

Exposé zur Magisterarbeit

# Implementierung einer netzwerkfähigen interaktiven stereoskopischen Visualisierungsumgebung

Ralf Baumbach, Matrikelnummer 219063, baumbach@kgw.tu-berlin.de

Technische Universität Berlin

Fakultät I

Institut für Sprache und Kommunikation

Fachgebiet Audiokommunikation

## Zusammenfassung

Ziel der Arbeit ist die Implementierung einer stereoskopischen Visualisierungsumgebung, die in ein bestehendes Versuchsframework zur dynamischen Binauralsynthese eingebettet werden kann. Die zu entwickelnde Software soll Probanden über die Darstellung stereoskopischer Umgebungen ein höheres Maß an Immersion in die künstliche Szene ermöglichen. Dafür werden auf die Innenseite einer umliegenden Sphäre stereoskopische Panoramabilder projiziert. Die Echtzeit-Steuerung der Anwendung soll über Kommandozeilenparameter sowie über OSC (Open Sound Control) erfolgen. Außerdem soll die Generierung einfacher 3D-OpenGL-Szenen möglich sein, die z.B. für einfache binaurale Hörversuche verwendet werden können. Ein weiteres Ziel der Arbeit ist die Erarbeitung eines Modell-Verfahrens zur Erstellung stereoskopischer Panoramabilder, die für die Nutzung der Visualisierungsumgebung nötig sind. Dieses Modell-Verfahren, soll die weitgehend computerunterstützte Aufnahme und Nachbearbeitung von hochauflösenden Stereo-Panoramen ermöglichen, wie sie zur Verwendung mit der Visualisierungsumgebung benötigt werden.

## Einleitung und Fragestellung

Für die Durchführung dynamischer binauraler Hörversuche wird eine Visualisierungsumgebung benötigt, die in der Lage ist stereoskopische Ausgaben für verschiedene Ausgabegeräte (Head Mounted Display: eMagine 3D Visor Z800, Dual-DVI: PI Sight, Fernseher) zu erzeugen. Folgende Anforderungen ergeben sich aus der benötigten Integration in ein bestehendes Versuchsframework:

Kommunikation über OSC (Open Sound Control): Alle in Echtzeit benötigten Eingaben an die Anwendung müssen per OSC möglich sein. So sollen im Fall des HMD (Head Mounted Display) die vom Trackingmodul gelieferten Daten per OSC übergeben werden. Ebenso soll ein Wechsel des dargestellten Panoramabilds möglich sein. Die verwendeten Ports und OSC-Pfade sollen auch per Kommandozeilenparameter veränderbar sein, um das System noch flexibler zu gestalten.

Plattformunabhängigkeit: Die Faltungseengine *fwonder* läuft unter Linux, während die Verwendung des eMagin HMDs Windows XP voraussetzt, wobei ein Linux-Treiber im Beta-Stadium vorliegt. Im Rahmen dieser Arbeit wird auch versucht das Gerät unter Mac OS X anzusteuern. Die zu implementierende Visualisierungsumgebung sollte demnach möglichst plattformunabhängig sein. Aufgrund der weiten Verbreitung und der weitläufigen Unterstützung durch gängige Grafikkarten wird für die 3D-Umgebung die Grafik-Bibliothek OpenGL verwendet und als Programmiersprache Python.

Besonderheiten der Peripheriegeräte: Während das HMD die Bilddaten im Page-Flipping-Format erwartet, d.h. alternierend je ein Frame pro Auge, erwartet der Fernseher die Bilddaten als Halbbilder (Interlaced). Das Dual-DVI-Gerät PI Sight erwartet einen kompletten Stream pro Auge. Ein hoher Grad an Modularität erleichtert eventuelle Erweiterungen und die Integration neuer Ausgabegeräte.

Headtracking: Für die Ausgabegeräte für die ein Positions-Tracking des Betrachters sinnvoll ist, sollen die getrackten Kopfbewegungen (über OSC) in Form von Pan- und Tiltparametern die berechnete Orientierung der dargestellten Szene verändern. Der Betrachter befindet sich dabei im Zentrum einer Sphäre, auf deren Innenseite ein 360°-Panoramafoto projiziert wird. Nur Kopfeigungen und -drehungen sind gestattet, nicht aber eine freie Bewegung im Raum.

OpenGL-Szenen: Durch die Übergabe noch zu definierender Kommandozeilenparameter sollen einfache OpenGL-Szenen standardisiert erstellt werden können. Dies kann z.B. in der Form von Szenen-Beschreibungen geschehen, die in einer externen Datei definiert werden, so dass der Anwendung als Argument nur der Name der Szenen-Datei übergeben werden muss. Aus dieser Datei wird dann das zu generierende Modell geladen und gerendert.

In einem weiteren Teil der Arbeit soll eine Aufnahmestrecke erstellt und deren Realisierung beschrieben werden, anhand welcher sich Stereo-Panoramabilder erzeugen lassen. Aufgrund der besonderen Anforderungen an Präzision und Einzelbildanzahl für die Erzeugung eines einzelnen Stereo-Panoramabilds soll die Aufnahme-prozedur so weit wie möglich automatisiert sein.

## Stand der Forschung

Generell lassen sich zwei Arten von Virtual-Reality-Systemen anhand der Anzahl der gleichzeitig möglichen Teilnehmer unterscheiden. Während die Verwendung von HMDs nur einen Benutzer zulässt, erlauben Mehrbenutzer-Systeme wie z.B. CAVE-Systeme (Cave Automatic Virtual Environment) [2] die gleichzeitige Teilnahme und Interaktion mehrerer Benutzer. Im Rahmen dieser Arbeit wird eine Ein-Benutzer-Umgebung entwickelt.

## Stereoskopische Aufnahmen

Um jedem Auge eine eigene Perspektive auf die betrachtete Szene zu präsentieren, müssen für die Erstellung je eines stereoskopischen Fotos zwei monoskopische Einzelbilder aus verschiedenen Perspektiven aufgenommen werden. Am einfachsten geschieht diese mit Hilfe einer Stereokamera. Die so erzeugten monoskopischen Einzelbilder werden dem Betrachter kombiniert dargeboten um einen stereoskopischen Eindruck zu erzeugen. In welcher Art die Kombination der Einzelbilder erfolgt, hängt sowohl vom gewählten Medium, als auch vom Darbietungskontext ab. Ob beim Betrachter wirklich eine Tiefenempfindung entsteht hängt neben dem Betrachtungsmedium maßgeblich von der Orientierung der zur Aufnahme verwendeten Kameras ab. So müssen diese nicht nur einen gemeinsamen Fokuspunkt und den gleichen Abstand zur Projektionsfläche (die Parallele der Achse zwischen beiden Kameralinsen) aufweisen, sondern außerdem mit den gleichen Kameraparametern aufgenommen werden (Belichtungszeit, Blende, Farbempfindlichkeit, Belichtung).

Für eine überzeugende stereoskopische Darstellung sind überdies auch bestimmte Regeln bei der Komposition der betrachteten Szene zu beachten. So sollte ein zu großer Tiefenbereich vermieden werden, weil sonst die Szenenelemente nicht mehr als Ganzes wahrgenommen werden können. Die Folge ist eine isolierte Wahrnehmung einzelner Bildebenen. Dieser Effekt wird als Bildzerfall bezeichnet [6].

Der Aufwand bei der Erstellung stereoskopischer Panoramen unterscheidet sich deutlich von der Erstellung monoskopischer Panoramen. Für letztere genügt es, eine begrenzte Anzahl von Fotos auf einem drehbaren Stativkopf oder sogar frei Hand zu machen und diese dann mit einer geeigneten Software zusammenzufügen (sog. stitching). Neuere Digitalkameras haben oft bereits einen eingebauten Panoramamodus, um dem Fotografen die Panoramaerstellung zu erleichtern. Für stereoskopische Panoramen dagegen sollten maximal  $1^\circ$  Kamera-Rotation zwischen jedem der Einzelbilder liegen [1]. Dies ergibt bei einem  $360^\circ$  Stereo-Panorama (2 Bilder pro  $1^\circ$ -Rotation) bereits 720 Einzelbilder. Von jedem dieser Bilder wird dann ein dem Rotationsintervall entsprechend breiter Streifen herausgeschnitten. Die Kombination dieser Einzelstreifen ergibt das eigentliche Stereo-Panorama. Die hohe Anzahl der Einzelbilder macht eine automatisierte Aufnahme nötig, wie sie z.B. durch professionelle, motorisierte Schwenkköpfe ermöglicht werden. Möglich ist auch ein Eigenbau, oder die Verwendung eines Drehtellers.

Peleg et al. stellen 1999 eine Methode zur Erstellung von stereoskopischen Panoramen vor, die mit einer einzigen rotierenden monoskopischen Videokamera auskommt (Stereo Mosaicing) [4]. Diese Methode ist weniger aufwendig als die Aufnahme mit zwei Kameras oder einer Kamera auf einer Stereoschiene, hat aber den Nachteil, dass die Auflösung gängiger digitaler Videokameras im Vergleich zu gängigen digitalen Fotokameras geringer ist, und somit nicht die gleiche Bildqualität erzielt werden kann. Die Methode des Stereo Mosaicing ist für diese Arbeit nutzbar, indem statt einer Videokamera und kontinuierlicher Drehung, eine digitale Fotokamera mit diskreten Rotationswinkeln verwendet wird. Auf diese Weise lässt sich die Anzahl der benötigten Einzelbilder um die Hälfte reduzieren.

Ebenfalls von Peleg et al. stammt der Vorschlag eine Stereo-Panorama-Aufnahme mit einem optischen Aufbau zu erzeugen, so dass für ein Stereo-Panorama nur zwei Aufnahmen

notwendig sind und keine Drehbewegung der Kamera erfolgen muss. Das vorgeschlagene, aber bisher nicht realisierte System kommt ohne bewegliche Teile aus und reduziert den Erstellungsaufwand deutlich [5].

Für weitere Möglichkeiten Stereo-Panoramen zu erzeugen wird auf die Literatur verwiesen [3].

## **Stereoskopische Ausgaben**

Die Ausgabe stereoskopischer Bild-Daten kann auf verschiedene Arten erfolgen, abhängig von den Anforderungen der verwendeten Ausgabegeräte und vom Format, in dem die Stereodaten vorliegen. Zur Betrachtung mit Hilfsmitteln wie z.B. Shutter-Brillen oder Head-Mounted-Displays wird u.a. zwischen Page-Flipping, Anaglyphen, Interlaced- und Dual-Stream-Ausgabe unterschieden. Daneben existieren auch Verfahren zur Visualisierung von Stereo-Szenen ohne Hilfsmittel, z.B. Parallel- oder Kreuzblick.

Die Erzeugung und Ausgabe stereoskopischer OpenGL-Szenen stellt gewisse Mindestanforderungen an die verwendete Hardware, vor allem an die im System arbeitende Grafikkarte. Native Unterstützung von Stereo-Renderern gibt es bislang nur von der Firma NVIDIA, die eigene Stereo-Treiber für bestimmte Grafikkartenserien anbietet. Diese Treiber haben die Aufgabe eine bestehende OpenGL-Szene in Echtzeit in eine stereoskopische Szene umzurechnen, indem der eigentliche Blickpunkt des Szenenbetrachters einmal leicht nach links und einmal leicht nach rechts verschoben wird. Dadurch ergibt sich ein stereoskopischer Blick auf die gerenderte Szene. Die nötigen Berechnungen laufen nach Aktivierung des Stereomodus direkt auf dem Grafikchip ab. So ist es möglich, aus allen Arten von OpenGL-Szenen Stereoansichten zu erzeugen und auszugeben. Es bedarf dabei keiner zusätzlichen Programmierung der Rendering-Engine. Für die Ausgabe stehen verschiedene Modi zur Verfügung, je nach gewünschtem oder geräteabhängigem Ausgabeformat (z.B. Page Flipping, Interlaced, Anaglyph).

Leider eignet sich diese Form der Stereo-Ausgabe nicht für die vorliegende Arbeit, da die zu rendernde OpenGL-Szene lediglich aus einer um den Betrachtungspunkt herumliegenden Sphäre besteht, auf dessen Innenseite das Panoramabild projiziert wird. Die eigentliche Tiefeninformation befindet sich also nicht in der OpenGL-Szene, sondern in den Stereo-Panoramen, die auf die Sphäreninnenseite projiziert werden. Das je nach Ausgabegerät benötigte Ausgabeformat muss also unabhängig von der nativen NVIDIA-Lösung erzeugt werden. So lassen sich in Grafikkarten mit Vierfachbuffer, d.h. mit je einem Front- und Backbuffer für jedes Auge, Ausgaben im Page-Flipping-Format generieren. Die Interlaced-Ausgabe stellt in OpenGL kein Problem dar, da lediglich jede zweite Bildzeile pro Frame dargestellt werden muss. Die Ausgabe auf einem Gerät mit Dual-VGA-Eingang erfolgt über die Erzeugung zweier, direkt nebeneinander gerendeter Szenen. Für die Umsetzung stehen verschiedene Medien-Frameworks zur Verfügung, die entweder auf OpenGL aufbauen wie das OpenGL Utility Toolkit (GLUT, FreeGLUT) oder OpenGL verwenden können wie Simple Directmedia Layer (SDL). Die Wahl des Frameworks hängt vor Allem von den benötigten Funktionalitäten ab.

## **Applikationssteuerung**

Das nachrichten-basierten Kommunikationsprotokoll OSC ermöglicht die Kommunikation zwischen Computern und Multimedia-Anwendungen über ein Netzwerk [7]. Es wird vor allem in Bereichen verwendet in denen eine Echtzeit-Steuerung notwendig ist, z.B. für das Senden von Steuerungskommandos über ein Netzwerk. Die Kommunikation im bestehenden Versuchsframework zur dynamischen Binauralsynthese erfolgt bereits über OSC. Für die Programmiersprache Python existieren OSC-Erweiterungen, welche für die zu erstellende Software verwendet werden können.

## **Methode**

Für die Erzeugung der stereoskopischen Panorama-Bilder wird eine Aufnahmestrecke entworfen, die von der eigentlichen Aufnahme bis zur Nachbearbeitung am Computer reicht. Am Ende der Strecke stehen Stereo-Panoramen zur Verfügung, die zur Verwendung mit der zu entwerfenden Software geeignet sind. Die Software zur Visualisierung wird in der plattformunabhängigen Programmiersprache Python realisiert. Für die Zielpunkte und Anforderungen der Software wird auf die Einleitung verwiesen. Begonnen wird mit der Implementierung eines Anaglyphrenderers, um die stereoskopischen Projektionen zu testen und zu verfeinern. Im Laufe der Arbeit werden dann die Module für die Ausgabe mit anderen Medien, wie Head-Mounted-Displays oder Shutterbrillen hinzugefügt. Die Software selbst soll so modular wie möglich gestaltet werden, damit es später mit möglichst wenig Aufwand möglich ist, neue Ausgabegeräte hinzuzufügen.

Die Entwicklung beider Teile, der Aufnahmestrecke und der Visualisierungssoftware erfolgt parallel.

## **Eigene Vorarbeiten**

Im Vorfeld der Arbeit wurden ausreichende Kenntnisse im Umgang mit Python, OpenGL, Stereo-Visualisierung und Kommunikation über OSC erworben. Außerdem erfolgte eine Recherche über die Möglichkeiten der rechnergestützten Aufnahme von Panorama-Serienbildern.

## **Arbeits- und Zeitplan**

1. Monat: Erstellung einer ersten Software-Version mit Anaglyphrenderer und Konzeption der Aufnahmestrecke
2. Monat: Erzeugung der ersten Stereo-Panoramen, während gleichzeitiger Weiterentwicklung der Software; Hinzufügen neuer Ausgabemodule; Dokumentation
3. Monat: Notwendige Nachbesserungen an Software und Aufnahmestrecke; Erzeugung weiterer Stereo-Panoramen; Dokumentation
4. Monat: Ausgiebiges Testen, Durchführung von Benchmarks und Abschluß der Dokumentation

## Literatur

- [1] BOURKE, PAUL D.: *Synthetic Stereoscopic Panoramic Images*. In: ZHA, HONGBIN, ZHIGENG PAN, HAL THWAITES, ALONZO C. ADDISON und MAURIZIO FORTE (Herausgeber): *VSMM*, Band 4270 der Reihe *Lecture Notes in Computer Science*, Seiten 147–155. Springer, 2006.
- [2] CRUZ-NEIRA, CAROLINA: *Virtual reality based on multiple projection screens: the cave and its applications to computational science and engineering*. Doktorarbeit, Chicago, IL, USA, 1995.
- [3] DUKE GLEDHILLA, GUI YUN TIAN, DAVE TAYLOR und DAVID CLARKE: *Panoramic imaging—a review*. *Computers Graphics*, 27:435–445, 2003.
- [4] PELEG, S und M BEN-EZRA: *Stereo panorama with a single camera*. In: *Computer Vision and Pattern Recognition Conf.*, Seiten 395–401, 1999.
- [5] PELEG, SHMUEL, Yael PRITCH und MOSHE BEN-EZRA: *Cameras for Stereo Panoramic Imaging*. *Computer Vision and Pattern Recognition*, IEEE Computer Society Conference on, 1:1208, 2000.
- [6] WIMMER, P: *Aufnahme und Wiedergabe stereoskopischer Videos im Anwendungsbereich der Telekooperation (Diplomarbeit)*. 2004.
- [7] WRIGHT, M; FREED, A; MOMENI A: *Open Sound Control: State of the Art 2003*. In: *International Conference on New Interfaces for Musical Expression*, Seiten 153–159, Montreal, 2003. OpenSound Control.