welt übernommen wurden: Tastatur, Maus und Joystick sind wohl die häufigsten Vertreter dieser Gattung, deren Grenzen aber nur schwer zu umreißen sind, da es viele Abwandlungen (z. B. den Trackball) gibt und die Art der Geräte stark divergiert (z. B. Gamepad, Stift). Darüber hinaus gibt es solche Geräte, die an die neuen Anforderungen angepasst und damit für die 3D-Welt reif gemacht wurden, wie es beispielsweise bei der 3D-Maus der Fall ist.

Geräte aus der  
2D-Welt

Ein weiterer Ansatz ist der *Two-Pointer Input*: Dabei werden zwei Eingabegeräte parallel eingesetzt. In der an der Brown University entwickelten Form dieser Technik handelt es sich um eine Maus – die allerdings in der nicht-dominanten Hand geführt wird – und einen Eingabe-Stift. Beide Geräte verfügen über Aktionsknöpfe, über die das System die Aktionen des Benutzers interpretiert. Aktionen werden mit Kombinationen von Bewegungen und Knopf-Drücken an beiden Geräten ausgeführt.

Two-Pointer  
Input

Das System der eingesetzten Eingabegeräte lässt sich insoweit verallgemeinern, als dass grundsätzlich jedes Gerät in die Virtualität übertragen werden kann, indem es aus der Realität dupliziert wird. Dies hat den entscheidenden Vorteil, dass ein gewisses taktiles Feedback sicher gestellt ist, da der Benutzer das virtuelle Gerät ja tatsächlich in der Hand hält.

Um Bewegungen in allen sechs Freiheitsgraden in der 3-Dimensionalität zu verfolgen und in den Computer zu übertragen, genügen herkömmliche Eingabegeräte, wie Kugel- bzw. optischen Mäuse, nicht den Anforderungen, sondern müssen um eine weitere Dimension erweitert werden. In der Regel wird dabei die ergänzend erforderliche Technik nicht nur für die zusätzliche Dimension eingesetzt, sondern direkt für alle Freiheitsgrade verwendet. Beim Registrieren der Bewegungen kommen in der Regel *Tracking-Systeme* zum Einsatz, unabhängig davon, ob der Benutzer ein Eingabegerät oder lediglich seine Gliedmaßen zur Interaktion verwendet. Das Tracking-System misst dabei die Position und/oder Ausrichtung eines Sensors, und empfängt dabei einen kontinuierlichen Strom von Eingabedaten. Daraus wird der darzustellende Sichtbereich (*view-frustum*, Abb. 4.2) ermittelt. Dabei wird die virtuelle Kamera auf die Position des getrackten Kopfes gesetzt. So kann die virtuelle

Tracking-  
Systeme

view-frustum

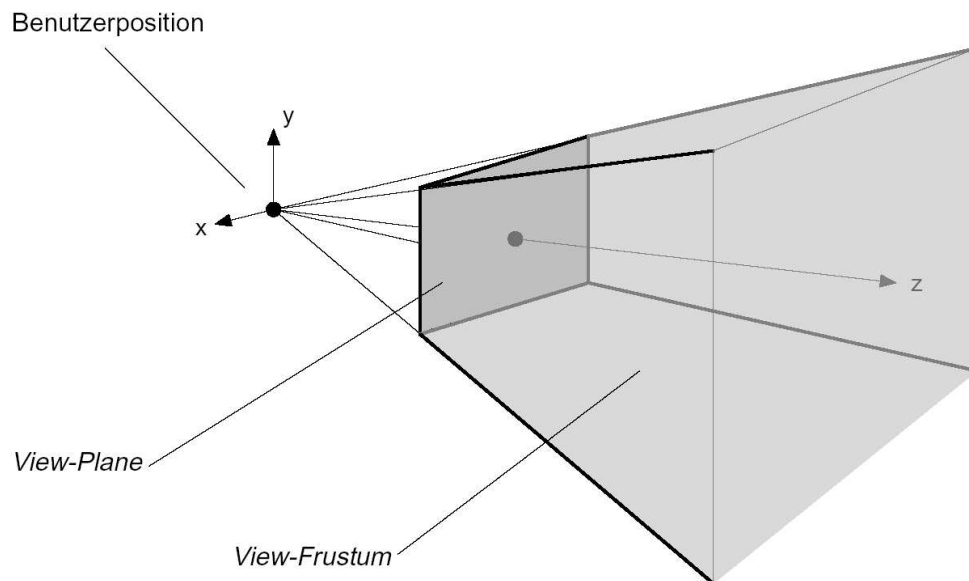


Abb. 4.2: View-frustum nach Kubica

Welt aus unterschiedlichen Perspektiven betrachtet werden, indem der Benutzer den Kopf bewegt. Es werden unterschiedliche Techniken zum Tracking unterschieden, z. B. magnetisch, mechanisch, akustisch, visuell und diverse Hybrid-Formen.<sup>24</sup> Bei den üblichen Tracking-Systemen werden die Empfänger in ausreichender Entfernung positioniert und an Kopf (Head-Tracking) oder Hand des Benutzers bzw. an dem zu trackenden Eingabegerät, z. B. einem Stift oder einer 3D-Maus (Device-Tracking), die Sensorpunkte fixiert.<sup>25</sup> Alternativ werden die Augen des Benutzers getrackt (Eye-Tracking). Wird die Position des Kopfes zur Darstellung der Szene verwendet, bedeutet dies gleichzeitig, dass die jeweils dargestellte Szene nur aus der Position des getrackten Kopfes vollständig korrekt betrachtet werden kann. Soll die Szene für einen zweiten Benutzer aus einer individuellen Perspektive betrachtbar werden, ist ein Tracking des zweiten Kopfes notwendig. Ebenso wird dann in der Regel die Darstellung einer zweiten Szene, z. B. mit einer verdoppelten Bildfrequenz, erforderlich.<sup>26</sup>

<sup>24</sup>Einen Überblick hierzu bietet Dieter Schmalstieg. *3D Interaction*, <http://www.ims.tuwien.ac.at/teaching/mmi/1-3dui.pdf>, 19.09.2004.

<sup>25</sup>Hier sollen einige Beispiele für solche Tracking-Verfahren aufgezeigt werden. So erlaubt es die an der Stanford University entwickelte *Two-Handed Direct Manipulation* dem Benutzer, die virtuellen 3D-Objekte direkt und natürlich mit beiden Händen auf der verwendeten Responsive Workbench zu bearbeiten. Dabei arbeitet der Benutzer zur Objektmanipulation entweder mit zwei pinch gloves oder einem Stift mit sechs Freiheitsgraden. Dem Benutzer stehen dabei Manipulationstechniken zur Verfügung, die ihm durch ein Menü an der Vorderseite der Workbench präsentiert werden, die aber auch über Gesten mit den Datenhandschuhen erreichbar sind. Ein Verfahren aus Wien, das unter dem Namen *Transparent Props for Virtual Desktop* bekannt geworden ist, basiert auf sogenannten transparent props einem getrackten Handheld-Stift und einem Pad. Die Objekte sind mit 3-dimensionalen Grafiken des Displays erweitert und erlauben so einen Vielzahl von Interaktionstechniken.

<sup>26</sup>Mehr Informationen zur sog. Stereoskopie finden sich in Abschnitt 4.2 auf Seite 38.

Der Notwendigkeit, die menschlichen Befehle in den Computer zu übertragen, müssen sich alle Eingabegeräte unterwerfen, doch über die Zeit haben sich interessante Ansätze entwickelt, dies zu bewerkstelligen. Die Entstehung neuer Eingabegeräte lässt sich in erster Linie zurückführen auf die Defizite der bisherigen. Für die Interaktion mit dem Computer, besonders in drei Dimensionen, sind die Unternehmen und Forschungseinrichtungen auf der Suche nach schnelleren, komfortableren und vor allem natürlicheren Mitteln zur Interaktion mit dem Computer. Daraus ist unter anderem die Erkennung von Gesten entstanden, die sich ebenfalls auf ein Tracking-Verfahren stützt. Hier wird versucht, die mit den Händen ausgeführten Bewegungen über Datenhandschuhe zu erfassen. Diese verfügen über druckempfindliche Punkte und werden zusätzlich mit den oben angesprochenen Sensoren ausgestattet, um ihre Position im Raum zu erfassen und in den Computer zu übertragen. Dieses, englisch als *gestural interaction* bezeichnete Verfahren erlaubt in einer Abwandlung auch das Erfassen der Körperhaltung (*postures*) des Benutzers. Seine Vorteile liegen vor allem darin, dass in vielen Fällen auf das Menü für die Interaktion verzichtet werden kann. Allerdings ist ein hoher Rechenaufwand für das in diesem Fall sehr komplexe Tracking erforderlich.

gestural and  
postural  
interaction

Andere Verfahren, die ohne Tracking auskommen, setzen gänzlich andere Eingabemethoden voraus, die nicht eins-zu-eins in die virtuelle Welt übertragen werden. Letzteres ist bei der Benutzung von Handheld-Devices der Fall, während für verschiedene Bewegungstechniken zum Beispiel Laufbänder (*treadmills*) oder Fahrräder eingesetzt.

treadmills,  
bicycles

Wiederum einen anderen Weg schlägt die *Spracherkennung* ein. Verfahren dieser Gruppe sind bereits seit längerer Zeit auch aus anderen Anwendungen bekannt, kommen bei der Interaktion in 3D-Räumen aber zu ungeahnter Bedeutung. Spracherkennung ist die ideale Ergänzung zu vielen anderen Interaktionsmethoden, da der Mund des Benutzers in den üblichen Systemen bisher nicht genutzt wird und somit zur Interaktion voll zur Verfügung steht und keine anderen Interaktionsmöglichkeiten behindert.

voice  
recognition

## 4.2 Ausgabegeräte

Auch für die Ausgabegeräte gilt selbstverständlich, dass diese so gut wie möglich auf die menschlichen Sinne abgestimmt sein müssen. Angefangen bei visuellen, über auditive hin zu taktilen wird derzeit bereits an olfaktorischen Ausgabegeräten gearbeitet, wenngleich letztere – ebenso wie dem Geschmackssinn zugewandte Geräte – noch am Anfang ihrer Entwicklung stehen. Beginnend bei einfachen Bildschirmen, die zunächst nur für 2-dimensionale Darstellungen verwendet und später mit unterschiedlichen Techniken auf die dritte Dimension erweitert wurden, wird heute auch versucht, Geräusche im Raum zu verteilen. Die Entwicklung ging hier von Mono- über Stereo- hin

3D-Sound

zu verschiedenen Surround-Sound-Techniken, soll im Folgenden aber nicht weiter betrachtet werden.

Bei der visuellen Darstellung von 3-dimensionalen Objekten und Welten wird zunächst zwischen monoskopischer und stereoskopischer Darstellung unterschieden. Dabei zeichnen sich die monoskopischen Systeme dadurch aus, dass sie nur ein Bild für beide Augen produzieren, wie es bei herkömmlichen Bildschirmen der Fall ist. Die Darstellung von 3D-Szenen erfolgt dabei über ein Mapping der 3D-Daten auf die 2D-Koordinaten. *Stereoskopische* Systeme machen es für den Benutzer in der Regel erforderlich, ein Gerät aufzusetzen, wie es bei der Verwendung von Shutter-Brillen der Fall ist. Die meisten dieser Verfahren gehen auf Charles Wheatstone zurück, der das erste Stereoskop entwickelte.<sup>27</sup> Stereoskopische Displays produzieren für jedes Auge ein einzelnes Bild und stellen sicher, dass dieses Bild auch nur von dem richtigen Auge wahrgenommen wird. Die Vielfalt der stereoskopischen Displays ist sehr groß und sie konnten sich in unterschiedlichen Anwendungsbereichen durchsetzen und haben ihre Stärken vor allem in solchen Fällen, in denen mehrere Benutzer mit individuellen Bildern zur Darstellung von Mehrdimensionalität versorgt werden müssen, z. B. bei Gruppenpräsentationen oder im Kino. Die bekanntesten Techniken umfassen die sogenannten Shutter-Brillen, die jeweils abwechselnd ein Bild für das linke und dann für das rechte Auge durchlassen, anaglyphische Brillen, die mit unterschiedlichen Farbkanälen für die beiden Augen arbeiten, und die Verwendung von polarisiertem Licht, das über die Brille gefiltert werden kann. Stereoskopische Display-Verfahren werden beispielsweise in Head-mounted Displays verwendet. Erst die Verwendung von *autostereoskopischen* Geräten macht es möglich, auf Geräte zu verzichten, die der Benutzer direkt an sich tragen müsste und konnte so bereits erheblich zur vergrößerten Akzeptanz von 3D-Anwendungen beitragen. Auch für die autostereoskopische Darstellung werden verschiedene Verfahren, teilweise in Kombination, eingesetzt.<sup>28</sup> „Die dargestellten Ansätze autostereoskopischer Displays zeigen, dass sich die berührungslose Visualisierung dreidimensionaler Szenen ohne Hilfsmittel von reinen Laborprototypen zu benutzbaren kommerziellen Produkten weiterentwickelt hat. [...] Für spezielle Anwendungen wie Präsentationen, medizinische Visualisierungen oder im Bereich Unterhaltung bieten diese Geräte klare Vorteile gegenüber 2D-Ansätzen oder konventionellen stereoskopischen Techniken.“<sup>29</sup>

Stereoskopie

<sup>27</sup>Gabriele Schmid. *Illusionsräume. Mesdags Panorama, Monets Seerosen, Boissonnets Hologramme und Kirchen der Gebrüder Asam. Konstruktionen und Vermittlungsstrategien*, (<http://www.eugwiss.hdk-berlin.de/schmid/diss/home.html>), 31.08.2004, S. Kap. III.20.

<sup>28</sup>Nick Holliman. *3D Display Systems*, (<http://www.dur.ac.uk/n.s.holliman/Presentations/3dv3-0.pdf>), 30.08.2004.

<sup>29</sup>Christian Geiger. 'Helft mir, Obi-Wan Kenobi. Stereoskopische Techniken und autostereoskopische Displays', *iX. Magazin für professionelle Informationstechnik*, (5/2004), S. 99–102.

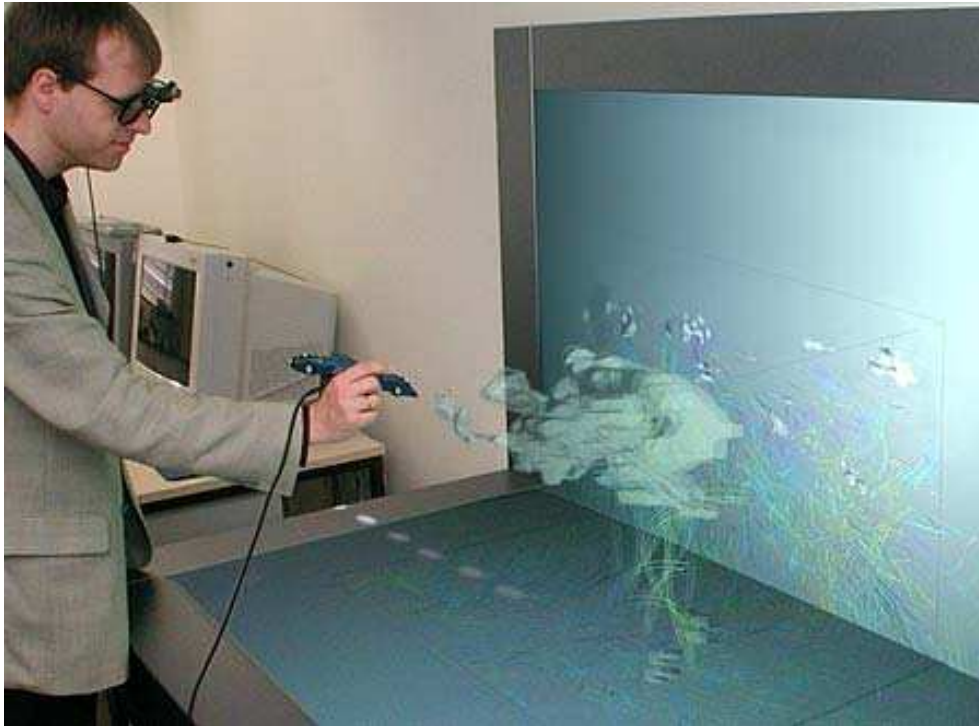


Abb. 4.3: Workbench am Rechenzentrum der Uni Hannover

Die klassischen Verfahren für die dreidimensionale Visualisierung basieren auf LCD-Shutterbrillen und Head-mounted Displays. *Shutter-Brillen* arbeiten mit Stereo-Monitoren zusammen, die über eine hohe Bildwiederholrate verfügen und somit schnell genug sind, um für jedes Auge abwechselnd ein eigenes Bild darzustellen. Sind Brille und Monitor korrekt synchronisiert, wird jeweils nur das korrekte Bild zum jeweiligen Auge durchgelassen und so die Dreidimensionalität visualisiert. Ähnlich funktionieren Brillen mit Polarisationsfilter, die das von zwei Projektoren unterschiedlich *polarisierte Licht* (vertikal vs. horizontal) filtern. Weiterentwicklungen, die ebenfalls den Einsatz von Shutter-Brillen erfordern, sind die *Workbench*<sup>30</sup> (Abb. 4.3) und das *CAVE* (Computer Assisted Virtual Environment). Während die Workbench in der Regel auf ein bis zwei Displays beschränkt ist, stellt ein CAVE bis zu 6 große Bildschirme oder Projektionsflächen zur Verfügung, die dem Benutzer das bisher beste Gefühl für maximale Immersion bieten, da sie tatsächlich einen vollständigen Raum bilden. In der Regel wird jede Projektionsfläche von zwei Projektoren angestrahlt, die auf die angeschlossenen Brillen abgestimmt sind. Auf diese Weise entsteht ein 3-dimensionales Bild. Die an der Universität Münster eingesetzte Workbench kann auch in eine horizontale Position gebracht werden, so dass sie sich besonders gut zur Visualisierung von Landschaften und zur Stadtplanung eignet (Abb. 5.1).

Shutter-Brillen

polarisiertes  
LichtWorkbench,  
CAVE

In akzeptabler Qualität, aber deutlich preisgünstiger und daher immer häufiger in kleineren Projekten eingesetzt, sind *Head-mounted Displays* (HMD,

Head-mounted  
Displays

<sup>30</sup>auch: Responsive Workbench, Holobench, Virtual Table





Abb. 4.4: Head-mounted Display nach Kölsch

Abb. 4.4), bei deren Einsatz der Benutzer zwei kleine Bildschirme und ggf. eine spezielle Optik direkt vor den Augen trägt. Er sieht also die natürlich Welt nicht (bzw. die Welt nicht natürlich). Dadurch ist der Vorteil einer recht starken Immersion gegeben.

## 5 3D-Stadtplanung

Die bisherigen Betrachtungen sollen nun noch einmal zusammen gefasst werden. Dabei werden die speziellen Anforderungen für die virtuelle 3D-Stadtplanung besonders berücksichtigt, wie sie im vorliegenden Projekt bei der Zusammenarbeit mit der Stadt Münster anzunehmen sind. Dabei ergeben sich in einigen Bereichen die Antworten auf Fragen nach Einsetzbarkeit und Effektivität sehr schnell. In anderen Bereichen ist eine genauere Betrachtung der Anforderungen bei der Umsetzung nötig bzw. setzen eine genaue Diskussion dieser Anforderungen mit den Partnern der Stadt Münster voraus.

Die Diskussion über den Einsatz von Interaktionsmetaphern hängt im vorliegenden Fall sehr stark von den verwendeten Ein- und Ausgabegeräten ab. Es ist derzeit davon auszugehen, dass im späteren Einsatz lediglich die übliche Büro-Hardware verwendet wird, an den beteiligten Arbeitsplätzen also in erster Linie ein handelsüblicher 2D-Monitor als Ausgabegerät, sowie eine Tastatur und eine Maus als Eingabegeräte zur Verfügung stehen werden. Lediglich in gesonderten Präsentationen könnten darüber hinaus gehende Geräte (Workbench, optisches Trackingsystem, Stereo-Display) eingesetzt werden, so dass der möglicherweise später auftauchende Bedarf nach tatsächlicher 3D-Darstellung nicht durch den Einsatz bestimmter Verfahren verhindert werden sollte. Für die Interaktionsmetaphern bedeutet dies zunächst, dass die, die mit speziellen Geräten (z. B. Virtual-Hand- und Ray-casting-Metaphern) verknüpft sind, zu Beginn nicht betrachtet werden müssen. Geht es um die Manipulation innerhalb der virtuellen Stadtlandschaft, werden sich voraussichtlich die genannten 3D-Metaphern (Virtual Sphere, Eckpunkte-Metapher) als nützlich erweisen und das leichte Interagieren unter Einsatz der Maus ermöglichen. Für virtuelle Spaziergänge bzw. Fahrten durch die Stadt bieten sich offensichtlich Walking und Driving an, die den Travel-Metaphern zuzurechnen sind. Gegebenenfalls ist auch das Flying als zusätzliche Option in Betracht zu ziehen, wenn beispielsweise Luftaufnahmen gemacht werden sollen, um diese mit realen Bildern zu vergleichen. Ebenso ist über die Verwendung der World-in-Miniature nachzudenken, die sich in ähnlichen Anwendungsfällen als äußerst hilfreiche Metapher erwiesen hat und dem Benutzer vor allem bei der Auswahl des aktuellen Blickwinkels helfen kann.

Im Zusammenhang mit der Verwendung der zur Verfügung stehenden Interaktionstechniken ist es zunächst unerlässlich, noch einmal auf den Einsatz geeigneter Constraints hinzuweisen: Bei einem Anspruch auf maximale Rea-

Einsatz von Hardware zur Interaktion

verwendbare Metaphern

Bedeutung von Constraints

litätsnähe, wie er im Projekt der interaktiven 3D-Stadtplanung zu stellen ist, ist die virtuelle Umgebung so zu planen und aufzubauen, dass in der realen Welt unmögliche Bewegungen und Dinge auch im VE unmöglich bleiben. Ein Haus sollte nicht mitten auf einer Straße platziert werden können oder in der Luft schweben und ein Spaziergänger sollte vor einer Wand stehen bleiben müssen statt durch sie hindurch zu gehen. Solche Constraints erhöhen gleichzeitig die Benutzerfreundlichkeit und Bedienbarkeit des Gesamtsystems.

Für den Einsatz von Interaktionstechniken bietet sich an, die in handelsüblichen 2D-Anwendungen eingesetzten Verfahren weitestgehend zu übernehmen, um den zusätzlichen Lernaufwand für die Mitarbeiter der Stadt Münster zu minimieren. Dabei geht es darum, sowohl die bekannten Klick-Techniken und vor allem auch Tastenkombinationen für Maus- bzw. Tastatureinsatz zu nutzen. Ebenso sollte die Anordnung von Elementen in Menüs den üblichen Konventionen folgen und in diesem, wie auch in allen anderen Bereichen dem Anspruch an durchgehende Konsistenz und Übereinstimmung mit den an anderer Stelle eingesetzten Techniken Rechnung tragen. Es ist vorgesehen, für die Gestaltung von Menüs die Qt-Frameworks der Firma Trolltech<sup>31</sup> einzusetzen. Dies bietet sich in erster Linie an für die Befehle zur System-Kontrolle (Anlegen von Projekten, Öffnen, Speichern, etc.), sowie für Kontext-Menüs, wie sie sich für den Einsatz in der direkten Objekt-Interaktion und Manipulation anbieten.

Nach den derzeitigen Formulierungen der Anforderungen an die zu erstellende Software ist davon auszugehen, dass im späteren Einsatz ein Schwerpunkt auf der Visualisierung der Stadt und – damit verbunden – der Präsentation der Baugebiete durch virtuelle Rundgänge erfolgen soll. Neben den bei den Metaphern angesprochenen Verfahren zur Bewegung im VE, ist daher der Navigation besondere Beachtung zu schenken. Um dem Benutzer jede mögliche Hilfe anzubieten, sind alle Navigationsverfahren auf ihre Einsatzmöglichkeit und eventuelle Vor- und Nachteile zu prüfen. Besonders die Nutzung eines Koordinatensystems, möglicherweise abgeglichen mit echten Weltkoordinaten des Global Positioning System (GPS), sowie eine ständige Visualisierung einer Übersichtskarte, ggf. verbunden mit der World-in-Miniature-Metapher oder einem virtuellen Kompass, können diesen Anforderungen gerecht werden.

Die Betrachtung der Möglichkeiten für die virtuelle 3D-Stadtplanung erweist sich als sehr wichtig und deutet zugleich an, welche zusätzlichen Prozesse zur Visualisierung und Interaktion derzeit noch nicht genutzt werden können weil sich die dafür erforderliche Hardware noch nicht ausreichend etabliert hat. Eine Erweiterung auf tatsächliche 3D-Hardware verspricht eine größere Immersion, die möglichen Kauf- und Mietinteressenten einen besseren Eindruck des Wohngebietes vermitteln und ihnen so zusätzliche Anreize geben könnten.

---

<sup>31</sup>[www.trolltech.com](http://www.trolltech.com)

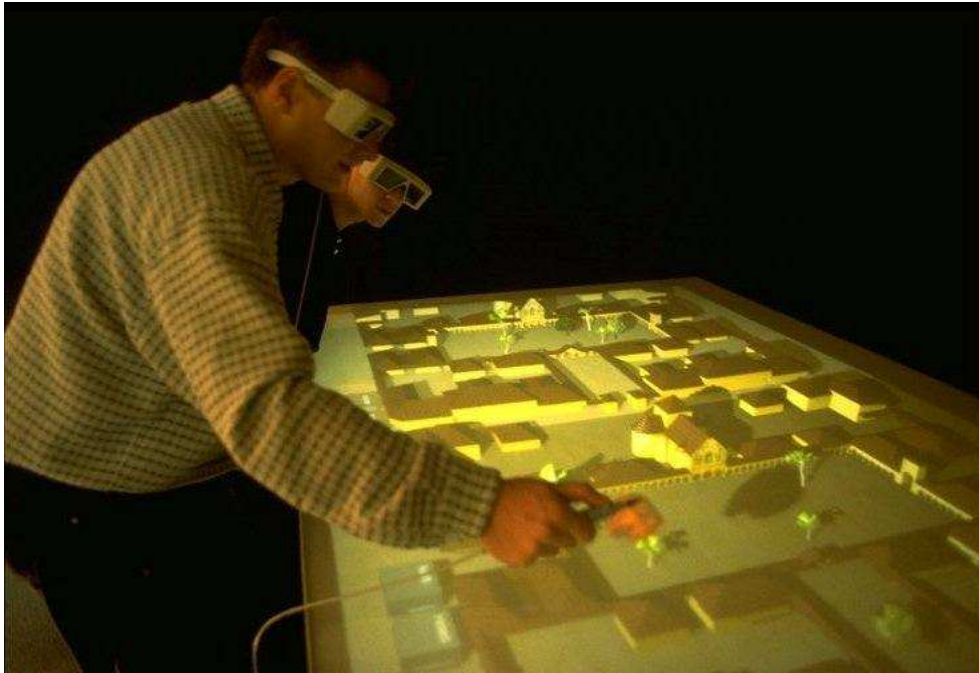


Abb. 5.1: Einsatz der Workbench zur Stadtplanung nach Klein und Merten

Die Entwicklung neuer Hardware und Software für die Visualisierung von Dreidimensionalität schreitet schnell voran und verspricht auch in Zukunft spannende Ideen und Verfahren. Die hier vorgestellten Metaphern, Techniken und Geräte zur Interaktion in dreidimensionalen, virtuellen Umgebungen geben einen Überblick über den aktuellen Stand der Technik und erfolgreich eingesetzte Methoden. Es zeigt sich, dass die Entwicklung bereits weit genug voran geschritten ist, um sinnvoll einsetzbare Produkte zu entwickeln, die dabei die in der Vergangenheit gemachten Fehler nicht mehr haben und so auch von den Benutzern akzeptiert werden können. Sofern bei der Entwicklung des Projektes die wesentlichen Ziele nicht aus dem Auge verloren werden und der Anforderungsrahmen nicht zu weit gesteckt wird, steht einer erfolgreichen Umsetzung aus Sicht der 3D-Interaktion nichts im Wege.

# Literaturverzeichnis

- Deutsches Universalwörterbuch*, (Mannheim, Wien, Zürich: Dudenverlag, 1989).
- Brockhaus. Die Enzyklopädie*, (Leipzig, Mannheim: F. A. Brockhaus, 1998), Band 14.
- Wahrig Fremdwörterlexikon*, (München: dtv, 1999).
- Baccei, Tom, *Das magische Auge*, (Ars edition, 1994).
- Barr, Michael, *Michael Barr's Embedded Systems Glossary*,  
(<http://www.netrino.com/Publications/Glossary/>), 21.09.2004.
- Bowman, Doug A. und Hodges, Larry F., *An Evaluation of Techniques for Grabbing and Manipulation Remote Objects in Immersive Virtual Environments*, (<http://people.cs.vt.edu/~bowman/papers/grab.pdf>), 08.09.2004.
- Center for Pervasive Computing, *Concepts and Technology for the Future*,  
(<http://www.pervasive.dk/>), 21.09.2004.
- Computer Graphics & Software Ergonomie, Carl von Ossietzky  
Universität Oldenburg, *Graffiti: 3D-Transformationen; Translation*,  
(<http://olli.informatik.uni-oldenburg.de/Graffiti3/graffiti/flow7/page3.html>),  
14.09.2004.
- Döllner, Jürgen, Kersting, Oliver und Hinrichs, Klaus, *Programmierbare, interaktive 3D-Karten zur Kommunikation raumbezogener Information*,  
([http://www.hpi.uni-potsdam.de/source/cgs/dhk\\_umwinf00\\_final.pdf](http://www.hpi.uni-potsdam.de/source/cgs/dhk_umwinf00_final.pdf)),  
13.09.2004.
- Geiger, Christian, 'Helft mir, Obi-Wan Kenobi. Stereoskopische Techniken und autostereoskopische Displays', *iX. Magazin für professionelle Informationstechnik*, (5/2004), S. 99–102.
- Hofstadter, Douglas R., *Gödel, Escher, Bach: an Eternal Golden Braid*, (Harvester Press, 1979).
- Holliman, Nick, *3D Display Systems*,  
(<http://www.dur.ac.uk/n.s.holliman/Presentations/3dv3-0.pdf>),  
30.08.2004.
- Jacob, Robert J. K., 'New Human-Computer Interaction Techniques', *Human-Machine Communication for Educational Systems Design*, (1994), S. 131–138.

- Kniss, Joe, Kindlmann, Gordon und Hansen, Charles, *Interactive Volume Rendering Using Multi-Dimensional Transfer Functions and Direct Manipulation Widgets*,  
(<http://www.cs.utah.edu/~jmk/papers/vis01/index.html>), 15.09.2004.
- Kubica, Roland P., *Verfahren zur Darstellung von virtuellen Welten*,  
([http://waste.informatik.hu-berlin.de/rpk/diplomarbeit/download/Darstellung\\_virt.\\_Welten.pdf](http://waste.informatik.hu-berlin.de/rpk/diplomarbeit/download/Darstellung_virt._Welten.pdf)), 28.09.2004.
- Mine, Mark R., Jr., Frederick P. Brooks und Sequin, Carlo H., *Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction*, (<http://www.cs.unc.edu/~mine/papers/minecows.pdf>), 16.09.2004.
- Poupyrev, I. et al., 'The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR', *Proceedings of the ACM Symposium on User Interface Software and Technology* (1996).
- Schmalstieg, Dieter, *3D Interaction*,  
(<http://www.ims.tuwien.ac.at/teaching/mmi/1-3dui.pdf>), 19.09.2004.
- Schmid, Gabriele, *Illusionsräume. Mesdags Panorama, Monets Seerosen, Boissonnets Hologramme und Kirchen der Gebrüder Asam. Konstruktionen und Vermittlungsstrategien*,  
(<http://www.eugwiss.hdk-berlin.de/schmid/diss/home.html>), 31.08.2004.
- Stoakley, R., Conway, M. und Pausch, R., 'Virtual Reality on a WIM: Interactive Worlds in Miniature', *Proceedings of CHI*, (1995), S. 265–272.
- Tufte, E. R., *Visual Design of the User Interface*, (IBM Corporation, Armonk, N. Y., 1989).
- Zachmann, Gabriel, *Virtual Cities*, ([http://web.informatik.uni-bonn.de/II/ag-klein/people/zach/projects/bonn/virtual\\_cities/](http://web.informatik.uni-bonn.de/II/ag-klein/people/zach/projects/bonn/virtual_cities/)), 21.09.2004.