

Intelligente Navigation in 3D-Welten: Zur Rolle graphischer Abstraktion

Antonio Krüger *

Christoph Stahl †

Zusammenfassung

Das Navigieren in komplexen 3D-Welten wird durch zwei Faktoren erheblich erschwert: Zum einen stellt die enorme Datenmenge große Anforderungen an die Hardware der Visualisierungsplattform, zum anderen wirkt der Detailreichtum, der diesen Aufwand erforderlich macht, oft ablenkend und kann im Extremfall dazu führen, daß der Benutzer den Überblick verliert und die Navigation von einem bekannten Startpunkt aus neu starten muß.

Dieser Beitrag diskutiert einen Lösungsvorschlag, der versucht, beiden Problemen gerecht zu werden. Durch die graphische Abstraktion von Modelteilen und der damit einhergehenden Vereinfachung des 3D-Modells werden Rechnerressourcen gespart und dem Navigierenden wird gleichzeitig ein Überblick ermöglicht, da potentiell ablenkende Details unterdrückt werden.

Während der Navigation wird der Detaillierungsgrad automatisch durch das System oder interaktiv durch den Navigierenden angepaßt. Ziel dieser semi-automatischen graphischen Abstraktion ist ein auf das Navigationsziel und auf die Leistung des visualisierenden Rechners möglichst optimal zugeschnittener Abstraktionsgrad.

1 Motivation

Die breite Markteinführung von 3D-Beschleunigerkarten auch für Standard PCs eröffnet vielen Benutzern nun die Möglichkeit, VRML 3D-Welten virtuell durchs Internet zu besuchen. Die Anreicherungen durch Animationen und Verhaltensschemata im Standard VRML 2.0 [VRM96] lassen diese Welten immer attraktiver werden und nach der zunächst nur spielerischen Testphase der letzten Jahre haben sich Anwendungen herauskristallisiert, die auch kommerzielles Interesse hervorrufen. Die Palette reicht von Konferenzen in virtuellen Räumen über virtuelle Kaufhäuser mit vielfältigem Angebot bis zur Simulation komplizierter Abläufe (zum Beispiel Fertigungsprozesse).

Ein Hauptproblem ist und bleibt allerdings die Navigation in komplexen Welten. Durch die steigende Rechenleistung steigt proportional auch das Bedürfnis nach mehr Detailreichtum und Realität. Dies führt oft zu einer sehr schwerfälligen und gehemmten Interaktion mit virtuellen Umgebungen und schreckt viele Benutzer ab. VRML bietet zur Linderung dieses Problems die Möglichkeit, Objekte abhängig von der Entfernung vom Betrachter in verschiedenen *Levels of Detail* (LOD) darzustellen. Dies schafft allerdings nur begrenzte Abhilfe, da Objekte in der Nähe des Betrachters in höchster Detailstufe sichtbar bleiben.

* Graduiertenkolleg Kognitionswissenschaft, Universität des Saarlandes

† Fachbereich Informatik, Universität des Saarlandes

Die Komplexität der 3D-Welten birgt jedoch noch eine weitere Schwierigkeit: Benutzer verlieren oft während der Navigation den Überblick und wissen nicht mehr wo sie sich befinden.

Wie in [BK96] gezeigt, verbessern *graphische Abstraktionen* Echtzeitanimationen, indem sie durch die Berücksichtigung von technischen und kognitiven Ressourcen sowohl die Animationen beschleunigen als auch ihre Aussagekraft verbessern, da der Betrachter nicht von irrelevanten Details abgelenkt wird.

In ähnlicher Weise können *graphische Abstraktionen* dabei helfen, Probleme der Navigation in komplexen 3D-Welten zu entschärfen. Abhängig von der Navigationsaufgabe wird dem Benutzer zunächst eine abstrahierte Version des Modells angeboten. Diese kann er nun durch Interaktion nach seinen Wünschen entlang der Objekthierarchie verfeinern, oder auch wieder abstrakter gestalten, und sich so eine ganz persönliche Ansicht verschaffen. Auf diesem Wege sind alle für ihn relevanten Details sichtbar, ohne daß er den Gesamtüberblick verlieren muß. Eine Kombination von abgespeicherten und bei Bedarf dynamisch zur Laufzeit generierten Abstraktionen erlaubt eine flexible Anpassung des Abstraktionsgrades an die jeweiligen Bedürfnisse und Ziele des Benutzers, der in der so entstandenen *Hypergraphik* (siehe dazu auch [RKSZ94]) frei navigieren kann.

2 Graphische Abstraktion

In diesem Abschnitt wird das Konzept der graphischen Abstraktion im Kontext dieser Arbeit erläutert. Resultierend aus der Tatsache, daß der Begriff „graphische Abstraktion“ im allgemeinen Sprachgebrauch mehrdeutig und vage ist, wird zunächst eine Definition des Begriffs präsentiert, die als Grundlage für ein Modell der graphischen Abstraktion dienen kann. Darauf aufbauend wird ein Vorschlag gemacht, wie graphische Abstraktionen verglichen und bewertet werden können.

Im Folgenden verstehen wir unter einer Abbildung ein Objekt, das mittels einer Transformation aus einem (Welt)-Objekt entsteht. Weiter gehen wir davon aus, daß jede Abbildung einen kommunikativen Akt darstellt und somit auch eine Intention und ein Ziel besitzt.

2.1 Terminologie

Unter einer *graphischen Abstraktion* verstehen wir im Kontext dieser Arbeit eine spezielle Art der Vereinfachung von Objektabbildungen.

Typische Beispiele für die Vereinfachung von Abbildungen sind:

- die Substitution von Farben durch Grauwerte an Teilen des Objekts.
- die Vereinheitlichung von Strichstärken des Objekts.
- die Vereinfachung der Kontour des Objekts.
- das Filtern und Verschmelzen von Objektteilen.
- das Skalieren von Objektteilen.

Für eine ausführlichere Beschreibung sei auf [Krü95] und [Str98] verwiesen.

Die Vereinfachung einer Objektabbildung ist genau dann eine graphische Abstraktion, wenn der Betrachter in der Lage ist, trotz der Vereinfachung der Abbildung, eine Verbindung zu einem ganz speziellen Weltobjekt oder aber einer ganz speziellen Klasse von Weltobjekten herzustellen.

Anhand einer Abbildung eines Romans soll dies verdeutlicht werden. Ein Photo dieses Romans wäre eine graphische Abstraktion des Romans, da der Betrachter eine Zuordnung von Abbildung und Weltobjekt nachvollziehen kann. Die Entfernung von Titel und Autor aus der Abbildung würde diese Zuordnung zerstören, aber trotzdem würde die Abbildung als Darstellung eines beliebigen Buches, also einer Klasse von Weltobjekten, erkannt werden. Wäre dies das Ziel der Vereinfachung, so handelte es sich bei der Abbildung immer noch um eine graphische Abstraktion des Romans. Erst wenn die Vereinfachung so stark wird, daß der Betrachter den Bezug zum Weltobjekt verliert, kann nicht mehr von einer graphischen Abstraktion gesprochen werden. Diese wäre der Fall, wenn der Roman als reines Rechteck dargestellt würde, das vom Betrachter nicht mehr als eine bestimmtes Weltobjekt oder einer Klasse von Weltobjekten erkannt werden könnte.

Weitere Information hingegen könnte diese Abbildung wieder zu einer graphischen Abstraktion machen, zum Beispiel durch einen entsprechenden Untertitel. Dies verdeutlicht die starke Kontextabhängigkeit von graphischen Abstraktionen. Der Betrachter, das kommunikative Ziel und die jeweilige Präsentationssituation sind wesentliche Faktoren, die die Interpretation einer graphischen Abstraktion beeinflussen.

Eine ähnliche Beschreibung graphischer Abstraktion findet sich auch in [Str98]. Hier wird allerdings eine allgemeinere Definition für Abstraktion in der Computervisualisierung angeboten, die auch weitere Informationsträger, wie zum Beispiel Text miteinschließt.

2.2 Abstraktionsgrade

Die Möglichkeit, verschiedene graphische Abstraktionen eines Objektes vergleichen zu können, ist eine weitere Voraussetzung, diese intelligent einsetzen zu können. Allgemeine Aussagen wie „Strichzeichnungen sind einfacher als photographische Abbildungen“ sind natürlich nicht ausreichend. Ein umfassender Ansatz sollte ein *syntaktisches* und ein *semantisches* Maß zur Verfügung stellen, das sowohl die reine Informationsmenge einer Abbildung, als auch die Art der von ihr übermittelten Information berücksichtigt.

Syntaktische Kriterien von Abbildungen sind:

- **Objektkomposition**

Dies bezieht sich auf die Anzahl von Teilobjekten aus denen ein Objekt besteht. Danach wäre die Abbildung \mathcal{A}_1 des Objekts \mathcal{O} einfacher als die Abbildung \mathcal{A}_2 , wenn die Abbildung \mathcal{A}_1 weniger Objektteile von \mathcal{O} darstellt als \mathcal{A}_2 .

- **Objektsilhouette**

die Silhouette in \mathcal{A}_1 ist dann einfacher als in \mathcal{A}_2 , wenn sie weniger konkave Teilelemente, weniger Unstetigkeiten oder weniger Teilsegmente aufweist.

- **Objekteigenschaften**

Darunter fallen zum Beispiel Farbe, Textur und Objektachsen. So wäre $\mathcal{A}1$ einfacher als $\mathcal{A}2$ wenn das Objekt \mathcal{O} mit weniger Farben dargestellt würde.

Diese syntaktischen Kriterien können, wie das Buchbeispiel aus dem vorherigen Abschnitt demonstriert, die Semantik der Abbildung beeinflussen, die darüber entscheidet, ob von einer graphischen Abstraktion gesprochen werden kann oder nicht. Das zugehörige semantische Maß kann mithilfe von Darstellungsklassen definiert werden: sind zwei Abbildungen $\mathcal{G}\mathcal{A}1$ und $\mathcal{G}\mathcal{A}2$ als graphische Abstraktionen von einem Objekt \mathcal{O} gegeben und ist $\mathcal{G}\mathcal{A}1$ identifizierbar als \mathcal{O} und $\mathcal{G}\mathcal{A}2$ nur kategorisierbar als ein beliebiges Objekt aus \mathcal{O} s Kategorie, so ist $\mathcal{G}\mathcal{A}2$ semantisch einfacher als $\mathcal{G}\mathcal{A}1$. Die Kontrolle über syntaktisches und semantisches Maß erlaubt eine sehr individuelle Anpassung der Vereinfachung einer Abbildung an das kommunikative Ziel der Graphik, den Benutzer, die technischen Begebenheiten des Ausgabemediums sowie den situativen Kontext der Abbildung.

3 Verwandte Arbeiten

Die hier von uns präsentierten Ideen entspringen einem interdisziplinären Rahmen, aufgespannt durch verwandte Arbeiten aus der KI, Computergraphik, Psychologie sowie der Kognitionswissenschaft. Wir verweisen allerdings an dieser Stelle für eine ausführlichere Grundlagenbeschreibung auf [Krü95] und fassen hier lediglich die Arbeiten zusammen, die unmittelbar mit den Themen der graphischen Abstraktion und der Navigation in 3D-Welten verbunden sind. Zunächst werden Ansätze vorgestellt, die versuchen, graphische Abstraktion und Vereinfachung rechnergestützt durchzuführen, um danach kurz auf Navigationsmetaphern für 3D-Welten einzugehen.

3.1 Automatische Graphische Abstraktion

Computergestützte Graphikgenerierung wird üblicherweise durch eine *Rendering-Pipeline* [FvDFH96] beschrieben. Ausgehend von einer Datenbank in der geometrische Daten der Welt und zusätzliche Information (wie zum Beispiel Lichtquellen und Texturen) gespeichert sind und einem Betrachterpunkt, wird sequentiell in mehreren Schritten eine zweidimensionale Darstellung erzeugt. Operationen, die zu Vereinfachungen der resultierenden Graphik führen und so auch graphisch abstrahieren, können prinzipiell an mehreren Stufen des Generierungsprozesses ansetzen. Analog zur Rendering-Pipeline entsteht so eine *Abstraktions-Pipeline*, in der schrittweise vereinfacht und abstrahiert wird.

Abbildung 1 skizziert die wichtigsten Ebenen. Auf der *Modellebene* werden die geometrischen Daten direkt manipuliert und vereinfacht. Diese passieren dann die *Generierungsebene*, die der Rendering-Pipeline entspricht. Durch Veränderung von diversen Parametern (zum Beispiel zur Beleuchtung, zur Texturdarstellung oder zur Darstellung von verborgenen Flächen) kann auch auf dieser Ebene Einfluß auf den Detaillierungsgrad des Resultats genommen werden, oder es werden Teile dem Generierungsprozeß hinzugefügt bzw. ersetzt. Letztendlich kann mithilfe von zweidimensionalen pixelbasierten Operatoren auch

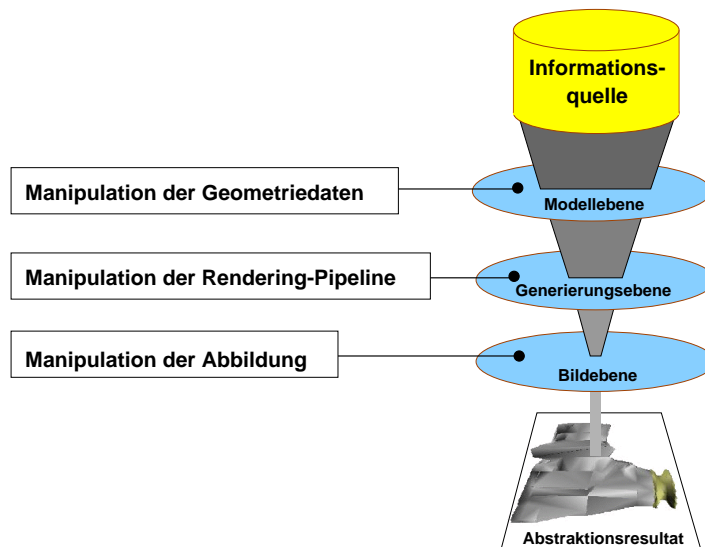


Abbildung 1: Ein Abstraktionsvorgang in drei Ebenen mit zugehörigen Manipulationen

auf der *Bildebene* direkt vereinfacht werden. Bekannte Beispiele hierfür sind Mal- und Zeichenfunktionen gängiger Bildbearbeitungssoftware.

Prinzipiell bedeutet ein Eingriff weiter oben in der Abstraktions-Pipeline größere Kontroll- und Einflußmöglichkeit auf das Endresultat. Dies gilt insbesondere für den automatischen Einsatz dieser Verfahren ohne Interaktion des Benutzers. So lassen sich auch die nachfolgend vorgestellten Verfahren hauptsächlich in den beiden oberen Ebenen der Abstraktions-Pipeline ansiedeln.

3.1.1 Graphische Vereinfachung in der Computergraphik

Die wachsende Komplexität dreidimensionaler Modelle und die damit verbundenen Schwierigkeiten der Echtzeitvisualisierung motiviert seit einiger Zeit eine Reihe von Forschern, nach Lösungsansätzen zu suchen, die eine Reduzierung dieser Daten mit akzeptablen visuellen Qualitätseinbußen versprechen. Eine Möglichkeit, dieses Problem zu entschärfen, ist die Verwendung mehrerer Detaillierungsgrade der zugrundeliegenden Modelldaten. Durch eine Vereinfachung der Polygonmodelle (*Multiresolution Modeling*) kann die Graphikgenerierung beschleunigt werden, da weniger komplexe Geometriedaten einen geringeren Rechenaufwand zur Folge haben. Verfahren zur automatischen Ableitung abstrahierter Modelle wurden in den letzten Jahren zunehmend untersucht [HG94]. Einige Ansätze basieren auf der Verschmelzung von aneinander angrenzenden Knoten oder Kanten. Hoppe [Hop96] schlägt vor, diese Vorgehensweise mithilfe einer Energiefunktion zu realisieren, die ein Maß für Datenreduktion und den daraus resultierenden Fehler repräsentiert. Die Minimierung dieser Funktion führt zu einer Lösung, die einen guten Kompromiß zwischen beiden

Gegenpolen bietet. Das Ergebnis wird als *Progressive Mesh* bezeichnet, und besitzt unter anderem die elegante Eigenschaft einer sehr kompakten Repräsentation.

Eine andere Methode [CVM⁺96] verwendet zwei Hüllen (*Simplification Envelopes*), um eine vereinfachte Repräsentation zu erhalten. Dabei liegt eine innerhalb und eine außerhalb des eigentlichen Objekts. Die Vereinfachung muß zwischen beiden Hüllen liegen und der Abstand zwischen den Hüllen bestimmt den Detaillierungsgrad des Resultats: je größer der Abstand desto einfacher das resultierende Modell.

Ein Verfahren, das die Topologie verändern kann wird von Rossignac und Borrel [RB93] propagiert. Alle Knoten des Modells werden mithilfe eines 3D-Gitters gefiltert. Dazu werden alle Knoten die im gleichen Gittersegment liegen durch einen Repräsentanten ersetzt. Wie auch beim Filtern von Signalen zu beobachten ist, neigt dieses Verfahren zur Aliasierung und bei großen Gitterabständen zu sehr vereinfachten Darstellungen.

Modifikationen des Algorithmus [KS97] parametrisieren den Filtervorgang, so daß der Detaillierungsgrad innerhalb des Gitters variabel wird. Auf diesem Wege gelingt es, gleichzeitig wichtige Details beizubehalten und unwichtige stärker zu filtern.

Ein anderes Teilgebiet der Computergraphik, das den Aspekt der Datenreduktion außer acht läßt, versucht, ausdrucksstarke künstlerisch wirkende Graphiken automatisch zu generieren. Modifizierte Renderer sind zum Beispiel in der Lage, klassische künstlerische Techniken wie Kupferstich oder eigenwillige Pinselführung [LS95] zu imitieren.

In [PS95] wird ein Renderer vorgestellt, der nicht nur Strichzeichnungen erzeugen kann, sondern auch durch die Wahl verschiedener Linienstile dem Resultat eine attentionale Fokusstruktur geben kann.

Die bisher vorgestellten Verfahren zur Vereinfachung von Graphiken konzentrieren sich entweder auf die Datenreduktion oder auf eine hohe Qualität des Resultats. Ein Ansatz, der versucht einen Mittelweg zu gehen, wird in [HL97] vorgeschlagen. Hier ist das Hauptziel zwar wieder die Verringerung des Rechenaufwands, aber dies wird unter der Berücksichtigung der *perzeptuellen Kosten* der vereinfachten Graphik gemacht. Dazu wird eine Hypothese bezüglich des Betrachterfokus aufgestellt und Vereinfachungen entsprechend bewertet: liegen diese im Fokus, so führen sie zu sehr hohen perzeptuellen Kosten und sollten besser vermieden werden. Ein probabilistischer Ansatz hilft zu entscheiden, welche Regionen des Bildes außerhalb des Fokus liegen und daher vereinfacht werden dürfen.

3.1.2 Graphische Abstraktion in der intelligenten Graphikgenerierung

Bis heute gibt es nur einige wenige Systeme, die versuchen graphische Abstraktionen automatisch oder semi-automatisch zu generieren, um so die Intention einer Graphik zu unterstreichen. Der erste Vorschlag in diese Richtung wurde im Rahmen des **APEX**¹ Systems [Fei85] gemacht. Die zugrundeliegende geometrische Domäne besteht aus achsenparallelen Quadern, die durch einen abgeschlossenen binären Verschmelzungsoperator vereinfacht werden können. Dazu berechnet **APEX** im voraus eine Abstraktionshierarchie, aus der zur Laufzeit eine dem Präsentationsziel entsprechende Abstraktion ausgewählt wird.

Das System *PROXIMA* [Krü95] erweiterte die mögliche geometrische Domäne um beliebige Quader, Zylinder und Flächen. Statt sich auf eine unflexible statische Vorausberechnung

¹An Experiment in the Automated Creation of Pictorial Explanations

der Abstraktionen zu verlassen, berechnet *PROXIMA* alle graphischen Abstraktionen in einem Anytime-Verfahren zur Laufzeit. Zusätzlich zum Verschmelzungsoperator werden verschiedene Linienstile zur Darstellung der Objekte verwendet (zum Beispiel das *Stricheln* von Objekten im Hintergrund). Ausgehend von einem Präsentationsziel benötigt das System Wissen über signifikante Attribute der zu abstrahierenden Modelle. Darauf basierend, parametrisieren Abstraktionsregeln dann den Verschmelzungsoperator.

Das Nachfolgesystem *AR \mathcal{P}* erweitert *PROXIMA* um eine explizite Suche und Repräsentation der Aufmerksamkeitsstruktur, sowie eine Evaluierung der gewonnenen Zwischenergebnisse mithilfe einfacher bildverarbeitender Techniken. Zusätzlich wurde die Menge der Abstraktionstechniken um eine modifizierte Version des Filterverfahrens von Rossignac (siehe Abschnitt 3.1.1) erweitert. *AR \mathcal{P}* versucht nicht nur, mithilfe von graphischer Abstraktion die Intention einer Graphik zu verbessern, sondern berücksichtigt ebenfalls die zur Verfügung stehenden Systemressourcen. Eine Koppelung mit dem Echtzeitanimationssystem *CATHI* [BK96] hat gezeigt, daß mithilfe graphischer Abstraktion im gleichen Zug der Inhalt einer Graphik verbessert werden kann und Rechnerressourcen geschont werden können.

3.2 Navigation in 3D-Welten

Die Verwendung der zusätzlichen dritten Dimension zur Visualisierung stellt für eine Reihe von Visualisierungs- und Navigationsaufgaben eine Bereicherung dar. Naturgemäß profitiert besonders die Darstellung von 3D-Modellen in Kombination mit 3D-Ausgabegeräten (zum Beispiel Shutter-Brillen) von dieser Erweiterung², aber auch bei der Darstellung von großen hierarchischen Strukturen erweist sich die dritte Dimension als vorteilhaft. Ein Beispiel hierfür sind die am XEROX Parc entwickelten *ConeTrees* [RCM93].

Die eigentliche Navigation wird bei den meisten 3D-Visualisierungstools durch das Bewegen der Kamera innerhalb der Modellwelt realisiert. Zu diesem Zweck kann oft zwischen verschiedenen Kamerametaphern umgeschaltet werden, die die Menge der zur Verfügung stehenden Freiheitsgrade auf eine sinnvolle Auswahl begrenzen. Zusätzlich besteht oft die Möglichkeit, durch 3D-Widgets oder durch direkte Manipulation der Objekte weitere Navigationshilfen anzusteuern, wie zum Beispiel das Einnehmen von vordefinierten Kamerapositionen oder die Veränderung von Materialeigenschaften, um zum Beispiel eine bestimmte Stelle zu markieren.

Durch die zunehmende Verbreitung des Internet-Standards VRML2.0 [VRM96] finden die entsprechenden Browser (wie zum Beispiel der Cosmoplayer von SGI) vermehrt Verwendung. Wie eingangs erwähnt, wird hier auf verschiedene Auflösungen der 3D-Modelle (LODs) zurückgegriffen, um die Navigation und Interaktion flüssiger und damit angenehmer zu gestalten. Dazu wird mit wachsender Entfernung des Betrachters vom Objekt auf detailärmere Versionen des Modells umgeschaltet. Dieses Konzept ist für den hier vorgestellten Einsatz von graphischen Abstraktionen leider zu unflexibel, da unser Ansatz die Verwendung graphischer Abstraktionen gerade aus der Nähe verlangt.

²Trotzdem gibt es eine ganze Reihe von Suchaufgaben, die besser zweidimensional unterstützt werden. Darunter fällt zum Beispiel nach [LPR97] die Dateiauswahl innerhalb eines Filesystems.

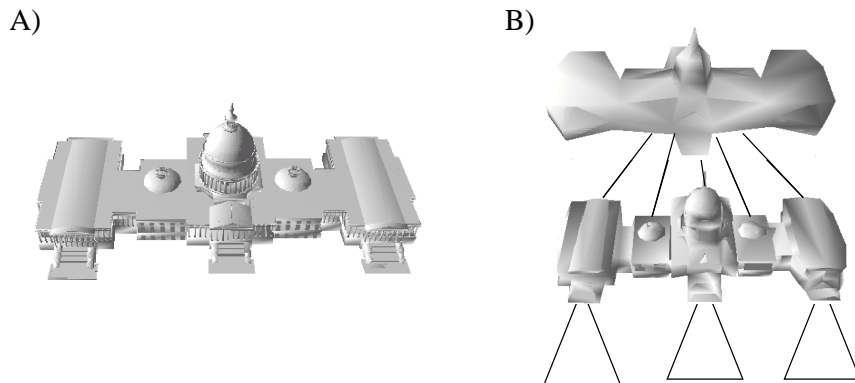


Abbildung 2: Beispiel für eine graphische Abstraktion

4 Navigation mithilfe graphischer Abstraktion

Die Navigation in komplexen virtuellen Welten erfordert ein beträchtliches Maß an *technischen* Ressourcen, um befriedigende Bildschirmraten und Antwortzeiten zu erhalten. Spezielle Graphikhardware mag dieses Problem in bestimmten Fällen entschärfen, spornt aber gleichzeitig dazu an, den Detaillreichtum der Welten zu erhöhen und so den Geschwindigkeitsvorteil wieder zu egalisieren. Diese Detailfülle überlastet oft nicht nur die Rechnerressourcen sondern oft auch die *kognitiven* Ressourcen des Betrachters.

Ein intelligenter Einsatz von graphischer Abstraktion kann hier helfen, beide Ressourcenarten zu entlasten. Insbesondere kann so eine Fokussierung auf relevante Regionen erzielt werden, ohne auf zusätzliche metagraphische Objekte zurückzugreifen. Wir schlagen dazu eine Koppelung üblicher Browser bzw. Visualisierungstools mit einem graphischen Abstraktionssystem vor (in diesem Fall mit dem System *AR \mathcal{P}* , siehe dazu auch Abschnitt 3.1.1).

Ausgangspunkt sind hierarchisch gegliederte 3D-Modelle wie zum Beispiel das Gebäude in Abbildung 2A. Wichtig ist, daß eine rekursive Beschreibung des Modells vorliegt, die das Modell in Teilkomponenten wie zum Beispiel Seitenflügel, Hauptkomplex und diese wieder in Säulen und Mauern zerlegt.

Aus dieser Objekthierarchie wird eine Abstraktionshierarchie von aufsteigenden Abstraktionsgraden der einzelnen Teilobjekte generiert. Der Benutzer kann innerhalb dieser Hierarchie navigieren und die Auflösung wählen, die seinen Bedürfnissen entspricht. Abbildung 2B zeigt die obersten beiden Ebenen einer Abstraktionshierarchie des Beispielmodells. Die symbolische Darstellung von Teilbäumen deutet an, daß die gesamte Hierarchie wesentlich weiter in die Tiefe gehen kann. Die Terminalen entsprechen hier der höchsten Detailstufe des Modells (Abbildung 2A).

Im Folgenden wird exemplarisch an diesem Beispiel erläutert, wie graphische Abstraktion das Navigieren und Explorieren von 3D-Modellen vereinfachen kann. Prinzipiell sind zwei Abstraktionsmodi während der Navigation denkbar. Während die *automatische* Abstrakti-

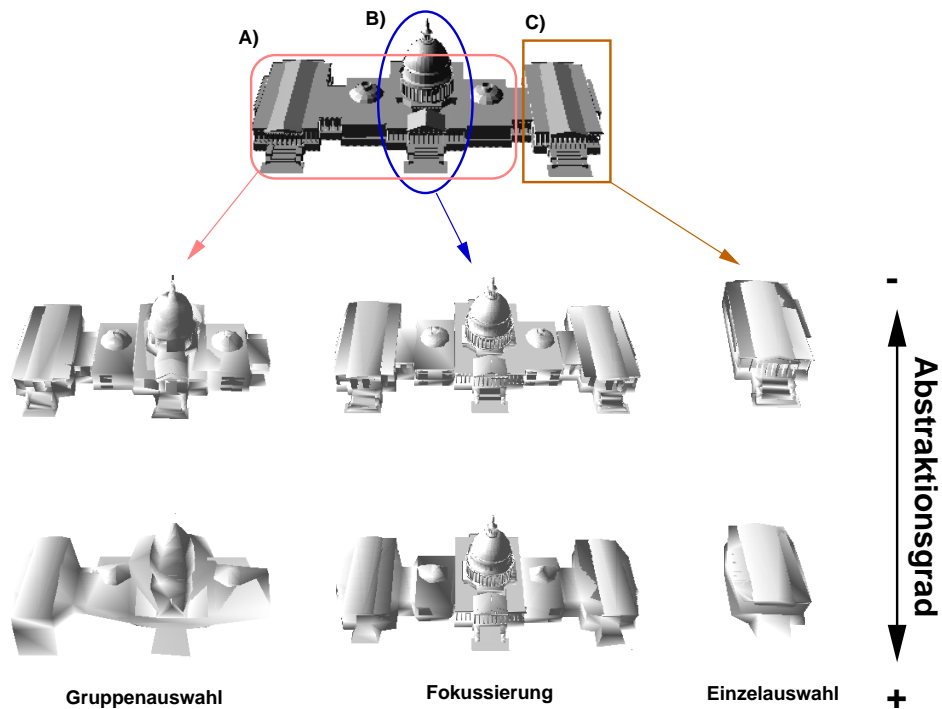


Abbildung 3: Drei Interaktionsmöglichkeiten zur Variation des Abstraktionsgrads

on ohne direkten Einfluß des Benutzers durchgeführt wird, kann dieser in einem zweiten Modus gezielt Abstraktionen von bestimmten Modellteilen in verschiedenen Abstraktionsgraden *anfordern*.

4.1 Automatische Abstraktion

Die automatische Abstraktion ist zunächst der Regelfall. Direkt nach Laden des Modells wird diese aktiviert, um möglichst früh Rechnerressourcen bei der Darstellung zu entlasten, gleichzeitig erhält der Betrachter eine Gesamtübersicht ohne viele Details. Ausgehend von der Objekthierarchie wird bottom-up vorgegangen. Dazu werden alle inneren Knoten der Objekthierarchie durch einen abstrakten Repräsentanten ersetzt, indem jeweils alle Nachfolger dieses Knotens zusammengefaßt und dann gemeinsam abstrahiert werden. Wird dies bis in die oberste Objektebene fortgeführt ergibt sich eine graphische Abstraktion des Gesamtmodells. In Abbildung 2B (oben) wird gezeigt, wie eine Anfangsabstraktion für das Gebäude aus Abbildung 2A aussehen könnte. Der initiale Abstraktionsgrad wird auf Basis von Geometrieeigenschaften, Vorgaben des Benutzers oder Annahmen des Systems über den Benutzer gewählt und zielt im einfachsten Fall darauf, eine Gesamtübersicht über die Geometrie des Modells zu geben. *AR \mathcal{P}* zum Beispiel berücksichtigt zu diesem Zweck

die konvexe Hülle des Objekts und seine Hauptachsen, so daß diese auch durch die graphischen Abstraktion keiner Änderung unterliegen. Möglich ist allerdings auch, daß der Benutzer sein Navigationsziel im voraus präzisiert. Beispielsweise könnte darauf hingewiesen werden, daß die Säulenkonstruktion untersucht werden soll. In der initialen Abstraktion müßten dann die entsprechenden Objekte sichtbar sein und vom Abstraktionsprozeß herausgenommen werden. Weiterführende Hinweise zu einer intelligenten Auswahl von Abstraktionsgraden finden sich in [BK97].

Trotzdem wird es nicht in allen Situationen möglich sein, die Intentionen und Navigationsziele des Benutzers vollständig und richtig zu erfassen. Aus diesem Grund ist das Konzept der interaktiven Veränderung der Abstraktionsgrade sinnvoll und notwendig.

4.2 Abstraktion auf Anfrage

Der Benutzer muß in der Lage sein, interaktiv die Abstraktionsgrade einzelner Objekte oder Objektteile zu variieren. Gerade dieses Mittel erlaubt es, eine persönliche Sicht der 3D-Welt zu erhalten, in der die wichtigen Details detailliert dargestellt und die unwichtigen unterdrückt werden. Wir schlagen dazu, wie in Abbildung 3 zu sehen, drei Interaktionsmöglichkeiten vor. Erstens: die individuelle Wahl eines Abstraktionsgrades für ein einzeln selektiertes Teilobjekt (Abbildung 3C). Zweitens: das Zusammenfassen und anschließende Verschmelzen mehrerer Objekte zu einem neuen abstrakteren Objekt (Abbildung 3A) und schließlich drittens: das Herausnehmen einzelner Objekte einer Gruppe aus dem Abstraktionsprozeß, um eine Fokussierung zu erreichen (Abbildung 3B). Die Auswahl dieser Aktionen erfolgt entweder über verschiedene Arten von Mausclicks oder, falls dies technisch problematisch ist (wie zum Beispiel bei VRML-Browsern) über kleine 3D-Widgets, die über eine virtuelle Konsole zugänglich sind. Eine Erweiterung um zusätzliche virtuelle Menüs würden dem Benutzer erlauben, sein Navigationsziel während der Navigation weiter zu präzisieren.

Die kombinatorische Vielfalt der so potentiell generierbaren Darstellungen eines 3D-Modells verbietet eine vollständige Vorausberechnung der graphischen Abstraktionen. Stattdessen sollte eine Kombination aus vorliegenden häufig verwendeten Abstraktionen und im Bedarfsfall dynamisch zur Laufzeit generierten Abstraktionen den Bedürfnissen des Benutzers entgegenkommen.

Der eleganteste Weg, diesem Ziel einen Schritt näher zu kommen, ist die Verbindung herkömmlicher 3D-Viewer und Browser mit einem graphischen Abstraktionssystem.

4.3 Koppelung von Browser und Abstraktionssystem

Aktuelle 3D-Browser bieten einen großen Funktionsumfang und obwohl oft ein Level-of-Details Konzept vorhanden ist, reichen wie in Abschnitt 3.2 erläutert, die Fähigkeiten nicht aus um die vorgestellten Abstraktionsmöglichkeiten zu realisieren. Eine Lösung dieses Problems ist die Koppelung mit einem graphischen Abstraktionssystem. Dazu eignen sich besonders VRML2.0-Browser (z.B. der Cosmoplayer von SGF³), da als Interface JAVA bzw. VRML-Script angeboten wird. Das Abstraktionssystem kann über eine Socket-

³siehe <http://www.cosmo.sgi.com>

Anbindung mit dem Browser verbunden werden. Dieser sendet Abstraktionsanfragen für einzelne Teilobjekte oder Gruppen von Objekten an das Abstraktionssystem, welches die geometrischen Vereinfachungen berechnet und anschließend an den Browser zurückliefert. Im Browser werden dann die ursprünglichen Modellteile durch die vereinfachten Varianten ersetzt. Ein Cache bzw. History-Mechanismus erlaubt die einfache Wiederverwendung von bereits generierten Abstraktionen, ohne daß diese neu berechnet werden müßten.

5 Ausblick

Der Einsatz graphischer Abstraktion zur Navigation in komplexen Welten ist ein neuartiges Konzept und wertet die Bedeutung graphischer Abstraktionen für 3D-Browser erheblich auf, da weit über das Level-of-Detail Konzept hinausgegangen wird.

Die vorliegende Arbeit versteht sich allerdings als erster Gedankenanstoß, wie eine Integrierung aussehen könnte. Vor allem die vorgeschlagenen Interaktionsmöglichkeiten müssen Gegenstand weiterer Untersuchungen sein. Hier sollten auch dreidimensionale Eingabegeräte Berücksichtigung finden. So sind benutzerdefinierte 3D-Viewports denkbar, die Regionen mit unterschiedlichen Abstraktionsgraden voneinander trennen.

Mit der Realisierung der Anbindung zwischen dem Abstraktionssystem $\mathcal{AR}\mathcal{P}$ und gängigen VRML2.0-Browsern wurde begonnen. Die anschließende Testphase wird weitere Erkenntnisse bringen, in welchen Anwendungsgebieten die Navigation mithilfe graphischer Abstraktion von Vorteil sein kann.

Literatur

- [BK96] Andreas Butz und Antonio Krüger. *Lean Modeling – The intelligent use of geometrical abstraction in 3D animations*. In: Wolfgang Wahlster, Hrsg., ECAI 96 12th European Conference on Artificial Intelligence. John Wiley & Sons, Ltd., 1996.
- [BK97] Andreas Butz und Antonio Krüger. *Zur Auswahl von Abstraktionsgraden*. In: Oliver Deussen und Peter Lorenz, Hrsg., *Simulation und Animation*, S. 147–158, Budapest, 1997. SCS Society for Computer Simulation Europe.
- [CVM⁺96] Jonathan Cohen, Amitabh Varshney, Dinesh Manocha, Greg Turk, Hans Weber, Pakaj Agarwal, Frederick Brooks und William Wright. *Simplification Envelopes*. In: *Proceedings of Siggraph'96*, New Orleans, 1996.
- [Fei85] Steven Feiner. *APEX: An Experiment in the Automated Creation of Pictorial Explanations*. *IEEE Computer Graphics and Applications*, 5(11):117–123, 1985.
- [FvDFH96] James D. Foley, Andries van Dam, Steven K. Feiner und John F. Hughes. *Computer Graphics: Principles and Practice*. The system programming series. Addison-Wesley, second edition in c. Auflage, 1996.

- [HG94] Paul Heckbert und Michael Garland. *Multiresolution Modeling for Fast Rendering*. In: Proceedings of Graphics Interface'94, S. 43–50, 1994.
- [HL97] Eric Horvitz und Jed Lengyel. *Perception, Attention, and Resources: A Decision-Theoretic Approach to Graphics Rendering*. In: Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence (UAI'97), S. 238 – 249, Providence, RI, USA, 1997.
- [Hop96] Hugues Hoppe. *Progressive Meshes*. In: Proceedings of Siggraph'96, New Orleans, 1996.
- [Krü95] Antonio Krüger. *PROXIMA: Ein System zur Generierung graphischer Abstraktionen*. Diplomarbeit, Universität des Saarlandes, Fachbereich für Informatik, Prof. Dr. W. Wahlster, 1995.
- [KS97] Tatjana Klajic und Christoph Stahl. *Ein System zur graphischen Abstraktion von Polygonmodellen*. Fortgeschrittenen Praktikum an der Universität des Saarlandes, 1997. URL: <http://w5.cs.uni-sb.de/bundy/>.
- [LPR97] Ulrich Leiner, Bernhard Preim und Stephan Ressel. *Entwicklung von 3D-Widgets – Überblicksvortrag*. In: Oliver Deussen und Peter Lorenz, Hrsg., Simulation und Animation, S. 147–158, Budapest, 1997. SCS Society for Computer Simulation Europe.
- [LS95] John Lansdown und Simon Schofield. *Expressive Rendering: A Review of Nonphotorealistic Techniques*. IEEE Computer Graphics and Applications, S. 29–37, May 1995.
- [PS95] Bernhard Preim und Thomas Strothotte. *Tuning Rendered Line-Drawings*. In: Winter School of Computergraphics, Plzen, S. 228–238, 1995.
- [RB93] Jarek Rossignac und Paul Borrel. *Multi-Resolution 3D Approximations for Rendering Complex Scenes*. Interner Bericht RC 17697, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, NY 10598, 1993.
- [RCM93] George Robertson, Stuart Card und Jock Mackinlay. *Cone Trees: Animated 3D Visualization of Hierarchical Information*. In: CHI '91, S. 189–202, April 1993.
- [RKSZ94] Thomas Rist, Antonio Krüger, Georg Schneider und Detlev Zimmermann. *AWI: A Workbench for Semi-Automated Illustration Design*. In: Proc. AVI '94. ACM Press, 1994.
- [Str98] Thomas Strothotte, Hrsg. *Abstraction in Interactive Computational Visualization: Exploring Complex Information Spaces*. Springer, 1998. to appear.
- [VRM96] VRML2.0. *Specification by Silicon Graphics Inc.*. <http://vrml.sgi.com/moving-worlds/index.html>, 1996.