

Evaluierung und beispielhafte Umsetzung von
Techniken und Methoden zur Stereoskopie

Projektarbeit

Januar 2005

Katja Einsfeld

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 2 |
| 2 | 3D Wahrnehmung | 3 |
| 2.1 | Das visuelle System | 3 |
| 2.2 | 3D Wahrnehmung | 3 |
| 2.2.1 | Hinweise zur Wahrnehmung der Tiefe | 4 |
| 2.2.2 | Parallel- und Schieltechnik | 6 |
| 2.3 | Techniken zur Immersion | 7 |
| 2.3.1 | Richtlinien für immersive künstliche 3D Bilder | 7 |
| 3 | Echte 3D Visualisierung - Stereoskopie | 8 |
| 3.1 | Begriffsbestimmungen | 8 |
| 3.2 | Die Geschichte der Stereoskopie | 9 |
| 3.3 | Überblick über Techniken und Funktionalität von 3D Anzeigen | 9 |
| 3.4 | Nähere Betrachtung zweier 3D-Systeme | 13 |
| 3.4.1 | 3D Bildschirm | 13 |
| 3.4.2 | PowerWall | 14 |
| 3.5 | Erzeugen von Stereopaaren | 15 |
| 3.5.1 | Begriffe zur Stereopaarerzeugung | 15 |
| 3.5.2 | Beobachtungen zur Parallaxe | 16 |
| 3.5.3 | Wenig zu empfehlende Methoden der Stereopaar-Erzeugung | 17 |
| 3.5.4 | Die Methode der asymmetrischen perspektivischen Projektion | 18 |
| 4 | 3D Programmierung | 21 |
| 4.1 | 3D Modellierung | 21 |
| 4.2 | OpenGL | 21 |
| 4.2.1 | OpenGL-Methoden zur Stereographie | 22 |
| 4.3 | Chromium | 22 |
| 4.3.1 | Chromium-Techniken zur Stereographie | 23 |
| 4.3.2 | Konfigurations-Skript zur Erzeugung von Stereopaaren mittels Chromium | 24 |
| 5 | Stereoskopie Demonstrator | 26 |
| 5.1 | Ideen und Zweck | 26 |
| 5.2 | Implementierung und Beurteilung der Stereoeffekte | 26 |
| 5.3 | Kritische Beurteilung und mögliche Lösungen | 29 |
| 6 | Schlusswort | 30 |

1 Einleitung

Wir leben und orientieren uns in einer dreidimensionalen Umgebung, die wir mittels unseres binokularen visuellen Systems auch dreidimensional wahrnehmen können. PC-Anwendungen, mit denen wir in zunehmendem Maße Aufgaben erledigen sollen, sind jedoch in den meisten Fällen auf die Zweidimensionalität des Bildschirms beschränkt. Auch sogenannte 3D-Programme und -Spiele projizieren räumliche Bilder auf die zweidimensionale Anzeige und erzeugen lediglich ein Flachbild, weshalb diese Techniken als 2,5-dimensional bezeichnet werden. Ein 3D-Eindruck kann, wie in Gemälden, trotzdem durch Techniken wie perspektivische Darstellung oder Schattenbildung entstehen. Da einem Punkt auf der Fläche verschiedene Punkte im Raum zugeordnet werden können, fällt bei unbekanntem Objekten bei dieser Art der Darstellung die Orientierung oft schwer, zumal die Interaktion meist wenig zur Orientierung beiträgt, da diese sehr gewöhnungsbedürftig oder stark eingeschränkt ist.

Die sogenannten "Echten 3D Technologien" dagegen, die künstlich raumgetreue Bilder erzeugen und auch als Stereoskopie bezeichnet werden, wollen ein wirklich dreidimensionales Bild, ein Raumbild, schaffen, das der Realität möglichst nahe kommen soll. Ziel ist es, so eine Arbeitsumgebung zu erzeugen, die es ermöglicht, sich wie in der realen Umgebung zu orientieren und mit dieser zu interagieren. Auf diese Weise soll eine intuitive und effiziente, weil der Realität nachempfundene, natürliche Benutzung gewährleistet sein. Das DFKI-Projekt @VISOR beschäftigt sich beispielsweise damit, einen 3D Schreibtisch als reale Umgebung für alltägliche PC-Arbeiten wie geordnete Datei-Ablage und Terminplanung zu erstellen, der weitere Möglichkeiten, die bisher der Realität vorbehalten waren wie Post-It-Markierungen, oder solche, die in realen Umgebungen problematisch sind, wie Verknüpfungen zwischen Objekten, zu entwickeln. Außerdem besteht die Hoffnung, dass multivariate Daten mit Hinzunahme der dritten Dimension besser als mit den bisher zur Verfügung stehenden zwei Dimensionen repräsentiert werden können.

Ein überzeugendes dreidimensionales Bild zu erzeugen, das die menschliche Wahrnehmung nicht reizt oder bei längerer Betrachtung zu Kopfschmerzen führt, ist deshalb besonders schwierig, weil wir es gewohnt sind unsere Umwelt dreidimensional wahrzunehmen und daher sehr hohe Anforderungen an realistische 3D Darstellungen haben. Gerade wenn längere Zeit mit der 3D-Umgebung gearbeitet werden soll, ist es wichtig, eine ergonomische Ansicht zu erzeugen.

Die vorliegende Arbeit soll einen kurzen Überblick über die Techniken und Methoden der Stereoskopie verschaffen und die Möglichkeiten einer bestimmten 3D-Anzeige, der PowerWall, anhand verschiedener Beispielapplikationen näher untersuchen. In Kapitel 2 wird in Hinblick auf die künstliche Nachbildung zunächst die natürliche 3D Wahrnehmung untersucht. Dabei stehen Fragen nach Bedingungen der Entstehung des 3D Eindrucks und Methoden zur Erzeugung einer immersiven Umgebung unter künstlichen Voraussetzungen im Mittelpunkt. Kapitel 3 klassifiziert die wichtigsten Stereoskopie-Verfahren und beschreibt deren technische Realisierung, wobei dann speziell der Aufbau und die Benutzung der PowerWall und die dazu verwendete Methode des Stereopaars sowie insbesondere die Berechnung dessen erklärt wird. Auf die für diese Arbeit nötigen Werkzeuge zur 3D Graphik Programmierung, OpenGL und Chromium, wird in Kapitel 4 eingegangen. In Kapitel 5 wird die Implementierung eines Demonstrators für die PowerWall beschrieben und der Wirkungsgrad seiner 3D-Effekte beurteilt. Das Schlusswort in Kapitel 6 schließlich wagt einen kurzen Blick in die Zukunft.

2 3D Wahrnehmung

2.1 Das visuelle System

Lichtstrahlen, die durch die Pupille in das menschliche Auge gelangen, werden von der Linse gebrochen. Muskeln am Auge ermöglichen es, die Wölbung der Linse, d.h. die Brennweite und damit den Fokus zu variieren, was als Akkomodation bezeichnet wird: Ist die Linse im Ruhezustand, werden die Strahlen nur unwesentlich gebrochen und wir können weit entfernte Objekte von denen aus fast parallele Strahlen ins Auge reflektiert werden, fokussieren. Um Gegenstände in der Nähe zu fokussieren, spannen sich die Ziliarmuskeln an und die Linse wölbt sich. Die Strahlen werden so stärker gebrochen. Der Abstand des fokussierten Objekts vom Betrachtungspunkt wird *Fokallänge* genannt. Es sind jedoch nicht nur die Gegenstände, die sich genau im Abstand der Fokallänge befinden scharf zu sehen, sondern auch solche, die sich in einem Bereich um die Fokallänge befinden. Die Größe dieses Bereiches ist die *Fokaltiefe*. Sie ist für nahe fokussierte Objekte sehr gering und für entfernte Objekte bedeutend größer. Gegenstände außerhalb des Bereichs der Fokaltiefe erscheinen unscharf.

Gleichzeitig mit der Akkomodation findet die Konvergenz, die gegenläufige Ausrichtung der Augen auf das Objekt im Fokus, statt. Blicken wir in die Ferne, so stehen die optischen Achsen der Augen parallel. Wenn wir dagegen ein nahes Objekt betrachten, drehen sich die Augen nach innen. Das menschliche Gehirn steuert Akkomodation und Konvergenz so, dass automatisch ein Zusammenhang zwischen Fokussierung und Konvergenzwinkel der Augen besteht.

Die Helligkeitsänderung der Strahlen, die schließlich auf die Netzhaut (Retina) treffen, wird von sich dort befindenden Rezeptoren registriert. Neuronen verarbeiten auf noch nicht völlig geklärte Weise diese Informationen auf verschiedenen Ebenen, sodass ein dreidimensionales mentales Modell unserer Umgebung entsteht. Das, was wir dann bewusst wahrnehmen ist von bisherigen Erfahrungen abhängig, eine Selektion, Vereinfachung und Veränderung dessen, was auf der Netzhaut erscheint. Auftretende Augen-Fehler können durch die Anpassungsfähigkeit des visuellen Systems ausgeglichen werden.

2.2 3D Wahrnehmung

Mit einem Auge sind Entfernungen schwer abschätzbar und wir erkennen nur ein 2D-Abbild unserer Umgebung. Wenn wir uns dagegen mit zwei Augen (binokular) umsehen, so nehmen wir nicht zwei getrennte, leicht unterschiedliche 2D-Bilder, sondern ein einziges Bild mit Tiefeninformation wahr. Physiologische Prozesse im Gehirn synthetisieren (verschmelzen) das rechte und linke Netzhautbild zu einem Raumbild.

Der mittlere menschliche Augenabstand, der in der Stereographie als Stereobasis bezeichnet wird, beträgt 60-70 mm. Dieser Abstand bewirkt, dass Objekte auf der linken Netzhaut an einer anderen Position als auf der rechten Netzhaut erscheinen. Bei gerader Haltung des Kopfes ergibt sich eine horizontale Verschiebung der Gegenstände, wobei diese von der Lage des Objekts im Fokus abhängt. Bei nahen fokussierten Objekten ist die Variation der horizontalen Distanzen der Punkte des Objektes auf beiden Netzhäuten größer als bei entfernten Objekten. Gegenstände, die mehr als 220 m entfernt liegen, können nicht mehr plastisch wahrgenommen werden.

Dass die Unterschiedlichkeit der beiden Netzhautbilder, die sogenannte binokulare Disparität, eine wichtige, aber nicht die einzige Methode unseres visuellen Systems zur Wahrnehmung von Tiefe ist, wird im folgenden Abschnitt erläutert. Dabei wird im Hinblick auf das Ziel dieser Arbeit, virtuelle räumliche Welten zu erzeugen, auch zu beachten sein, welche dieser Hinweise an unser Gehirn auch künstlich erzeugt werden können, um einen Tiefeneindruck

zu schaffen.

2.2.1 Hinweise zur Wahrnehmung der Tiefe

Werden bei der Darstellung dreidimensionaler Objekte bestimmte Techniken angewandt, so geben diese unserem visuellen System Hinweise, wie die Darstellung räumlich wahrgenommen werden kann. Die ersten sechs dieser Techniken, die uns räumliche Ausdehnung vortäuschen sollen, sind auch für Darstellungen in 2D Bildern zu gebrauchen. Werden ausschließlich diese Hinweise verwandt, nehmen wir ein Flachbild wahr, können uns darauf abgebildete Gegenstände aber räumlich vorstellen.

- **Perspektive:** Bei Darstellungen mit Linearperspektive schneiden sich parallele Linien, die nicht parallel zur Projektionsebene verlaufen, in einem Fluchtpunkt. Am verbreitetsten ist die Ein-Punkt-Perspektive, bei der die Projektionsebene die xy -Ebene ist, vor allem Gegenstände mit Kanten parallel zu den Koordinatenachsen dargestellt werden und daher nur die Kanten parallel zur z -Achse sich in dem Fluchtpunkt treffen. In der darstellenden Kunst sind außerdem die Zwei- und Drei-Punkt-Perspektive verbreitet. Die Größe der dargestellten Gegenstände ändert sich umgekehrt proportional zu deren Entfernung vom Beobachtungspunkt. Objekte, die in der Szene weiter hinten liegen, werden dichter zusammen und kleiner dargestellt, als vergleichbare Objekte im Vordergrund. Der mittels perspektivischer Projektion darstellbare Bereich der 3D Umgebung hat die Form eines Pyramidenstumpfes, dessen Spitze im Projektionszentrum liegt.
- **Größe bekannter Objekte:** Aus unserer Erfahrung wissen wir, wie die Größenverhältnisse zwischen uns bekannten Objekten sind. Wenn ein großer Gegenstand genauso groß wie ein kleinerer Gegenstand auf der Netzhaut abgebildet wird, so können wir daraus schließen, dass der größere Gegenstand weiter hinten im Raum angeordnet sein muss, was natürlich eng mit der perspektivischen Darstellung zusammenhängt.
- **Detail:** Von uns näheren Teilen des Raums können wir mehr Details wie Musterung oder Oberflächenstrukturen erkennen, als bei entfernteren Objekten. Texturen in der Entfernung erscheinen nur noch verschwommen.
In der Computergraphik kann perspektivische Projektion angewandt werden, um einen Raum auf den Bildschirm zu projizieren. Da entfernte Objekte so kleiner erscheinen und daher weniger Pixel auf der Anzeige zur Verfügung haben, sind sie automatisch weniger detailliert.
- **Verdeckung:** Wenn ein Gegenstand einen anderen teilweise verdeckt, so folgern wir, dass das halb verdeckte Objekt hinter dem ersteren liegen muss. Dieser Tiefenhinweis wird in der Computergraphik durch das Entfernen von vom Betrachtungspunkt weg zeigenden Flächen und durch Speicherung und Nutzung der Tiefeninformation im Tiefenpuffer realisiert. Teile von Objekten, die auf eine Fläche projiziert werden, für die schon ein Eintrag im Tiefenpuffer existiert, werden nur dann in den Puffer übernommen, wenn sie näher zum Beobachtungspunkt liegen als das bereits registrierte Objekt.
- **Licht und Schatten:** Objekte in der Ferne erscheinen weniger hell, etwas verschwommen und verblaut im Vergleich zu nahen Objekten. Der Verblauungseffekt, der eintritt, weil das blaue Licht am leichtesten die Atmosphäre durchdringt, wurde schon lange von Künstlern zur Darstellung weiter Landschaften genutzt.
Bei beleuchteten Szenen tragen die Schatten von Objekten, die auf den Untergrund oder andere Objekte fallen, zur Wahrnehmung der Anordnung bei und die Verteilung des

Lichts auf dem Objekt selbst hilft bei dem Erkennen seiner Form. Außerdem können Entfernungen durch Abnahme des Einflusses von Lichtquellen abgeschätzt werden. All diese Effekte können mit Methoden der Computergraphik nachgebildet werden, wobei jedoch Aufwand und Nutzen abgewogen werden müssen.

- **Relative Bewegung:** Bei Animationen auf Bildschirmen oder Projektionsflächen kann außerdem zur Erzeugung eines Tiefeneindrucks benutzt werden, dass Objekte, die weiter entfernt sind, sich scheinbar langsamer bewegen als dem Beobachter nahe Objekte, die vorbeizuraschen scheinen. Dieser Hinweis auf räumliche Ausdehnung kann auch dann genutzt werden, wenn der Beobachter sich um die Szene herum bewegt oder die Szene gedreht wird. Man spricht hierbei auch von *Bewegungs-Parallaxe*, da sich bei der Bewegung die Winkel und Abstände der Objekte zueinander und zu der Beobachtungspose ändern und so Tiefenhinweise geben.

Wie das Bild von Maurits Cornelis Escher zeigt, kann es beim (vorsätzlich) unsachgemäßen Gebrauch dieser Hinweise auch zu Täuschungen kommen, wenn beispielsweise die Objekte in der Szene eine andere Größe oder Form haben als wir mittels unserer Erfahrung vermuteten oder perspektivische Gesetze außer Kraft gesetzt werden.

Zusätzlich zu den bereits genannten Tiefenhinweisen, die eher auf Erfahrung und psychologischen Effekten beruhen, gibt es noch einige Hinweise, die durch die Physik des menschlichen Wahrnehmungsprozesses bestimmt sind. Diese können bei gewöhnlichen, flächigen 3D Darstellungen nicht benutzt werden, da die Augen sich auf die Entfernung zu einer Fläche einstellen und somit keine unterschieden Netzhautbilder entstehen. (Eine Ausnahme hiervon bilden Parallel- und Schieltechnik, die im Anschluss erläutert werden.) Durch Benutzung aller oder der wichtigsten dieser physikalischen Tiefenhinweise unterscheidet sich die "Echte 3D Visualisierung" oder Stereoskopie von der flächigen 3D Darstellung. Das künstliche Erzeugen dieser Tiefenhinweise bewirkt bei guter Realisierung, dass unserem visuellen System vorgetäuscht werden kann, eine räumliche Ausdehnung und nicht mehr eine Darstellung eines Raums auf einer Fläche wahrzunehmen.

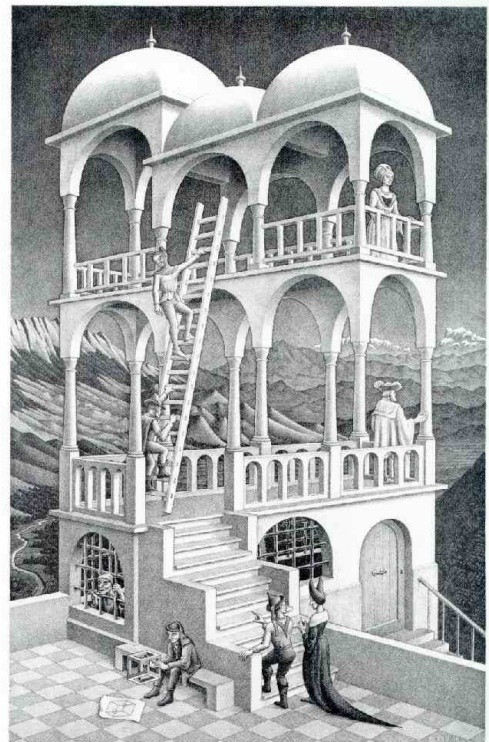


Bild 1: Bild von Escher, bei dem mit den Tiefenhinweisen gespielt wird.

- **Binokulare Disparität:** Da unsere Augen auf einer horizontalen Achse durch eine gewisse Distanz getrennt sind und wir daher die Szene von zwei leicht verschiedenen Standpunkten gleichzeitig betrachten, erscheinen auf der rechten und linken Netzhaut leicht verschiedene Bilder, die dann vom visuellen System zu einem dreidimensionalen Eindruck verarbeitet werden. Dieser für die Stereoskopie wichtigste Tiefenhinweis kann künstlich dadurch erzeugt werden, dass von einer räumlichen Szene von den beiden Augenpunkten aus jeweils eine perspektivische Projektion durchgeführt wird. Diese beiden Bilder, die Halbbilder genannt werden, müssen dann so angeordnet sein, dass jedes Auge nur das für dieses Auge vorgesehene Bild sieht. Wenn ein Auge nicht

komplett von dem anderen Halbbild abgeschirmt werden kann, treten Geisterbilder auf (engl.: Ghosting): Das Bild ist leicht verschwommen doppelt zu erkennen. Werden die Bilder vertauscht, so kann sich der Tiefeneffekt umkehren.

Wie genau die beiden Halbbilder berechnet werden und mit welchen Techniken sie dem jeweiligen Auge zugeordnet werden können, wird in Kapitel 3 erklärt.

- **Akkomodation:** Wenn wir ein nahes Objekt fokussieren wollen, muss das Auge durch Muskelkraft die Fokallänge der Linse so einstellen, dass die Strahlen von diesem Objekt so gebrochen werden, dass sie auf der Netzhaut scharf erscheinen. Die zur Fokussierung nötige Muskelspannung, die Akkomodation, gibt dem Gehirn Hinweise darüber, wie weit das Objekt entfernt ist. Dieser Tiefenhinweis kann nur dann für die Stereoskopie genutzt werden, wenn das Projektionsgerät eine Projektion auf ein räumliches oder sich bewegendes Objekt zulässt. Wenn dagegen mit einer Projektionsfläche gearbeitet wird, kann der Akkomodations-Tiefenhinweis keine Tiefeninformationen transportieren.

- **Konvergenz:** Zur Wahrnehmung eines nahen Gegenstandes rotieren die Augen ebenfalls durch Muskelkraft ein wenig nach innen, bei weit entfernten Szene-Abschnitten stehen die Achsen parallel. Auch diese Muskelbewegung ist vom Gehirn als Tiefenhinweis nutzbar.

Akkomodation und Konvergenz verändern sich natürlicherweise gleichzeitig. Es gibt, wie in Kapitel 3 beschrieben wird, jedoch auch Stereoskopie-Techniken, bei denen die Akkomodation von der Konvergenz entkoppelt ist, was jedoch die Wahrnehmungsqualität nicht negativ beeinflussen muss, da binokulare Disparität meist der dominantere Tiefenhinweis ist.

Nicht bei allen Darstellungstechniken können alle Tiefenhinweise genutzt werden und auch bei bestimmten Szenen oder Objekten können nicht alle Hinweise zum Tiefeneindruck beitragen, was aber auch nicht unbedingt nötig ist. Einige Hinweise können dominanter als andere sein, aber auch das kann von Szene zu Szene variieren.

2.2.2 Parallel- und Schieltechnik

Parallel- und Schieltechnik werden im Englischen als *free viewing*-Techniken bezeichnet, da es mit etwas Übung ohne Hilfsgeräte möglich ist, zwei auf einer Fläche sichtbare Halbbilder zu einem räumlichen Eindruck zu verbinden. Diese Methoden sind recht nützlich, um einen ersten Stereoeindruck zu bekommen, eignen sich jedoch nicht für den dauerhaften Gebrauch, da sie für unser visuelles System zu anstrengend sind. Bei der Parallel-Technik werden beide Augen wie beim Blick in die Ferne parallel, also jedes Auge auf das dafür vorgesehene Einzelbild, ausgerichtet. Gleichzeitig muss aber der Fokus auf die nahe Fläche und nicht in die Ferne abgestimmt sein. Zur Anwendung der Schieltechnik werden die Halbbilder umgekehrt angeordnet. Die betrachtende Person blickt in die Mitte zwischen die Bilder und versucht durch Schielen, die Einzelbilder zu einem Bild zusammenrücken zu lassen.

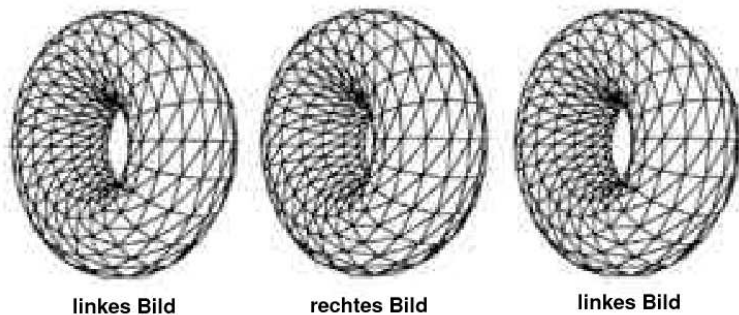


Bild 2: Das erste und zweite Bild dient zum Ausprobieren der Paralleltechnik, das zweite und dritte für die Schieltechnik

Bei der Parallel-Technik werden beide Augen wie beim Blick in die Ferne parallel, also jedes Auge auf das dafür vorgesehene Einzelbild, ausgerichtet. Gleichzeitig muss aber der Fokus auf die nahe Fläche und nicht in die Ferne abgestimmt sein. Zur Anwendung der Schieltechnik werden die Halbbilder umgekehrt angeordnet. Die betrachtende Person blickt in die Mitte zwischen die Bilder und versucht durch Schielen, die Einzelbilder zu einem Bild zusammenrücken zu lassen.

2.3 Techniken zur Immersion

Immersion bedeutet Eintauchen in eine Umgebung. Im Zusammenhang dieser Arbeit hat der Anspruch, eine immersive Anwendung zu schaffen zur Konsequenz, dass diese nicht nur ihren Zweck erfüllen sowie praktisch und ergonomisch zu benutzen sein soll, sondern darüber hinaus ist zu gewährleisten, dass Benutzende sich möglichst umgehend in die Anwendungsumgebung hineindenken und die Wahrnehmung und Aufmerksamkeit ungeteilt auf den Zweck der Anwendung lenken können. Insbesondere soll es nicht nötig sein, sich während des Benutzens Gedanken über die Funktionalität oder gar die Art der Programmierung zu machen. Dazu ist es im Allgemeinen unabdinglich, dass die Anwendung in Echtzeit arbeitet und intuitive, der Realität nachempfundene, weitgehende und einfach zu benutzende Interaktionsmöglichkeiten bietet. Überzeugende und realitätsnahe graphische und akustische Ausgabe können das Eintauchen in die Anwendung unterstützen, aber auch vom eigentlich Zweck der Applikation ablenken und folglich die Immersion in zielgerichtetes Arbeiten behindern.

Für die im Rahmen dieser Arbeit zu erstellende stereoskopische Anwendung sind keine weitgehenden Interaktionsarten gefordert - die Immersion soll ausschließlich aufgrund der räumlichen Darstellung ein Eintauchen in die künstliche 3D Umgebung ermöglichen. Dazu sind folgende Punkte zu beachten:

2.3.1 Richtlinien für immersive künstliche 3D Bilder

- Da wir es gewohnt sind, zu einem Zeitpunkt nur in einem beschränkten Tiefenbereich scharf zu sehen und so abwechselnd verschiedene Bereiche fokussieren, sollte der Tiefenbereich, also der Abstand zwischen Nah- und Fernpunkt, in dem die künstliche räumliche Umgebung aufgebaut wird, nicht zu groß sein. Wenn dieser Bereich zu groß gewählt wird, tritt ein Effekt, der in der Stereoskopie als Bildzerfall bezeichnet wird, auf: Das Bild ist nicht mehr als ganzes wahrzunehmen, Vorder- oder Hintergrund scheinen sich aufzulösen. Alternativ wäre es möglich, einen Bereich scharf und die übrigen Bereiche unscharf darzustellen. Dies kann jedoch der Immersion entgegen und unrealistisch wirken, da das Auge vergeblich versuchen wird, den unscharfen Bereich zu fokussieren. Nur wenn eine Interaktion des Programms mit der Akkomodation des Auges möglich wäre, könnte also ein größerer Tiefenbereich sinnvoll eingesetzt werden. (Bei 3D-Techniken, bei denen Akkomodation als Tiefenhinweis nutzbar ist, tritt diese Problematik nicht auf.)
- 3D-Bilder die weit aus einer Projektionsfläche herauszuragen scheinen, sind anstrengend zu betrachten, da das Auge den Fokus auf die Projektionsfläche einstellt, der Konvergenzpunkt wie das Objekt aber näher sein muss. Entspannter ist die 3D Wahrnehmung wenn sich das Objekt scheinbar hinter der Projektionsfläche befindet und dieses wie durch ein Fenster angesehen werden kann (Scheinfenster-Technik). Dem entgegen steht, dass nach vorne herausragende Objekte wesentlich mehr beeindrucken können.
- Der Nahpunkt (minimale Distanz von Szeneteilen zur Beobachungsposition) sollte nicht zu nahe an der Beobachungsposition liegen, da Objekte, die sehr nahe vor dem Auge liegen, schwer realitätsnah räumlich darzustellen sind und, da das Auge sich auf dieses Objekt fokussieren würde, alles andere unscharf erscheinen müsste. Die Mitte zwischen Beobachungsposition und Projektionsfläche kann ein Anhaltspunkt für den Nahpunkt sein. In der Literatur werden noch viele andere Richtlinien zur Positionierung des Nahpunkts genannt, was zeigt, dass dies von Anwendung zu Anwendung neu abgewogen werden muss. Wenn also der Darstellungsbereich für die räumliche Szene am Nahpunkt abrupt endet, sollte darauf geachtet werden, dass sich

keine Objekte durch die Ebene, die parallel zur Projektionsfläche durch den Nahpunkt verläuft, dem Beobachtungspunkt nähern und so zunächst halb dargestellt werden und schließlich ganz verschwinden. Auch an den anderen Rändern des Darstellungsbereichs sollte es vermieden werden, Gegenstände, die nur noch halb in diesem Bereich liegen darzustellen, da abgeschnittene Darstellungen nicht immersiv sind. Wenn die Projektionsfläche groß ist und das Blickfeld im Normalfall eher auf das Zentrum, als auf die Ränder des Darstellungsbereichs gerichtet ist, ist es jedoch möglich, dass Objekte diesen Bereich über die obere, untere, rechte oder linke (jedoch nie über die hintere oder vordere) Randfläche verlassen. Damit zusammenhängend ist ein weites mögliches Blickfeld als in großem Maße immersionsfördernd anzusehen.

- Besonders gut ist der immersive räumliche Eindruck der Darstellung, wenn es möglich ist, die Kopfposition oder gar den Beobachtungspunkt zu verändern und durch Herumlaufen das 3D Objekt zu erkunden. Dies ist jedoch, wie in Kapitel 3 näher beschrieben wird, nicht bei allen Stereoskopie-Techniken uneingeschränkt möglich. Bei einigen Techniken, wie drehende Projektionsflächen, können die Darstellungen ohne zusätzlichen Aufwand von allen Seiten betrachtet werden, bei anderen Techniken wird die Augenposition vom Programm registriert und die Darstellung für diese Position aktualisiert. Wenn, wie bei anderen Systemen, diese Interaktion mit der Augenposition nicht möglich ist, verzerrt sich die räumlich Umgebung beim Umherlaufen und wirkt so der Immersion entgegen.

3 Echte 3D Visualisierung - Stereoskopie

3.1 Begriffsbestimmungen

In der Literatur zu 3D Visualisierung und Stereoskopie werden diese Begriffe nicht einheitlich benutzt, weshalb hier zunächst geklärt werden soll, in welchem Sinn sie in dieser Arbeit verwandt werden. Wie in der Einleitung, in Kapitel 1, und im Abschnitt 2.2.1 über Tiefenhinweise bereits erwähnt, unterscheidet sich die herkömmliche 3D Visualisierung, die Flachbilder erzeugt und besser als 2,5D Visualisierung bezeichnet werden sollte, von der Raumbilder erzeugenden "echten 3D Visualisierung" dadurch, dass bei letzterer auch physikalische Tiefenhinweise zum Raumeindruck beitragen. Insbesondere die binokulare Disparität, die Verschiedenheit der Bilder auf der linken und rechten Netzhaut, spielt dabei eine wichtige Rolle. Binokulare Disparität tritt natürlich bei Betrachtung realer Gegenstände, bei Betrachtung der Ausgabe von Visualisierungstechniken wie sich drehende Spiegel und bei Verfahren, die explizit zwei Bilder erzeugen und diese dem jeweiligen Auge zuordnen, wie dem Anaglyphen-Verfahren, jedoch nicht bei der Betrachtung von Flachbildern auf.

Stereo, kommt vom griechischen Wort *stereos* und bedeutet *fest* oder im weiteren Sinne *körperlich, räumlich*. Danach, nach [5] und weiteren Quellen bezeichnet Stereoskopie die Gesamtheit der Verfahren zur Aufnahme und Wiedergabe von raumgetreuen Bildern und kann daher gleichbedeutend mit "echter 3D Visualisierung" benutzt werden. Andere Autoren wie [6] verstehen unter Stereoskopie lediglich die, allerdings bedeutende, Gruppe von 3D Visualisierungsverfahren, die mit Stereobildpaaren arbeiten, also solche, die für jedes Auge ein Bild erzeugen und nicht die volumetrischen Verfahren wie drehende Spiegel.

Im Rahmen dieser Arbeit soll, um weitere Verwirrung zu vermeiden, Stereoskopie im ursprünglichen Sinn, gleichbedeutend mit "echter 3D Visualisierung" benutzt werden.

3.2 Die Geschichte der Stereoskopie

Schon lange vor der Photographie, im 4. Jh. v. Chr. befasste sich der griechische Mathematiker Euklid mit Stereobildpaaren und räumlichem Seheindruck. Jedoch erst 1833 erfand der Engländer Sir Charles Wheatstone das erste Stereoskop und die erste Beliebtheitswelle erlebte die Stereoskopie mit dem Fortschritt der Photo-Technik um 1860. Zu Beginn des 20. Jh. geriet die Stereoskopie fast in Vergessenheit und bekam erst um 1950, als Stereokameras für den Kleinbildfilm und das View Master System (siehe Abbildung 3) auf den Markt kamen, neuen Aufwind.

Danach gab es immer wieder kleinere Begeisterungsschübe für stereoskopische 3D-Darstellungen, beispielsweise durch Kinofilme im Anaglyphen-Verfahren. Erst in den letzten Jahren, da die Computer-Technologie soweit fortgeschritten ist, dass stereoskopische Echtzeit-Darstellungen möglich wurden, gewinnt die Stereoskopie wieder an Bedeutung.

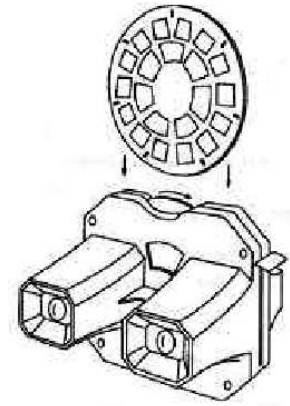


Bild 3: View Master

3.3 Überblick über Techniken und Funktionalität von 3D Anzeigen

Es gab und gibt eine Vielzahl von sehr unterschiedlichen Systemen und Verfahren zur Anzeige von Raumbildern. McAllister [1] und Halle [3] beschreiben eine Auswahl davon. Ein Überblick über die diesen Geräten und Verfahren zu Grunde liegenden Methoden, die sich in verschiedene Kategorien einteilen lassen, soll dieser Abschnitt geben. Daran anschließend werden dann zwei 3D Visualisierungssysteme, die für diese Arbeit von Bedeutung sind, näher beschrieben.

Grundsätzlich gilt für alle 3D Visualisierungssysteme, dass das Raumbild nicht einfach in der Luft entstehen kann, sondern ein Darstellungsmedium auf der Sichtlinie vom Betrachtungspunkt aus in Richtung 3D Szene liegen muss. Dieses Medium kann ein von einer Lichtquelle ausgesandtes Licht reflektierendes Material, wie eine Projektionswand oder eine sich rotierende Fläche, oder ein selbst Licht aussendendes Objekt, wie ein spezieller Monitor oder rotierende kleine Lichtquellen, sein. Dabei kann das Darstellungsmedium je nach Verfahren vor, hinter oder in der dargestellten Szene liegen. Die Größe der 3D Szene ist durch das Medium beschränkt.

Volumetrische Techniken: Techniken dieser Kategorie zeichnen sich dadurch aus, dass das Darstellungsmedium den Raum ausfüllt bzw. sich darin bewegt, weshalb auch von multiplanaren Techniken gesprochen wird. So gibt es beispielsweise sich parallel zu einem Bildschirm bewegende Spiegel, deren Bewegung mit dem Monitor synchronisiert ist. Der Betrachtungspunkt ist auf der gleichen Seite des Spiegels wie der Monitor. Das 3D Bild entsteht hinter dem Spiegel, indem der Bildschirm je nach Entfernung des Spiegel mehr oder weniger weit entfernte Ebenen der Szene anzeigt. Ähnlich funktionierende Systeme gibt es mit rotierenden Spiegeln, mit Spiegeln, die sich nur wenig bewegen aber dafür mit Hilfe von Membranen den Fokus verändern können, mit drehender Helix und mit rotierenden LEDs, wobei die letzten beiden schwer anzusteuern sind. Bei all diesen Techniken sollte die Bewegung schnell genug sein, damit das Bild nicht flimmert und beim Einsatz von CRT-Monitoren muss darauf geachtet werden, dass diese nicht zu lange nachleuchten, da andernfalls verschmierte Bilder entstehen.

Nicht verbreitet, aber möglich ist auch die 3D Darstellung mit einem volumenfüllenden Medium, dessen innere Punkte beispielsweise mit einem Laser zum Leuchten gebracht werden können.

Nachteil der volumetrischen Techniken ist, dass die Raumbilder transparent sind, weshalb der

Tiefenhinweis der Verdeckung nicht wirksam genutzt werden kann. Aus diesem Grund werden diese Techniken meist nur zur Darstellung von Drahtgittermodellen genutzt. Ein Vorteil aber ist, dass es im Gegensatz zu anderen Methoden möglich ist, sich in einem gewissen Bereich oder, sogar ganz um das Raumbild herum zu bewegen. Außerdem können mehrere beobachtende Personen gleichzeitig, von verschiedenen Punkten aus die Szene betrachten. Volumetrische Verfahren sind *autostereoskopisch*, d.h. das Raumbild ist ohne Hilfsmittel wie Brillen oder ähnliches wahrnehmbar.

Ein weiteres Charakteristikum, das diese Techniken von anderen unterscheidet, ist, dass keine Entkopplung von Akkomodation und Konvergenz stattfindet, diese beiden Tiefenhinweise also wie bei der Betrachtung realer Gegenstände konsistent sind. Raumbilder, die diese Eigenschaft aufweisen werden als *massiv* (engl. *solid*) bezeichnet.

Holographie: Bei dieser Technik wird ein 3D Objekt mit einem speziellen Film aufgenommen, indem ein Laserstrahl auf das Objekt gerichtet wird und die von diesem Objekt reflektierten und abgelenkten Lichtwellen zusammen mit einem Referenzlaserstrahl, der direkt auf den Film strahlt, Interferenzmuster in Form von Ringen auf dem Film erzeugt. Wird nach der Entwicklung wieder ein Laser auf den Film gerichtet, so nehmen beim Durchqueren des dort aufgezeichneten Interferenzmusters die Wellenfronten der Strahlen wieder genau die Form an, die sie bei der Aufzeichnung hatten, weshalb sich ein räumliches Bild des Gegenstandes aufbaut. Das Bild kann in einem gewissen Bereich von verschiedenen Punkten aus betrachtet werden. Bewegte Bilder sind mit dieser Methode nicht zu erzeugen.

Wie bei den volumetrischen Techniken entsteht ein massives Bild, d.h. Akkomodation und Konvergenz sind nicht entkoppelt, obwohl, im Gegensatz zu den volumetrischen Techniken, vom Darstellungsmedium kein Raum aufgespannt wird. Die Holographie kann deshalb zwischen den volumetrischen Techniken und den nachfolgend beschriebenen Parallaxe-Techniken eingeordnet werden.

Parallaxe-Techniken: Die Parallaxe misst die Unterschiedlichkeit der bei der Betrachtung einer 3D-Szene entstehenden beiden Netzhautbilder. Parallaxe-Techniken, bei denen folglich der Tiefenhinweis der binokularen Disparität im Vordergrund steht, arbeiten mit flachen Darstellungsmedien, sodass das Auge den Fokus auf diese Fläche einstellt und somit die Akkomodation von der Konvergenz entkoppelt ist und keine massiven, sondern nur noch sogenannte *virtuelle* Raumbilder entstehen. Da der Tiefenhinweis der binokularen Disparität stärker ist als der der Akkomodation werden trotzdem räumliche Szenen wahrgenommen.

Wie bei Parallaxe-Techniken trotz des flächigen Darstellungsmediums je nach Entfernung der Objekte vom Beobachtungspunkt unterschiedliche Parallaxen und damit auch binokulare Disparität entsteht, soll beispielhaft das Parallaxe-Barrier-Panoramagramm in Abbildung 4 verdeutlichen:

Ein auf bestimmte Weise arrangiertes Bild, das von hinten beleuchtet wird, ist in einem kurzen Abstand mit vertikal verlaufenden schmalen, undurchsichtigen Barrieren teilweise verdeckt, so dass bei einem Blick in die verbleibenden Öffnungen nur gewisse Bereiche sichtbar sind. Da beide Augen aus einem anderen Winkel auf die Öffnungen sehen, kann, bei geeigneter streifenweiser Anordnung des Bildes, binokulare Disparität simuliert werden und so der Tiefeneindruck entstehen. Zusätzlich es es noch möglich die Streifen so anzuordnen, dass bei leichter

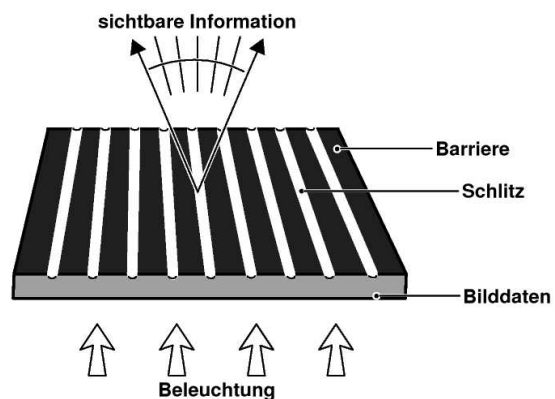


Bild 4: Parallaxe-Panoramagramm

Drehung des Kopfes und Veränderung des Sichtwinkels zur Fläche, andere, unter der Barriere liegende Streifen sichtbar werden. Dadurch können leicht rechts bzw. links von der ursprünglichen Position liegende Teile der 3D Szene gezeigt werden, also der Eindruck eines Panoramas entstehen. Übersteigt die Drehung des Kopfes einen bestimmten Winkel, so wiederholen sich die sichtbaren Streifen und das Panorama scheint auf eine andere, bereits gesehene Sicht zu springen.

Genau analog dazu funktioniert das Linsenraster-Panoramagramm (engl. lenticular sheets): Statt undurchsichtiger Barrieren sind bei dieser Technik halbzylinderförmige schmale Linsen aus Plastik vertikal angeordnet. Dadurch werden die Lichtstrahlen auf darunter liegende Bildstreifen so gebrochen, dass der gleiche Effekt wie oben beschrieben entsteht.

Stereopaar-Techniken: Stereopaar-Techniken sind die wichtigste Art von Parallaxe-Techniken, die zudem für die echte 3D Visualisierung eines der meist genutzten Verfahren sind. Dies hat, wie bereits im Abschnitt 3.1 erklärt, zur Folge, dass der Begriff der Stereoskopie von einigen Autoren auf Stereopaar-Techniken eingeschränkt wird. Die Idee der Stereopaar-Techniken ist, binokulare Disparität zu simulieren, indem jedem Auge ein speziell dafür konstruiertes Flachbild zugeordnet wird, das nur von diesem Auge gesehen werden kann. Die beiden sogenannten Halbbilder, also das rechte und das linke Halbbild, die zusammen als Stereopaar bezeichnet werden, können, bei fachgerechter Realisierung, vom visuellen System zu einem räumlichen Eindruck synthetisiert werden, da ja auch die Bilder, die bei Betrachtung realer Gegenstände auf der Netzhaut erscheinen lediglich Flachbilder sind. Die entstehenden Raumbilder sind *virtuell* da das Auge auf die beiden Flachbilder fokussiert bleibt, die Akkomodation also nicht zum Tiefeneindruck beiträgt. Bei der Wahrnehmung realer Objekte konvergieren die Augachsen auf den Punkt, der fokussiert wird. Hier jedoch wird die Akkomodation von der Konvergenz entkoppelt: Während die Akkomodation auf die Bildebene fixiert ist, rotieren die Augen je nach scheinbarer Entfernung eines Punktes in der Szene unterschiedlich stark nach innen. Da dies nicht dem gewohnten Verhalten der Augenmuskulatur entspricht, ist bei einigen Menschen eine kurze Eingewöhnungszeit nötig; bei wenigen bleibt die 3D Wahrnehmung virtueller Raumbilder auch nach dieser Zeit problematisch.

Wie die beiden Halbbilder berechnet werden können, wird in Abschnitt 3.5 besprochen. Aus dieser Betrachtung wird sich ergeben, dass eine Bewegung des Betrachtungspunktes vor der Darstellungsfläche zu unrealistischen Verzerrungen der 3D Szene führt, wenn diese nicht unter Registrierung des Betrachtungspunktes neu berechnet.

Es gibt eine Vielzahl verschiedener Stereopaar-Techniken, die sich vor allem dadurch unterscheiden, wie das linke und rechte Halbbild den Augen präsentiert werden, sodass jedes Auge nur das jeweils dafür vorgesehene Bild sehen kann. Wenn diese Trennung der beiden Halbbilder nicht komplett möglich ist, wenn also das rechte Auge auch Teile des linken Bildes sehen kann und umgekehrt, entstehen *Geisterbilder* (engl. *Ghosting*). Die Szene ist dann in einer leicht verschobenen, schwachen Kontur doppelt zu sehen. Wegen der großen Bedeutung dieser Techniken, werden sie im Folgenden weiter klassifiziert:

- **Sukzessivstereoskopie** (engl. field sequential- bzw. time multiplexed techniques): Bei dieser Technik werden von der Graphikhardware abwechselnd mit möglichst hoher Frequenz (120 Hz) das rechte und linke Halbbild erzeugt und angezeigt. Ein anderer Mechanismus sorgt dafür, dass die Augen im gleichen Rhythmus freigegeben oder blockiert werden, sodass das rechte Bild nur in das rechte Auge gelangt und das linke nur in das linke Auge. Im Gegensatz zu den vorher beschriebenen Verfahren, bei denen kein brillenähnliches Hilfsgerät nötig war und die deshalb als *autostereoskopisch* oder brillenlos bezeichnet werden, sind diese Verfahren nun *nicht autostereoskopisch* oder brillengebunden, um die Zuordnung jedes Halbbildes zu dem jeweiligen Auge zu erreichen.

Dabei kann weiter zwischen aktiven und passiven Brillen unterschieden werden. Für die Sukzessivstereoskopie sind vor allem aktive Brillen, für die nachfolgend beschriebene Parallelstereoskopie häufig passive Brillen im Gebrauch.

aktive Brillen: Aktive Brillen werden mit der Graphikhardware synchronisiert und bekommen z.B. durch einen elektronischen Impuls mitgeteilt, wann das rechte und wann das linke Halbbild angezeigt wird, wann also das rechte und wann das linke Auge blockiert werden muss. Die Blockierung kann beispielsweise dadurch erfolgen, dass die Brille für jedes Auge mit einem LCD ausgerüstet ist, dessen Kristalle durch Anlegen einer geringen Spannung so ausgerichtet werden können, dass keine Lichtstrahlen mehr durch diese Fläche gelangen. Diese Lichtstrahlen können wie bei einem gewöhnlichen 2,5D Graphiksystem von einer Anzeige oder von einer vom Projektor angeleuchteten Projektionswand ausgehen.

- **Parallelstereoskopie** (engl. time parallel techniques): Bei diesen Verfahren werden die beiden Halbbilder gleichzeitig angezeigt. Mittels unterschiedlicher optischer Techniken können diese dann wiederum dem jeweiligen Auge zugeordnet werden. Wenn diese Trennung nicht komplett möglich ist, tritt Ghosting auf, was bei dauerhafter Betrachtung zu Kopfschmerzen führen kann. Wegen der Vielfalt dieser Techniken, werden im Folgenden einige Beispiele gegeben:

Beim im Abschnitt 3.2 vorgestellten ViewMaster werden die beiden Bilder mit einer Linse vergrößert direkt vor dem jeweiligen Auge angeordnet. Bei den ebenfalls schon lange bekannten Anaglyphen-Verfahren, mit denen allerdings nur Graustufen-Raumbilder erzeugt werden können, wird das eine Halbbild in rot, das andere auf der gleichen Fläche in grün dargestellt und eine rot-grün Brille trennt diese beiden Bilder mehr oder weniger präzise voneinander. Sehr ähnlich, aber wesentlich fortschrittlicher lässt sich Parallelstereoskopie in Verbindung mit den unten beschriebenen passiven Brillen anwenden. Diese bei der *PowerWall* angewandte Technik wird unter 3.4.2 nochmals genauer beschrieben.

Ähnlich funktionierend wie der ViewMaster, aber auch für bewegte Bilder geeignet, ist das sogenannte *head mounted display*. Dieses in Form eines Helms mit je einem kleinen Display für jedes Auge getragene Gerät kann mittels *position tracking* die aktuelle Position an das System melden, das daraus eine veränderte Sicht auf die 3D Szene berechnen und darstellen kann. Die Schwierigkeit bei diesem Interaktionsmechanismus ist jedoch, dass das System sehr schnell reagieren muss, also der Aufbau der neuen Szene der Bewegung nicht zu sehr nachhängen darf, wenn dies als realistischer Effekt empfunden werden soll. Ein weiteres Problem des *head mounted display* ist die geringe mögliche Auflösung der beiden kleinen Anzeigen.

Auch die in Abschnitt 2.2.2 bereits beschriebene Parallel- und Schiel-Technik kann als Parallelstereoskopie bezeichnet werden. Als letztes seien noch die beiden für Panoramagramme genutzten, unter den allgemeinen Parallaxe-Verfahren bereits erwähnten, Techniken der Parallaxe-Barriere- und Linsenraster-Anzeigen in Bezug auf Stereopaare genannt. Die beiden Halbbilder werden in vertikale Streifen zerteilt und so ineinander geschoben, dass bei einem Blick durch die Barrieren bzw. durch die Linsen das rechte Auge nur Streifen des rechten Halbbildes sieht und das linke Auge nur die des linken Halbbildes. Mit moderner Technologie, die im Abschnitt 3.4.1 zum 3D Bildschirm näher erläutert wird, ist dieses Verfahren auch bei in Verbindung mit CRTs oder LCD-Displays zur Darstellung bewegter Raumbilder anwendbar.

passive Brillen: Im Gegensatz zu Systemen mit aktiven Brillen ist es hier nötig, dass das Projektions- oder Anzeigegerät das davon ausgehende Licht polarisiert. Für linkes und rechtes Halbbild werden unterschiedliche, zueinander orthogonale Polarisationsach-

sen verwendet, wobei lineare oder zirkuläre Polarisierung möglich ist (siehe Abschnitt 3.4.2 zur *PowerWall*). Dies kann durch sogenannte Polfiltervorsätze vor Projektoren erreicht werden. Die Brille enthält ebenfalls Polfilter, wobei der des rechten Auges orthogonal zu dem des linken angeordnet wird. Nur die Lichtstrahlen einer Polarisationsrichtung können durch diesen Filter gelangen, so dass linkes und rechtes Halbbild exklusiv dem jeweiligen Auge zugeordnet werden können.

Passive Brillen benötigen aufgrund der Polarisierungstechnik im Gegensatz zu den aktiven Brillen keine Synchronisation und sind deshalb billiger, was für Mehrbenutzer-Systeme sehr vorteilhaft ist. Jedoch hat auch diese Technologie einige Nachteile: Wenn der Kopf zu weit gedreht wird, werden bei linearer Polarisierung die Bilder umgekehrt zugeteilt und es entsteht ein *pseudostereoskopisches* Raumbild: die Tiefenverhältnisse erscheinen umgekehrt. Zirkuläre Polarisierung erlaubt einen größeren Bewegungsspielraum des Kopfes. Werden Projektoren und eine Projektionswand benutzt, so ist darauf zu achten, dass diese Wand das Licht nicht wieder entpolarisiert. Dies wird von dem sogenannten "silver screen", einer teuren Projektionswand mit spezieller Beschichtung geleistet. Bei der Benutzung von CRTs und passiven Brillen ist es wichtig, dass der Bildschirm nicht zu lange nachleuchtet, da sonst Ghosting auftreten kann. Ein weiterer Nachteil gegenüber aktiven Brillen ist die durch Verwendung von Polfiltern verringerte Transmission: Je nach Ausführung gelangen nur 30-60% des Lichtes durch diese Filter, weshalb starke Projektoren nötig sind.

3.4 Nähere Betrachtung zweier 3D-Systeme

3.4.1 3D Bildschirm

Der sogenannte 3D Bildschirm, ist ein auf den ersten Blick gewöhnlich aussehender Flachbildschirm. Durch eine, mit dem unter Abschnitt 3.3 erläuterten autostereoskopischen Parallaxe-Barriere-System vergleichbare Technik wird es jedoch möglich, bewegte Raumbilder zu erzeugen, die die Höhe und Breite des Monitors haben sowie etwa 10 cm hinter die Bildschirmfläche und 10 cm vor diese reichen.

Abbildung 5, die der Herstellerseite [8] entnommen ist, zeigt den schematischen Aufbau des Systems: Die beiden Halbbilder werden in viele schmale, vertikale Streifen zerteilt und abwechselnd so angeordnet, dass bei einem Blick durch die Lichtstrahlen brechende Schicht (engl.: *beam splitter*) jedes Auge nur die ihm zugeordneten Streifen sehen kann. Diese setzen sich dann auf der Netzhaut wieder zu vollständigen Halbbildern zusammen, die dann schließlich einen Tiefeindruck vermitteln. Der beam splitter

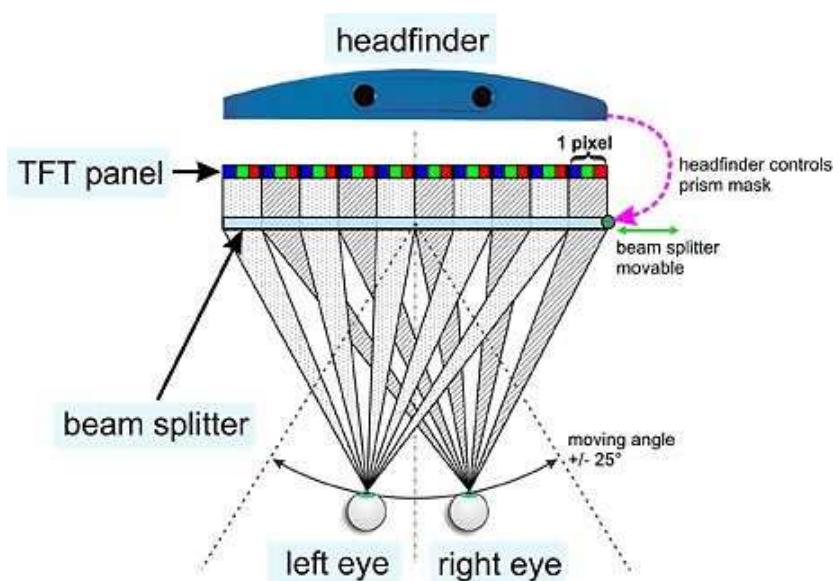


Bild 5: Funktionsprinzip des 3D-Bildschirms

Der beam splitter

muss dazu in einem bestimmten Abstand von der Ebene der Bilddaten angeordnet sein. Da sich bei horizontaler Bewegung des Beobachtungspunktes der Sichtwinkel zum Monitor ändert und damit andere Teile der Bilddaten sichtbar werden sollten, ist der 3D Monitor an der oberen Kante mit zwei Sensoren, die die Position der beiden Augen registrieren, ausgerüstet. Das System berechnet dann aus der aktuellen Position der Augen die dafür nötige Anordnung der Halbbilderstreifen und des bewegbaren beam splitters. Auf diese Weise ist es möglich, sich in einem Bereich von $\pm 25^\circ$ vor dem Monitor hin- und her zu bewegen und die 3D-Szene von verschiedenen Punkten aus zu betrachten. Durch die dabei auftretende Bewegungs-Parallaxe kann das visuelle System einen zusätzlichen Tiefenhinweis nutzen. Bei zu schneller Bewegung vor dem Monitor kann das Bild darauf jedoch nicht schnell genug reagieren.

Nachteil des in vertikalen Streifen ineinander geschobenen Stereopaars ist, dass dadurch die horizontale Auflösung im Vergleich zu einem herkömmlichen Monitor gleicher Größe halbiert wird. Dadurch wird es insbesondere schwierig, Details wie beispielsweise Schrift in annehmbarer Qualität darzustellen. Ein neben dem 3D Bildschirm platzierter normaler Zusatzmonitor kann als Ergänzung dienen, wenn abwechselnd herkömmliche Arbeitsoberflächen und Raumbilder angezeigt werden sollen.

3.4.2 PowerWall

Die PowerWall ist ein mit der Technik der brillengebundenen Parallelstereoskopie realisiertes 3D-System, das sowohl mit passiven als auch mit aktiven Brillen ausgeführt werden kann.¹ Auch im Aufbau gibt es große Unterschiede. Beispielhaft soll nun die in der für diese Arbeit genutzte Ausführung der PowerWall beschrieben werden.

Das das Verfahren passiver Brillen nutzende System ist so aufgebaut, dass der Raum durch die Projektionsfläche, die etwa die Größe einer gewöhnlichen Zimmerwand hat, zerteilt wird. Durch die Größe der Projektionswand wird ein weiter Blickwinkel ermöglicht, was die Immersion fördert. Im, von den Betrachtenden aus gesehen, hinteren Teil stehen drei mit moderner Graphikhardware ausgerüstete Arbeitsrechner, die über die Netzwerk-Software Chromium (siehe Abschnitt 4.3) verbunden sind. Die Berechnung der darzustellenden 3D Szene wird, um es etwas vereinfacht darzustellen (nähere Informationen in späteren Abschnitten), so aufgeteilt, dass ein Rechner die Koordination übernimmt und die anderen jeweils ein Halbbild des Stereopaars berechnen. Ebenfalls im hinteren Teil des Raumes befinden sich zwei mit Polarisationsfiltern versehene Projektoren, von denen einer das rechte und der andere das linke Halbbild mit zueinander orthogonalen Polarisationsachsen übereinander auf die nicht entpolarisierende Projektionswand werfen. Wenn die beiden Halbbilder auch nur zu kleinen Anteilen entpolarisiert werden und deshalb nicht mehr vollständig getrennt werden können, tritt Ghosting auf. Diese Wand, die mit einer besonderen Beschichtung versehen ist, muss zudem die Eigenschaft haben, dass die beiden Bilder auf der anderen Seite dieser Wand, möglichst ohne zu große Intensitätsverluste und Farbverschiebungen, für die mit passiven Brillen ausgerüsteten Betrachtenden sichtbar sind, während die Projektoren und die Rechner nicht wahrgenommen werden sollen. Ohne Brille erscheint das Bild schwammig, da zwei leicht verschiedene Bilder übereinander liegen. (Dies ist auch der Grund, warum es technisch schwierig ist, Flachbilder, die den mit der PowerWall erzielbaren 3D-Effekt realistisch darstellen, zu photographieren.) Mit der passiven Brille, die auf der rechten und linken Seite jeweils nur das für diese Seite polarisierte Licht durchlässt, entsteht der Stereoeffekt und nach einer kurzen Eingewöhnungsphase wird eine 3D Umgebung wahrgenommen.

¹Leider werden teilweise auch große Projektionsflächen, die keine Stereobilder erzeugen, als PowerWall bezeichnet.

Das aus dem Prospekt eines Herstellers [7] entnommene Bild 6 zeigt, in welcher Größenordnung der Intensitätsverlust beim Passieren des Polarisationsfilters

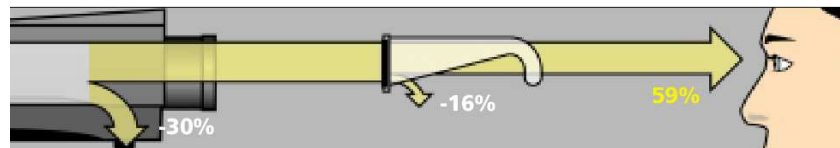


Bild 6: Verlust von Lichtintensität

in den Projektoren und der Polbrille der Benutzenden liegen kann. Auch durch die Projektionswand tritt eine vom Material abhängige, nicht näher spezifizierte Transmissionsverminderung auf. Ein Gesamt-Intensitätsverlust von 41% ist ein im Vergleich zu anderen Systemen guter Wert, der jedoch trotzdem durch leistungsstarke Projektoren ausgeglichen werden muss.

Das hier beschriebene PowerWall-System funktioniert mit zirkularer Polarisation. Den Unterschied zu linearer Polarisation soll Bild 7, das ebenfalls dem oben zitierten Prospekt entnommen ist, verdeutlichen. Bei linearer Polarisation verlaufen die durch den Polarisationsfilter gelangenden Lichtwellen nur in einer Ebene. Für das linke und rechte Halbbild werden dann zueinander orthogonale Ebenen verwandt. Das Problem

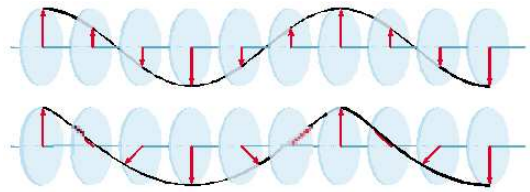


Bild 7: Polarisation: oben linear, unten zirkular

an dieser Technik ist jedoch, dass bei Neigung des Kopfes und damit der Polbrille die beiden Halbbilder nicht mehr eindeutig von einer Seite der Brille blockiert und von der anderen durchgelassen werden können. Bei noch stärkerer Neigung kann sich die Zuordnung der Halbbilder sogar umkehren. Bei zirkularer Polarisation tritt diese Problematik nicht auf, da der Drehsinn der Lichtwellen, der für das linke und rechte Halbbild wieder orthogonal zueinander liegt, auch bei Neigung des Kopfes gleich bleibt. Dieser höhere Freiheitsgrad muss jedoch mit Polarisationsfiltern höherer Präzision erkauft werden. Bei Neigung des Kopfes um 90° geht der Stereoeffekt aber dennoch verloren, da das 3D-Bild nur horizontale Parallaxe enthält. Weil die beschriebene PowerWall (noch) nicht über Sensoren für den Blickpunkt verfügt, muss das Raumbild wie in Abschnitt 3.5 erläutert wird, beim Umherlaufen vor der Projektionsfläche je nach Standpunkt unterschiedlich verzerrt erscheinen. Das gleiche gilt für mehrere Betrachtende an unterschiedlichen Punkten. Auch andere Interaktionsmechanismen wie *Force-Feedback* (das Spüren von Widerstand in einem Handschuh beim Greifen nach Gegenständen in der Szene) sind bisher nicht implementiert.

3.5 Erzeugen von Stereopaaren

Da Stereopaar-Techniken in der Stereographie und auch in dieser Arbeit eine entscheidende Rolle spielen, beschäftigt sich dieser Abschnitt mit der Berechnung der Halbbilder. Dazu sind zunächst einige Begriffe zu klären.

3.5.1 Begriffe zur Stereopaarerzeugung

Die Fläche, auf der die beiden Halbbilder angezeigt werden (z.B. die PowerWall), wird als *Stereofenster* bezeichnet. Diese Fläche ist gewöhnlich parallel zu der Ebene, die die beiden Augen schneidet und die senkrecht zur Standard-Blickrichtung liegt. Vor, in und hinter diesem Fenster kann sich die virtuelle 3D Szene ausdehnen. Ein bestimmter Punkt in der 3D Szene wird im Allgemeinen im linken und rechten Halbbild an zwei leicht unterschiedlich

angeordneten Stellen dargestellt. Diese beiden Punkte in den Halbbildern, die vom gleichen Punkt der 3D Szene stammen werden *homologe* (einander entsprechende) *Punkte* genannt. Als *Parallaxe* wird der Abstand p der homologen Punkte voneinander bezeichnet. Der Abstand der homologen Punkte kann durch Übereinanderlegen der Halbbilder bestimmt werden. Ebenfalls Parallaxe oder, um es von der ersten Bedeutung zu unterscheiden, Parallaxe-Winkel wird der aus der Parallaxe p gewonnene Wert $\theta = 2 * \text{atan}(p/(2 * d))$ genannt, wobei d der Abstand von der Betrachtungs- zu der Projektionsebene ist.²

3.5.2 Beobachtungen zur Parallaxe

Davon ausgehend, dass die beiden Augen nur auf einer horizontalen Achse verschoben sind, ergibt sich auch in der Realität nur eine horizontale Parallaxe. Erst wenn wir den Kopf zur Seite neigen, kann binokulare Disparität in einer anderen als der horizontalen Richtung zum Tiefeneindruck beitragen. Bild 8 verdeutlicht in einer Sicht von oben auf die beiden betrachtenden Augen, die 3D Szene und die Projektionsfläche wie horizontale Parallaxe entsteht. Es wird unterschieden zwischen positiver Parallaxe, wenn der betrachtete Punkt hinter der Fläche liegt und negativer Parallaxe, wenn der Punkt davor liegt. Bei positiver Parallaxe liegen die Punkte, die zum rechten Halbbild gehören rechts von denen, des linken Halbbilds. Bei Punkten mit negativer Parallaxe sind die homologen Punkte auf der Projektionsebene horizontal vertauscht. Je näher die Punkt an der Projektionsfläche bzw. dem Stereofenster liegt, desto kleiner ist der Betrag der Parallaxe. Liegt der Punkt direkt auf der Ebene der Projektionswand, so entsteht Nullparallaxe, d.h. der Punkt wird in beiden Halbbilder an der gleichen Stelle abgebildet.

Wenn wir einen festen Punkt in der 3D Szene betrachten und uns von diesem entfernen, so wird der Betrag der Parallaxe geringer, nähern wir uns dagegen diesem Punkt, so vergrößert sich der Abstand der homologen Punkte im Stereofenster. Wenn die 3D-Szene bei Änderung des Betrachtungspunkts nicht angepasst wird, die Parallaxe für einen Punkt folglich immer gleich bleibt, so scheint sich der Punkt mitzubewegen: Bei negativer Parallaxe gilt nach dem Strahlensatz $a/b = p/i$, wobei die Variablen a und b wie in Bild

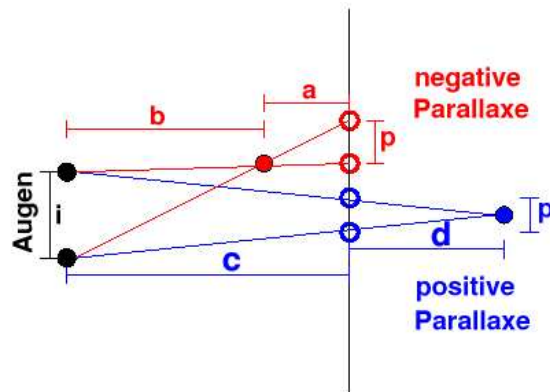


Bild 8: Positive und negative Parallaxe

8 eingezeichnet für die senkrecht zum Stereofenster gemessenen Abstände zwischen Stereofenster und Punkt, sowie zwischen Punkt und Augenebene, p für Parallaxe und i für interokulare Distanz stehen. An dieser Formel wird ersichtlich, dass bei konstanter Parallaxe das Verhältnis zwischen a und b gleich bleibt, d.h. wenn man versucht auf einen Punkt vor der Projektionsfläche zuzulaufen, so wird der Abstand zwar immer geringer, man kann den Punkt aber nicht erreichen, es sei denn man würde in das Stereofenster greifen. Bei positiver Parallaxe gilt ebenfalls nach dem Strahlensatz $c/(c + d) = p/i = k = \text{const}$. Daraus erhält man durch Umformen: $c/d = k/(1 - k) = \text{const}$. Aus dieser Gleichung folgt, dass sich bei gleich bleibender Parallaxe der Abstand von den Augen zum Stereofenster um den gleichen Faktor vergrößert oder verkleinert in dem sich der Abstand von Stereofenster und Punkt vergrößert oder verkleinert. Sowohl bei negativer als auch bei positiver Parallaxe

²In anderen Zusammenhängen wird die Parallaxe auch als Winkel zwischen zwei Graden, die von verschiedenen Standorten auf einen Punkt gerichtet sind, definiert.

gilt also, dass eine Veränderung des Abstands zur Projektionsebene um den Faktor x bei gleich bleibender Parallaxe, den absoluten Abstand eines Punktes von der Ebene ebenfalls um den Faktor x vergrößert bzw. verkleinert. Analog dazu kann erklärt werden, warum und in welchem Maße bei seitlicher Verschiebung des Beobachtungspunktes Punkte vor der Projektionsfläche in die gleiche Richtung mitgehen und Punkte hinter der Projektionsfläche in die andere Richtung. Diese Punktebewegungen werden als immersionsstörende Verzerrungen der Gesamtszene wahrgenommen. Zur Berechnung des Stereopaars sollte von einem Beobachtungspunkt in einer vernünftigen Distanz und im Zentrum der Projektionsfläche ausgegangen werden.

3.5.3 Wenig zu empfehlende Methoden der Stereopaar-Erzeugung

Um nun konkret ein Stereopaar zu berechnen, also die Funktionalität unserer Augen oder wenigstens die einer Stereokamera zu simulieren, gibt es verschiedene Methoden, die mehr oder weniger gute Ergebnisse erzielen. Da die perspektivische Projektion die Funktionalität des Auges gut nachbildet, sollte diese in jedem Fall angewandt werden. Ebenfalls unumgänglich ist es, zwei verschiedene Projektionszentren, die dem rechten und linken Auge entsprechen, zu nutzen und damit zwei Projektionen durchzuführen. Eine naheliegende Idee ist, von beiden Augpunkten aus eine perspektivische Projektion durchzuführen. Verlaufen die Projektionsachsen einfach senkrecht zur Betrachtungsebene und ist der Projektions-Pyramidenstumpf symmetrisch darum angeordnet, so taucht dabei das Problem auf, dass Teile der Szene nur in dem einen oder nur in dem anderen Bild festgehalten werden. Es entspricht zwar der Realität, dass an den seitlichen Rändern der linken und rechten Netzhaut Teile der Szene registriert werden, die das jeweils andere Auge nicht sieht, dies kann aber nicht bei 3D Systemen mit begrenzter Projektionsfläche genutzt werden, da bei einem Blick zur Seite, an den Rand der Projektionswand im Zentrum des Blickfeldes kein Stereopaar, sondern nur ein plötzlich abgeschnittenes und ein anderes, noch weiter reichendes Halbbild wahrgenommen werden könnte. Ziel sollte es aber sein, die komplette Projektionsfläche für das Stereopaar auszunutzen, um dem Kopf Bewegungsfreiheit zu ermöglichen. Das Zuschneiden der beiden auf diese Weise erzeugten Halbbilder, so wie es in der Stereophotographie praktiziert wird, wäre zwar eine mögliche Lösung, ist jedoch nicht ganz einfach korrekt auszuführen. Andere, in der computergesteuerten Erzeugung von Stereopaaren verbreitetere Methoden werden in Bourke [4] beschrieben.

Bei der in Abbildung 9 dargestellten Rotationsmethode (im englischen Sprachgebrauch: *toe-in*) wird die 3D Szene vor der perspektivischen Projektion für das linke Halbbild linksherum um den Punkt im Fokus und für das rechte Halbbild rechtsherum um den gleichen Punkt gedreht und zwar jeweils um die Hälfte des Winkels, unter dem sich die linke und rechte Strecke vom Auge zu dem Punkt schneiden. Dann wird mit dem Punkt zwischen den beiden Augen als Projektionszentrum für beide rotierten Szenen jeweils eine symmetrische perspektivische Projektion durchgeführt.

Die mit dieser Methode erzeugten Stereopaare mögen für kleinere Szenen ihren Zweck erfüllen, bei einer großen PowerWall und mit zunehmendem Abstand vom Zentrum des Stereofensters wird die auftretende *vertikale Parallaxe* jedoch immer störender. Vertikale Parallaxe be-

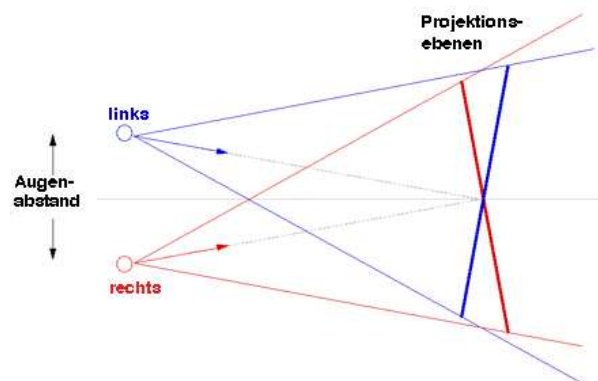


Bild 9: Stereobildpaare durch Rotation

deutet, dass die homologen Punkte in den beiden Halbbildern nicht nur unterschiedliche x- sondern auch unterschiedliche y-Koordinaten haben, was den Betrachtenden Kopfschmerzen bereiten kann oder, bei extremer vertikaler Parallaxe, sogar verhindert, das das Stereopaar zu einem 3D Eindruck verbunden werden kann. Das Auftreten dieses Effekts kann damit erklärt werden, dass nach der Rotation für das linke Halbbild ein Punkt P eine andere Entfernung von der Projektionsebene hat als nach der Rotation für das rechte Halbbild. Wenn danach die perspektivische Projektion durchgeführt wird, wird P in dem Fall, in dem er weiter entfernt ist weiter zur Mitte hin projiziert als im anderen Fall, weshalb daraus unterschiedliche y-Koordinaten resultieren.

Um vertikale Parallaxe zu vermeiden, kann nach Hodges [2] statt perspektivischer Projektion orthogonale Projektion angewandt werden. Diese hat jedoch den Nachteil, dass die horizontale Parallaxe unbegrenzt ist. Um eine Parallaxe größer als die interokulare Distanz, die die Augen zu zwingen versucht, sich nach außen zu drehen, zu vermeiden, muss der zur Darstellung der 3D-Szene nutzbare Bereich der z-Achse stark beschränkt werden. Wenn diese Beschränkung erfolgt und daher nur noch kleine Parallaxe-Winkel auftreten, kann die Rotation durch die effizientere Scherung des Raumes angenähert werden.

3.5.4 Die Methode der asymmetrischen perspektivischen Projektion

Eine andere, wesentlich empfehlenswertere Methode, die keine vertikale Parallaxe erzeugt und den z-Bereich nicht so stark einschränkt, ist die der asymmetrischen perspektivischen Projektion mit parallelen Achsen und zwei Zentren (im englischen Sprachgebrauch: *off-axis*), wie sie in Abbildung 10 dargestellt ist. Asymmetrische Projektion bedeutet, dass der Bereich um die Projektionsachse nicht auf allen Seiten gleich groß ist. Dies ist nötig, damit auf der Ebene, auf der der Fokus liegt, der gleiche Bereich der 3D-Szene für das linke und für das rechte Projektionszentrum sichtbar ist. Dadurch entsteht auch automatisch und ohne dass Verschieben oder Zuschneiden der Halbbilder nötig wäre Nullparallaxe im Bereich des Fokus.

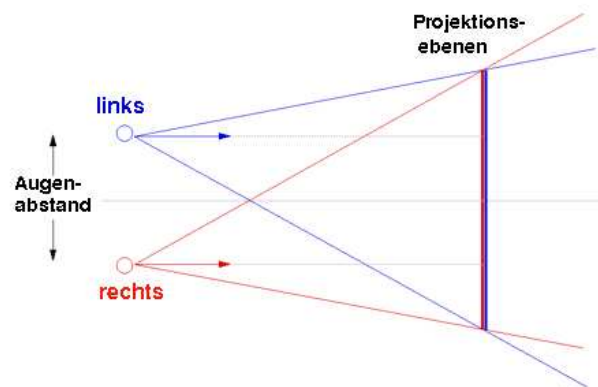


Bild 10: asymmetrische perspektivische Projektion

Die Methode der asymmetrischen perspektivischen Projektion ist die mit vertretbarem Aufwand für ebene Projektionsflächen geeignetste Methode und wurde auch in dieser Arbeit eingesetzt. Bei der Realisierung ergeben sich zwei verschiedene Möglichkeiten: Zum einen kann die Berechnung des Stereopaars und die dazu nötige Projektion bereits in der Software durchgeführt werden. Problematisch kann hierbei sein, dass nicht alle Graphik-Systeme asymmetrische perspektivische Projektion unterstützen. Welche Befehle OpenGL dafür und für allgemeine Stereo-Anwendungen bereit hält wird in Abschnitt 4.2.1 erläutert. Zum anderen ist es möglich, gewöhnliche Graphikanwendungen, die eine 3D-Szene darstellen, bei der Ausführung so zu manipulieren, dass ein Stereopaar daraus entsteht. Abschnitt 4.3.1 beschreibt, wie dies mittels der Netzwerk- und Graphik-Software Chromium realisiert wird.

Vorüberlegungen

In beiden Fällen sind für die asymmetrische perspektivische Projektion mit parallelen Achsen und zwei Zentren die gleichen, im Folgenden beschriebenen Überlegungen und Berechnungen nötig. Zunächst sei festgelegt, dass die Projektionsebene parallel zur x-y-Ebene des

Koordinatensystems ausgerichtet sei und sich auf der negativen z-Achse befinde. Die Projektionsebene ist gleichzeitig die Ebene, auf die fokussiert werden soll. Je nach gewünschter Anordnung kann die 3D Szene um die Projektionsebene herum aufgebaut werden. Der Betrachtungspunkt sei der Ursprung des Koordinatensystems mit Blickrichtung zur negativen z-Achse. Durch Hinzunahme des Hochvektors in Richtung positiver y-Achse ist die Ausrichtung komplett festgelegt. Die beiden Zentren der perspektivischen Projektion entstehen durch Translation in einer Ebene parallel zur Projektionsebene durch den Ursprung und senkrecht zum Hochvektor, sodass das linke Projektionszentrum auf der negativen x-Achse liegt und das rechte auf der positiven und beide durch den Abstand der interokularen Distanz *augdist* (etwa 6,5 cm) getrennt sind. Neben der Ausrichtung der Kamera und der interokularen Distanz ist noch die physikalische Projektionsfläche und deren Umgebung als feste Größe gegeben. Im Fall der PowerWall beispielsweise bestimmen die Höhe und Breite der Projektionswand, sowie der davor liegende sichtbare Raum entscheidend die Möglichkeiten der Raumbilddarstellung. (Weil die für den Stereoeffekt nötige Parallaxe von der interokularen Distanz anhängt und diese in Metern gemessen wird, ist es, um Verwirrung zu vermeiden, sinnvoll, sich die Ausmaße der Szene ebenfalls in der Einheit Meter vorzustellen.)

Für die Berechnung der Halbbilder einer speziellen 3D-Szene müssen außerdem deren Dimensionen und Anordnung im Raum bekannt sein. Insbesondere werden die Fokallänge *fokaldist* (Distanz der fokussierten Projektionsebene vom Ursprung, also von der Betrachtungsposition), die Distanz der Nah- und Fern-Clipping-Ebenen *nah* und *fern* vom Ursprung und die Höhe (Ausdehnung in y-Richtung) der Szene auf der Ebene, auf der der Fokus liegt, *fokaldisthoehe*, benötigt. Empfohlen wird, die Ausdehnung der Szene an die des vorhandenen Ausgabegeräts anzupassen. In Beispiel der PowerWall sollte also das Verhältnis von Szenenhöhe, -weite und Ausdehnung vor *fokaldist*-Ebene in der gleichen Größenordnung wie das Verhältnis von PowerWall-Höhe, -Weite und Raum vor der Wand liegen. Wenn entgegen dieser Empfehlung die Szene im Vergleich zur x- und y-Richtung eine sehr große z-Ausdehnung hat, vor der PowerWall jedoch nicht genügend Platz ist, um eine Beobachtungsposition in dieser Relation einzunehmen, so verkleinert sich, wie oben erklärt, der Abstand der Punkte zu der Projektionsebene und die Szene erscheint in z-Richtung komprimiert. Zur Vermeidung zu großer negativer Parallaxen durch sehr nahe liegende Objekte, die bei entkoppelter Akkommodation unrealistisch wirken, kann gefordert werden, dass die Nah-Clipping-Ebene mindestens die Hälfte der Distanz der Projektionsebene zum Ursprung hat. Andere Richtlinien zur Wahl sinnvoller Werte für *fokaldist* und *fokaldisthoehe* besagen, dass der Blickwinkel, der durch die Projektion festgelegt wird zwischen 45 und 60° betragen soll.

Wenn, bei sich grob entsprechenden Größenverhältnissen von Höhe, Breite und Tiefe des virtuellen Raums und der Ausgabegeräte, die Projektionsebene im Bildraum in der Referenzeinheit Meter eine andere Größe als die physikalisch vorhandene Ausgabeebene hat und das Verhältnis in y-Richtung mit $verhaeltnis = powerwallhoehe/fokaldisthoehe$ quantifiziert werden kann, so werden bei der Projektion auf die reale Projektionswand die Objekte der Szene, die Abstände zueinander und zum Ursprung, sowie die erzeugten Parallaxen der beiden Halbbilder um den Faktor *verhaeltnis* vergrößert oder verkleinert. Da sich jedoch die interokulare Distanz der betrachtenden Person bei der Projektion nicht verändert, steht diese nun nicht mehr im richtigen Verhältnis zu den errechneten Parallaxen. Um dieses Problem ohne viel Aufwand zu lösen, kann das Stereobild statt mit der interokularen Distanz *augdist* mit dem Wert $offset = augdist/verhaeltnis$ erzeugt werden.

Berechnung der Projektionsparameter

Bevor die Projektion durchgeführt wird, muss das Projektionszentrum auf der x-Achse um $offset/2$ verschoben werden. Alternativ kann die Szene in die entgegengesetzte Richtung translatiert werden, also für das linke Halbbild um $offset/2$ nach rechts und für das rechte nach links. Die Art der asymmetrischen perspektivischen Projektion, d.h. die Größe des

Projektions-Pyramidenstumpfs, kann durch die sechs Parameter *oben*, *unten*, *rechts*, *links*, *nah* und *fern* festgelegt werden. Die letzten beiden Parameter beschreiben die Distanz der Nah- und Fern-Clipping-Ebenen, die gegeben sind. Die ersten vier Parameter beschreiben die Ausdehnung der Nah-Clipping-Ebene um die z-Achse, wodurch die Größe des Blickfeldes nach allen Seiten festgelegt wird. Zu deren Berechnung sind die Hilfsgrößen $BzuH = \text{powerwallbreite}/\text{powerwallhoehe}$ (Verhältnis von Breite zu Höhe) und $s = \text{nah}/\text{fokaldist}$ (Faktor aus dem Strahlensatz) nötig. Die obere und untere Kante der Nah-Clipping-Ebene ergeben sich nach dem Strahlensatz durch $\text{oben} = s * \text{fokaldisthoehe}$

Wie in Abbildung 11 zu sehen ist, unterscheiden sich die Parameter *links* und *rechts* für das linke und rechte Halbbild, da das Zentrum der Projektion verschoben ist. Ohne Translation wären *links* und *rechts* die Randpunkte des roten Bereichs, die durch den Abstand $s * \text{fokaldisthoehe} * BzuH$ getrennt sind. In diesem Fall wäre $\text{links} = -s * \text{fokaldisthoehe}/2 * BzuH$. Für das linke Halbbild kommt wegen der Translation nach links der Term $\text{offset}/2$ hinzu. Wegen des veränderten Winkels des links begrenzenden Sichtstrahls zur Fokalebene muss außerdem der in der Abbildung 11 rot eingezeichnete, nach dem Strahlensatz berechnete Bereich $(\text{fokaldist}-\text{nah})/\text{fokaldist} * \text{offset}/2 = (1-s) * \text{offset}/2$ von der Strecke abgezogen werden, sodass sich schließlich $\text{links} = -s * \text{fokaldisthoehe}/2 * BzuH + \text{offset}/2 - (1-s) * \text{offset}/2 = 1/2 * s * (-\text{fokaldisthoehe} * BzuH + \text{offset})$ ergibt. Der rechte Rand des linken Halbbildes liegt analog bei $\text{rechts} = 1/2 * s * (\text{fokaldisthoehe} * BzuH + \text{offset})$. Für das rechte Halbbild erhält man nach analoger Rechnung $\text{links} = 1/2 * s * (-\text{fokaldisthoehe} * BzuH - \text{offset})$ und $\text{rechts} = 1/2 * s * (\text{fokaldisthoehe} * BzuH - \text{offset})$.

In dem obigen Ansatz wird die Größe des Blickfeldes durch die y-Ausdehnung der Ebene im Fokus, fokaldisthoehe bestimmt. In anderen Anwendungen kann es sinnvoll sein, das Blickfeld direkt über die Größe der Nah-Clipping-Ebene festzulegen. In diesem Fall ist statt fokaldisthoehe die Größe nahhoehe gegeben, wodurch sich sofort $\text{oben} = \text{nahhoehe}/2$ und $\text{unten} = -\text{oben}$ ergeben. Für das linke Auge berechnet sich links zu $\text{links} = 1/2 * (-\text{nahhoehe} * BzuH + s * \text{offset})$ und die übrigen Größen analog. Dieser Ansatz ist jedoch mit Vorsicht zu genießen, da es bei Ausdehnung der Szene bis zum linken oder rechten Rand der Nah-Clipping-Ebene, zu einer sogenannten *Window-Violation* kommt: Auf Bild 11 ist zu erkennen, dass der Bereich vor der Fokaldistanz-Ebene, der zum linken Halbbild beiträgt nicht deckungsgleich mit dem des rechten Halbbildes ist. Objekte oder Teile von Objekten, die nicht im Schnitt beider Bereiche liegen, erscheinen nur in einem Halbbild, was störende, unrealistische Effekte zur Folge hat.

Einen Beitrag zur Steigerung der Realitätsnähe der Darstellung kann die Betrachtung der Lage des Beobachtungspunktes relativ zur Ausgabefläche liefern. Beim Beispiel einer Power-Wall von etwa 2,3 Höhe und einer durchschnittlichen Augenhöhe der betrachtenden Personen von 1,7m liegt der Beobachtungspunkt deutlich über dem Mittelpunkt der y-Ausdehnung, der standardmäßig als Projektionszentrum dient. Um, zur Erzeugung einer immersiveren Darstellung, Projektionszentrum und Beobachtungspunkt zusammenfallen zu lassen, bietet sich folgender Trick an: Die Szene wird zunächst um $-(\text{augenhoehe} - \text{powerwallhoehe}/2)/\text{verhaeltnis}$ in y-Richtung translatiert, damit der Ursprung auf Augenhöhe liegt. (Die Skalierung mit $1/\text{verhaeltnis}$ ist aus dem gleichen Grund wie bei der Errechnung von offset aus augdist anzuwenden.) Nun müssen die Projektionsparameter *unten* und *oben* noch angepasst werden, da die zu projizierenden Bereiche oberhalb und unterhalb des Zentrums nicht mehr notwendi-

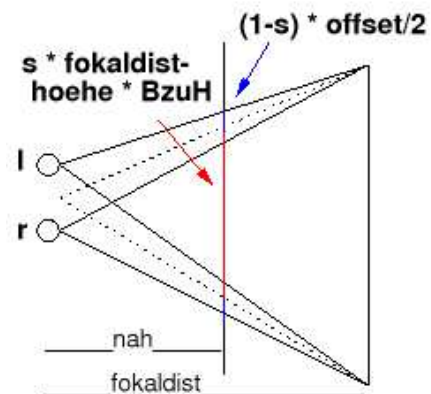


Bild 11: Berechnung von *links* und *rechts*

gerweise identisch sind, die Projektion also auch in vertikaler Richtung asymmetrisch wird. *unten* und *oben* errechnen sich durch $z = \text{augenhoeh}/\text{powerwallhoehe}$, $\text{help} = s * \text{fokaldisthoehe}$, $\text{unten} = -\text{help} * z$ und $\text{oben} = \text{help} * (1-z)$. Wenn diese Art der Projektion auch in großen Teilen der realen Wahrnehmung entsprechen mag, so ist sie dennoch keine exakte Nachbildung, sondern lediglich eine Annäherung, da beispielsweise die Projektionsachsen in der Realität nicht fest als senkrecht zur y-Achse der Szene angenommen werden können.

Eine übersichtliche Darstellung des kompletten Verfahrens in Form eines Python-Skripts zur Erzeugung der Halbbilder aus einer 3D-Szene mittels der Netzwerk- und Graphik-Software Chromium ist in Abschnitt 4.3.1 zu finden.

Nach der möglichst korrekten Berechnung der Halbbilder ist es wichtig, diese richtig darzustellen oder richtig zu registrieren, wie in Stereoskopie-Fachkreisen gesagt wird, denn verzerrte oder verschobene Halbbilder können unter Umständen nicht oder nur schwer zu einem 3D-Eindruck verknüpft werden. Deshalb müssen die Geräte, die die Halbbilder erzeugen, also beispielsweise zwei Projektoren genau eingestellt werden. Die Halbbilder sollten möglichst keine Helligkeitsunterschiede aufweisen, die gleiche Größe haben, ohne Verzerrung, an genau der richtigen Stelle (bei der Powerwall z.B. exakt aufeinander) dargestellt werden und so, dass sie mittels passiver Brillen scharf voneinander getrennt werden können und kein Ghosting auftritt (z.B. durch exakte Polarisation).

4 3D Programmierung

Dieses Kapitel soll keinen Überblick über alle Methoden der 3D-Programmierung geben, sondern lediglich die Programme und Werkzeuge, die im Rahmen dieser Arbeit verwandt wurden, kurz vorstellen.

4.1 3D Modellierung

Die Modellierung von 3D-Objekten und -Szenen kann entweder mit Hilfe von graphischen Primitiven und eigenen Datenstrukturen von Hand im Programmcode und in Hilfsprogrammen erfolgen. Ein solches Programm ist beispielsweise *3D-Studio*. Mit diesem 3D-Modellierungs-, Animations- und Raytracing-Programm werden Objekte im *3DS-Format* erzeugt. Dank der großen Beliebtheit von *3D Studio* sind solche Objekte bereits in relativ großer Vielfalt und teilweise frei im Internet zu finden. Methoden, die die wichtigsten Informationen der *.3DS-Files* lesen und als OpenGL-Objekte importieren können, finden sich ebenfalls im Internet. Für das im Rahmen dieser Arbeit erstellte Demonstrationsprogramm wurden zwei fertige *3DS-Objekte* mit Methoden aus [11] in OpenGL-Objekte verwandelt.

4.2 OpenGL

OpenGL [9] ist eine von Silicon Graphics entwickelte Hardware- und betriebssystemunabhängige Bibliothek, die es als Interface zur Graphikhardware ermöglicht, dreidimensionale Graphiken zu erstellen, zu bearbeiten und interaktiv darzustellen. (GL steht für "Graphics Library".) Grundlegende Tipps und Tutorials zur Erstellung von 3D-Animationen mit OpenGL sind zahlreich im Internet zu finden, beispielsweise unter [10] oder [11].

Das OpenGL-Graphik-System funktioniert wie eine Zustandsmaschine: Es gibt zahlreiche Zustände und Modi, die gesetzt werden können und so die Arbeitsweise des Systems beeinflussen. Ein weiteres Charakteristikum des Systems ist die Pipeline-Abarbeitung der Befehle: Zunächst wird die Geometrie als mathematische Beschreibung intern dargestellt. Diese wird dann mittels Transformationen in 3D angeordnet und in verschiedenen Schritten je nach Zu-

stand der entsprechenden Systemvariablen mit Farbe, Licht und Textur versehen. Schließlich wird die Geometrie rasterisiert, also in Werte im Framebuffer und zuletzt Pixel auf dem Bildschirm verwandelt.

OpenGL funktioniert nach dem client-server-Modell, wobei der Client das Programm ausführt und Graphik-Befehle erzeugt und der Server diese Befehle erhält und ausführt, was die Ausführung von OpenGL-Applikationen in einem Netzwerk ermöglicht. Beim Einprozessor-Betrieb ist dieser gleichzeitig Client und Server. Das Drei-Rechner-Netzwerk, das für diese Arbeit verwandt wurde, ist über ein mittels der Netzwerk-Software Chromium kommunizierendes Netzwerk verbunden, das nach diesem Modell arbeitet und im Abschnitt 4.3 beschrieben wird.

4.2.1 OpenGL-Methoden zur Stereographie

Wenn der Stereoeffekt bereits in der Software realisiert werden soll und dies durch die Graphik-Hardware des Rechners unterstützt wird, kann das Erzeugen des Stereopaars in OpenGL mit speziell dafür vorgesehenen Befehlen durchgeführt werden. Zunächst muss mit `glutInitDisplayMode(GLUT_DOUBLE|GLUT_RGB|GLUT_DEPTH|GLUT_STEREO)` der Stereo-Modus aktiviert werden. Damit ist es, wenn auch die Graphikkarte das erlaubt, möglich, in vier verschiedene Puffer zu rendern: `GL_BACK_LEFT`, `GL_BACK_RIGHT`, `GL_FRONT_LEFT` und `GL_FRONT_RIGHT`. Mittels `glDrawBuffer(...)` kann einer dieser vier Puffer explizit als aktueller Zeichen-Puffer ausgewählt werden. Für die zur Erzeugung der Stereobilder notwendige asymmetrische perspektivische Projektion hält OpenGL den Befehl `glFrustum(left, right, bottom, top, near, far)` bereit, der die Ausdehnung des Projektions-Pyramidenstumpfs durch die sechs Parameter festlegt. `left`, `right`, `bottom` und `top` sind die Abmessungen der Nah-Clipping-Ebene um den Nullpunkt der xy-Ebene, `near` ist der Abstand der Nah-Clipping-Ebene und `far` der Abstand der Fern-Clipping-Ebene vom Projektionszentrum. Diese Software-Stereo-Methode bietet sich vor allem bei Ein-Rechner-Anwendungen mit Stereo-Ausgabegeräten wie Head-Mounted-Displays oder 3D Bildschirmen an. Wenn jedoch wie in der Anwendung der Powerwall zwei Projektoren sinnvollerweise mit je einem Rechner angesteuert werden, also ein Mehrrechner-System vorliegt, bei dem ein Rechner das linke Halbbild und ein anderer das rechte Halbbild erzeugt, bieten sich andere Methoden an. Der Vorteil der Software-Stereo-Lösung liegt jedoch darin, dass die Stereo-Parameter auf einfache Weise interaktiv verändert werden können.

4.3 Chromium

Netzwerke aus handelsüblichen PCs mit Graphikkarte werden in zunehmendem Maße als Hochleistungs-Graphikcomputer benutzt. Auch für die in dieser Arbeit beschriebene Powerwall wird von einem Drei-Rechner-Netzwerk gesteuert. Ermöglicht wird dies durch Chromium [12], ein Open Source Interface zur Unterstützung verteilten Rechnens von Graphik-Programmen.

Dabei sind verschiedene Arten von Parallelität und viele Konfigurationen möglich. Sowohl Graphik-Programme, die für verteiltes Rechnen optimiert sind, als auch beliebige andere Programme können über ein Chromium-Netzwerk parallel verarbeitet werden. Es ist möglich ganz verschiedene Arten von Ausgabegeräten wie große Projektionswände, die in Kacheln mit je eigenem Projektor aufgeteilt sind, Ein-Monitor-Systeme oder Stereosysteme, mit Chromium anzusteuern. Außerdem kann man Chromium auch auf einem einzigen Rechner benutzen und die Parallelität lediglich simulieren.

Ein Chromium-Netzwerk besteht aus Knoten, die als gerichteter azyklischer Graph angeordnet sind. Jeder Knoten ist ein Prozess, der SPUs (Stream Processing Units) lädt und

als Kette nacheinander anordnet. Graphik-Befehle, die diesen Knoten erreichen müssen die Kette der SPUs durchlaufen und werden je nach Art und Funktionalität der SPUs einfach weitergereicht, abgewandelt oder abgearbeitet. Es gibt beispielsweise SPUs, die Graphikobjekte sortieren, solche die Transformationen vornehmen und solche, die die gerade durchgeführten Befehle als Text ausgeben. Am Ende einer SPU-Kette sollte sich immer eine Render-SPU befinden, die bis dahin noch nicht abgearbeitete Befehle ausführt. Die SPUs, die ihre Eigenschaften auch an Kind-SPUs vererben können, implementieren Teile der OpenGL-Bibliothek. Implementierte Befehle werden von der SPU abgearbeitet, die anderen werden weitergeleitet. Dadurch werden die OpenGL-Befehle im Chromium-Netzwerk nicht direkt mit Hilfe der Standard-OpenGL-Bibliothek ausgeführt, sondern weitergeleitet, möglicherweise verändert und dann schließlich in einem Chromium-Knoten zum Teil durch Befehle der Chromium internen OpenGL-Bibliothek dynamisch ersetzt oder an System-OpenGL-Bibliothek gesandt und abgearbeitet.

Die Wurzel des Chromium-Netzwerks, von dem alle OpenGL-Befehle ausgehen, wird *app*-Knoten (von engl. application, Anwendung) oder auch Client-Knoten genannt, alle übrigen sind *server*-Knoten. Der Aufbau eines Chromiums-Netzwerks, mit allen Knoten, verschiedenen SPUs, sowie deren Eigenschaften werden in Konfigurations-Dateien festgelegt. Diese sind Python-Scripts³, die alle gewöhnlichen Python-Konstrukte erlauben, wodurch es möglich ist, die Chromium-Konfiguration an sich ändernde Bedingungen anzupassen. Um ein Graphik-Programm mit Hilfe von Chromium auszuführen, wird ein sogenannter *mothership*-Python-Prozess, dem die gewünschte Konfigurationsdatei und andere Parameter wie beispielsweise der Name des auszuführenden Programms übergeben wird, gestartet. Dann werden je nach Konfiguration noch die Prozesse für den app-Knoten mittels *crappfaker* und ein oder mehrere server-Knoten mittels *crserver* gestartet. Der mothership-Prozess hat Überblick über den Aufbau des Netzwerks und kann den Knoten Aufträge senden.

Bedauerlicherweise funktioniert Chromium noch nicht fehlerfrei. So ist beispielsweise die Verwendung von Texturen in Display-Listen (OpenGL Konstrukt zur effizienten Befehlsabarbeitung) in Verbindung mit einer TileSort-SPU problematisch. Die Schwierigkeiten, die im Rahmen dieser Arbeit mit Chromium entstanden sind, werden in Kapitel 5 genauer beschrieben.

4.3.1 Chromium-Techniken zur Stereographie

Möchte oder kann mensch den Stereoeffekt nicht in Software (siehe 4.2.1) realisieren, so bietet Chromium die Möglichkeit, den Stereoeffekt nachträglich mittels Befehlsfolgenmodifikation in einer SPU zu erzwingen. Dazu wird die *tilesort*-SPU missbraucht. Diese wird normalerweise genutzt, um den Darstellungsbereich großer Ausgabeflächen zu unterteilen, die Geometrie nach Zugehörigkeit zu den einzelnen Bereichen zu sortieren und diese dann für jeden Bereich an einen anderen Knoten weiterzuleiten, um die einzelnen Teile der Ausgabefläche verteilt auf verschiedenen Rechnern zu erzeugen. Im Fall der Stereobildgenerierung wird die *tilesort*-SPU so konfiguriert, dass zwei Bereiche, die beide die Größe des gesamten Darstellungsbereichs haben, entstehen. Die gesamte Geometrie wird also an beide auf den Knoten der SPU folgende Knoten geschickt, wobei jedoch vorher eine Modifikation der Graphikbefehle entsprechend der asymmetrischen perspektivischen Projektion zur Erzeugung von Halbbildern erfolgt, sodass der eine Knoten die Befehle für das rechte und der andere die Befehle für das linke Halbbild erhält.

Die für die im Rahmen dieser Arbeit benutzte drei-Rechner-Powerwall-Konfiguration nötige Funktionalität lässt sich wie folgt beschreiben:

An das zentrale Chromium-Objekt `cr = CR()`, mit dem der Mothership-Prozess gestartet

³Python ist eine objektorientierte Programmiersprache, die mit Perl oder Java verglichen werden kann

werden kann, wird zunächst ein app- oder Client-Knoten `clientNode = CRApplicationNode()`, der die SPU `tilesortspu = SPU('tilesort')` enthält, angehängt. Durch das Setzen der ViewMatrix dieses Knotens kann das Projektionszentrum des ursprünglichen Programms modifiziert werden, indem der angegebene Wert (Translation zum rechten und linken Projektionszentrum) auf den durch das Programm vorgegebenen multipliziert wird. Außerdem wird eine Projektions-Matrix angegeben, die die vom Programm vorgeschriebene Projektionsmatrix ersetzt. Dann werden zwei Server-Knoten `node = CRNetworkNode()` (für das rechte und linke Halbbild je einen), die jeweils eine Render-SPU enthalten an das Chromium-Objekt angehängt und mit der Tilesort-SPU verknüpft.

Wegen der zentralen Bedeutung der Befehlsmodifikation zur Erzeugung von Stereobildern durch das Chromium-Konfigurations-Skript, ist dieses für das im Kapitel 5 beschriebene Beispielprogramm Murmelbahn hier vollständig dargestellt. Das Grundgerüst für dieses Skript stammt von einem im Chromium-Paket enthaltenen, jedoch mit Fehlern behafteten Beispiel-Skript für Stereographie.

Mit dem hier vorgestellten Konfigurations-Skript ist im Gegensatz zur Software-Stereo-Lösung keine interaktive Veränderung der Stereoparameter möglich. Chromium bietet zwar das Paket *CRUT* zur Interaktion mit Programmen an, jedoch ist auch dieses Chromium-Feature für das Betriebssystem Windows noch mit Fehlern behaftet und deshalb effektiv nicht nutzbar.

4.3.2 Konfigurations-Skript zur Erzeugung von Stereopaaren mittels Chromium

```
import sys #Chromium-interne Hinweise
sys.path.append( "../server" )
from mothership import *
import crmatrix

FORCE_STEREO = 0 #Flag: 1-> erzwungenes Stereo, 0-> Software-Stereo

#Annahmen:
#-Betrachtungspunkt ist Ursprung, Blickrichtung neg.
    z-Achse, hoch-Vektor: pos. y-Achse
#-Szene liegt auf neg. z-Achse, Dimensionen bekannt
#-Höhe der Szene in Fokaldistanz sollte in gleicher Größenordnung
    wie Fokaldistanz sein
#Allgemeine Parameter (für alle PowerWall-Anwendungen gleich)
augdist = 0.064 #Einheiten in Meter
TileWidth = 1280 #Pixel
TileHeight = 1024 #Pixel
powerwallhoehe = 2.3 #Meter
augenhoehe = 1.7

#Murmelhahndimensionen: Zentrum 20 entfernt, darum 6 in jede Richtung,
    inklusive Rahmen: y-Ausdehnung von 20
#Anwendungsspezifische Parameter (für jede Anwendung anpassen)
#Aufrufnahme der Anwendung inklusive Übergabe-Parameter
DEMO = "Debug\\murmelbahn2 -translate 1.1"
crbindir = r'D:\powerwalltests\Murmelbahn2' #Verzeichnis der Anwendung
nah = 7.0
fern = 40.0
fokaldist = 20.0
fokaldisthoehe = 20.0

for arg in sys.argv[1:]:
    if arg == "-force": #chromium erzwingt Stereo durch Manipulation der Projektion
```

```

        FORCE_STEREO = 1
    elif arg[0] != "-":
        DEMO = arg
        print "--> Note: you may have to tweak the projection parameters!"
    else:
        print "Unknown option %s" % arg
        sys.exit(1)
if FORCE_STEREO:
    print "--> Forcing %s into stereo mode via config file." % DEMO
else:
    print "--> Running %s in conventional stereo mode." % DEMO
cr = CR()
cr.MTU( 1024*1024 )

tilesortspu = SPU("tilesort")
tilesortspu.Conf('bucket_mode', 'Broadcast') #vollständige Geometrie weiterleiten
tilesortspu.Conf('stereo_mode', 'Passive')

clientnode = CRApplicationNode( '192.168.11.120' )
clientnode.StartDir( crbindir )
if FORCE_STEREO:
    dummy = 42
else:
    DEMO += " -s" #Software-Stereo aktivieren
clientnode.SetApplication( DEMO )
clientnode.AddSPU( tilesortspu )

EyeNames = [ "links", "rechts" ]
ipAddresses = [ '192.168.11.121', '192.168.11.122' ]

# erst linkes, dann rechtes Auge
for eye in range(2):
    renderspu = SPU( 'render' )
    renderspu.Conf( 'fullscreen', 1)

    node = CRNetworkNode( ipAddresses[eye] )
    node.AddTile( 0, 0, TileWidth, TileHeight)

    if FORCE_STEREO:
        verhaeltnis = powerwallhoehe / fokaldisthoehe
        offset = augdist / verhaeltnis
        z = augenhoehe/powerwallhoehe

        v = crmatrix.CRMATRIX()
        if eye == 0:
            v.Translate(+offset/2, -(augenhoehe-powerwallhoehe/2)/verhaeltnis, 0)
        else:
            v.Translate(-offset/2, -(augenhoehe-powerwallhoehe/2)/verhaeltnis, 0)
        node.Conf('view_matrix', v.ToList())

    BzuH = float(TileWidth) / float(TileHeight)
    s = nah / fokaldist

    help = s * fokaldisthoehe #Nahhoehe
    unten = -help * z #"zweidrittelsicht"

```

```

oben = help * (1-z)

p = crmatrix.CRMATRIX()
if eye == 0: #links
    links = 0.5 * s * (-fokaldisthoehe * BzuH + offset);
    rechts = 0.5 * s * (fokaldisthoehe * BzuH + offset);
else: #rechts
    links = 0.5 * s * (-fokaldisthoehe * BzuH - offset);
    rechts = 0.5 * s * (fokaldisthoehe * BzuH - offset);
p.Frustum(links, rechts, unten, oben, nah, fern)
node.Conf('projection_matrix', p.ToList())
node.Conf('stereo_view', EyeNames[eye])
node.AddSPU( renderspu )
cr.AddNode( node )
tilesortspu.AddServer( node, protocol='tcpip', port=7000 + eye)
cr.AddNode( clientnode )
cr.Go()

```

Gestartet werden kann die Anwendung mit erzwungenem Stereo-Effekt, indem folgendes ausgeführt wird:

1. Das Mothership mit dem Konfigurations-Skript starten
`python D:\powerwalltests\murmelmahn2\murmelmahn23.conf -force`
2. Auf den Server-Rechnern jeweils einen Chromiumserver starten mit
`crserver`
3. Auf dem Rechner, auf dem auch das Mothership läuft, den Chromium-"Applicationfaker" starten
`crappfaker`

5 Stereoskopie Demonstrator

5.1 Ideen und Zweck

Ziel des praktischen Teils dieser Arbeit war die Implementierung eines Stereoskopie Demonstrators. Dieses 3D-Graphik-Programm sollte für die Anwendung der PowerWall auf der unter Abschnitt 3.4.2 beschriebenen Rechner- und Projektionsanordnung ausführbar sein und ergonomische, stereographische, bewegte Bilder und möglichst eindrucksvolle, immersive Tiefeneffekte erzeugen. Dabei ging es auch darum, herauszufinden, wie der Tiefeneindruck verstärkt werden kann und welche Arten der Darstellung den Tiefeneindruck mindern oder gar völlig zerstören.

5.2 Implementierung und Beurteilung der Stereoeffekte

Wie bereits in anderen Abschnitten beschrieben, ist es auf verschiedenen Wegen möglich, Stereobildpaare für die Projektion auf die PowerWall zu erzeugen. Da wegen der unter 3.4.2 beschriebenen Drei-Rechner-Konfiguration sowieso Kommunikation zwischen diesen Rechnern notwendig ist und eine Synchronisation der Erzeugung von rechtem und linkem Halbbild auf verschiedenen Rechnern problematisch ist, wurde die im Abschnitt 4.3.1 erläuterte Variante des Chromium-Netzwerks gewählt, bei der ein gewöhnliches Graphik-Programm mittels der Chromium-Konfigurations-Datei einmal für das rechte und einmal für das linke Halbbild entsprechend manipuliert wird und diese veränderten Graphikbefehle an den jeweiligen Rech-

ner weitergeleitet werden.

Eine zentrale Bedeutung bei dieser Art der Stereobildpaar-Erzeugung kommt dabei der in der Konfigurationsdatei festgelegten Berechnungsweise des rechten und linken Halbbilds sowie der Wahl der dafür nötigen Parameter zu. Für beides werden in der Literatur sehr unterschiedliche Hinweise gegeben, was auch daran liegt, dass Verfahren und Parameter oft durch Ausprobieren für die jeweilige Anwendung angepasst wurden, sodass beispielsweise mit unrealistischen Parametern wie einem Augenabstand von zwanzig Metern gearbeitet wird, wenn dies zu dem gewünschten Effekt führte. Mit der Zielsetzung die vielen, sich gegenseitig beeinflussenden Parameter zu entwirren und gesuchte Parameter von gegebenen Parametern logisch abzuleiten, um ein Verfahren zu entwickeln, das für verschiedene Szenen automatisch ein im Sinne des Stereoeffekts und der Ergonomie korrektes Stereobildpaar erzeugt, habe ich mich für die im Abschnitt 3.5 beschriebene Vorgehensweise entschieden und diese in der unter 4.3.1 erläuterten Weise in einer Chromium-Konfigurations-Datei umgesetzt.

Da jedoch der entstehende Stereoeffekt in erheblichem Maße davon abhängt, welche Geometrien wie von der Graphikapplikation erzeugt werden, soll nun noch auf die eigentliche, mit Hilfe der OpenGL-Bibliothek geschriebene Software des Stereoskopie-Demonstrators eingegangen werden. Nach einigen Experimenten mit verschiedenen 3D Szenen haben sich zwei verschiedene Applikationen, in denen die gesammelten Erfahrungen eingeflossen sind, herauskristallisiert.

Die *Murmelbahn*, von der Bild 12⁴ eine Ansicht zeigt, ist eine spiralförmige Struktur, auf der eine Kugel herabrollt. Zunächst verfügte die Applikation weder über den dunkleren Rahmen noch die umgebende Kiste, wodurch es schwierig war, die Position der Bahn im Raum festzustellen. Durch den dunklen Rahmen auf der Fokaldistanz-Ebene hat das Auge einen Referenzpunkt. Eine Überdeckung des Rahmens durch Teile der Murmelbahngeometrie bietet zudem einen zusätzlichen Tiefenhinweis. Der Rahmen sollte in einer dunklen Farbe gehalten sein, damit er sich möglichst gut in den real gegebenen Rand der PowerWall einfügt und der Eindruck entsteht, dass Objekte, die über den Software Rand ragen, tatsächlich nach vorne über den realen PowerWall-Rahmen ragen würden.

Als hilfreich für die Tiefenwahrnehmung hat sich außerdem ein Einfärbung der Innenseite der Bahn in einer anderen Farbe sowie die Beleuchtung der Szene mit einem Licht von schräg oben und einem von schräg unten erwiesen. In Bild 12 ist die Bahn durch einen Befehlszeilenparameter ein wenig in Richtung der positiven z -Achse translatiert, da der Stereoeffekt von aus der Projektionsfläche herausragenden Objekten beeindruckender ist, als der von sich nach hinten ausdehnenden Szene-Bereichen, was jedoch auch nicht übertrieben werden sollte. Außerdem ist bei diesem Bild zu sehen, welcher Effekt eintritt, wenn das Projektionszentrum nicht in der Mitte belassen wird, sondern nach oben verschoben wird, um dem Verhältnis von Augenhöhe und PowerWall-Höhe und der daraus resultierenden realistischen Sicht auf die Szene gerecht zu werden. Der Stereoeffekt wird dadurch noch einmal stärker. Andererseits kann es zu Irritationen kommen, wenn die betrachtenden Personen eine von der angenommenen Augenhöhe stark unterschiedliche Beobachtungsposition haben.



Bild 12: Murmelbahn mit Projektionszentrum auf Augenhöhe

⁴Die hier dargestellten Bilder der Stereoapplikationen sind Screenshots von einem gewöhnlichen PC, da das Abphotographieren von der Darstellung auf der Powerwall wegen der Übereinanderprojektion von rechtem und linkem Abbild nicht möglich ist und eine nachträgliche Bildmanipulation zwar beeindrucken aber den Stereoeffekt letztlich trotzdem nicht transportieren kann

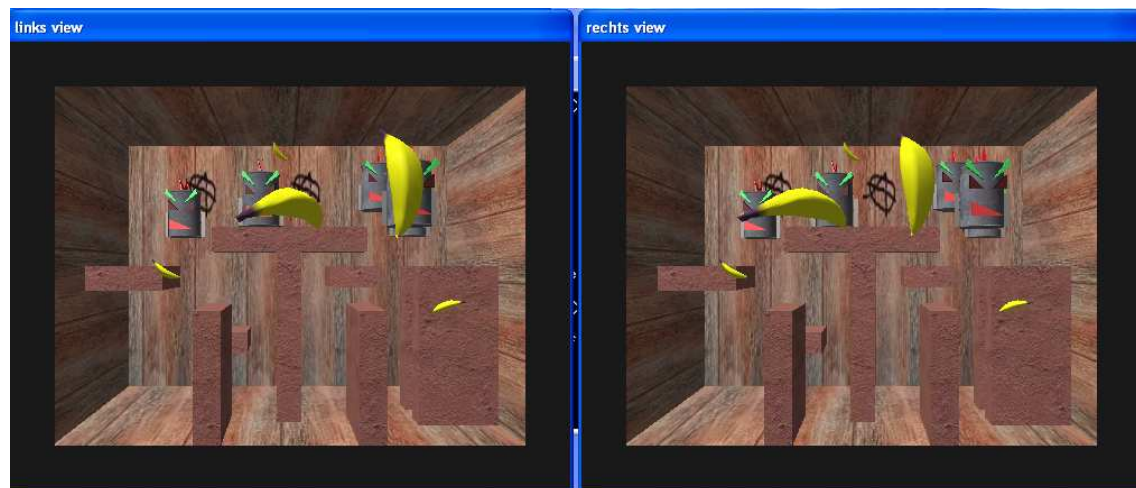


Bild 13: Linkes und rechtes Bild der Stereo-Animation

Die zweite Applikation, die als Stereographie Demonstrator genutzt werden kann ist in Abbildung 13 als rechtes und linkes Halbbild einer Momentaufnahme gezeigt. Hierbei handelt es sich um eine längere Animation, die verschiedene Stufen durchläuft und es so erstens ermöglicht eine langsame Gewöhnung der betrachtenden Personen an die Szene zu gewährleisten und zweitens verschiedene Effekte nacheinander demonstrieren kann. Wie bei der Murmelbahn-Anwendung wird dem Auge als Bezugspunkt für die Fokaldistanz-Ebene ein dunkel eingefärbter Rahmen geboten. Die Animation beginnt mit einer sich in abgewandelten Sinuswellen bewegenden Nurbs-Fläche (aus der OpenGL Utility Bibliothek (GLU)), die als sich öffnender Vorhang zu einem dahinter liegenden Raum fungiert. Wichtig für die räumliche Wahrnehmung des Vorhangs sind dabei die Lichteffekte auf dessen Oberfläche. Der sich dahinter auftuende Raum ist komplett texturiert, was durch die Größe der Textur sowie insbesondere durch den Verlauf der Holzmaserung in Richtung Fluchtpunkt zum Tiefeneindruck beiträgt. Außerdem vermindern die Texturen sonst zu stark auftretende Kontrastunterschiede, die zu Ghosting führen würden. Im Vergleich zur Murmelbahn, bei der fast nur gekrümmte Elemente auftreten, kann festgestellt werden, dass die ausschließlich entlang der Koordinatenachsen ausgerichteten Szene-Elemente in der perspektivischen Darstellung einen starken Raumeffekt ergeben.

Wenn die als 3DS-Objekte geladenen Tonnen sich vor-, hinter und über diese Szene-Elemente bewegen trägt zusätzlich die Bewegungsparallaxe zum Tiefeneindruck bei. Nach einer gewissen Zeit, die der betrachtenden Person ermöglichen soll, sich an die Ausdehnung der Szene hinter der PowerWall zu gewöhnen, beginnt das Eindringen der Tonnen und später der ebenfalls als 3DS-Objekte geladenen Bananen in den vorderen Bereich der Szene. Hierbei, wie beim Eindringen beliebiger Objekte in den Raum vor der Projektionswand, ist zu beachten, dass nur die Teile der Szene dargestellt werden können, die sich im Projektions-Pyramidenstumpf, der auf den Beobachtungspunkt zuläuft, befinden. Deshalb fliegen die Bananen von den Tonnen genau auf das Projektionszentrum zu. Die Animation testet dabei zwei verschiedenen Modi: Zunächst nähern sich die Bananen den betrachtenden Personen auf direkter Flugbahn und verschwinden dann plötzlich, kurz vor dem Projektionszentrum. Im zweiten Modus bewegen sich die Bananen in einer flachen Parabel-Kurve auf die z-Achse zu, bis kurz vor den Betrachtungspunkt, dann fliegen sie in einer sehr steilen Parabel-Kurve von der z-Achse weg und verlassen den Projektions-Pyramidenstumpf an den seitlichen Rändern. Der zweite Modus wirkt auf die meisten BetrachterInnen eindrucksvoller, obwohl die Flugobjekte nicht so dicht an sie herankommen, wie im ersten Modus, da das plötzliche Verschwinden der Bananen direkt vor dem Beobachtungspunkt, ohne dass eine Kollision bemerkbar wäre, die Immersion stärker stört, als das Verlassen des Blickfeldes über die Seitenflächen.

Der Unterschied der Parallaxe von Objekten im Vordergrund und Objekten im Hintergrund so stark, dass es bemerkbar ist, wie das Auge die Konvergenz beim Betrachten der verschiedenen Szenenabschnitte verändern muss, was einen starken Tiefeneindruck erzeugt. Durch das extreme Eindringen der Bananen in den Raum vor der Projektionswand wird eine sehr große Fokustiefe (scharf wahrgenommener Bereich) aufgespannt, was in nicht unerheblichem Maße unrealistisch ist. Da es sich hierbei jedoch nur um schnell durch den Raum fliegende Gegenstände und keine dauerhaft dort stationierten Szene-Teile handelt, hat das visuelle System wenig Gelegenheit, sich an diesem Missstand zu stören.

Die Anwendung wurde zunächst mit Display-Listen erstellt. Diese verbinden und optimieren die darin enthaltenen Graphik-Befehle beim Erstellen der Liste. Wird die Liste dann aufgerufen, so können diese Befehle sehr effizient ausgeführt werden, was insbesondere für die Mehrfachdarstellung der Tonnen und Bananen sinnvoll ist. Da die Chromium-TileSort-SPU damit erheblich Probleme hat und insbesondere Display-Listen in Verbindung mit Texturen nicht realisierbar waren, musste die Anwendung von Display-Listen stark eingeschränkt werden. Um den damit verbundenen Performance-Verlust ein wenig auszugleichen, wurden Vertex-Arrays für die 3D-Objekte erstellt. Diese werden wie die Display-Listen einmal erzeugt und können dann durch Verschiebung des lokalen Koordinatensystems zur Erzeugung der Objekte an verschiedenen Stellen im globalen Koordinatensystem genutzt werden, wobei der Vorteil gegenüber des Renderings ohne Vertex-Arrays darin besteht, dass mit bedeutend weniger Funktionsaufrufen das Objekt entsteht. Dies hat das Problem der Texturen in Chromium gelöst und beim Start über ein Chromium-Konfigurations-Skript für einen Einzelrechner problemfrei funktioniert, jedoch ergab sich bei der Ausführung der Applikation über das Netzwerk der PowerWall eine neue Schwierigkeit: Die Daten der Vertex-Arrays verbleiben beim Client, der diese für die Erstellung jedes Frames zu beiden Servern senden muss, wodurch die Frame-Rate unbefriedigend niedrig ausfiel. Eine Lösung für dieses Problem bieten die *Vertex Buffer Objects (VBOs)*, die in der Erweiterung zu OpenGL für moderne Graphikhardware mittels `#include <GL\gltext.h>` verfügbar sind. VBOs erstellen Objekte, die Daten wie Vertex-Arrays verwalten. Diese Objekte erhalten eine ID, über die auf die Objekte zugegriffen werden kann, ähnlich wie bei Texture-Objekten. Der Vorteil gegenüber Vertex-Arrays ist, dass die Daten nun auf der Server-Seite gespeichert werden und der Client nur die VBO-ID senden muss, um dem Server den nötigen Graphik Befehl mitzuteilen, was das Problem der niedrigen Frame-Rate im Chromium-Netzwerk gelöst hat.

Durch das Laden einer Textur, die erst im Laufe der Animation benötigt wird, bereits zu Beginn auf einer nicht sichtbaren Fläche konnte außerdem die kurze Verzögerung beim erstmaligen Erscheinen der Textur vermieden werden (engl.: texture-preloading).

5.3 Kritische Beurteilung und mögliche Lösungen

Es ist unbefriedigend, dass für eine korrekte Texturdarstellung über das Chromium-Netzwerk auf Display-Listen verzichtet werden muss, da die Darstellungsgeschwindigkeit darunter sehr leidet. Eine Lösung wäre es, eine spezielle Stereo-SPU für Chromium zu schreiben, die nur die dafür nötige Funktionalität und nicht alle weiteren, fehlerträchtigen Möglichkeiten der Tilesort-SPU bietet.

Ein großer Nachteil des Stereoskopie Demonstrators auch in Bezug auf die Immersion ist es, dass noch keine Interaktion möglich ist und eine Bewegung des Beobachtungspunktes zu Verzerrungen der Szene und nicht zu dem gewünschten Blickpunktwechsel führt. Insbesondere Objekte, die in horizontaler Ausrichtung etwa auf Augenhöhe liegen, können wegen der fehlenden vertikalen Parallaxe nur durch Neigung des Kopfes oder Auf- und Abbewegung des Beobachtungspunktes räumlich wahrgenommen werden. Die Auf- und Abbewegung des Kopfes wäre durch geeignete Interaktionsmechanismen realistisch nachbildbar, die Neigung

des Kopfes ist jedoch problematischer künstlich anzunähern, da die horizontale Parallaxe der Halbbilder in vertikale Parallaxe übergehen müsste und gleichzeitig die Zuordnung der Bilder zu dem richtigen Auge gewährleistet sein muss.

Noch realitätsnäher und immersiver könnte die Animation durch das Darstellen von Schatten und durch Kollisionsbehandlung werden. Andererseits demonstriert gerade das Sichtbarwerden der gelegentlichen Objektverschneidungen die Qualität der Stereosicht. Außerdem könnte das Hinzufügen von zu der Animation passenden Soundeffekten die Immersion in erheblichem Maße steigern.

Eine Herausforderung wäre es weiterhin, gerade bei Szenen mit großer z -Ausdehnung, die Fokustiefe, also den scharf dargestellten Bereich zu beschränken und andere Bereich, mittels jittering unscharf darzustellen. Wie bereits in Kapitel 2.3.1 erläutert, ist dies jedoch schwer ohne Interaktion mit der Akkomodationsbewegung der Augen möglich.

6 Schlusswort

Mit Hilfe der PowerWall und des Stereoskopie Demonstrators lässt sich zeigen, welche Möglichkeiten eine Stereoansicht bietet. Dass diese mehr Informationen als eine herkömmliche Darstellung transportiert, wurde bei der Betrachtung der Animation mit den Tonnen auf der PowerWall festgestellt, als kleine logische Fehler wie das Schneiden von Objekten miteinander oder das Springen auf nicht vorhandene Mauerstücke deutlich erkennbar wurden, die bei der Entwicklung auf einem gewöhnlichen PC nicht wahrnehmbar waren. Wenn die Anfangsschwierigkeiten wie die Findung einer allgemeingültigen Methode zur Erzeugung von Halbbildern, wozu diese Arbeit beigetragen haben will, oder die Probleme mit Chromium überwunden sind, stehen sicherlich viele neue Möglichkeiten offen, die in den unterschiedlichsten Anwendungsgebieten Nutzen finden könnten. Es sollte jedoch bei all der Euphorie über die realistischen Darstellungen bedacht werden, dass die Realität immer am realistischen bleiben wird.

Referenzen

- [1] D. McAllister. Display Technology: Stereo & 3D Display Technologies, North Carolina State University
- [2] L. Hodges. Computing Stereoscopic Views. Stereo Computer Graphics an Other True 3D Technologies, McAllister, Princeton University Press, 1993
- [3] M. Halle. Autostereoscopic display and computer graphics, ACM SIGGRAPH, 1997
- [4] P. Bourke. 3D Stereo Rendering Using OpenGL (and GLUT), <http://astronomy.swin.edu.au/pbourke/opengl/stereogl/>, 2002
- [5] <http://www.stereoskopie.de>
- [6] <http://www.stereoskopie.com>
- [7] <http://www.barco.com/> (Hersteller moderner Stereoskopie-Ausrüstung)
- [8] http://www.seereal.com/DE/products_principle_js.de.htm
- [9] OpenGL Programming Guide (The Red Book), Addison-Wesley, 1997
- [10] <http://nehe.gamedev.net/> (Tutorials und Tipps zur OpenGL-Programmierung)
- [11] <http://www.GameTutorials.com> (ebenfalls Tutorials u.a. zur OpenGL-Programmierung)
- [12] <http://www.chromium.sourceforge.net/doc/>