

DOES IT HAVE TO BE 3D?

Zum Phänomen der 3D-Technologie in
Computerspielen

VON DENNIS RAY VOLLMER

„The way it's meant to be played“

Werbeslogan des Grafikkartenherstellers Nvidia

I BRILLE – HANDSCHUH – GRAFIKKARTE?

Wer bei dem Stichwort ‚3D-Computerspiele‘ an etwas merkwürdig aussehende User mit Shutterbrille und Datenhandschuh denkt, hat eine vermutlich hinfällige Cyberutopie der körperlichen Vernetzung mit dem Computer vor Augen.¹ Ganz gleich, ob Sie Actionspiele wie etwa *Oblivion* (Bethesda, 2006) (Abb. 1), Strategiespiele a la *Warcraft 3* (Blizzard, 2002) (Abb. 2) oder Adventurespiele wie *Dreamfall* (DTP, 2006) (Abb. 3) bevorzugen, auf dem *Personal Computer* werden die generierten Spielwelten von einer Technologie modelliert und visualisiert, die seit 1995 als 3D-Grafikkarte bekannt ist.



Abb. 1: Screenshot *Oblivion*.

- I Weder das eine noch das andere Interface hat sich auf populärem Gebiet, also jenseits der wissenschaftlichen Anwendung für die Schnittstelle User-Computer durchgesetzt. Stattdessen interagiert der Spieler heute über einen Verbund aus bewegungsaufzeichnender Kamera, Display und Computerprogramm wie etwa bei *Eye Toy* für Sonys Playstation 2. Auch moderne stereoskopische Monitore setzen auf die Kopplung mit einer Kamera, etwa um die Position des Kopfes eines Betrachters bestimmen zu können.



Abb. 2: Screenshot Warcraft 3.



Abb. 3: Screenshot Dreamfall.

Diese Dreidimensionalität von Computerspielen ist jedoch mehr als ein beiläufiges ästhetisches Phänomen. Denn spätestens seit der Entwicklung von Vektorbildschirmen beginnt sich die Computergrafik als reine Ausgabe auf dem Bildschirm von ihrem Datenraum zu entkoppeln.² Im Gegensatz zu optischen Medien wie

2 Die Unverbundenheit von Computergrafik und Datenraum hinsichtlich des Computers muss natürlich in der grundsätzlichen Differenz von Eingabe, Verarbeitung und Ausgabe gesehen werden. Die Entwicklung der Computergrafik ist dennoch in ihrem Beginn eng

der Filmkamera besteht keine Notwendigkeit einer linearperspektivischen Darstellung – warum müssen Spiele also 3D sein?³ Umso interessanter ist daher die Durchsetzung dieser Technologie im Bereich der Spiele und in der Architektur des PCs. Mit eigener *Graphic Processing Unit* (GPU), eigenem Speicher sowie ihren fest installierten Grafikpipelines, die wie Produktionsstraßen die innere Programmebene, die *Game Engine*, der Spielwelten mit ihrer Ausgabe auf dem Display verbinden, bildet die 3D-Grafikkarte eine inzwischen selbstverständliche Hardwareeinheit des PCs.

Der Slogan „The way it’s meant to be played“ des Grafikkartenherstellers Nvidia artikuliert beispielhaft die Diskurse – die Wünsche und Ansprüche von Usern – hinsichtlich Realismus, Freiheit und Linearität im Spiel und auch Wahlmöglichkeiten von Perspektiven, welche sich an die Hardwareentwicklung richten. Seitdem stehen Computerspiele mit der hardwareseitigen Implementation von geometrischen Berechnungen, Abtastverfahren und auch fotorealistischen Effekten im Austausch. Mit 3D-Grafikkarten hat sich eine Tradition des technisch automatisierten Sehens in der Architektur des PCs verdinglicht und nahezu alle Spielgenres auf eine gemeinsame perspektivische Modellierungsgrundlage gestellt. Die Genealogie der 3D-Computerspiele muss daher auch als eine Geschichte ihrer Hardware und der mit ihr verbundenen Diskurse gelesen werden – gerade auch im Hinblick auf eine kritische Theorie.

II ENEMY IN RANGE! PERSPEKTIVISCHE SPIELE MIT LINIE UND FLÄCHE

Und diese Wechselbeziehung zwischen Hardware und Software beginnt vergleichsweise spät, denn Computerspiele, die eine dreidimensionale Darstellung auf den Bildschirm zeichnen, gab es schon, bevor der Hersteller *3dfx* 1995 den ersten so genannten ‚3D-Beschleuniger‘ auf den Markt brachte. Bereits 1980 ‚schoss‘ man etwa in *Ataris Battlezone* (Atari, 1980)⁴ aus der Sicht eines Panzercockpits auf bewegte Panzer.

mit ihrem Display – Kathodenstrahlröhre, Radar, Sketchpad – verbunden, wie unter anderem Claus Pias in seiner Genealogie der Computergrafik beschreibt: Pias, Claus: „Punkt und Linie zum Raster“, in: Brüderlin, Markus (Hrsg.): *Ornament und Abstraktion*, Köln 2001, S. 64-69.

- 3 Eine Fragestellung, die bisher nur selten aufgenommen wurde, siehe etwa Rouse, Richard III: „Do Computer Games Need to be 3D?“, in: *ACM SIGGRAPH Computer Graphics*, Jg. 32, Nr. 2, 1998, S. 64-66.
- 4 *Battlezone* wurde 1980 zunächst als Arcadespiel auf einem Vektordisplay veröffentlicht. Weitere Umsetzungen für verschiedene Spiele-Plattformen etwa für den Atari 2600 oder Commodore C64 folgten 1982. Es ist allerdings nicht das erste Arcadespiel in Vektorgrafik. Am MIT programmierte Steve Russel schon 1961 das berühmte *Spacewar* auf dem *Personal Digital Processor*.

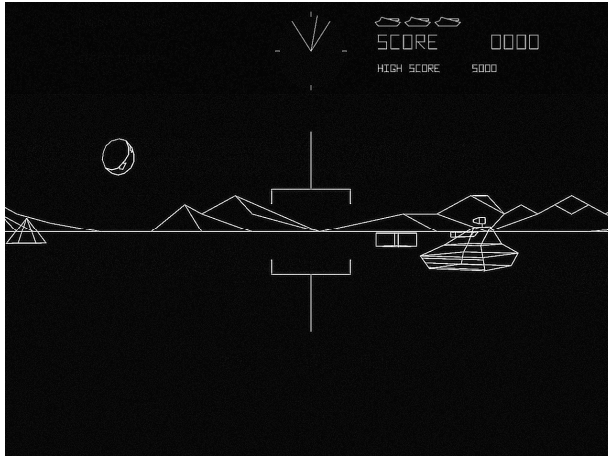


Abb. 4: Screenshot *Battlezone* (Arcade Version): Vektorgrafik.

Battlezone nahm das spätere Genre des *First-Person-Shooter* vorweg: Zentralperspektive und Fadenkreuz auf Höhe des Fluchtpunkts bildeten die Handlungsachse des Users in einer 360-Grad-Sicht mit unbestimmbarer Blickweite.

Fahrzeuge, Hindernisse in Form von Pyramiden und Quadern bestanden aus einfachen Linienmodellen, deren Größe sich proportional zur Annäherung oder Entfernung des Betrachters veränderte und damit perspektivisch glaubhaft blieb. Doch auch ein atmosphärischer Mond und Bergketten im Hintergrund waren in derselben Vektorgrafik modelliert – selbst wenn die Zeichnung der Linien und Umrisse noch ohne Berechnung einer Grafikkarte erzeugt und das Display längst kein Vektorbildschirm mehr, sondern der heimische Monitor oder Fernseher war.

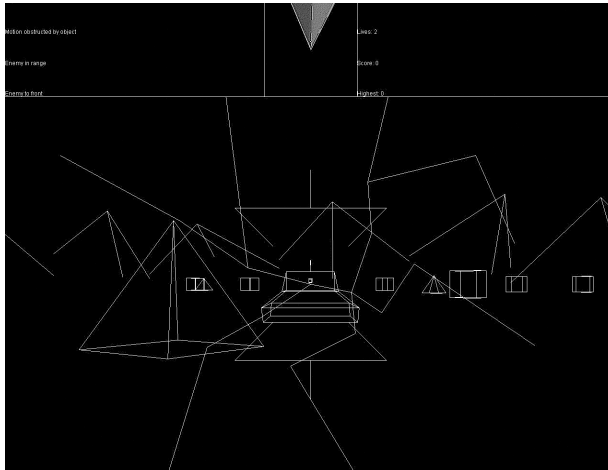


Abb. 5: Screenshot *Battlezone* (Emulation für Java): Der Bildschirm zersplittert beim Abschuss.

1982 flog man mit Microsofts *Flight Simulator 1.0*⁵ über ein Gitternetz aus grünen Linien, welche Landschaft und ebenfalls Hügel am Horizont zeichneten. (Abb. 6) Die Welt bestand hauptsächlich aus Drahtgittermodellen (*Wireframe*). Das Cockpit auf dem Bildschirm informierte über Position, Höhe und Treibstoff. Dennoch konnte der aus Längen- und Breitengitter erzeugte Tiefeneindruck ein Problem darstellen: Die Distanz zum Boden lies sich nämlich visuell nur solange einschätzen, wie man in der Lage war, sie aus Horizontlinie und Vertikalen, die auf einen Fluchtpunkt hin konvergieren, zu konstruieren. Parallel verlaufende Vertikalen kündigten in der Regel den Crash an, der nur durch gelerntes Ablesen der Instrumentenanzeige des simulierten Cockpits zu verhindern war. Geriet aber der Horizont aus dem Blick und man stürzte mit seiner Cessna ab, so zeigten optisch die Risse in der Scheibe an, wann man den Boden erreicht hatte (zumindest in der ersten Auflage des *Simulators 2.0*), denn man befürchtete, dass der Eindruck von ‚zerbrochenen Scheiben‘ zu viel Realismus für potenzielle Cessnakäufer darstellen könnte und entfernte sie in späteren Versionen.

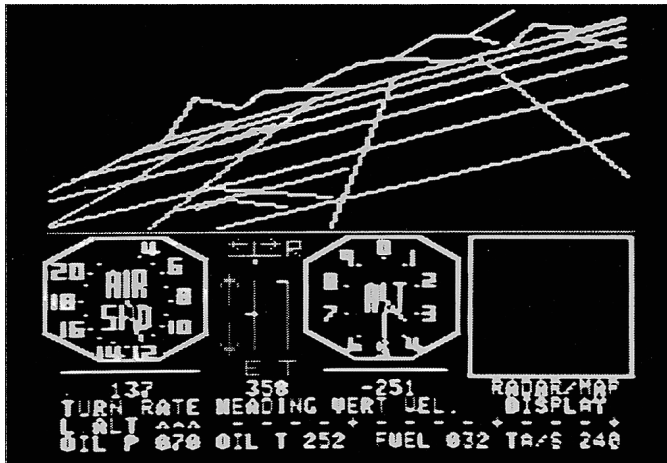


Abb. 6: Screenshot Microsofts *Flightsimulator 1.0*: Flug über das Koordinatengitter.

Vektorgrafik in Kopplung mit dem Radar diente ursprünglich dazu, Koordinaten von Objekten abzulesen und dem Computer über *Lightpen* mitzuteilen, ob es sich um Freund oder Feind handelte – das Projekt *Whirlwind* am MIT setzte sie so zum ersten Mal ein. *Battlezone* hingegen überließ die Freund-Feinderkennung nicht der Wahrnehmung des Spielers: Die Nachricht „Enemy in Range!“ auf dem

5 Der genaue Zeitpunkt der ‚Geburt‘ des *Flight Simulator* variiert von Quelle zu Quelle; so gibt Jos Gruppung das Jahr 1980 an (<http://fshistory.simflight.com/fsh/index.htm>, 10.09.2007), während Microsoft selbst 1982 als Jahr der Veröffentlichung nennt (<http://www.microsoft.com/presspass/press/2004/jan04/01-08GHHomeSoftwarePR.msp>, 10.09.2007).

Bildschirm instruierte ihn, dass sich der Feind in Abschussweite befand.⁶ Auch hier zeigte die Einbindung menschlicher Wahrnehmung ihre Tücken. Sobald sich die Linien der verschiedenen transparenten, texturlosen Drahtmodelle aus der Sicht des Cockpits perspektivisch überschneiden, konnte das Auge kaum klären, ob der Feind nun vor oder hinter einem Objekt stand. Allenfalls der Druck auf die Feuertaste verschaffte Eindeutigkeit in dieser Frage, denn das Wissen um die genauen Aufenthaltsorte der Modelle waren dem Programm vorbehalten, das so bestimmen konnte, ob Figuren und Geschosse miteinander kollidierten oder nicht.

Die ‚Battlezone‘ war zudem ein mathematisch unendlicher (Spiel-)Raum, dessen Bergkettensilhouette am Horizont eine allenfalls symbolische Grenze bildete und dem Spieler versicherte, dass irgendwo in der schwarzen Leere des Bildschirms ein Ende sein würde – das allerdings nicht zu erreichen war.⁷ Demgegenüber zeigte ein regelmäßiges, über einen zunächst planen Boden⁸ gelegtes Koordinatengitter in Microsofts *Flight Simulator* die zurückgelegte Wegstrecke an. Und blieb damit dem Grundgedanken des einstigen Zusammenschlusses aus Vektorgrafik, -bildschirm und Radar treu, welches Grafik als Ergebnis von Messungen und damit als Instruktion für den Betrachter einsetzte.⁹ So konnte der ‚Pilot‘ über die durch Land- bzw. Weltkarten gestützte Computergrafik, die tatsächliche Dauer, Entfernung und genaue Wegstrecke etwa der Flugstrecke von Paris nach New York mit jeweiligem Flugzeugtyp nachvollziehen. Auch Flüge an berühmte Schauplätze, etwa über die *Statue of Liberty* oder unter der *Golden Gate Bridge* hindurch, zählten zu den Attraktionen neuerer Versionen.¹⁰

Beide Spiele nutzten Grafik auch zur Erzeugung atmosphärischer Eindrücke wie Mond und Berge. Der wirksame mathematisch-geometrische Realismus wurde nach und nach ergänzt um einen fotorealistischen. Dank *Color Graphics Adapter*, dem 1981 von IBM geschaffenen Grafikkartenstandard, wandelte sich die Ansicht bald in eine vierfarbige Landschaft mit einer Auflösung von 320x200 Pixeln. Von Version zu Version verfeinerte sich das Wegenetz aus Längen und Breiten und wurde mit flächigen Texturen überzogen, die zwar eine fotorealistische Ansicht auf das nachgebildete Objekt lieferte, eine Orientierung am Gitter jedoch nicht mehr möglich machte.

-
- 6 Das kleine Radar im oberen Teil des Sichtfensters sollte dieser ehemaligen Nutzung von Vektorgrafiken noch Tribut.
 - 7 Seitdem sind in zahllosen Spielen Bergketten meist ein untrügliches Zeichen für die Grenze eines Spielfeldes.
 - 8 Spätere Versionen, die mit der Weltkarte operierten, gingen entsprechend von der Weltkugel aus.
 - 9 So bezeichnet Pias (wie Anm. 2), S. 65 den Zusammenschluss aus Vektorgrafik, -bildschirm und Radar gegenüber heutiger Computergrafik als ein eigenes Dispositiv, dessen Muster verstanden werden muss als eine „Notation, die zur Aufführung von Bewegungen instruiert“.
 - 10 Inklusive Crash-Landung in dieselbigen. In den neuen Versionen wird allerdings verhindert, dass man etwa in die *Twin Towers* von New York fliegt.

III DER FOTOREALISTISCHE RAUM: SPRITES UND BITMAPS

Vielleicht hätte das ‚Gitternetz‘ des Rasterbildschirms die ‚Entdecker‘ der Zentralperspektive, Brunelleschi und Alberti,¹¹ in größere Aufregung versetzt als die etwas grobe Geometrie aus Vektoren. Dabei lieferten Renaissancemalerei und Fotografie das Wissen und die Algorithmen, nach denen perspektivische Ansichten und das reflektierende Licht auf die zweidimensionale Fläche des Displays zu übertragen seien. Es bedurfte dazu weder der Modellierung virtueller noch der Vermessung oder Aufzeichnung realer Objekte: Rasterbildschirme und pixelorientierte Grafikauflösungen bildeten, Punkt für Punkt, eine detaillierte Matrix, anhand derer eine Grafik aufgebaut und ausgegeben werden konnte.

So erzählte etwa *Myst* (Brøderbund, 1993) seine Geschichte mit faszinierenden, aber starren linearperspektivischen Bildern, die visuell an die Konvention der Fotografie anschlossen. Ihre Interaktivität bestand darin, das entscheidende Bildelement zu finden und anzuklicken, mit dem die Erzählung – und damit das nächste Bild – fortgesetzt wurde.



Abb. 7: Screenshot *Myst*: Faszinierender Fotorealismus.

Day of the Tentacle (Lucas Arts, 1992) nutzte dagegen eine Technik aus *Bitmaps* und *Sprites*, um mittels perspektivischen, comichaften Hintergrundzeichnungen und überlagernden Figuren einen Tiefeneindruck zu erzeugen. Über *Sprites*, d.h.

11 Bekanntermaßen gilt Brunelleschi als ‚Vater‘ der Linearperspektive (1425). Die Idee regelhafter, homogener Darstellung nach geometrisch-optischen Gesetzen findet sich bereits im Traktat „Opus Majus“ (1260) des Franziskanermönchs Roger Bacon. Alberti schloss später an die Ergebnisse Brunelleschis und der mittelalterlichen *Perspectiva* an, und führte sie weiter als künstlerische Methode aus, verband sie aber auch zu einem Darstellungskode sittlicher Gesten und kohärenter Ordnung. Edgerton, Samuel Y.: *Die Entdeckung der Perspektive*, München 2002, S. 9 spricht daher von der „Wiederentdeckung der Linearperspektive“.

grafische Objekte, die sich in einen bestehenden Bildschirminhalt einfügen lassen, lies sich ebenfalls ein Bewegungseindruck von Figuren erzeugen.



Abb. 8: Screenshot *Day of the Tentacle*: Perspektivischer Hintergrund und überlagernde Sprites täuschen Tiefe vor.

Dies erforderte allerdings viele unterschiedliche Varianten einer Figur, die – ähnlich wie beim filmischen *Stop Motion* Verfahren – in eine glaubwürdige Bewegungssequenz gebracht werden mussten. Und ebenso erforderte dies die Einrichtung eines separaten Bildspeichers auf der Grafikkarte: Im *Framebuffer* wird dabei ein für die Ausgabe fertiges Bild abgelegt, während ein zweites berechnet wird, sodass ein flüssiger Bildaufbau erzielt werden kann. Von 1981 bis 1995 etablierten sich verschiedene Grafikkartenstandards bis zum *Video Graphics Array* (VGA) mit wachsender Auflösung und Grafikspeicher, die auf den Bereich Grafikanwendung und natürlich Computerspiele spezialisiert waren.

Diese auf Bildschirmauflösung basierenden Verfahren setzten perspektivisch keine Grenzen: Sie boten Ich-Perspektive, Draufsicht wie in *Sim City I* (Brøderbund, 1989) oder isometrische Ansichten wie *Diablo* (Blizzard, 1997). Gegenüber Vektor- oder Wireframegrafiken stellten sie jedoch eine Zäsur dar: Deutlicher müssen die durch *Sprites* und Bildpunktbearbeitung entstehenden Grafiken als ‚Bilder‘ verstanden werden, die als reine, beliebige Ausgabe mit ihrem Datenraum zunehmend unverbunden sind. Und wie Albertis Gitternetz unter dem fertigen Bild verschwindet, verbirgt auch diese Form der Computergrafik die Schemata ihrer Entstehung – die unterschwellige, grundsätzliche Unverbundenheit der benachbarten Bildschirmpunkte.¹² Darstellungskonventionen etwa des Films oder des Comics bildeten eine semantische Klammer, anhand derer ein User die Ob-

12 Weshalb etwa Weibel, Peter: „Zur Geschichte und Ästhetik der digitalen Kunst“ (http://www.aec.at/de/archives/festival_archive/festival_catalogs/festival_artikel.asp?ProjectID=9369, 10.09.2007) die Ursprünge des digitalen Bildes in den Kunstbewegungen des 20. Jahrhunderts ansetzt, „welche die Ruptur mit der klassischen Bildauffassung vorangetrieben haben“.

jekte der Bildschirmoberfläche erkennt und versteht – und seine Handlungen ausführen kann. So täuschten diese Computergrafiken das Auge „mit dem Schein oder Bild eines Bildes, während die Pixelmenge aufgrund ihrer durchgängigen Adressierbarkeit in Wahrheit die Struktur eines Textes aus lauter Einzelbuchstaben aufweist“¹³.

IV PSEUDO 3D- UND POLYGONWELTEN

1992 machte *Castle Wolfenstein 3D* (Id, 1992) eine Form der Spielansicht berühmt, die wohl am stärksten das Verständnis von 3D-Computerspielen geprägt hat: die *First-Person-* oder *Ego-Perspektive*. Ausgerichtet von einem Distanzpunkt in Augenhöhe der Spielfigur erstreckt sich eine zentralperspektivische Sicht, welche die Welt und ihre Objekte in eine konsistente Anordnung strukturiert. Im Vergleich zu heutigen Spielen müssen *Wolfenstein 3D* und besonders das spätere *Doom* (Id, 1993) als *Pseudo-3D* oder *2,5-D* Konzepte (Abb. 9) bezeichnet werden, denn sie erzeugten ihre perspektivischen Oberflächen auf der Basis von flachen Polygonfiguren und Texturen, die auch intern im Computer als Flächen und nicht als Räume abgelegt waren. Zur perspektivisch korrekten Wiedergabe einer Szene nutzte *Doom* das *Binary Space Partitioning*, d.h., es teilte sie in mehrere Abschnitte ein und erstellte eine Reihenfolge für die darzustellenden Elemente. Der so genannte *Painter's Algorithm* zeichnete anhand dieser Datenstruktur die Szene von hinten nach vorne und bestimmte so, welche Objekte für das Betrachterauge sichtbar und welche zu verdecken waren, ohne dass für die einzelnen Bildobjekte eine Raumkoordinate berechnet werden musste.



Abb. 9: Screenshot *Doom*: Pseudo 3D. Der Raum unterhalb der Brücke ist eine unbegehbare Textur.

13 Kittler, Friedrich: „Computergrafik. Eine halbtechnische Einführung“, in: Wolf, Herta (Hrsg.): *Paradigma Fotografie. Fotokritik am Ende des fotografischen Zeitalters. Band 1*, Frankfurt a.M. 2002, S. 178-194, hier S. 179.

Descent (Interplay, 1994) war eines der ersten *Games*, das tatsächlich dreidimensionale Polygonfiguren nutzte, welche noch rein von der Software berechnet wurden.



Abb. 10: Screenshot *Descent*: Eines der ersten *Games* aus Dreiecken.

1995 erschien die *NVI* des Herstellers Nvidia und eine der ersten 3D-Beschleunigerkarten, welche das Dreieck als einfachste Form der euklidischen Geometrie zum Aufbau von Computerspielwelten *hardwired* verfügbar machte und dazu in der Lage war, Bézierkurven eigenständig zu berechnen. Die erste spezialisierte 3D-Grafikkarte, die *Voodoo Graphics* von 3dfx, folgte 1996. Wie die *NVI* verfügte sie über eine eigene *Graphic Processing Unit* (GPU), die Algorithmen für Geometrie und fotorealistische Darstellungen implementiert hatte. Diese Algorithmen wurden zum Teil bereits in den 1960 und 70er Jahren u.a. von Sutherland, Henri Gouraud¹⁴ und John Warnock¹⁵ entwickelt. Die Karte berechnete und speicherte Texturen¹⁶ (*Texture Mapping*), schattierte diese (*Gouraud Shading*), entfernte verdeckte Oberflächen anhand ihrer Raumkoordinaten (*Hidden Surface Removal*), bestimmte die Transparenz von einander überschneidenden Texturen (*Alpha Blending*), Farbverläufe (*Dithering*) und sie glättete so genannte ‚Treppeneffekte‘ (*Anti-*

-
- 14 Der Erfinder des *Gouraud Shading*, eines Algorithmus zur Schattierung von Polygonoberflächen.
- 15 Der nicht nur Mitbegründer von *Adobe* ist, sondern auch den gebräuchlichsten Algorithmus für das Herausrechnen versteckter Flächen in einer Computergrafik (*Hidden Surface Removal*) schrieb.
- 16 Statt diese, wie Grafikkarten zuvor, ‚nur‘ im Speicher abzulegen.

aliasing) bei der Ausgabe auf dem Monitor. Als zusätzliche Steckkarte musste sie allerdings mit einer herkömmlichen VGA-Grafikkarte verbunden werden, um auch den ‚normalen‘ Betrieb zu ermöglichen.

Diese ‚neue‘ Generation von Grafikkarten führte abermals einen Paradigmenwechsel in der Darstellung herbei: Sie verliehen der geometrischen Polygonmodellierung (und auch bestimmten Effekten des Fotorealismus) eine feste Rechenstütze. Spiele wie *Tomb Raider* (Eidos, 1996) ergänzten die Ansichten um die so genannte Schulterperspektive oder *Third-Person-Perspective*.

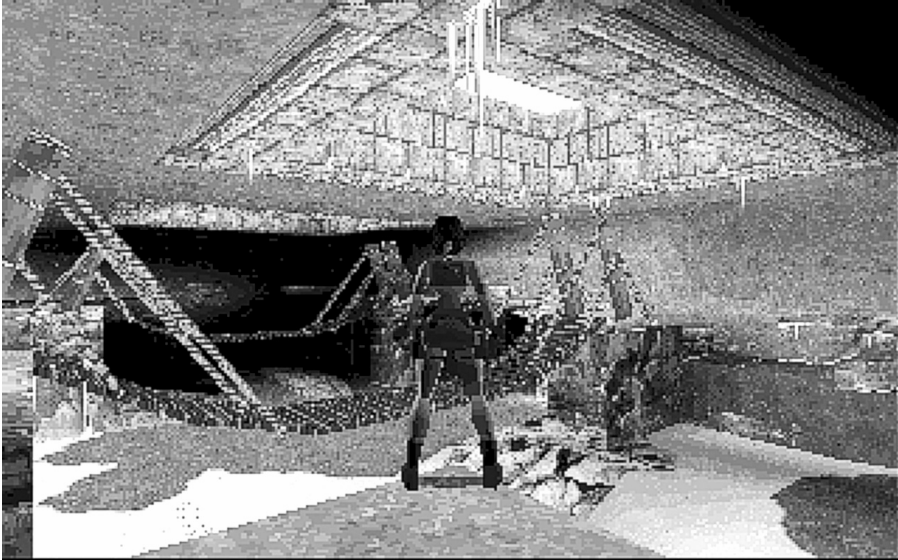


Abb. 11: Screenshot *Tomb Raider*: Berühmt für die ‚Schulterperspektive‘.

Die Frage des Betrachterstandpunkts ist seitdem eine Frage der Präferenz des Users, insofern die Polygonmodellierung der Spielwelt theoretisch gestattet, ebenso aus der Ich-Perspektive zu handeln wie aus der typisch isometrischen Sicht etwa der Strategiespiele.¹⁷

17 Entsprechend flexibel spielt man z.B. *Spellforce 2* (Jowood, 2005) aus der Ego-, Schulter- oder Vogelperspektive.



Abb. 12: Screenshot Warcraft 3: Strategischer Blick.



Abb. 13: Screenshot Warcraft 3: Blick aus der Third-Person-Perspective.

Die Auswahl der Betrachterposition ist damit ebenso wichtig wie die Steuerung der Spielfigur.¹⁸ Alternative, nichtgeometrische 3D-Varianten, wie etwa die *Voxel-*

18 Manovich, Lev: *The Language of New Media*, Cambridge 2001, S. xvi und S. 273ff. vergleicht diese gesteigerte Mobilität des Betrachterauges mit Vertovs ‚KinoGlaz‘-Theorie (1924) und der ‚New Vision‘ Bewegung der 1920er Jahre. Der Spieler im virtuellen Raum ist ein Flaneur (Benjamin).

Technik¹⁹, sind bis heute dagegen selten, da sie von Grafikkarten nicht unterstützt werden und so deutlich auf die Leistungsstärke der CPU und des verfügbaren Hauptspeichers angewiesen sind.



Abb. 14: Screenshot Delta Force 2 (Novalogic, 1999) nutzt volumetrische Elemente (Voxel) zur Darstellung.

V 3D-TECHNIK

Betrachtet man den technischen Aufbau der 3D-Grafikerzeugung, lassen sich drei Stufen unterscheiden: Applikation, Geometrie sowie *Rendering*. Die erste Stufe der Applikation legt allgemeine Daten für alle Szenen und Ereignisse fest wie etwa die Eigenschaften der Spielwelt und der dargestellten Objekte (ob diese beweglich sind oder fixiert), ob und wann bestimmte Ereignisse eintreffen und über welche Möglichkeiten und Eingaben ein Benutzer verfügt.

Die zweite Stufe der Geometrie erstellt die Polygonmodelle für eine Szene, baut die Szene maßstabgetreu auf, setzt einen Betrachterwinkel und berechnet die Beleuchtung. Die Scheitelpunkte oder *Vertices* der Polygone beinhalten verschiedene Informationen über ihre Position innerhalb des Koordinatensystems, die Farbe, die Intensität der Beleuchtung oder Angaben über die Textur. Die Größe der zu erzeugenden Welt und ihrer Objekte wird zunächst voneinander getrennt auf Idealmodelle hin berechnet und im Anschluss proportional auf eine Betrachterposition hin transformiert. Diese Transformationsphase richtet Welt und Objekte mittels einer Sehpyramide auf einen Blickwinkel hin aus und passt sie

¹⁹ Voxel = *Volumetric Element* – ein Verfahren zu Darstellung dreidimensionaler Objekte, das nicht von einem Polygongitter, sondern von aneinander gefügten volumetrischen Punkten ausgeht.

in Form einer Szene maßstabgetreu an. Dabei bestimmt die Länge der Pyramide die gewünschte Sichtweite und die aktuell zu berechnende Ausschnittsgröße. Alle Polygone, die sich außerhalb dieses Bereiches befinden, werden herausgeschnitten und die Szene erneut berechnet. Anschließend wird der Ausschnitt mit Hilfe von *Raytracing* oder – weitaus gebräuchlicher – *Lightmaps*²⁰ beleuchtet. Im anschließenden *Clipping*-Prozess werden solche Bereiche eines Objekts aus den weiteren Rechenverfahren ausgeklammert, die aus der gewählten Betrachterperspektive heraus nicht sichtbar bzw. verdeckt sind.

Die letzte Stufe der Pipeline stellt das *Rendering* dar und meint die Umwandlung der dreidimensionalen Szene in eine zweidimensionale Auflösung aus Bildpunkten. Dies umfasst gleich mehrere Prozesse: die Erkennung verdeckter Pixel, die Texturierung, die Schattierung bzw. Färbung der einzelnen Pixel sowie das ‚Glätten‘ von so genannten Treppeneffekten.²¹ Die Objekte der Szene werden also in zweidimensionale Dreiecke zerlegt und in ihren Konturen gezeichnet. Spezielle Algorithmen übersetzen geometrische Figuren in das Punkt-für-Punkt-Raster der zweidimensionalen Darstellung, so etwa der Bresenham-Algorithmus²² für Linien und Kreise oder der De Casteljau-Algorithmus²³ für Bézierkurven. Anhand der in den *Vertices* enthaltenen Tiefeninformationen lässt sich die Reihenfolge bei überlagernden Flächen bestimmen. Die Texturkoordinaten der *Vertices* bestimmen, in welcher Position eine Textur über ein Dreieck gelegt werden muss. Mittels verschiedener Filtermethoden wird für die Umrechnung einer Textur in Pixel ein durchschnittlicher Farbwert ermittelt. Jedes Pixel enthält in der folgenden Rasterisierungsphase auch eine Texturkoordinate sowie einen Farbwert. Nach dieser Rasterung wird die 3D-Szene schließlich als zweidimensionale Fläche auf den Bildschirm ausgegeben.

Neue Verfahren, die auf der Ebene eines jeden einzelnen Scheitelpunkts ansetzen, wie etwa der *Vertex Shader* der *Geforce3*-Karten oder die Berechnung von physikalisch-korrekten Bewegungsabläufen, haben diese Produktionskette inzwischen erweitert und setzen die Implementierung von Geometrie und Fotorealismus in Hardware fort. Prinzipiell kann nahezu jeder Bereich innerhalb des Produktionsprozesses weiter segmentiert werden.

20 D.h. Texturen, die Informationen über Lichtquelle, -intensität, -farbe beinhalten und zusätzlich über die für das Objekt vorgesehene Textur gelegt werden. Das ‚realistischere‘ Verfahren der *Radiosity*, d.h., der Berücksichtigung der gegenseitigen Reflexion von Objekten bei der Beleuchtung ist in der Regel jedoch nicht einbezogen.

21 Durch farbliche Annäherung der Pixel, die an die zu ‚glättende‘ Linie oder Fläche angrenzen. Neuste Generationen von Grafikkarten verfügen inzwischen über eine gesonderte *Antialiasing*-Einheit, welche auch Computergrafiken älterer Spiele nachbearbeitet, die Kantenglättung nicht vorgesehen haben.

22 Jack E. Bresenham entwickelte in den 1960er Jahren für IBM Algorithmen zur Darstellung von Linien und Kreisen auf einem Computerdisplay.

23 Paul de Casteljau, Mathematiker und Ingenieur bei Citroën, entwickelte diese Annäherungsverfahren von Polygonketten an eine Bezierkurve.

VI DIE SYMBOLISCHE MASCHINE: 3D UND DATENSTRUKTUR

3-D Technik hat Computergames erneuert, jedoch anders, als die von Shutterbrille und Datenhandschuh geprägten Vorstellungen einer haptisch erfahrbaren virtuellen Welt hätten vermuten lassen: Sie bietet eine fotorealistische Spielwelt der optischen Bewegungsfreiheit indem sie neben der horizontalen und vertikalen Bewegung auch solche in die Bildschirmtiefe ermöglicht.

Wo zuvor Beobachtungs- und Handlungsperspektiven in der Regel genretypisch in Ich- (z.B. Action), Erzähler- (z.B. Adventure) und isometrischer Perspektive (Strategie) unterschieden waren, ist es dem Nutzer nun möglich, seine Blick- und Handlungsachse stets in die spielerisch optimale Position zu stellen. 3-D Technik richtet die Welt immer neu auf diese Spielbedürfnisse hin aus.

Und sie zwingt ihm diese Entscheidung auch auf: Das Ringen mit der „Kamera“ um den besten Aktionswinkel ist ein Teil des Spielens geworden, genauso wie das Flanieren durch einen fotorealistischen, detailreichen virtuellen Raum, das sich um die „eigentliche“ Spielanforderungen oder Narration nicht kümmert. Der dreidimensionale, kontinuierliche Spielraum stülpt sich dem User über, der nun „realistische“ Wegstrecken ablaufen, -fahren oder -fliegen muss - und keinen „Teleporter“ nehmen kann.

Die skizzierte Entwicklung von Grafik und Hardware sollte zeigen, dass die ‚Wende‘ zur 3D-Modellierung mittels Polygonen keineswegs kontinuierlich oder selbstverständlich ist. Weder Computerspiele bzw. Software noch Hardware haben alternative Verfahren völlig ausgeschlossen, sondern bestimmte Methoden der Erzeugung fotorealistischer Effekte und Berechnung geometrischer Modellierung synthetisiert, die sich dann weiter zu populären Verfahren verdichtet haben. Handelt es sich bei dieser Synthese von Fotorealismus und 3D-Modellierung vielleicht einfach um Kontingenz? Ich denke nicht. Vielmehr scheinen die Paradigmenwechsel in der Grafik mit sich verändernden Auffassungen des Computers selbst verbunden zu sein. Hartmut Winkler verdeutlicht eine solche ‚Genealogie‘ eines sich wandelnden Computerverständnisses und seiner Diskurse: vom Computer als Rechenmaschine und Steuerungsinstrument in den 1960er und 1970er Jahren, zum Computer als Multimediemaschine in den 1980er bis 1990er Jahren bis hin zum Computer als Medium.²⁴ Die von Winkler vorgestellten Betrachtungsweisen gehen mit bestimmten Zeiträumen einher, in denen sich auch Grafik und Hardware veränderten: Die Ära der Rechenmaschine nutzte Vektorbildschirme und Wireframe-Grafik, die Multimedia-Phase Rasterbildschirme und pixelgenauen Fotorealismus.

Ausgehend von dieser Überlegung, scheint das ‚neue‘ Verständnis des Computers genau in dieser Fähigkeit der Konstruktion und Simulation von Modellen zu

24 Vgl. Winkler, Hartmut: „Medium Computer. Zehn populäre Thesen zum Thema und warum sie möglicherweise falsch sind“, in: Engell, Lorenz/Neitzel, Britta (Hrsg.): *Das Gesicht der Welt. Medien in der digitalen Kultur*, München 2004, S. 203-213.

bestehen.²⁵ Oder mit Winkler formuliert, ist der Computer ein Medium *und* eine symbolische Maschine, die „in der Lage ist, Signifikanten programmgesteuert automatisch zu prozessieren“²⁶ – und damit sowohl Instruktion, Vor-Schrift im Sinne Flussers ist, als auch Probehandeln ermöglicht. Zwar bezieht sich Winkler nicht direkt auf die Entwicklung von 3D-Technologie, sondern auf den medienwissenschaftlichen Diskurs, jedoch haben gerade 3D-Grafikkarten das Ausagieren und Permutieren von bereits zu Zeichen gewordenen Prozessen, d.h. von Modellen und – tiefer liegend – von Algorithmen, zum präferierten Gebiet ihrer Anwendungen gemacht.²⁷ Und insofern ihren vermeintlichen ‚Bildern‘ nun die Macht eines mathematischen Modells – einer operational modellierten ‚Realität‘, die sowohl zur Nachbildung als auch zur Vorlage etwa von zukünftigen realen Objekten dienen kann – unterliegt, kann 3D-Technologie nicht einfach als Fortsetzung einer gesteigerten Bildwerdung der Oberflächen aufgefasst werden.

Wenn die Medienwissenschaft die Entwicklung der Computergrafik und die Emergenz der *Games* etwas ungnädig als „Verbergung von Hardware und Software durch die Phantasmagorie von buntem Plastik und geschütztem Code“²⁸ verstanden hat, also auf die unterschwellig entstehende Zäsur zwischen Datenraum und Ausgabe kritisch verwies, so rückt die symbolische Maschine beide Ebenen näher zusammen, indem sie 3D-Modellierung als Modus des einen und Linearperspektive als Modus des anderen in ein ineinander übersetzbares Verhältnis bringt.

Allerdings ist dieses Verhältnis regelhafte Übersetzung von Modellen in Oberflächen immer noch weit entfernt von einer „Verwandlung von Datenstrukturen in eine sinnlich wahrnehmbare Spiegelwelt“²⁹, die Astrid Deuber-Mankowsky am Beispiel virtueller Welten beschreibt. Zwar können Daten als über die Sinne erfahrbare Zeichen für den User ebenso ‚real‘ werden wie das, was er aus Sinneseindrücken als Realität erfährt, die fotorealistischen Polygonwelten täuschen jedoch nicht darüber hinweg, dass sie an ihrem Rahmen, dem Display enden. Auch die Technik- und Werbediskurse um Effekte und Bildschirmauflösungen betonen diese symbolische Differenz. Sie bilden damit zumindest einen Einwand, Fiktion und Realität auf der Ebene der Sinneserfahrung völlig zu vermischen und müssen aus diesem Grund ebenso in die Tradition der „Illusionstechniken des Fiktiven“³⁰ gestellt werden.

25 Die nicht wirklich neu ist, sondern eben vorgängige Nutzungsweisen der Simulation und digitalen Fotografie synthetisiert – Verfahrensweisen, die so Normalusern zuvor nicht zur Verfügung standen.

26 Winkler (wie Anm. 24), S. 211.

27 Sogar Microsofts neues Betriebssystem „Vista“ greift auf 3D-Technologie zurück.

28 Pias, Claus: *Computer Spiel Welten*, München 2002, S. 5.

29 Deuber-Mankowsky, Astrid: *Lara Croft. Modell, Medium, Cyberheldin*, Frankfurt a.M. 2001, S. 48.

30 Krämer, Sibylle: „Vom Trugbild zum Topos. Über fiktive Realitäten“, in: Iglhaut, Stefan u.a. (Hrsg.): *Illusion und Simulation. Begegnung mit der Realität*, Ostfildern 1995, S. 130-

Deuber-Mankowskys Darstellung einer ‚sinnlich-wahrnehmbaren Spiegelwelt‘ berücksichtigt ebenfalls nicht, dass die Beziehung zwischen Datenstruktur und Ausgabe kaum in dem Maße regelhaft ist, wie z.B. die optischen Gesetze, nach denen ein Spiegel sein Umfeld reflektiert.³¹ Lässt man ‚Bilder‘ etwa der fraktalen Geometrie außer Acht, so beziehen sich die wahrgenommenen Zeichen überwiegend nicht direkt auf ihre Datenstruktur, sondern im hohen Maße auf die Darstellungskonventionen des Films, der Fotografie, des Comics oder der Musik. So ließe sich mit Volker R. Grassmuck erwidern:

Nur der Teil der Welt, der bereits mediatisiert ist, erscheint auf dem Medienhorizont, und er erscheint in der sensorischen Bandbreite, mit der jeweils der Medienkonsument an den künstlichen Horizont angeschlossen ist.³²

Hinzu tritt, dass diejenigen Bereiche, die als Datenstruktur ihren Weg in die Wahrnehmung finden, denjenigen gegenüberstehen, welche weder visualisiert, noch gehört, noch ertastet werden können oder sollen. Die Betonung der Sinnlichkeit virtueller Welten eignet sich daher nicht, dem von der Medienarchäologie hervorgehobenen problematischen Status des Menschen als Systemstelle eines Computersubjektes entgegenzutreten.³³ Umgekehrt aber vernachlässigt die medienarchäologische Kritik an verbergenden grafischen Oberflächen, dass, angesichts der starken Technik- und Werbediskurse zur 3D-Hardware, die immer wieder das hinter dem fotorealistischen Bild liegende Gittermodell entblößen, von einer ‚Täuschung‘ über die Bedingungen der Entstehung von Bildschirmoberflächen kaum zu sprechen ist.

Ungeachtet der immer noch bestehenden Grenzen zwischen Fiktion und Realität, Daten und Wahrnehmung, bleiben die Umbrüche in der Computergrafik im Hinblick auf eine mögliche kritische Theorie weiterhin interessant: Does it have to be 3D? Bereits die oben angesprochene Linearperspektive Albertis hatte von Beginn an einen theologischen Disput sowie Ideale der Ordnung und des ‚Guten‘ implementiert.³⁴ Als nicht weniger ‚ideologisch‘ beschreibt Sibylle Krämer die Funktion von Zentralperspektive und virtuellen Welten:

137, hier S. 137 weist darauf hin dass „[d]ie traditionellen Illusionstechniken des Fiktiven einher[gehen] mit Inszenierungen der symbolischen Differenz. Es gibt Theaterbühnen, Bilderrahmen, Fernsehschirme.“

- 31 Ihrer Auseinandersetzung fehlt leider insgesamt ein Technikbezug, welcher die unterschiedlichen Konstruktionsbedingungen virtueller Welten berücksichtigen könnte.
- 32 Grassmuck, Volker R.: „Die Turing Galaxis. Das Universal-Medium als Weltsimulation“, in: *Lettre International*, Jg. 28, Nr. 1, 1995, S. 48-55, hier S. 48.
- 33 Vgl. Pias (wie Anm. 28), S. 7.
- 34 Edgerton (wie Anm. 11), S. 33 führt dazu aus: „Die perspektivische Szenerie selber war bereits eine visuelle Metapher für diese höhere Existenz [von *virtù*, *onore* und *nobiltà*]“.

Während die Zentralperspektive suggeriert, daß real ist, was Bezug auf einen externen Beobachter hat, wird die virtuelle Realität zum Sinnbild dafür, daß der Standpunkt des externen Beobachters – wird er absolut gesetzt – selber illusorisch ist.³⁵

Virtuelle Welten scheinen, vielleicht wenig überraschend, mit Vorstellungen der Welt und des Menschen verbunden zu sein, die sich nicht allein auf der Ebene der Repräsentation äußern, sondern bereits auf der Ebene ihrer unterliegenden Struktur, der mathematischen Modellierung selbst. Ein einfacher Bezug auf die Oberflächendarstellung ist folglich nicht ausreichend, um eine (ideologie-)kritische Betrachtung von 3D-Technologien und ihrer ‚Realität‘ zu ermöglichen.

Betrachtet man die verschiedenen Ausformungen des Computers – wie etwa Winkler sie anführt – jeweils als Dispositive im foucaultschen Sinne, so müsste eine kritische Theorie der 3D-Technologie und der Spiele – wenn nicht sogar der symbolischen Maschine selbst – zukünftig fragen, welche die möglichen Formen der Erkenntnis, des Wissens und der Macht der jeweiligen Implementierung von Algorithmen in Hardware bilden und welche Diskurse zu ihnen geführt haben – „The way it’s meant to be played“.

35 Krämer (wie Anm. 30), S. 137.