



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelorarbeit

Carola Christoffel

**Modifikation der Software einer Wellenfeldsyntheseanlage zur
Wiedergabe fokussierter Quellen in Abhängigkeit der
Zuhörerposition**

*Fakultät Technik und Informatik
Studiendepartment Informatik*

*Faculty of Engineering and Computer Science
Department of Computer Science*

Carola Christoffel

**Modifikation der Software einer Wellenfeldsyntheseanlage zur
Wiedergabe fokussierter Quellen in Abhängigkeit der
Zuhörerposition**

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Angewandte Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Wolfgang Fohl Erstprüfer
Zweitgutachter: Prof. Dr. Philip Jenke Zweitprüfer

Eingereicht am: 13.05.2014

Carola Christoffel

Thema der Arbeit

Modifikation der Software einer Wellenfeldsyntheseanlage zur Wiedergabe fokussierter Quellen in Abhängigkeit der Zuhörerposition

Stichworte

Wellenfeldsynthese, Tonquellen, Positionsverfolgung, Softwaremodifikation, Echtzeitanwendung

Kurzzusammenfassung

In dieser Arbeit wird eine Modifikation an der Software sWONDER vorgenommen, um ein zuhörerbasiertes Rendering fokussierter Tonquellen bei der Wellenfeldsynthese zu erreichen. Die Position des Zuhörers wird dabei von einem Trackingsystem erfasst. Neben der Implementierung des Vorhabens wird die Funktionsweise einiger Komponenten von sWONDER im Detail beschrieben. Zur Auswertung des modifizierten Rendering-Verfahrens werden zwei Tests am laufenden System durchgeführt. Anschließend werden die Ergebnisse dieser Tests interpretiert.

Carola Christoffel

Title of the paper

Modification of the software of a wave field synthesis system to render focused sources depending on the listener's position

Keywords

Wave field synthesis, Sound Sources, Position Tracking, Software Modification, Real-Time Application

Abstract

This thesis describes a modification of the sWONDER software. The aim is to enable the wave field synthesis system to render focused sound sources based on the position of the listener. In order to achieve this, the listener's position is tracked by a visual tracking system. Aside from the implementation there will also be an in-depth description of several components of sWONDER. There will be two tests under live conditions to evaluate the modified rendering functionality. Finally, the results of those tests will be interpreted.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Problemstellung	1
1.2. Ziel der Arbeit	2
1.3. Ähnliche Lösungsansätze	2
1.4. Motivation	2
1.4.1. Erweiterung der akustischen Umgebung	3
1.4.2. Umsetzung von audiobasierten Spielen	3
1.4.3. Basis für andere Arbeiten und Projekte	3
1.5. Themenabgrenzung	3
1.6. Struktur der Arbeit	4
2. Wellenfeldsynthese	5
2.1. Die Idee	5
2.2. Physikalischer Hintergrund	5
2.3. Einschränkungen bei der Verwendung einer WFS-Anlage	8
2.4. Anwendungsgebiete	8
2.4.1. Räumlicher Klang zu Filmen	9
2.4.2. Nachbildung von Räumlichkeiten	9
2.4.3. Wellenfeldsynthese bei Videokonferenzen	9
2.5. Interaktionsmöglichkeiten	9
2.5.1. Steuerung durch Gesten	10
2.5.2. Visualisierung durch Augmented Reality	10
2.5.3. Erzeugung virtueller Instrumente	11
3. Die Software sWONDER	12
3.1. Komponenten der Software	12
3.2. Konfigurationsdateien	14
3.3. Kommunikation durch Open Sound Control	14
3.4. Klassifikation von Tonquellen	15
3.4.1. Punkt- und Linearquellen	15
3.4.2. Fokussierte Tonquellen	16
4. Das Labor der HAW Hamburg	17
4.1. Die Systemumgebung des Labors	17
4.2. Verwendung der Software sWONDER	18
4.2.1. Auswahl und Verteilung der Komponenten	18

4.2.2.	Rendering fokussierter Tonquellen	19
4.3.	Das ARTTRACK System	21
5.	Modifikation der Software sWONDER	23
5.1.	Idee und Anforderungen	23
5.2.	Vorgehensweise	23
5.2.1.	Überblick über die Software gewinnen	23
5.2.2.	Modifikation des Rendering-Verfahrens	24
5.2.3.	Verwendung des Open Sound Control Protokolls	25
5.2.4.	Anbindung des ARTTRACK Systems	25
5.3.	Implementation	25
5.3.1.	Beschreibung des Aufbaus von sWONDER-Komponenten	25
5.3.1.1.	Aufbau der Komponente cwonder	26
5.3.1.2.	Aufbau der Komponente twonder	26
5.3.2.	Neues Rendering-Verfahren für eine festen Zuhörerposition	27
5.3.3.	Einführung einer neuen OSC-Nachricht	29
5.3.3.1.	Definition der OSC-Nachricht	29
5.3.3.2.	Modifikation der OSC-Server	30
5.3.4.	Einbinden des Zuhörers und seiner Position	31
5.3.4.1.	Die Klasse Listener und ListenerArray	31
5.3.4.2.	Aktualisieren der Zuhörerposition (ListenerMoveCommand)	31
5.3.4.3.	Übergabe an das Rendering	32
5.3.5.	Modifikation der Schnittstelle zum ARTTRACK System	32
5.3.5.1.	Verwendung eines Targets als Position des Zuhörers	32
5.3.5.2.	Darstellung der Zuhörerposition als virtuelle Tonquelle	33
5.4.	Ergebnisdarstellung	34
6.	Testen des Zuhörerbasierten Renderings	36
6.1.	Messung der Lautstärke um fokussierte Tonquellen	36
6.2.	Hörtest zur Lokalisierbarkeit fokussierter Tonquellen	39
6.2.1.	Versuchsaufbau	39
6.2.2.	Konvertieren der Daten in das Koordinatensystem der WFS-Anlage	41
6.2.3.	Auswertung der Ergebnisse	42
7.	Schluss	46
7.1.	Zusammenfassung und Fazit	46
7.2.	Ausblick	46
	Glossar	48
	Abbildungsverzeichnis	50
	Literaturverzeichnis	51

Anhang A. Informationen zu den Tests des zuhörerbasierten Renderings	54
A.1. Einzeldaten der Lautstärkemessung	54
A.2. SuperCollider Skript des Hörtests	55
A.3. Einzeldaten des Hörtests	56

1. Einleitung

Auditive Erfahrungen sind stets mit einer räumlichen Erfahrung verbunden [WT06]. In Landschaften, Gebäuden oder urbanen Räumen werden Geräusche und Klänge aus unterschiedlichen Richtungen und Entfernungen wahrgenommen. Diese Hörerlebnisse sind durch das Prinzip der Wellenfeldsynthese reproduzierbar. Die Wellenfeldsynthese ermöglicht eine räumliche Wiedergabe beliebigen Audiomaterials. Diese Arbeit beschäftigt sich mit einem Teilaspekt, welcher sich bei der Verwendung der Wellenfeldsynthese ergibt.

1.1. Problemstellung

Damit das Prinzip der Wellenfeldsynthese angewendet und eine räumliche Wiedergabe realisiert werden kann, ist eine Wellenfeldsyntheseanlage (WFS-Anlage) notwendig. Mit dieser ist es möglich, virtuelle Tonquellen zu erzeugen und frei im Raum zu platzieren. Die Anlage besteht aus mehreren Lautsprechermodulen, welche in einer Reihe oder in einer geschlossenen Form angeordnet sind. Im Labor der HAW Hamburg bilden die Module ein Rechteck. Der Zuhörer steht und bewegt sich innerhalb der von den Lautsprechermodulen eingeschlossenen Fläche. Die Position einer virtuellen Tonquelle ist dabei vollkommen unabhängig von den Positionen der Lautsprecher. Sie kann außerhalb oder innerhalb der von den Lautsprechern eingeschlossenen Fläche liegen. Liegt die Position einer virtuellen Tonquelle außerhalb dieser Fläche, wird sie stets korrekt wahrgenommen. Für den Zuhörer klingt es so, als würde sich eine reale Schallquelle hinter den Lautsprechern befinden. Befindet sich eine virtuelle Tonquelle innerhalb der von den Lautsprechern eingeschlossenen Fläche (auch fokussierte Tonquelle genannt), kann diese vom Zuhörer nicht aus allen Richtungen wie eine reale Schallquelle wahrgenommen werden. Bewegt sich der Zuhörer in einen bestimmten Bereich, kann er die virtuelle Tonquelle nicht mehr lokalisieren und nimmt den Schall aus einer falschen Richtung wahr. Die Gründe dafür werden im Kapitel 2.2 erläutert.

1.2. Ziel der Arbeit

Das Ziel dieser Arbeit ist es, virtuelle Tonquellen, welche sich innerhalb der von den Lautsprechern eingeschlossenen Fläche befinden, für einen Zuhörer aus allen Richtungen korrekt wahrnehmbar und somit lokalisierbar zu machen. Dazu wird die Software sWONDER¹, welche für die Wellenfeldsynthese im Labor der HAW Hamburg verwendet wird, modifiziert. Ein im Labor aufgebautes Trackingsystem wird verwendet, um die Position des Zuhörers zu erfassen. Bewegt sich der Zuhörer im Raum, wird anhand seiner aktuellen Position das Rendering fokussierter Tonquellen angepasst.

1.3. Ähnliche Lösungsansätze

Eine mögliche Lösung des Problems, welches in dieser Arbeit behandelt wird, wird in einem Artikel von De Vries et al. vorgestellt [DV⁺08]. Sein Ansatz beinhaltet ebenfalls das Erfassen der Zuhörerposition durch ein Trackingsystem. Mehrere Themen werden in dem Artikel behandelt, darunter die Auswahl der aktiven Lautsprecher und der Ankunftszeitpunkt der Schallwelle bei dem Zuhörer. Auch eine für mehrere Zuhörer optimierte Lösung wird vorgestellt.

Die Software namens Sound Scape Renderer² bietet ebenfalls die Möglichkeit, Wellenfeldsynthese zu betreiben. Zusätzlich kann sie für zwei weitere Hörverfahren, binaurale Synthese und Ambisonics, verwendet werden [GS12]. Diese Software wurde bereits um ein Zuhörerbasiertes Rendering, welches sich an dem Artikel von De Vries [DV⁺08] orientiert, erweitert. Die Erweiterung ist in der aktuellen Version der Sound Scape Renderer Software verfügbar und trägt den Namen „reference offset“, da die Position des Zuhörers beim Rendering auch als Referenzpunkt bezeichnet wird [M⁺14].

1.4. Motivation

Die Motivation dieser Arbeit besteht darin, eine begehbare akustische Umgebung für einen Zuhörer zu schaffen, bei der es keine Einschränkungen bei der Platzierung von virtuellen Tonquellen gibt. In dieser können virtuelle Tonquellen unmittelbar in die Nähe des Zuhörers bewegt werden, um den Zuhörer kreisen oder genau auf die Position des Zuhörers gesetzt werden. Dabei werden alle virtuellen Tonquellen, auch fokussierte Tonquellen, von dem Zuhörer stets wie reale Schallquellen wahrgenommen. Im Folgenden werden mögliche Anwendungen der Modifikation, wie sie in dieser Arbeit durchgeführt wird, vorgestellt.

¹<http://sourceforge.net/projects/swonder/>

²<http://spatialaudio.net/ssr/>

1.4.1. Erweiterung der akustischen Umgebung

Die akustische Umgebung, welche überwiegend durch virtuelle Tonquellen außerhalb der Lautsprecher realisiert wird, kann nun uneingeschränkt auf die Fläche innerhalb der Lautsprecher erweitert werden. Es ist möglich, räumliche Komposition für die WFS-Anlage zu entwerfen, welche die innere Fläche mit einbezieht. Während die virtuellen Tonquellen durch den Raum schweben, kann sich der Zuhörer durch die Geräuschkulisse bewegen. Für eine Gruppe von Zuhörern ist die korrekte Lokalisation jedoch nur eingeschränkt möglich.

1.4.2. Umsetzung von audiobasierten Spielen

Der Zuhörer kann sich frei innerhalb der Anlage bewegen und die Positionen der virtuellen Tonquellen werden stets korrekt wahrgenommen. Durch diese Tatsache ist es für den Zuhörer möglich, fokussierte Tonquellen auch ohne visuelle Unterstützung zu lokalisieren, was für audiobasierte Spiele genutzt werden kann. Dabei können verschiedene Spielprinzipien umgesetzt werden. Der Zuhörer muss beispielsweise die virtuellen Tonquellen suchen und sich auf deren Positionen stellen. Findet er die exakten Positionen, wird das als Erfolg interpretiert. Oder er muss virtuellen Tonquellen, welche sich auf ihn zu bewegen, ausweichen.

1.4.3. Basis für andere Arbeiten und Projekte

Das Zuhörerbasierte Rendering kann in bereits bestehende Arbeiten integriert werden und auch eine Basis für zukünftige Arbeiten und Projekte sein. Im Labor wird beispielsweise eine Gestensteuerung zur Positionierung von virtuellen Tonquellen verwendet [NOG12], welche von dem neuen Rendering profitieren kann. Die Person, die die Gestensteuerung anwendet, kann die Positionen der virtuellen Tonquellen nicht mehr verlieren, wenn sie innerhalb der Lautsprecher platziert werden. Durch das Zuhörerbasierte Rendering sind sie stets lokalisierbar. Die durch das neue Rendering-Verfahren entstehenden Möglichkeiten können in zukünftigen Arbeiten genutzt oder weiterentwickelt werden.

1.5. Themenabgrenzung

Diese Arbeit greift nicht in das allgemeine Prinzip der Wellenfeldsynthese ein. Lediglich die Software sWONDER, welche die Wellenfeldsynthese umsetzt, wird an einigen Stellen modifiziert. Dies betrifft nicht die Berechnungen zur Erzeugung korrekt synthetisierter Wellenfronten, welche in der Software durchgeführt werden. Auch eine Berücksichtigung mehrerer Zuhörer wird nicht implementiert.

1.6. Struktur der Arbeit

In Kapitel 2, 3 und 4 werden die Grundlagen erläutert, welche für das Verständnis dieser Arbeit notwendig sind. Das zweite Kapitel gibt einen Überblick über das Thema Wellenfeldsynthese. Dabei wird das Prinzip erläutert und einige Beispiele, welche die Anwendung der Wellenfeldsynthese zeigen, werden aufgezählt. Im dritten Kapitel wird das Labor der HAW Hamburg vorgestellt. In diesem Raum befindet sich die WFS-Anlage, an der die Modifikation der Software getestet wurde. Das vierte Kapitel zeigt den Aufbau und die Funktionsweise der Software sWONDER, an der die Modifikation vorgenommen wird. Das nachfolgende fünfte Kapitel dient einer detaillierten Beschreibung der Vorgehensweise, durch die das Zuhörerbasierte Rendering erreicht wurde. Neben der Modifikation von sWONDER ist eine Anpassung der Schnittstelle zwischen Trackingsystem und WFS-Anlage notwendig. Des Weiteren werden in diesem Kapitel Teile des Programmcodes der sWONDER Software im Detail beschrieben. In Kapitel 6 wird die Durchführung zweier Tests, welche die Qualität des neuen Rendering-Verfahrens prüfen sollen, beschrieben und die Ergebnisse werden ausgewertet. Der erste Test ist eine Lautstärkemessung, welche um fokussierte Tonquellen herum durchgeführt wurde. Darauf folgt ein Hörtest, der mit mehreren Probanden durchgeführt wurde, um die Lokalisationsfähigkeit fokussierter Tonquellen zu ermitteln. Das siebte und letzte Kapitel fasst diese Arbeit zusammen und zeigt einen Ausblick auf weitere mögliche Arbeiten.

2. Wellenfeldsynthese

Das Prinzip der Wellenfeldsynthese zählt zu den Grundlagen dieser Arbeit. Dieses Kapitel bietet eine Einführung in das Themengebiet und den physikalischen Hintergrund. Dabei werden die wichtigsten Begriffe im Zusammenhang mit der Wellenfeldsynthese erläutert. Auch die Einschränkungen einer WFS-Anlage werden aufgezeigt. Im zweiten Abschnitt werden mögliche Anwendungsgebiete sowie Interaktionsmöglichkeiten mit der Wellenfeldsynthese vorgestellt.

2.1. Die Idee

Die Wellenfeldsynthese (kurz WFS) ist ein Verfahren zur Erzeugung virtueller Tonquellen. Diese besitzen vergleichbare Eigenschaften zu realen Schallquellen, obwohl sich an der Position einer virtuellen Tonquelle kein schallemittierendes Objekt befindet. Durch diese Technik können virtuelle akustische Umgebungen geschaffen werden. Bei mehrkanaligen Wiedergabesystemen gibt es die Möglichkeit, durch das Verteilen eines Signals auf zwei Kanäle eine Phantomschallquelle zu bilden. Diese kann eine beliebige Position zwischen den beiden Lautsprechern einnehmen. Der Nachteil ist, dass die räumliche Wirkung nur an einem bestimmten Punkt, dem „sweet spot“, erzielt werden kann. Befindet sich der Zuhörer nicht an diesem Punkt, geht die räumliche Wirkung verloren. Bei der Wellenfeldsynthese verändert sich die Position der virtuellen Tonquellen nicht, wenn sich der Zuhörer durch den beschallten Raum bewegt. Dies wird erzielt, indem korrekt synthetisierte Schallfelder gebildet werden, während die Stereophonie auf einer akustischen Illusion beruht [WT06]. Zur Umsetzung einer Wellenfeldsynthese werden Lautsprecherarrays, welche aus vielen aneinander grenzenden Lautsprechern bestehen, benötigt. Die Lautsprecher müssen einen möglichst geringen Abstand zueinander haben. Das Verfahren der Wellenfeldsynthese wurde im Jahr 1988 von Berkhout an der TU Delft erfunden und wird seitdem weiterentwickelt.

2.2. Physikalischer Hintergrund

Die Wellenfeldsynthese basiert auf dem Huygensschen Prinzip, welches besagt, dass jeder Punkt einer Wellenfront der Ausgangspunkt einer neuen Elementarwelle ist. Diese Elementar-

wellen überlagern sich wiederum zu einer neuen Wellenfront. In Abbildung 2.1 ist dargestellt, wie sich dieses Prinzip auf die Wellenfeldsynthese übertragen lässt. Eine reale Schallquelle, welche sich hinter den Lautsprechern befindet, bildet ringförmige Wellenfronten, die sich gleichmäßig in alle Richtungen um die Schallquelle ausbreiten. Für eine virtuellen Tonquelle an derselben Position übernimmt das System die Berechnung der Wellenfronten. Dabei können die Lautsprecher als Punkte auf einer Wellenfront angesehen werden, wie sie von Huygens beschrieben werden. Sie müssen die Elementarwellen bilden, welche sich überlagern und die gewünschte Wellenfront zu der virtuellen Tonquelle synthetisieren. Dazu berechnet das System die Amplituden der Elementarwellen und mit welcher Verzögerung die einzelnen Lautsprecher aktiv werden müssen. Bei diesem Verfahren kann nur der Teil der ringförmigen Wellenfront synthetisiert werden, der sich durch den Zuhörerbereich bewegt. Die Auswahl und Anzahl der aktiven Lautsprecher hängt von der Position der virtuellen Tonquelle ab. In der Abbildung sind die Lautsprecher auf der linken Seite aktiv, da sie die Elementarwellen erzeugen.

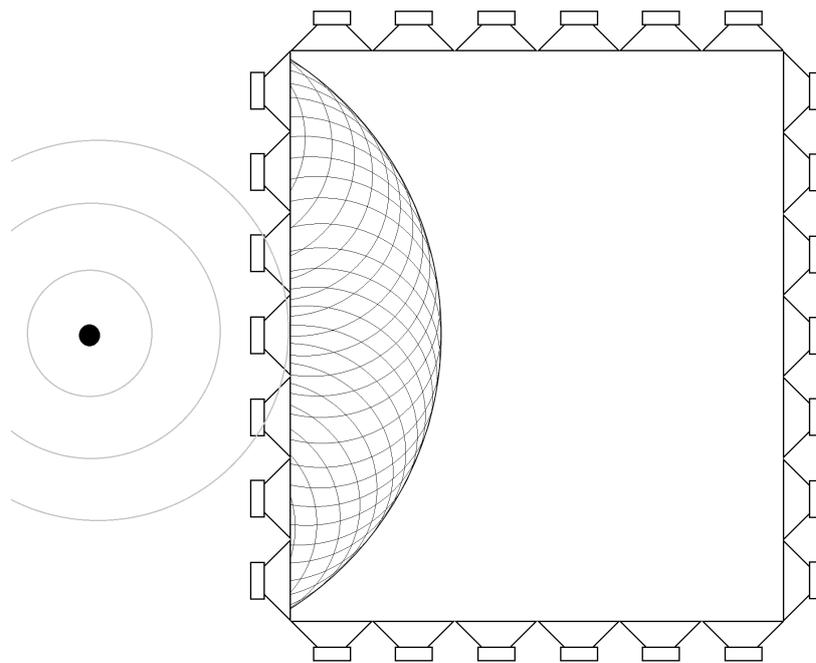


Abbildung 2.1.: Prinzip der Wellenfeldsynthese - Synthese einer Wellenfront

Virtuelle Tonquellen innerhalb der Lautsprecher können ebenfalls nachgebildet werden. Dazu erzeugt ein Teil der Lautsprecher eine Wellenfront mit entgegengesetzter Krümmung. Diese wird im Folgenden als konkav bezeichnet, da sie von den aktiven Lautsprechern aus gesehen konkav ist. Diese konkave Wellenfront läuft auf einen virtuellen Punkt zu, dem Punkt

der virtuellen Tonquelle, und breitet sich von dort aus wieder als konvexe Wellenfront aus. Da sich die Wellenfront in diesem Fall auf einen Punkt fokussiert, werden virtuelle Tonquellen innerhalb der Lautsprecher auch als fokussierte Tonquellen bezeichnet. Die erzeugten Wellenfronten bei einer fokussierten Tonquelle sind in Abbildung 2.2 dargestellt. In diesem Bild ist ein Teil der Lautsprecher auf der linken Seite aktiv. Die mathematischen Regeln, die bei dieser Technik angewendet werden, wurden erstmals in der Masterthesis von Jansen vorgestellt [Jan97].

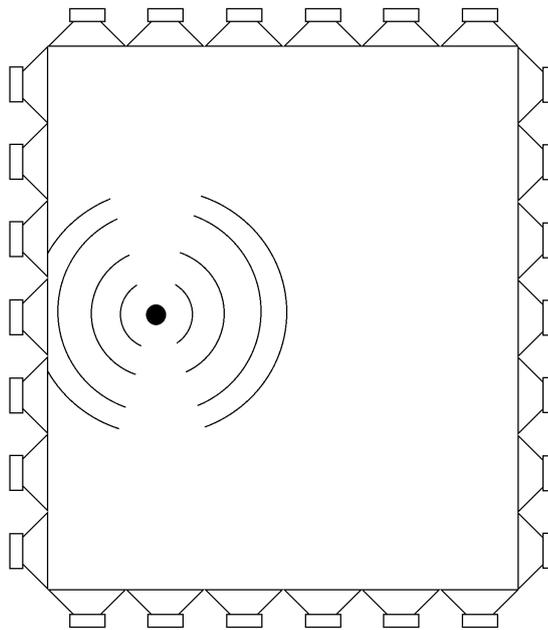


Abbildung 2.2.: Wellenfronten einer fokussierten Tonquelle

Fokussierte Tonquellen sind nicht von jeder beliebigen Positionen aus korrekt lokalisierbar. Bei einer realen Schallquelle breitet sich eine Wellenfront aus, bewegt sich also von dem Punkt der Schallquelle weg. In diesem Fall müsste sich die Wellenfront von dem fokussierten Punkt aus in Richtung der Lautsprecher bewegen. Bei der Wellenfeldsynthese gibt es jedoch immer einen Bereich, in dem sich die Wellenfront in die entgegengesetzte Richtung bewegt, das heißt von den Lautsprechern aus in Richtung des fokussierten Punktes. Dies ist der Bereich der konkaven Wellenfront, welcher stets zwischen den aktiven Lautsprechern und der fokussierten Tonquelle liegt. Befindet sich der Zuhörer in diesem Bereich, erhält er widersprüchliche Informationen [AS08]. An dieser Position kann die virtuelle Tonquelle nicht mehr wahrgenommen werden. Befindet sich der Zuhörer zwischen den aktiven Lautsprechern und der fokussierten Tonquelle, nimmt er den Schall aus der Richtung der aktiven Lautsprecher wahr und nicht

mehr aus der Richtung der fokussierten Tonquelle.

Bei einer geschlossenen Anordnung von Lautsprechern ist dies lediglich ein Standortproblem, da die aktiven Lautsprecher bei einer fokussierten Tonquelle frei wählbar sind. In dem Szenario in Abbildung 2.2 ist es ebenso möglich, dass die Lautsprecher auf der rechten Seite die Wellenfronten bilden, auch wenn die Position der fokussierten Tonquelle dieselbe ist. In diesem Fall verschiebt sich der Bereich, in dem der Zuhörer die fokussierte Tonquelle nicht korrekt hört, in die rechte Hälfte des Raumes. Das beschriebene Problem entsteht dadurch, dass nicht bekannt ist, wo sich der Zuhörer befindet. Außerdem kann sich die Position des Zuhörers ständig ändern, da er sich innerhalb der Lautsprecher frei bewegen kann.

2.3. Einschränkungen bei der Verwendung einer WFS-Anlage

Da der Abstand zwischen den Lautsprechern möglichst gering sein muss, sind selbst bei der Beschallung einer kleinen Fläche viele Lautsprecher notwendig. Hieraus resultieren die in der Regel hohen Herstellungs- bzw. Anschaffungskosten einer WFS-Anlage. Es existieren bereits Ansätze, die Anzahl der Lautsprecher bei Anwendung der Wellenfeldsynthese zu reduzieren [N⁺08]. Dies ist jedoch nur möglich, wenn die Lautsprecher direkt um den Zuhörer platziert werden, wodurch keine Bewegung des Zuhörers innerhalb der beschallten Fläche mehr möglich ist. Des Weiteren wird viel Rechenleistung benötigt, da für jeden einzelnen Lautsprecher ein eigenes Signal berechnet wird [Baa04].

Eine konventionelle WFS-Anlage ist durch die Anordnung der Lautsprecher auf die horizontale Ebene begrenzt. Virtuelle Tonquellen können dabei nicht in der Höhe verändert werden. In der Theorie ist es jedoch möglich, eine dreidimensionale Wellenfeldsynthese zu betreiben. Die Einschränkung auf die horizontale Ebene ist eher ein praktischer Ansatz [TW04]. Aufgrund der Anordnung der Lautsprecher gibt es auch einige systembedingte Grenzen, beispielsweise gelingt die Wellenfeldsynthese in den Ecken und bei virtuellen Tonquellen unmittelbar vor den Lautsprechern nur unvollkommen [WT06].

2.4. Anwendungsgebiete

Es gibt mehrere mögliche Anwendungsgebiete, in der die Wellenfeldsynthese genutzt werden kann. In diesen Gebieten wird weiterhin geforscht, eine praktische Anwendung ist aufgrund der im vorigen Kapitel genannten Einschränkungen nicht ohne Weiteres möglich.

2.4.1. Räumlicher Klang zu Filmen

Die räumliche Akustik der Wellenfeldsynthese kann mit einem Video oder Film verbunden werden, beispielsweise in einem Kino. Objekte aus dem Film erhalten virtuelle Tonquellen, welche sich synchron zu den Objekten bewegen. Eine Schwierigkeit besteht darin, Bewegungen oder Positionen von Objekten auf der zweidimensionalen Leinwand in den dreidimensionalen Raum zu übertragen. Ein 3D-Film kann die räumliche Wirkung unterstützen, da bei diesem auch ein Tiefeneindruck entsteht. In einem Artikel über die Verbindung von 3D-Film und Wellenfeldsynthese beschäftigen sich André et al. mit der bestmöglichen Platzierung der virtuellen Tonquellen zu einem 3D-Film [A⁺13].

2.4.2. Nachbildung von Räumlichkeiten

Mit der Wellenfeldsynthese ist es möglich, dass jeder Raum klingen kann wie ein Konzertsaal [Spo04]. Dies wird erreicht, indem die akustischen Eigenschaften von Räumen durch die Wellenfeldsynthese nachgebildet werden. Für Open-Air Konzerte kann beispielsweise die Akustik so verändert werden, dass es für die Zuhörer so klingt, als ob sie sich in einem virtuellen Raum und nicht unter freiem Himmel befinden würden. Auch die Atmosphäre eines Fußballstadions kann direkt in einen Zuhörerraum übertragen werden.

2.4.3. Wellenfeldsynthese bei Videokonferenzen

Bei einer Videokonferenz findet eine Übertragung von Bild und Ton statt, um mit anderen Personen zu kommunizieren, welche sich nicht im selben Raum befinden. Verschiedene Systeme zur Umsetzung von Videokonferenzen werden auf dem Markt angeboten. Eine räumliche Soundwiedergabe zu einer Videokonferenz ist mit einer WFS-Anlage möglich. Veröffentlichungen von De Bruijn et al. und Lopez et al. beschäftigen sich mit der Verwendung der Wellenfeldsynthese im Bereich der Videokonferenz [DBB03, L⁺10b]. In dem Artikel von Lopez et al. werden verschiedene Prototypen vorgestellt. Vorteile der Wellenfeldsynthese in diesem Bereich sind eine leichtere Identifikation des Sprechenden und eine realistischere Umsetzung der Konferenz [DBB03].

2.5. Interaktionsmöglichkeiten

Im Folgenden werden einige Interaktionsmöglichkeiten mit den virtuellen Tonquellen der Wellenfeldsynthese vorgestellt. Der Fokus liegt dabei auf einer direkten, „natürlichen“ Interaktion

mit den virtuellen Tonquellen. Andere Formen der Interaktion, beispielsweise das Versenden von Nachrichten an die Software, werden in diesem Kapitel nicht berücksichtigt.

2.5.1. Steuerung durch Gesten

Eine Form der Interaktion mit der Wellenfeldsynthese ist die Manipulation der virtuellen Tonquellen per Gestensteuerung. Dazu ist ein Trackingsystem notwendig, welches die unterschiedlichen Gesten und Handbewegungen erkennen kann. Fohl und Nogalski beschreiben eine Gestensteuerung, mit der es möglich ist, die Positionen virtueller Tonquellen zu verändern [FN13]. Dabei wurden zwei Arten von Gesten zur Positionierung entwickelt. Mit einer der beiden Gesten kann eine Tonquelle auf horizontaler Ebene in einer Kreisbahn um die Person verschoben werden, welche die Geste ausführt. Mit einer weiteren Geste kann die Entfernung der virtuellen Tonquelle zur Person vergrößert oder verkleinert werden. Dabei ist es auch möglich, die virtuelle Tonquelle in den Raum hinein zu ziehen, wodurch sie zur fokussierten Tonquelle wird. Ein Zuhörerbasiertes Rendering ist jedoch nicht integriert. Vor der Ausführung einer Geste muss die Handfläche in Richtung der virtuellen Tonquelle zeigen. Eine visuelle Darstellung ist dabei nicht notwendig, da die virtuellen Tonquellen in den meisten Fällen auch nach Gehör lokalisiert werden können.

Müller et al. haben ebenfalls eine Form der Gestensteuerung entwickelt, welche als direkt beschrieben wird [M⁺14]. Diese ist so umgesetzt, dass eine Person den Eindruck hat, virtuelle Tonquellen „anfassen“ zu können. Die virtuellen Tonquellen können in der Luft gegriffen, bewegt und in der Luft platziert oder in verschiedene Behälter fallen gelassen werden. Wird einer dieser Behälter anschließend bewegt, bewegt sich die virtuelle Tonquelle mit dem Behälter mit, sodass der Eindruck entsteht, sie befände sich „in“ dem Gefäß. Bei dieser Form der Interaktion müssen sich die virtuellen Tonquellen in unmittelbarer Nähe zur Person befinden, weshalb ausschließlich fokussierte Tonquellen verwendet werden. Die Basis des entwickelten Systems bildet die Software Sound Scape Renderer, in der ein Zuhörerbasiertes Rendering fokussierter Tonquellen bereits implementiert ist.

2.5.2. Visualisierung durch Augmented Reality

Die durch die Wellenfeldsynthese generierten virtuellen Tonquellen sind unsichtbar, da sich an deren Positionen keine realen Objekte befinden. Mit Hilfe von Augmented Reality können die virtuellen Tonquellen in Form von grafischen Darstellungen in die reale Umgebung eingefügt werden. Dieser Ansatz wird in einem Artikel von Melchior et al. erläutert und ist in Form eines Augmented Reality Systems umgesetzt [MLDV05]. In diesem System sind mehrere rote Kreise

die Darstellung einer virtuellen Tonquelle. Die Augmented Reality Darstellung wird darüber hinaus mit einer Manipulation von virtuellen Tonquellen kombiniert. In einem Prototypen des angestrebten Systems kann der Träger eines Head-Mounted Displays virtuelle Tonquellen sehen und mit Hilfe eines virtuellen „Spatens“, den der Träger in der Hand hält, in ihrer Position verschieben. Das fertige System soll einer Person alle Steuerungsmöglichkeiten bieten, die für eine Audioproduktion benötigt werden. Dazu gehören beispielsweise das Verändern der Eigenschaften virtueller Tonquellen oder auch die Manipulation der Audiospuren, welche den virtuellen Tonquellen zugeordnet sind.

2.5.3. Erzeugung virtueller Instrumente

Das Spielen virtueller Instrumente kann beispielsweise mit Hilfe eines Trackingsystems realisiert werden. Um die Klänge zusätzlich genau an den Positionen der virtuellen Instrumente hörbar zu machen, sodass das Spielen des Instruments realistischer erscheint, kann die Wellenfeldsynthese genutzt werden. Eine Umsetzung dieser Idee ist das Projekt „Grainstick“, eine interaktive Installation, welche von Leslie et al. entwickelt wurde [L⁺10a]. Diese Installation ist an die Funktionsweise eines Regenmachers (engl. rainstick) angelehnt. Zwei sich gegenüberstehende Personen halten jeweils einen Controller in der Hand. Zwischen den beiden Controllern befindet sich der virtuelle Regenmacher. Bewegt eine Person den Controller nach oben und die andere Person gleichzeitig nach unten, bewegen sich die „Körner“ von einem Ende zum anderen Ende des Stabs, also von einem Controller zum anderen. Die Geräusche der Körner werden in Form von fokussierten Tonquellen, welche sich entlang des Stabs bewegen, hörbar gemacht.

3. Die Software sWONDER

Zur Realisierung der Wellenfeldsynthese im Labor wird die Software sWONDER verwendet. Diese wurde im Fachgebiet Audiokommunikation an der TU Berlin unter der Leitung von Prof. Dr. Weinzierl entwickelt. Die Software ermöglicht es, Audiospuren auf virtuelle Tonquellen zu legen und diese an beliebige Positionen im Raum zu verschieben. Des Weiteren richtet sich die Software an Komponisten und Sounddesigner, da sie Werkzeuge anbietet, um räumliche Kompositionen zu entwerfen [BP04]. Im folgenden Kapitel wird der Aufbau und die Funktionsweise der Software beschrieben. Der Schwerpunkt liegt dabei auf den notwendigen Komponenten für die Wellenfeldsynthese, einen vollständigen Überblick über die Software bietet die Dissertation von Marije Baalman [Baa08].

3.1. Komponenten der Software

sWONDER ist eine verteilte Anwendung. Sie besteht aus mehreren Komponenten, die auf verschiedenen Rechnern gestartet werden können, da sie über das Netzwerk miteinander kommunizieren. Jede Komponente besitzt eine fest definierte Aufgabe innerhalb der Gesamtanwendung. Im folgenden werden die Komponenten nacheinander vorgestellt.

cwonder ist die zentrale Kontrolleinheit. Sie ist für die Informationsweitergabe innerhalb der verteilten Anwendung zuständig. Deshalb muss jede Komponente beim Start der Anwendung eine Verbindung zu cwonder aufbauen. Sobald eine Komponente eine Nachricht an cwonder sendet, werden die ausgelesenen Daten an bis zu vier verschiedene Streams weitergeleitet. Die Art der Information bestimmt, welche Komponente diese benötigt und somit auch, an welchen Stream diese gesendet werden muss.

Die Rendering Einheit heißt twonder. Diese Komponente ist notwendig, um die virtuellen Tonquellen hörbar zu machen. Die Spuren eines Audioplayer sind der Input und werden als Output auf virtuelle Tonquellen abgebildet. Jede virtuelle Tonquelle benötigt eine korrekt synthetisierte Wellenfront, welche aus den Elementarwellen der einzelnen Lautsprecher entsteht. Dazu berechnet die Komponente die Verzögerung und Amplitude jeder Elementarwelle. Für die Berechnungen benötigt die Komponente Informationen zu den Positionen der Lautsprecher und der virtuellen Tonquellen. Es ist möglich, mehrere Instanzen von twonder zu starten und

3. Die Software sWONDER

diesen unterschiedliche Lautsprechersegmente, welche in der Konfiguration definiert sind, zuzuordnen. Werden die Instanzen anschließend auf mehreren Rechnern gestartet, kann so die Rechenlast verteilt werden.

Die Komponente xwonder bietet eine grafische Darstellung der Audioumgebung. Lautsprecher und virtuelle Tonquellen werden hier als zweidimensionale Grafik dargestellt und somit sichtbar gemacht. Da es nicht möglich ist, virtuelle Tonquellen in der Höhe bzw. entlang der z-Achse zu verschieben, ist diese Darstellung ausreichend. Die Oberfläche ist zusätzlich eine Schnittstelle, die es dem Benutzer ermöglicht, mit den virtuellen Tonquellen zu interagieren. Dazu gehört beispielsweise das Verschieben virtueller Tonquellen mit dem Mauszeiger. Auch die Eigenschaften einer virtuellen Tonquelle können verändert werden. In der Abbildung 3.1 ist die Oberfläche von xwonder zu sehen.

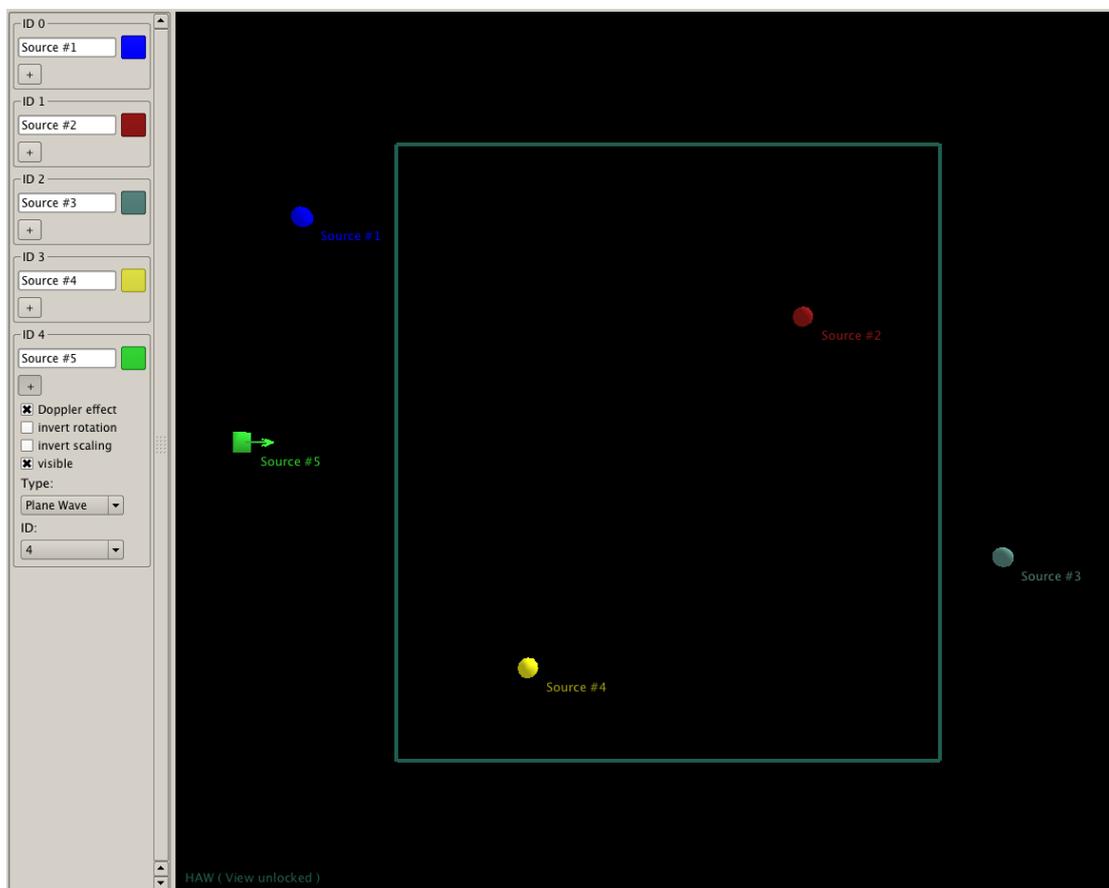


Abbildung 3.1.: Die grafische Oberfläche der Komponente xwonder

3.2. Konfigurationsdateien

Eine Konfiguration der verteilten Anwendung kann über die dafür vorgesehenen Konfigurationsdateien vorgenommen werden, welche im XML-Format geschrieben sind. Die Anwendung bietet eine Konfigurationsdatei pro Komponente, um Werte festzulegen, die sich während der Ausführung des Programms nicht ändern. Beispiele hierfür sind der Name des Raums, der Projektpfad, die maximale Anzahl an Tonquellen oder der Abstand zwischen zwei Lautsprechern. Beim Start einer Komponente wird deren Konfigurationsdatei einmalig ausgelesen. Zusätzlich gibt es die sogenannte Array Konfiguration für twonder. In diesen Dateien werden die Position, Ausrichtung und Länge der einzelnen Lautsprechersegmente definiert. Werden mehrere Instanzen von twonder verwendet, erhält jede Instanz ihre eigene Array Konfiguration. In dieser Konfiguration ist festgelegt, welche Instanz von twonder welches Lautsprechersegment verwaltet.

3.3. Kommunikation durch Open Sound Control

Open Sound Control (OSC) ist ein Protokoll, welches in Sound- und Multimedia-Installationen zur Kommunikation über das Netzwerk verwendet wird. OSC bietet einige Verbesserungen gegenüber dem MIDI-Protokoll [Fra08]. Dazu zählen beispielsweise das dynamische Adressierungsschema und die Möglichkeit, zukünftige Ereignisse im Voraus zu senden. Dies wird durch die Übertragung eines Zeitstempels realisiert.

OSC-Nachrichten haben folgenden Aufbau: Erst erfolgt die Angabe der Adresse, dann eine Definition der Parameter (Anzahl und Datentypen) und anschließend die Angabe der Werte der Parameter [WFM03]. OSC-Nachrichten werden auch in der Software sWONDER zur Kommunikation verwendet. Dazu wird die OSC-Bibliothek liblo¹ verwendet. Die Komponenten müssen untereinander über OSC kommunizieren, um jederzeit einen konsistenten Informationsstand zu erreichen. Das Versenden von OSC-Nachrichten ist auch eine Möglichkeit, das Programm von außen zu steuern, ohne die Oberfläche von xwonder zu verwenden. Dazu ist ein Programm notwendig, welches OSC-Nachrichten versenden kann. Dazu kann beispielsweise das Programm SuperCollider verwendet werden. Eine OSC-Nachricht zur Neupositionierung einer Tonquelle lautet in SuperCollider:

```
NetAddr("192.168.3.1", 58100).sendMsg("/WONDER/source/position", 0, -0.69, 1.92);
```

Diese Nachricht sagt aus, dass die virtuelle Tonquelle mit der ID 0 an die Position mit der x-Koordinate -0.69 und der y-Koordinate 1.92 gesetzt werden soll. Die Definition der Datentypen

¹<http://liblo.sourceforge.net/>

wird in sWONDER nicht mit versendet. Eine Übersicht aller in sWONDER implementierten OSC-Nachrichten befindet sich in der Dissertation von Marije Baalman [Baa08].

3.4. Klassifikation von Tonquellen

In sWONDER können Tonquellen auf verschiedene Arten klassifiziert werden. Diese Klassifizierung kann an der Form der Wellenfront einer virtuellen Tonquelle vorgenommen werden, sowie an der Position der virtuellen Tonquelle im Raum.

3.4.1. Punkt- und Linearquellen

In sWONDER sind zwei Arten von Tonquellen implementiert: Punktquellen und Linearquellen. Punktquellen besitzen eine feste Position im Raum und werden in xwonder als Punkt dargestellt. Eine Punktquelle ist die Nachbildung einer realen Schallquelle, bei der sich die Schallwellen gleichmäßig in alle Richtungen ausbreiten. Die von den Lautsprechern erzeugte Wellenfront ist gekrümmt.

Linearquellen werden in xwonder als Viereck mit einem senkrecht darauf stehenden Pfeil dargestellt. Die Position einer Linearquelle in xwonder ist irrelevant für den Höreindruck, wichtig ist nur die Ausrichtung des Pfeils. Die Lautsprecher bilden bei einer Linearquelle eine Wellenfront, die einer Geraden gleicht und den Raum komplett durchquert. Der Pfeil zeigt die Richtung an, in die sich die Wellenfront bewegt. Bewegt sich der Zuhörer durch den Raum, bleibt die Richtung, aus der er den Schall hört, immer gleich. Schallquellen dieser Art kommen in der Natur nicht vor. Ein ähnlicher Effekt kann jedoch durch eine Punktquelle erzeugt werden, deren Position sehr weit außerhalb des Raumes liegt.

In Abbildung 3.2 sind die Wellenfronten von Punktquellen und Linearquellen dargestellt.

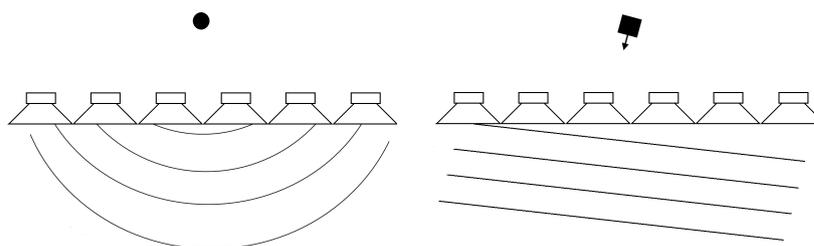


Abbildung 3.2.: Wellenfronten einer Punkt- und Linearquelle

3.4.2. Fokussierte Tonquellen

Der Benutzer hat die Möglichkeit, Tonquellen an eine beliebige Position innerhalb oder außerhalb des Raumes, in dem sich die Lautsprecher befinden, zu verschieben. Die Position kann auch weit außerhalb des Raumes liegen und ist nicht durch die Wände des Raumes oder ähnliches begrenzt, da jedes beliebige Schallfeld erzeugt werden kann. Liegt die Position einer Tonquelle innerhalb der von den Lautsprechern aufgespannten Fläche, ist es eine fokussierte Tonquelle. Ein als „focus limit“ bezeichneter Wert in der Konfigurationsdatei von twonder bestimmt, wie das Rendering fokussierter Tonquellen abläuft. Die dort angegebene Zahl gibt die maximale Distanz an, die eine fokussierte Tonquelle zu einem Lautsprecher haben kann, damit der Lautsprecher noch zum Rendering der fokussierten Tonquelle beiträgt. Das bedeutet, dass die Lautsprecher, die die geringste Distanz zur virtuellen Tonquelle haben, am Rendering beteiligt sind. Am Rendering beteiligt bedeutet, dass diese Lautsprecher die Elementarwellen erzeugen, welche für die Synthetisierung der Wellenfront notwendig ist. Bei fokussierten Tonquellen tritt das in Abschnitt 2.2 genannte Problem zur Lokalisation auf.

4. Das Labor der HAW Hamburg

Der Raum in der HAW Hamburg, in dem die WFS-Anlage und alle für den Betrieb der Software notwendigen Rechner stehen, wird in dieser Arbeit als Labor bezeichnet. In diesem Kapitel werden die unterschiedlichen Systeme und die Verwendung der Software sWONDER im Labor vorgestellt.

4.1. Die Systemumgebung des Labors

Die WFS-Anlage der Firma Four Audio besteht aus 26 Lautsprechermodulen, welche in Form eines Rechtecks an einem Gerüst aufgehängt sind. Die Fläche innerhalb der Lautsprecher beträgt ca. 5x6 Meter. Jedes Audiomodul besteht wiederum aus 26 Lautsprechern und besitzt 8 Spuren, somit stehen insgesamt 676 Lautsprecher und 208 Spuren zur Verfügung. Es werden vier Rechner verwendet, auf denen die Software als verteilte Anwendung gestartet wird: ein Mac, ein WFS-Server und zwei Nodes. Der Mac ist die Benutzerschnittstelle, auf diesem wird das Startskript ausgeführt.

Als weiteres System ist ein kamerabasiertes Trackingsystem der Firma Advanced Realtime Tracking GmbH (ARTTRACK System¹) installiert. Es besteht aus sechs Infrarotkameras, welche eine Fläche von 4x4 Metern abdecken. Der Bereich des Trackingsystems befindet sich komplett innerhalb der Lautsprecher, ist aufgrund der begrenzten Reichweite der Kameras jedoch kleiner als der Bereich der WFS-Anlage. In Abbildung 4.1 ist ein Modell der Systeme im Labor dargestellt. Auf der Abbildung ist zusätzlich eine Powerwall zu sehen, welche in dieser Arbeit nicht genutzt wird.

¹<http://www.ar-tracking.com/products/tracking-systems/arttrack-system/>

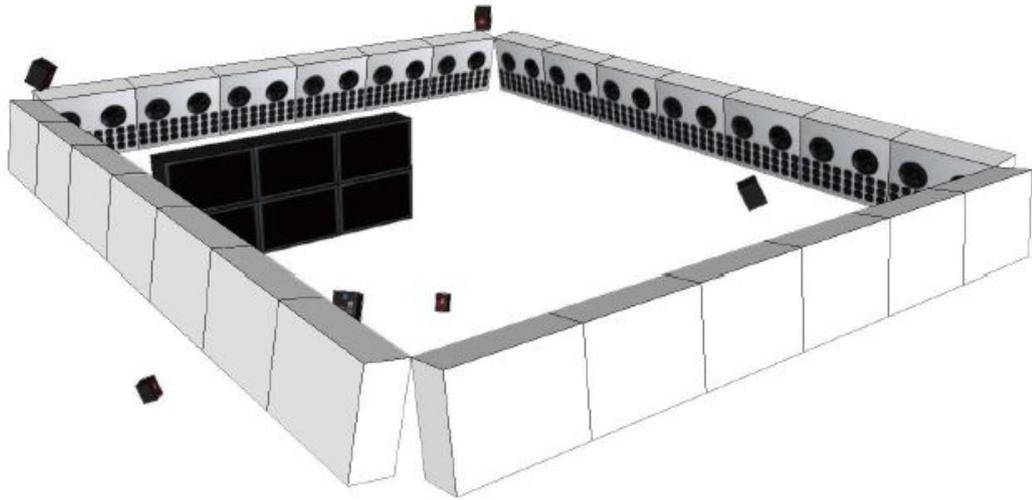


Abbildung 4.1.: 3D-Modell der Systeme im Labor [Nog12]

4.2. Verwendung der Software sWONDER

Die Komponenten der Software sWONDER laufen verteilt auf mehreren Rechnern. In diesem Kapitel wird die Verteilung der Komponenten sowie deren Konfiguration im Bezug auf fokussierte Tonquellen dargestellt.

4.2.1. Auswahl und Verteilung der Komponenten

Die Auswahl der Komponenten im Labor beschränkt sich auf cwonder, twonder und xwonder. Da der Mac die Benutzerschnittstelle ist, wird hier xwonder gestartet. Auf dem WFS-Server befindet sich cwonder, das bedeutet, alle OSC-Nachrichten werden dort empfangen. Die beiden Nodes sind für das Rendering zuständig. Auf jedem Node werden sieben Instanzen von twonder gestartet. Die Gesamtheit der Lautsprechermodule wird somit von vierzehn twonder-Instanzen verwaltet. In Abbildung 4.2 ist die Verteilung der Komponenten auf den Rechnern dargestellt, hier sind jedoch nicht alle Instanzen von twonder abgebildet. Die mit Dante beschrifteten Pfeile stellen das Dante Audio Netzwerk dar. Dante ist ein performantes System, welches die Übertragung von digitalen Audiodaten über das TCP/IP-Protokoll ermöglicht. Diese Daten werden von einer Audio Software, beispielsweise Ardour, an twonder übertragen und von dort aus an die Lautsprecher weitergeleitet.

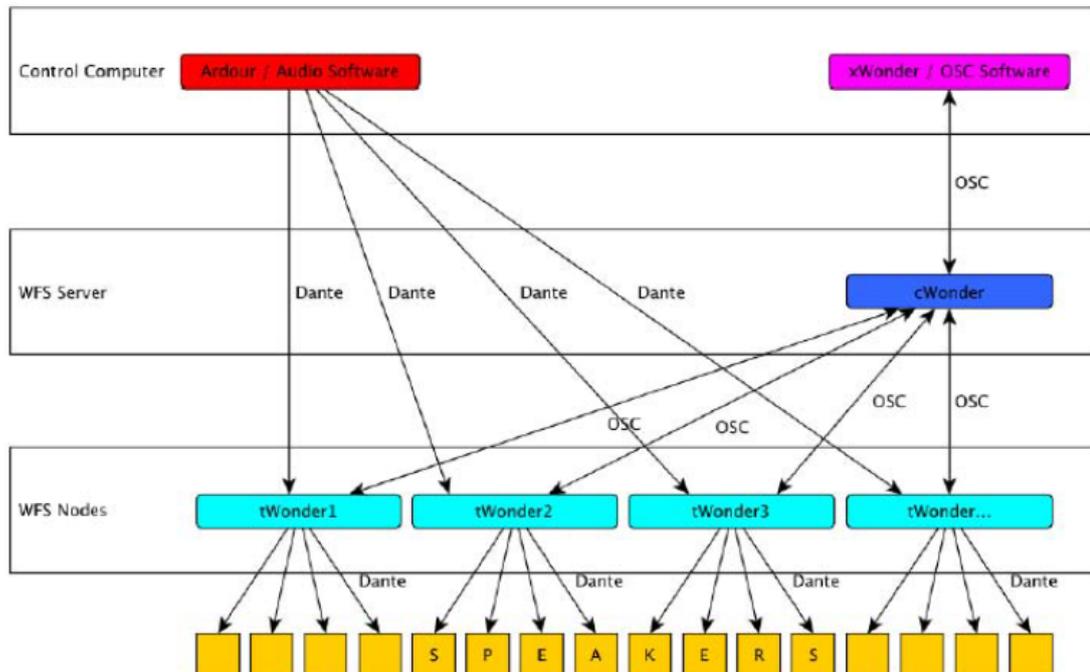


Abbildung 4.2.: Verteilung der Komponente von sWONDER im Labor [Fou]

Zum Starten der verteilten Anwendung wird ein Startskript verwendet, welches die Komponenten cwonder und twonder startet und miteinander verbindet. Anschließend kann xwonder manuell auf dem Mac gestartet werden.

4.2.2. Rendering fokussierter Tonquellen

Der Wert „focus limit“, welcher in der Konfigurationsdatei von twonder festgelegt werden kann, bezieht sich auf das Rendering fokussierter Tonquellen. Er wird in Abschnitt 3.4.2 genauer erläutert. Im Labor ist das „focus limit“ auf 2.8 Meter gesetzt. Das entspricht dem Abstand eines Lautsprechermoduls zum Mittelpunkt der beschallten Fläche. Da die Lautsprechermodule in einem Rechteck aufgehängt sind, erfüllen jedoch nur die Lautsprecher der beiden längeren Seiten diese Bedingung. In Abbildung 4.3 ist für vier verschiedene Lautsprecher, welche sich an den markierten Stellen befinden (jeweils in der Mitte jeder Seite des Rechtecks), dargestellt, bis zu welcher Grenze eine fokussierte Tonquelle von diesen Lautsprechern gerendert wird.

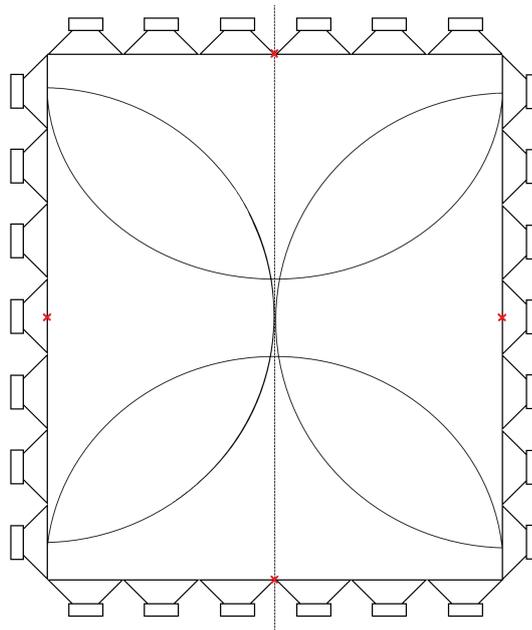


Abbildung 4.3.: Visualisierung des Wertes „focus limit“

In der Abbildung stellt ein Halbkreis den Bereich dar, in dem sich eine fokussierte Tonquelle befinden muss, damit der Lautsprecher, der sich in der Mitte des Kreises befindet, zum Rendering der fokussierten Tonquelle beiträgt. Der Radius des Halbkreises ist durch den Wert „focus limit“ festgelegt. Wie in der Abbildung zu sehen, ist das Rendering im mittleren Bereich des Raumes nicht konsistent, da die Lautsprecher ein Rechteck und kein Quadrat bilden. Befindet sich eine fokussierte Tonquelle genau im Mittelpunkt des Raumes, werden Lautsprecher der linken und rechten Seiten aktiv, nicht jedoch Lautsprecher an den kürzeren Seiten des Rechtecks. Wird eine fokussierte Tonquelle über die Mittellinie des Raumes geschoben, „springt“ das Rendering von der einen auf die andere Lautsprecherseite um. Es bietet sich an, die Mitte des Raumes als Grenze für das Rendering zu nehmen, wenn nicht bekannt ist, wo der Zuhörer steht. Denn so ist der Bereich, in dem der Zuhörer eine fokussierte Tonquelle korrekt hört, möglichst groß. Die Problematik, welche sich beim Rendering von fokussierten Tonquellen ergibt, wird in Abschnitt 2.2 erklärt. Befindet sich der Zuhörer im Mittelpunkt der Fläche, hört er alle fokussierten Tonquellen korrekt. In Abbildung 4.4 ist beispielhaft für eine virtuelle Tonquelle dargestellt, an welcher Position der Zuhörer die virtuelle Tonquelle korrekt und an welcher er diese nicht korrekt wahrnehmen würde. Da sich die fokussierte Tonquelle im Bild in der linken Hälfte der Anlage befindet, sind die Lautsprecher auf der linken Seite aktiv. Diese erzeugen die Elementarwellen für die konkave Wellenfront.

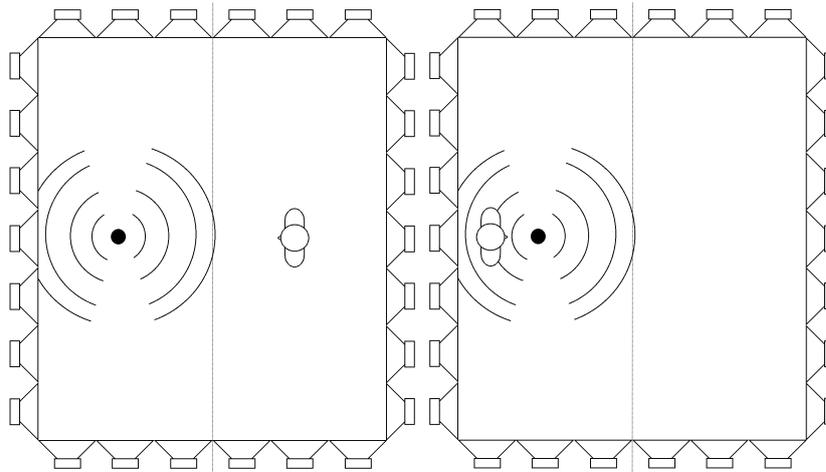


Abbildung 4.4.: Position des Zuhörers für das korrekte und nicht korrekte Hören einer fokussierten Tonquelle im Labor

4.3. Das ARTTRACK System

Die sechs Kameras des ARTTRACK Systems, welche im Labor installiert sind, senden Infrarotlicht aus. Objekte mit einer reflektierenden Oberfläche werden von den Kameras erkannt, da sie das von den Kameras ausgesendete Licht reflektieren [Adv14]. Für das Tracking werden kugelförmige Marker verwendet, welche eine stark reflektierende Oberfläche besitzen. Wird eine Reflexion von mehreren Kameras erkannt, kann durch die Überschneidung auf den Bildern der Kameras die Position eines Markers im dreidimensionalen Raum berechnet werden. Zusätzlich gibt es Targets, welche einen Körper aus mehreren Markern bilden. Diese sind in einer bestimmten Konstellation angeordnet, sodass die Kameras zwischen verschiedenen Targets unterscheiden können. Zusätzlich kann das System die Ausrichtung eines Targets im Raum bestimmen, da sich die Abstände zwischen den Markern nicht verändern, auch wenn das Target gedreht wird.

Die Software des Trackingsystems heißt dtrack und läuft auf einem Rechner im Labor. Über diese kann das System gestartet und gestoppt werden. In Abbildung 4.5 ist die Oberfläche der Software dargestellt. Sie bietet verschiedene Sichten auf das laufende System. Eine Sicht ist die grafische Darstellung der Kamerabilder. Auf dieser werden die von den Kameras erfassten Reflexionen durch Targets als Kreuze dargestellt. Ein rotes Kreuz bedeutet, dass die Reflexion nur von einer Kamera erkannt wurde, ein gelbes Kreuz, dass die Reflexion von mehreren Kameras erkannt wurde. Andere Einstrahlungen, die nicht durch Marker oder Targets hervorgerufen werden, werden in Form von Kreisen dargestellt.

4. Das Labor der HAW Hamburg

Eine weitere Sicht ist die Anzeige der aktuellen Positions- und Rotationsdaten eines Targets. Die Positionsdaten werden als x, y und z und die Rotationsdaten als Rx, Ry und Rz bezeichnet. Damit das System die Position und Ausrichtung eines Targets im Raum berechnen und aktualisieren kann, ist es notwendig, dass das Target von mehreren Kameras erkannt wird. Die rot hinterlegten Targets wurden vom System noch nicht erkannt und besitzen somit noch keine Daten.

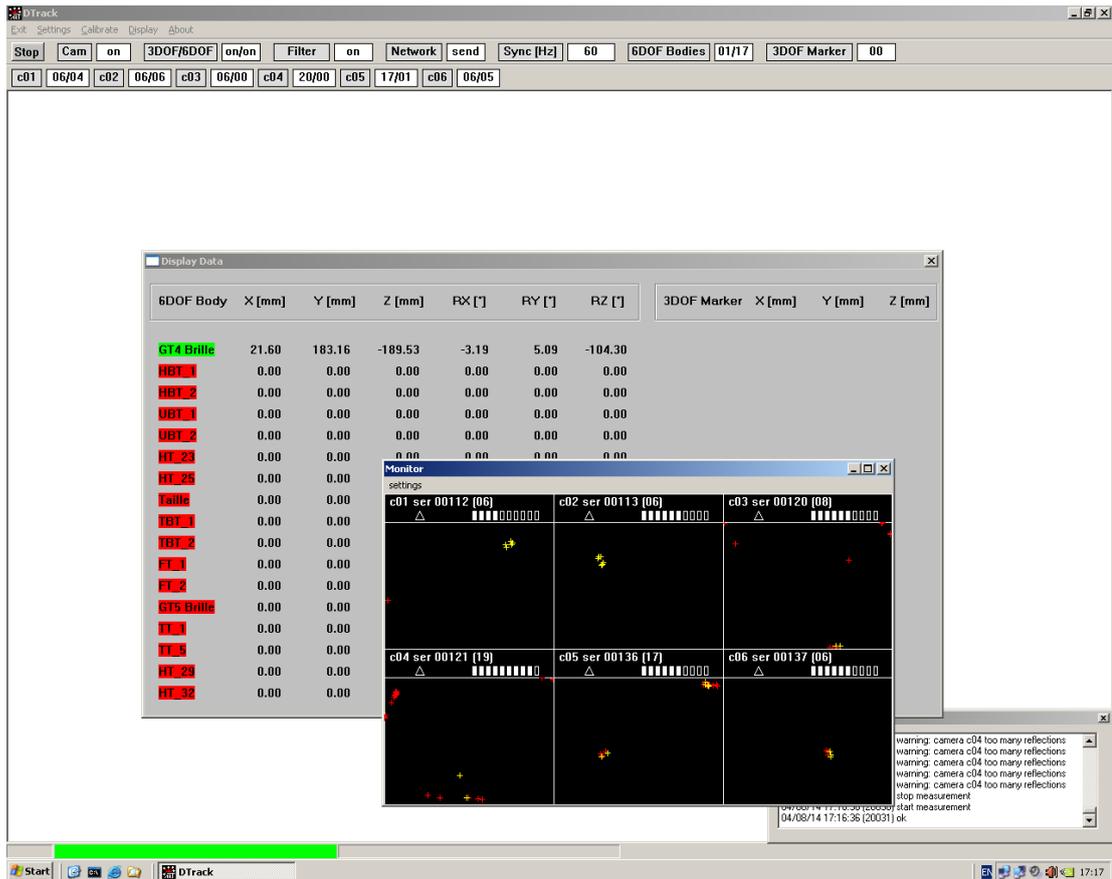


Abbildung 4.5.: Grafische Oberfläche der Software dtrack

5. Modifikation der Software sWONDER

Der praktische Teil dieser Arbeit besteht darin, die bestehende Software sWONDER so zu modifizieren, dass ein Rendering fokussierter Tonquellen in Abhängigkeit der Zuhörerposition möglich ist.

5.1. Idee und Anforderungen

In der aktuellen Version von sWONDER wird das Rendering durch die Lautsprecher übernommen, deren Distanz zur fokussierten Tonquelle am geringsten ist. So ist der Bereich, in dem ein Zuhörer stehen muss um die fokussierte Tonquelle korrekt zu hören, möglichst groß. Es gibt jedoch immer einen Bereich, in dem der Zuhörer die Tonquelle nicht lokalisieren kann und den Ton aus einer falschen Richtung wahrnimmt. Die Idee ist es, die Software so zu modifizieren, dass ein Zuhörer fokussierte Tonquellen von allen Positionen aus korrekt wahrnehmen kann, selbst wenn er sich um diese herum bewegt. Eine Anforderung an die modifizierte Software ist ein neues Rendering-Verfahren, das sich an der Position des Zuhörers orientiert. Die Konstellation von Tonquelle und Zuhörer entscheidet darüber, welche Lautsprecher das Rendering übernehmen. Eine weitere Anforderung ist die sofortige Reaktion auf Änderungen der Zuhörerposition. Das OSC-Protokoll kann diese Anforderung erfüllen, da es für Echtzeit-Anwendungen geeignet ist [WFM03].

5.2. Vorgehensweise

Bevor Änderungen an der Software vorgenommen werden können, muss eine Vorgehensweise erarbeitet werden.

5.2.1. Überblick über die Software gewinnen

Der erste Schritt besteht darin, einen Überblick über das Programm zu gewinnen. Die Struktur der Software ist komplex, da eine Vielzahl von unterschiedlichen Komponenten existiert. Jede einzelne dieser Komponenten repräsentiert wiederum ein Zusammenspiel mehrerer Klassen.

Ein wichtiger Schritt dieser Arbeit besteht darin, die relevanten Stellen zur Umsetzung des Vorhabens zu finden. Dabei sind die Schlüsselworte Rendering, fokussierte Tonquellen und auch OSC hilfreich.

Ein tieferes Verständnis der einzelnen Komponenten ist notwendig, um den Radius der Suche einzugrenzen. In einem Handbuch, welches in Form einer Textdatei in dem Softwarepaket enthalten ist, wird die Funktionsweise jeder Komponente grob beschrieben. Diese Informationen können für eine erste Analyse genutzt werden, gehen jedoch nicht auf technische Details ein. Diese Arbeit beschränkt sich auf die Komponenten `cwonder` und `twonder`. `cwonder` ist notwendig, da nur diese Komponente Informationen in Form von OSC-Nachrichten von außen empfangen und an andere Komponenten weiterleiten kann. `twonder` ist die Rendering Einheit und somit für das Rendering fokussierter Tonquellen zuständig. Als nächstes muss die Verbindung zwischen `cwonder` und `twonder` analysiert werden. Dazu ist ein Einblick in die Funktionsweise der Klassen notwendig. Da sich das Rendering-Verfahren in `twonder` befindet, wird hier die Entscheidung getroffen, ob eine Tonquelle fokussiert ist oder nicht. Dagegen spricht die Tatsache, dass mehrere Instanzen von `twonder` gleichzeitig gestartet werden können und jede Instanz nur einen Teil der Lautsprecher verwaltet. Dies kann durch einen Blick in die Array Konfiguration im Labor bestätigt werden. Für die Definition fokussierter Tonquellen ist jedoch das Wissen über die Lage und Größe des gesamten Innenraums zwischen den Lautsprechern notwendig. In der Dissertation von Marije Baalman wird dieser Innenraum als „render polygon“ bezeichnet [Baa08]. Die Definition des Renderpolygons steht in der Konfigurationsdatei von `cwonder`. Im Labor wird das Polygon beispielsweise durch die Angabe der Koordinaten der vier Eckpunkte beschrieben, da hier der Innenraum zweidimensional betrachtet ein Rechteck ist. Diese Information wird bei der Initialisierung von `cwonder` an jede Instanz von `twonder` weitergegeben, bevor das Rendering beginnt. Somit kennt jede Instanz die Fläche des Renderpolygons und kann berechnen, ob sich eine Tonquelle darin befindet. Wenn dies der Fall ist, ist es eine fokussierte Tonquelle und das Rendering muss entsprechend angepasst werden.

5.2.2. Modifikation des Rendering-Verfahrens

Um eine Wiedergabe fokussierter Tonquellen in Abhängigkeit der Zuhörerposition zu realisieren, muss der Vorgang des Renderings analysiert werden, wie er momentan im Programmcode von `twonder` implementiert ist. Danach kommt es zu konkreten Überlegungen, nach welchen Prinzipien das neue Rendering implementiert werden muss. Damit der Zuhörer jede fokussierte Tonquelle korrekt lokalisieren kann, muss er sich immer im Bereich der konvexen Wellenfront befinden. Das ist die Wellenfront, welche sich ab dem Punkt der fokussierten Tonquelle in

Richtung Zuhörer bewegt. Diese Bedingung ist erfüllt, wenn die Lautsprecher, die sich vom Zuhörer aus gesehen „hinter“ der fokussierten Tonquelle befinden, das Rendering übernehmen. Je nach Zuhörerposition sind es andere Lautsprecher, auf die diese Bedingung zutrifft.

5.2.3. Verwendung des Open Sound Control Protokolls

Eine Möglichkeit, dem Programm die aktuelle Zuhörerposition mitzuteilen, ist das Versenden von OSC-Nachrichten. Die Informationen auf diese Weise zu Versenden hat mehrere Gründe. Da das Programm bereits auf Basis des OSC-Protokolls kommuniziert, ist die Funktionsweise des Sendens und Empfangens dieser Art von Nachrichten bereits implementiert. Außerdem kann sich die Zuhörerposition ständig ändern, daher ist es notwendig, dass das Programm in Echtzeit darauf reagiert.

5.2.4. Anbindung des ARTTRACK Systems

Das kamerabasierte Trackingsystem wird verwendet, um die Position des Zuhörers zu ermitteln. Dazu muss der Zuhörer ein Target bei sich tragen. Die Position des Targets wird von den Kameras erfasst und per Multicast an das interne Netzwerk gesendet. Diese Informationen müssen erst verarbeitet werden, in diesem Fall ist eine Umrechnung vom Koordinatensystem des Trackingsystems in das der WFS-Anlage notwendig. Anschließend müssen sie in Form von OSC-Nachrichten an sWONDER gesendet werden. An dieser Stelle kann das Programm MoWeC verwendet werden, welches von Malte Nogalski im Rahmen seiner Bachelorarbeit an der HAW Hamburg entwickelt wurde. Es stellt eine Schnittstelle zwischen ARTTRACK System und WFS-Anlage dar [Nog12].

5.3. Implementation

Nachdem analysiert wurde, welche Schritte nacheinander erfolgen müssen, um das gewünschte Ergebnis zu erhalten, erfolgt die konkrete Implementation. Dazu wird die Software sWONDER, welche open-source verfügbar ist, in der aktuell erhältlichen Version (Version 3.1.0) modifiziert.

5.3.1. Beschreibung des Aufbaus von sWONDER-Komponenten

Im Folgenden werden die beiden für die Modifikation relevanten Komponenten, cwonder und twonder, im Detail beschreiben.

5.3.1.1. Aufbau der Komponente cwonder

Die Komponente cwonder besitzt einen OSC-Server (oscinctrl.cpp), welcher für das Empfangen von OSC-Nachrichten zuständig ist. Hier werden alle OSC-Nachrichten definiert, die cwonder empfangen kann. In jeder Definition muss eine Callback-Methode angegeben werden, welche aufgerufen wird, sobald die entsprechende Nachricht eintrifft.

Die ausführbare Methode von cwonder heißt main (in main.cpp). Diese Methode liest die Konfigurationsdatei von cwonder aus, startet cwonder als Daemon und erzeugt anschließend eine Instanz des OSC-Servers.

Die Callback-Methoden, welche bei eintreffenden OSC-Nachrichten aufgerufen werden, befinden sich in der Klasse, die den gleichen Namen trägt wie die Komponente selbst (cwonder.cpp). Da die Nachrichten von cwonder über Streams an andere Komponenten weitergeleitet werden, sind Methoden implementiert, die diese Verbindungen erstmalig aufbauen (beispielsweise renderStreamConnect). Bei Betrachtung der Callback-Methoden (diese Methoden beginnen mit „set“) wird deutlich, dass eine OSC-Nachricht nicht immer an alle Streams gesendet wird, da nicht jede Information für jede Komponente von Relevanz ist. Beispielsweise sendet cwonder die Informationen, die für eine Visualisierung notwendig sind, an den sogenannten Visual Stream. An diesem ist die Komponente xwonder angemeldet. Informationen, die für das Rendering notwendig sind, werden über den Render Stream an twonder gesendet.

5.3.1.2. Aufbau der Komponente twonder

Die Komponente twonder besitzt alle Informationen bezüglich des Renderings. Beispielhaft seien hier die Klassen für die Lautsprecher (speaker.cpp) und die virtuellen Tonquellen (source.cpp) genannt. Ein Lautsprecher besitzt eine Position und eine Normale, beides beschrieben durch x-, y- und z-Koordinate. Die Normale zeigt die Ausrichtung des Lautsprechers. Eine virtuelle Tonquelle kann eine Punktquelle (PointSource) oder eine Linearquelle (PlaneWave) sein. Eine Punktquelle besitzt eine Position beschrieben durch einen zweidimensionalen Vektor, welcher aus einer x- und y-Koordinate besteht.

In der source-Klasse ist die Methode implementiert, welche das Rendering durchführt (calcDelayCoeff). Sie ist zweimal implementiert, einmal für Punkt- und einmal für Linearquellen, da das Rendering der beiden Quellentypen unterschiedlich verläuft. In dieser Methode wird die Verzögerung und Amplitude einer Elementarwelle berechnet, welche von einem Lautsprecher erzeugt wird. Vorab werden verschiedene Prüfungen durchlaufen, da die Berechnung eventuell nicht durchgeführt werden muss. Das ist beispielsweise der Fall, wenn die Tonquelle eine fokussierte Tonquelle ist und sie sich außerhalb des „focus limit“ für diesen Lautsprecher

befindet. Die Prüfung, ob eine Tonquelle fokussiert ist oder nicht, findet auch in dieser Klasse statt (`PointSource::isFocused`). Sie ist nur für Punktquellen implementiert, da eine Linearquelle nicht fokussiert sein kann.

Die Hauptklasse besitzt den gleichen Namen wie die Komponente (`twonder.cpp`). In der ausführbaren Methode (`main`) wird ähnlich wie bei `cwonder` die Konfigurationsdatei von `twonder` ausgelesen und die Komponente als Daemon gestartet. Der OSC-Server zum Empfangen von OSC-Nachrichten wird ebenfalls gestartet und die OSC-Nachrichten, die `twonder` annehmen kann, werden definiert. Neben der `main`-Methode kann der Großteil der restlichen Methoden in folgende Gruppen aufgeteilt werden: `SourceAggregate`, `SourceArray`, `SpeakerArray`, `Commands` und `OSC-Handler`. Die Unterteilung ist durch Kommentare gekennzeichnet und dient der besseren Übersicht über den Programmcode.

Das `SourceAggregate` repräsentiert eine `Source`, also eine virtuelle Tonquelle, mit zusätzlichen Eigenschaften. Jedem `SourceAggregate` ist eine Punktquelle oder Linearquelle zugeordnet. `SourceArray` und `SpeakerArray` repräsentieren Listen, in denen alle `SourceAggregates` bzw. Lautsprecher gespeichert werden. Fügt der Benutzer eine neue virtuelle Tonquelle hinzu, so wird ein neues `SourceAggregate` erstellt und dem `SourceArray` hinzugefügt. Das `SpeakerArray` wird bereits beim Start der Komponente mit `Speaker`-Objekten, welche für die Lautsprecher stehen, befüllt und verändert sich während des Programmverlaufes nicht mehr.

Die `Commands` sind Befehle, die als Reaktion auf bestimmte OSC-Nachrichten ausgeführt werden. Ein `MoveCommand` setzt beispielsweise eine virtuelle Tonquelle auf eine neue Position. Es ist möglich, den OSC-Nachrichten Zeitstempel mitzugeben, welche angeben, wann ein solcher Befehl ausgeführt werden soll.

Die `OSC-Handler` sind die `Callback`-Methoden und werden aufgerufen, sobald OSC-Nachrichten eintreffen. Jede OSC-Nachricht besitzt eine entsprechende `Handler`-Methode, die die Argumente der Nachricht ausliest. Je nach Typ der Nachricht werden die Änderungen entweder durch die Methode selbst übernommen oder ein `Command` wird gestartet, der die Änderungen durchführt. Eine weitere Methode in dieser Klasse ist die `process`-Methode. Diese iteriert über das `SourceArray` und das `SpeakerArray` und ruft für jede Konstellation aus Lautsprecher und virtueller Tonquelle einmal die `Rendering`-Methode auf.

5.3.2. Neues Rendering-Verfahren für eine festen Zuhörerposition

Im ersten Schritt wird das `Rendering` so modifiziert, dass es sich an die Position des Zuhörers anpasst. Da noch keine OSC-Nachrichten bezüglich der Zuhörerposition übertragen werden, wird zunächst ein beliebiger Punkt im Koordinatensystem in den Programmcode eingetragen. Dieser soll die Zuhörerposition repräsentieren. Zum Testen des neuen `Rendering`-Verfahrens

muss sich der Zuhörer auf diese angegebene Position stellen. An dieser Stelle müssen alle fokussierten Tonquellen aus der korrekten Richtung wahrgenommen werden können. Das Rendering-Verfahren ist in der Methode `calcDelayCoeff` der Klasse `source.cpp` implementiert, welche zu `t wonder` gehört. Die Berechnungen zur Verzögerung und Amplitude einer Elementarwelle werden nicht verändert, sondern die Entscheidung, welche Lautsprecher das Rendering übernehmen. Im ursprünglichen Rendering-Verfahren wird zunächst die Distanz zwischen der fokussierten Tonquelle und dem aktuellen Lautsprecher berechnet. Da die Positionen der fokussierten Tonquelle und des Lautsprechers als Ortsvektoren interpretiert werden können, werden diese voneinander abgezogen. Das Ergebnis ist der Vektor, der die virtuelle Tonquelle und den Lautsprecher miteinander verbindet. Anschließend wird die Länge dieses Vektors berechnet. Diese entspricht der Distanz zwischen virtueller Tonquelle und Lautsprecher. Ist die Distanz kleiner als der Wert „focus limit“ aus der Konfigurationsdatei, trägt der Lautsprecher zum Rendering der fokussierten Tonquelle bei und die Berechnungen werden durchgeführt. Ist die Distanz größer als das „focus limit“, trägt er nicht zum Rendering bei. Das modifizierte Rendering orientiert sich nicht an dem „focus limit“ aus der Konfigurationsdatei. Der Wert wird somit nicht mehr benötigt. Die Berechnung der Distanz zwischen fokussierter Tonquelle und Lautsprecher wird jedoch beibehalten. Auf die gleiche Art und Weise wird die Distanz zwischen der Zuhörerposition und dem Lautsprecher berechnet. Ist die Distanz zwischen fokussierter Tonquelle und Lautsprecher kleiner als die Distanz zwischen Zuhörer und Lautsprecher, trägt der Lautsprecher zum Rendering bei. Ist die Distanz zwischen fokussierter Tonquelle und Lautsprecher jedoch die größere, befindet sich der Zuhörer zwischen fokussierter Tonquelle und Lautsprecher. Deshalb darf der Lautsprecher in diesem Fall nicht zum Rendering beitragen. Für den Lautsprecher auf der gegenüberliegenden Seite sind die Distanzverhältnisse umgekehrt, deshalb wird dieser zum Rendering beitragen. In Abbildung 5.1 ist das neue Rendering-Verfahren dargestellt. Die aktiven Lautsprecher sind grau eingefärbt, für sie ist die Distanz zur fokussierten Tonquelle kleiner als die Distanz zum Zuhörer. Dort, wo der grau eingefärbte Bereich in den weißen Bereich übergeht, befindet sich der Punkt, an dem die Distanzverhältnisse gleich sind. Der weiße Bereich zeigt die nicht aktiven Lautsprecher, für sie ist die Distanz zur fokussierten Tonquelle größer als die Distanz zum Zuhörer.

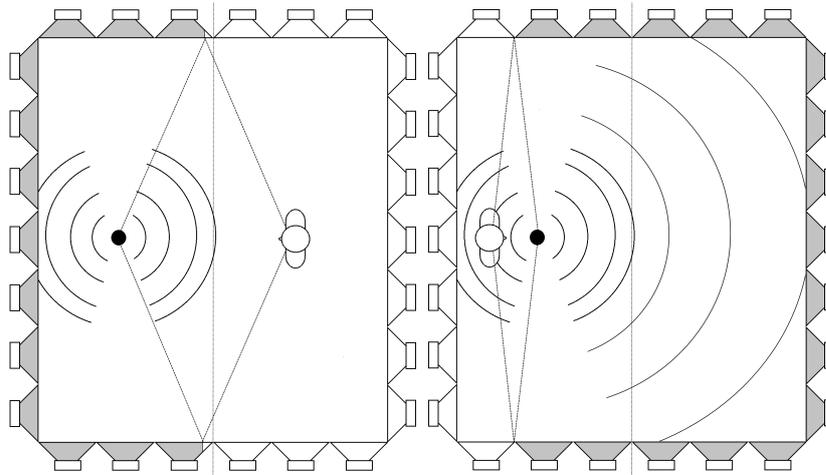


Abbildung 5.1.: Darstellung der aktiven Lautsprecher für unterschiedliche Zuhörerpositionen bei dem neuen Rendering-Verfahren

5.3.3. Einführung einer neuen OSC-Nachricht

Um sWONDER die Position des Zuhörers, welche vom Trackingsystem erfasst wird, mitzuteilen, muss eine neue OSC-Nachricht zu den bereits bestehenden hinzugefügt werden. Diese wird zunächst an cwonder gesendet und von dieser Komponente an twonder weitergeleitet.

5.3.3.1. Definition der OSC-Nachricht

Der Aufbau der neuen OSC-Nachricht orientiert sich an der Nachricht, die für die Positionsänderung von virtuellen Tonquellen bestimmt ist. Die Adresse dieser Nachricht lautet „/WONDER/source/position“. Für die neue Nachricht wird die Adresse „/WONDER/listener/position“ gewählt. Die Definition der Parameter ergibt sich aus den Informationen, welche übertragen werden müssen. Da eine Position übertragen werden soll, enthält die Nachricht zwei Parameter für die Übertragung der x- und y-Koordinate. Im Hinblick darauf, dass die Software in Zukunft mehrere Zuhörer berücksichtigen könnte, wird zusätzlich eine ID mitgegeben. Wenn jeder Zuhörer eine eigene ID besitzt, kann beim Rendering zwischen mehreren Zuhörern unterschieden werden. Ein Beispiel für eine OSC-Nachricht mit dem beschriebenen Aufbau könnte so aussehen:

```
/WONDER/listener/position, 0, 0.25, 2.5
```

Diese Nachricht enthält die Information, dass sich der Zuhörer mit der ID 0 an der Position mit der x-Koordinate 0.25 und der y-Koordinate 2.5 befindet.

5.3.3.2. Modifikation der OSC-Server

cwonder und twonder können nur die OSC-Nachrichten empfangen, die in den zugehörigen OSC-Servern definiert sind. Deshalb müssen die Server bei der Einführung einer neuen OSC-Nachricht um deren Definition erweitert werden.

Der OSC-Server und dessen Methoden sind in der Header-Datei `oscin.h`, welche sich im Ordner `include` befindet, definiert. Die Klasse ist eine Wrapperklasse für den OSC-Server von `liblo`. Die konkrete Implementierung dieser Methoden in `cwonder` befindet sich in der Datei `oscinctrl.cpp`. Darunter befindet sich eine Methode namens `addMethods`, in der alle OSC-Nachrichten definiert werden. Diese Methode wird um eine zusätzliche Zeile erweitert:

```
addMethod( "/WONDER/listener/position", "iff", listenerPositionHandler, this );
```

Die erste Zeichenkette steht für die Adresse, die zweite für die Datentypen der Parameter. Der String „iff“ bedeutet, dass eine ID vom Typ `integer` und zwei Koordinaten in Form von `float`-Werten übertragen werden. An dritter Stelle wird die Callback-Methode angegeben, welche aufgerufen wird, wenn eine OSC-Nachricht mit diesem Aufbau empfangen wird. Diese Methode, welche den Namen `listenerPositionHandler` trägt, wird anschließend so implementiert wie die bereits vorhandenen Callback-Methoden anderer OSC-Nachrichten. Allgemein ist die Funktionsweise der Callback-Methoden so, dass sie das Weiterleiten der Nachricht an die Streams anstoßen und anschließend eine Antwort an den Empfänger zurücksenden. Diese Antwort enthält den String „/WONDER/reply“. Sie dient dem Empfänger als Bestätigung, dass die OSC-Nachricht bei `cwonder` angekommen ist. In der `listenerPositionHandler` Methode wird zunächst eine weitere Methode aufgerufen (`setListenerPosition`), welche dafür zuständig ist, die Nachricht an ausgewählte Streams weiterzuleiten. Die Nachricht wird an den `Render Stream` gesendet, da `twonder` ein Empfänger dieses Streams ist, und an den `Visual Stream`, da es zukünftig visuelle Darstellungen der Zuhörerposition geben könnte. `xwonder` ist ebenfalls ein Empfänger des `Visual Streams`, erkennt die Nachricht jedoch nicht, da sie in `xwonder` nicht definiert ist. Die Methode `setListenerPosition` prüft vor dem Senden an die Streams, ob die Parameter der Nachricht korrekt sind. Ist die Zuhörer-ID kleiner als `null`, wird die Nachricht nicht weitergeleitet und der Rückgabewert der Methode ist `1`. Liegt der Parameter im zulässigen Bereich (größer oder gleich `Null`), wird die Nachricht weitergeleitet und der Wert `0` zurückgegeben. Diese Zahl gibt an, dass keine Fehler aufgetreten sind. Der Rückgabewert wird wieder an den `listenerPositionHandler` übergeben. Diese Methode formuliert nun die Antwortnachricht für den Empfänger und integriert den Rückgabewert darin. Auf diese Weise kann dem Empfänger zusätzlich mitgeteilt werden, ob die Nachricht von `cwonder` an die Streams weitergeleitet wurde oder nicht.

Auch im OSC-Server von `twonder` muss diese Nachricht implementiert werden. Eine Instanz

des OSC-Servers wird in der main-Methode in twonder.cpp erzeugt. Hier werden auch die OSC-Nachrichten und die zugehörigen Callback-Methoden definiert. Folgende Zeile wird hinzugefügt:

```
oscServer->addMethod( "/WONDER/listener/position", "iff", oscListenerPosHandler );
```

Der Methodenaufruf ist der gleiche wie in cwonder. Diesmal ist der Name der Callback-Methode `oscListenerPosHandler`, um die Namenskonvention in twonder beizubehalten. Diese Methode liest die Parameter der OSC-Nachricht aus und führt einen Command aus, um die Position des Zuhörers zu aktualisieren. Die Implementierung dieses Commands wird in Abschnitt 5.3.4.2 beschrieben.

5.3.4. Einbinden des Zuhörers und seiner Position

Der nächste Schritt besteht darin, den Zuhörer und dessen Position in das Programm zu integrieren. Die Position muss stets aktualisiert werden, wenn OSC-Nachrichten zur aktuellen Zuhörerposition eintreffen. Außerdem muss die Position des Zuhörers bis in die Rendering-Methode durchgereicht werden, damit das Rendering anhand der aktuellen Position durchgeführt werden kann.

5.3.4.1. Die Klasse Listener und ListenerArray

Da das Programm objektorientiert aufgebaut ist, wird eine neue Klasse für den Zuhörer eingeführt, welche `Listener` genannt wird. Der `Listener` besitzt eine Position in Form einer x- und y-Koordinate. Im Hinblick auf mehrere Zuhörer wird auch eine neue Klasse `ListenerArray` implementiert, welche eine Liste von mehreren Zuhörern darstellt. Der Zuhörer an der ersten Stelle besitzt die ID 0. Bei der Erstellung des `ListenerArray`s kann die gewünschte Anzahl an Zuhörern übergeben werden. Das `ListenerArray` wird in der main-Methode von twonder erstellt. Als Parameter, welcher die Anzahl der Zuhörer angibt, wird eine 1 übergeben. Im Konstruktor des `ListenerArray`s wird anschließend ein `Listener` erstellt und in die Liste eingefügt. Als Defaultwert für die Position der `Listener`-Instanz wird der Mittelpunkt der WFS-Anlage im Labor angegeben. Diese Position wird erst dann aktualisiert, wenn die erste OSC-Nachricht zur aktuellen Zuhörerposition eintrifft.

5.3.4.2. Aktualisieren der Zuhörerposition (ListenerMoveCommand)

Die Position des Zuhörers wird per `ListenerMoveCommand` neu gesetzt. Die Callback-Methode der neuen OSC-Nachricht erzeugt eine Instanz dieser Klasse und startet den Command mit `execute`. Dann wird der `Listener` mit der entsprechenden ID in dem `ListenerArray` ausgewählt

und seine Positionsdaten werden neu gesetzt. Dies geschieht jedoch nur, wenn ein Listener mit der angegebenen ID vorhanden ist.

5.3.4.3. Übergabe an das Rendering

Da nun die Position des Zuhörers gespeichert und per OSC-Nachricht aktualisiert werden kann, muss sie an die Rendering-Methode (`calcDelayCoeff`) übergeben werden. Für den Fall, dass es in Zukunft mehrere Zuhörer gibt, wird das gesamte `ListenerArray` an die Rendering-Methode gegeben. In der `process`-Methode von `twonder` wird für jede Kombination von Lautsprecher und Tonquelle eine Methode namens `getDelayCoeff` aufgerufen. Diese ruft wiederum die Methode `calcDelayCoeff` auf, welche das Rendering durchführt. Das `ListenerArray` wird an die Methode `getDelayCoeff` übergeben und bis in die Methode `calcDelayCoeff` durchgereicht. Dazu müssen die Signaturen der beiden Methoden um den Parameter des `ListenerArrays` erweitert werden. Diese Erweiterung muss sowohl bei deren Definitionen in der Header-Datei (`source.h`), als auch bei deren Implementierung (`source.cpp`) vorgenommen werden. Die Position des Zuhörers ist zwar nur bei Punktquellen relevant, doch die Methodensignaturen müssen für beide Quellentypen erweitert werden. Das liegt daran, dass sowohl Punkt- als auch Linearquellen von derselben Oberklasse `Source` ableiten und diese die Methodensignaturen vorgibt. Zum Zeitpunkt des Renderings ist noch nicht bekannt, ob eine virtuelle Tonquelle eine Punkt- oder Linearquelle ist, deshalb müssen die Methoden bereits in der Oberklasse `Source` definiert sein. Das Rendering von Linearquellen ändert sich jedoch nicht, da das `ListenerArray` hier nicht verwendet wird. In der Methode `calcDelayCoeff` der Punktquelle wird der Listener an der Stelle 0 aus dem Array genommen und dessen Position abgefragt. Das anschließende Rendering wird anhand dieser Zuhörerposition durchgeführt.

5.3.5. Modifikation der Schnittstelle zum ARTTRACK System

Neben der Modifikation der Software sWONDER muss das Programm MoWeC, welches die Schnittstelle zwischen Trackingsystem und WFS-Anlage bildet, erweitert werden. Ziel dabei ist es, die Position des Zuhörers zu verfolgen und als OSC-Nachricht an `cwonder` zu schicken.

5.3.5.1. Verwendung eines Targets als Position des Zuhörers

Das Trackingsystem bietet mehrere Targets, welche von den Infrarotkameras erkannt werden. Trägt der Zuhörer eines dieser Targets bei sich, kann seine Position im Raum verfolgt werden. Für diesen Zweck wird ein Brillentarget verwendet, das in Abbildung 5.2 dargestellt ist. Diese Art von Target bietet den Vorteil, dass der Zuhörer es nicht ständig in der Hand halten muss.

Zusätzlich befindet es sich stets auf der Höhe des Kopfes bzw. der Ohren des Zuhörers. Diese Position ist für das Hören einer virtuellen Tonquelle relevant.



Abbildung 5.2.: Das Brillentarget

Die Software MoWeC, welche in Java geschrieben ist, wird um eine zusätzliche Komponente erweitert. Diese wird als PersonTracker bezeichnet. Es ist bereits möglich, die Position eines Targets zu erfassen und in das Koordinatensystem der WFS-Anlage umzurechnen. Auch das Erstellen und Senden von OSC-Nachrichten an cwonder ist bereits implementiert. Dazu wird das Framework JavaOSC¹ verwendet. Der PersonTracker erfasst die Position des Targets, rechnet sie in das Koordinatensystem der WFS-Anlage um und sendet sie in Form der neu definierten Nachricht (/WONDER/listener/position) an sWONDER. Die Zuhörer-ID ist dabei immer 0, da mehrere Zuhörer noch nicht berücksichtigt werden. Da das Trackingsystem 60 Bilder pro Sekunde erfasst, wird die Nachricht auch so oft an cwonder gesendet.

5.3.5.2. Darstellung der Zuhörerposition als virtuelle Tonquelle

Eine visuelle Darstellung der Zuhörerposition wird in dieser Arbeit nicht umgesetzt. Der MoWeC bietet jedoch die Möglichkeit an, eine virtuelle Tonquelle einem Target zuzuordnen. Das bedeutet, dass die aktuelle Position des Targets mit der Adresse „/WONDER/source/position“ versendet wird. Diese Nachricht setzt eine virtuelle Tonquelle immer an die Position des Targets, wodurch sich die virtuelle Tonquelle synchron zu dem Target bewegt. In xwonder kann so die Position des Targets als virtuelle Tonquelle dargestellt werden. Trägt ein Zuhörer das Target bei sich, kann gleichzeitig das Zuhörer-basierte Rendering sowie das Target-Source-Matching aktiviert werden. Die Position des Zuhörers wird dann in Form einer virtuellen Tonquelle in

¹<http://www.illposed.com/software/javaosc.html>

xwonder angezeigt. Da die virtuelle Tonquelle, welche sich an der Zuhörerposition befindet, nur als visuelles Feedback dient, wird dieser keine Audiospur zugeordnet.

5.4. Ergebnisdarstellung

Das Ergebnis, welches sich aus der Durchführung aller beschriebenen Schritte ergibt, ist ein zuhörerbasiertes Rendering in der Software sWONDER. Zur Umsetzung werden zwei verschiedene Systeme benötigt, die WFS-Anlage und das ARTTRACK System. Die Position des Zuhörers wird vom Trackingsystem erfasst und an das Netzwerk gesendet. Das Programm MoWeC greift die Information ab und sendet sie in Form einer OSC-Nachricht an cwonder. cwonder erkennt den Inhalt der Nachricht und leitet diese an twonder weiter. In dieser Komponente wird die Zuhörerposition aktualisiert. Das neue Rendering erfolgt in Abhängigkeit der aktuellen Zuhörerposition und aktiviert die notwendigen Lautsprecher. Eine Übersicht des Ablaufs innerhalb der Systeme ist in Abbildung 5.3 dargestellt.

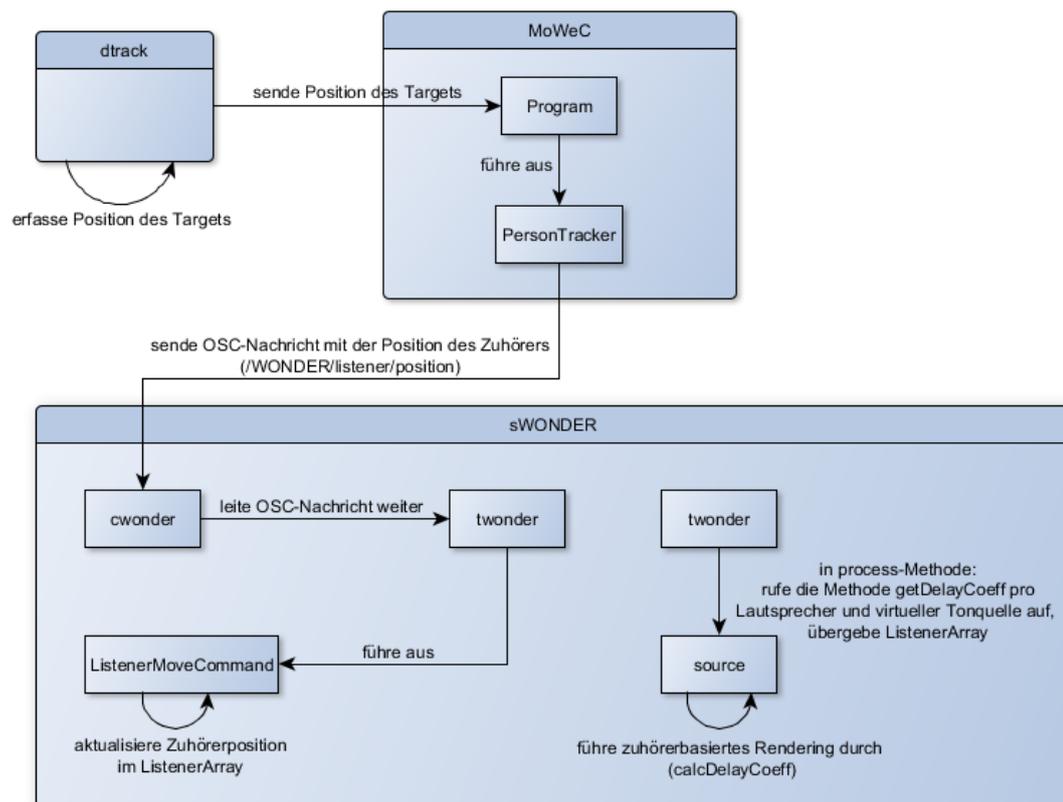


Abbildung 5.3.: Diagramm zum Ablauf des Zuhörerbasierten Renderings

6. Testen des Zuhörerbasierten Renderings

In diesem Kapitel werden mehrere Aspekte des entwickelten Rendering-Verfahrens beleuchtet. Dazu wird die Durchführung und Auswertung zweier Tests beschrieben. Ein Test besteht aus der Messung der Lautstärke an verschiedenen Positionen um eine fokussierte Tonquelle. Ein weiterer Test ist ein Hörtest mit mehreren Probanden, welcher durchgeführt wurde, um die Lokalisationsfähigkeit fokussierter Tonquellen durch das Gehör zu ermitteln.

6.1. Messung der Lautstärke um fokussierte Tonquellen

Das neue Rendering-Verfahren bewirkt, dass nicht mehr die der fokussierten Tonquelle am nächsten, sondern je nach Zuhörerposition unterschiedliche Lautsprecher aktiv sind. Um etwaige Schwankungen in der Lautstärke um eine fokussierte Tonquelle herum zu ermitteln, wird eine Lautstärkemessung durchgeführt. Dazu wird ein Mikrofon an acht verschiedenen Stellen um die fokussierte Tonquelle herum platziert. Ein Target wird am Mikrofonstativ befestigt, welches normalerweise für die Ermittlung der Zuhörerposition verwendet wird. So kann das Zuhörerbasierte Rendering verwendet werden und das Mikrofon nimmt den Ton in der Lautstärke auf, die der Zuhörer an derselben Position auch wahrnehmen würde. An jeder Position werden die ersten Sekunden eines Musikstücks, welches auf der fokussierten Tonquelle abgespielt wird, aufgenommen. Die aufgenommenen Tracks werden in dem Programm Cubase gespeichert. An jeder der acht Stellen wird das Mikrofon im Abstand von 80 cm, 100 cm und 120 cm zur fokussierten Tonquelle aufgestellt, somit wird der Track dreimal pro Stelle aufgenommen.

Die Messung wird für zwei unterschiedlichen Positionen der fokussierten Tonquelle durchgeführt. Die erste Position der fokussierten Tonquelle ist der Mittelpunkt der Fläche innerhalb der Lautsprecher, die zweite Position liegt näher an einer Ecke des Rechtecks. In Abbildung 6.1 sind die Positionen der fokussierten Tonquelle und die Positionen des Mikrofons dargestellt. Die unterschiedlichen Abstände des Mikrofons zur fokussierten Tonquelle sind in dem Bild nicht abgebildet.

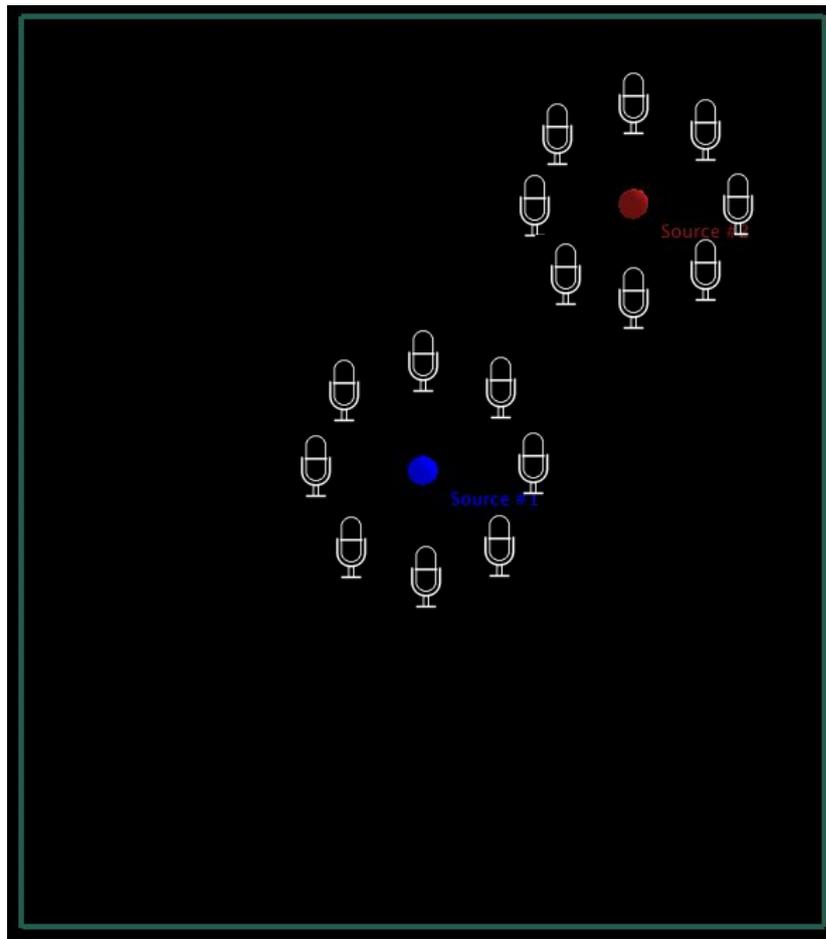


Abbildung 6.1.: Positionen der fokussierten Tonquelle und des Mikrofons bei der Lautstärkemessung

Das Musikstück beginnt mit einem lauten Geräusch. Die Lautstärke dieses Geräuschs in dB wurde in Cubase für jeden einzelnen Track entnommen, wobei der Lautstärkereglер auf 0,0 dB eingestellt war. Abbildung 6.2 zeigt die Auswertung der Lautstärkemessung um die fokussierte Tonquelle in der Mitte des Raumes, Abbildung 6.3 zeigt die Auswertung um die fokussierte Tonquelle, welche näher in der Ecke des Rechtecks lag.

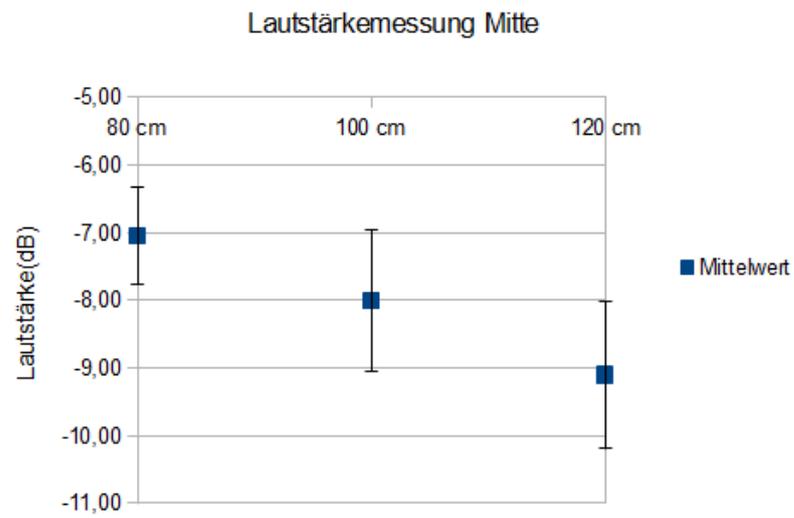


Abbildung 6.2.: Auswertung der Lautstärkemessung (Mitte)

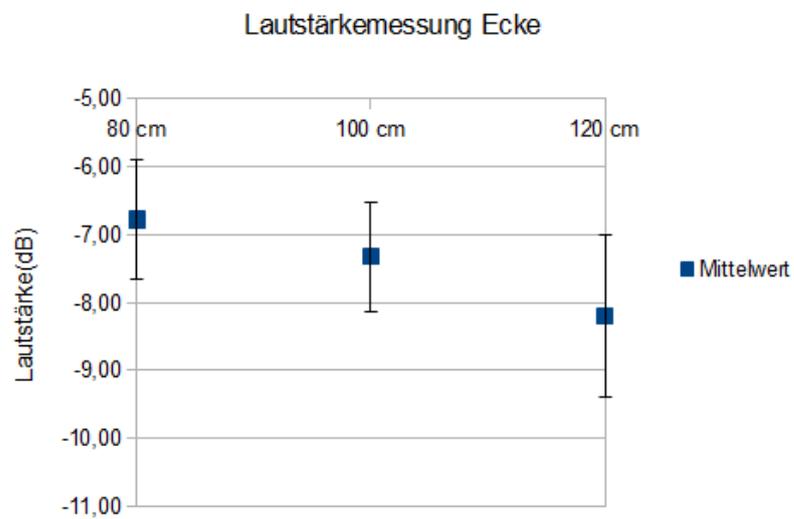


Abbildung 6.3.: Auswertung der Lautstärkemessung (Ecke)

Die Diagramme zeigen die Mittelwerte und die Standardabweichungen der gemessenen Lautstärke um die fokussierte Tonquelle. Die x-Achse zeigt die verschiedenen Abstände, die das Mikrofon zu der fokussierten Tonquelle hatte.

Die Mittelwerte in beiden Diagrammen zeigen, dass es in der Nähe der fokussierten Tonquelle lauter wurde und je weiter entfernt das Mikrofon aufgestellt wurde, desto leiser wurde es. Zu den Messungen, welche an den verschiedenen Stellen um die fokussierte Tonquelle im gleichen Abstand durchgeführt wurden, kann keine konkrete Aussage getroffen werden. Die Standardabweichungen zeigen, dass es Schwankungen in der Lautstärke um die fokussierte Tonquelle gab. Es lässt sich jedoch kein großer Unterschied zwischen den beiden Positionen der fokussierten Tonquelle sehen. Die Standardabweichung bei 100 cm ist beispielsweise im ersten Diagramm größer, bei 120 cm ist sie im zweiten Diagramm größer.

6.2. Hörtest zur Lokalisierbarkeit fokussierter Tonquellen

Die modifizierte Software soll eine Lokalisierbarkeit von fokussierten Tonquellen ermöglichen. Dies wurde in einem Hörtest mit 21 teilnehmenden Personen getestet. Bei dem Test geht es um eine Lokalisation von fokussierten Tonquellen durch das Gehör, weshalb die Testpersonen keine visuelle Unterstützung durch xwonder erhalten. Der Test besteht aus zwei Teilen, in denen unterschiedliche Anforderungen an die Testpersonen gestellt werden. Im ersten Teil muss die Testperson den Kopf in Richtung der fokussierten Tonquelle drehen und im zweiten Teil muss sie sich auf die Position der fokussierten Tonquelle stellen. Ein Distanztest wird dabei nicht vorgenommen, da es bereits Artikel gibt, welche bestätigen, dass fokussierte Tonquellen alleine durch das Gehör nicht von virtuellen Tonquellen hinter den Lautsprechern unterschieden werden können, sofern sich der Zuhörer nicht durch den Raum bewegen darf [K⁺04]. Jede Person setzt ein Target in Form einer Brille auf, wodurch das zuhörerbasierte Rendering ermöglicht wird. Die vom Trackingsystem erfassten Daten zur Positions- und Drehrichtung des Targets werden auf einem Rechner angezeigt und erfasst. Sind die Tests abgeschlossen, werden die Ergebnisse ausgewertet.

6.2.1. Versuchsaufbau

Im ersten Teil des Hörtests stellt sich die Testperson auf den Mittelpunkt der Fläche, welche von den Lautsprechern eingeschlossen wird. Eine fokussierte Tonquelle wird durch das Versenden einer OSC-Nachricht an eine bestimmte Stelle platziert und die Musik wird gestartet. Die Testperson wird aufgefordert, den Kopf in Richtung der fokussierten Tonquelle zu drehen. Hat sich die Testperson für eine Richtung entschieden, muss sie diese Kopfstellung beibehalten und

6. Testen des zuhörerbasierten Renderings

die Daten des Trackingsystems werden per Screenshot gespeichert. Danach wird die fokussierte Tonquelle auf eine neue Position gesetzt und das Verfahren wird wiederholt. Die Positionen der fokussierten Tonquelle wurden so gewählt, dass sie sich in der Nähe der linken oberen Ecke (in der grafischen Darstellung von xwonder) befinden. Insgesamt wird die fokussierte Tonquelle an drei verschiedene Stellen platziert. Als nächstes wechselt die Testperson ihre Position, sie stellt sich jetzt in den linken oberen Bereich. Von dieser Position aus muss die Testperson wieder den Kopf in die Richtung der fokussierten Tonquelle bewegen. Die fokussierte Tonquelle wird an dieselben drei Positionen gesetzt wie im ersten Durchlauf. Die Abbildung 6.4 zeigt die beiden Positionen der Testperson im ersten und zweiten Durchlauf und die Positionen der fokussierten Tonquelle in xwonder. Das Bild dient nur der Darstellung des Testszenarios und entspricht nicht dem Testszenario selbst, da sich die Testperson sowie die fokussierte Tonquelle immer nur auf einer der Positionen zur Zeit befand. Insgesamt ergeben sich 6 Konstellationen von Testperson und fokussierter Tonquelle. Source #1-#3 zeigen die Positionen der fokussierten Tonquelle in beiden Durchläufen.

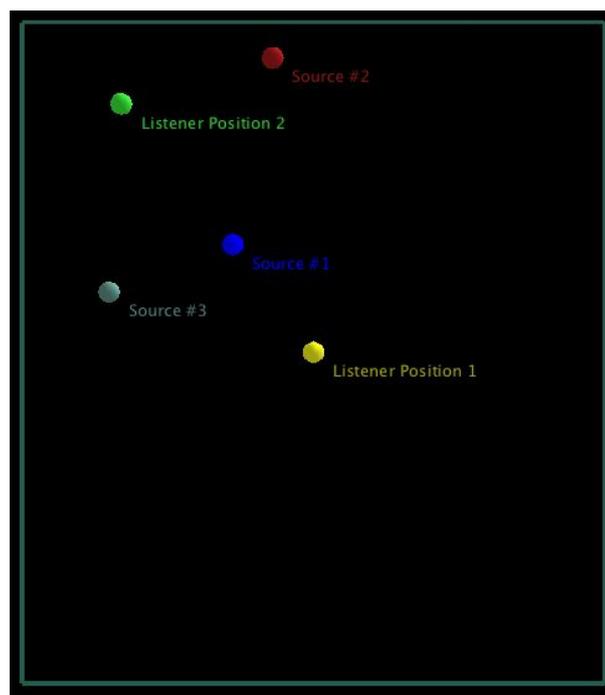


Abbildung 6.4.: Positionen der fokussierten Tonquelle im Hörtest (1. Teil)

Im zweiten Teil des Tests wird die Testperson aufgefordert, die fokussierte Tonquelle zu finden und sich auf deren Position zu stellen. Die Testperson wird zunächst aufgefordert, sich

außerhalb der von den Lautsprechern eingeschlossenen Fläche aufzustellen. Die fokussierte Tonquelle wird an eine Stelle innerhalb der Lautsprecher platziert. Die Testperson bewegt sich nun auf diese zu und stellt sich möglichst genau auf die Position der fokussierten Tonquelle. Hat sich die Testperson für eine Position entschieden, bleibt sie an dieser Stelle stehen und die Daten des Trackingsystems werden per Screenshot gespeichert. Anschließend wird die fokussierte Tonquelle auf eine andere Position gesetzt und die Testperson muss die neue Position ausgehend von der alten Position finden. Insgesamt wird die fokussierte Tonquelle an drei verschiedene Stellen platziert. Die Abbildung 6.5 zeigt die verschiedenen Positionen der fokussierten Tonquelle.

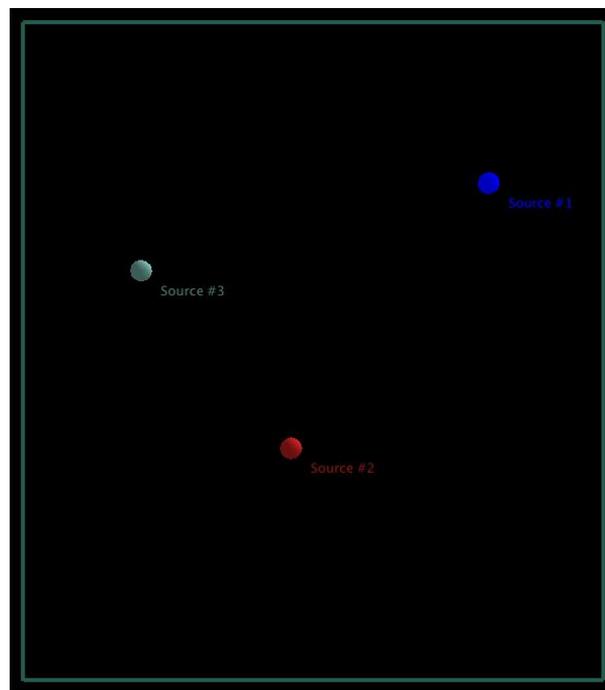


Abbildung 6.5.: Positionen der fokussierten Tonquelle im Hörtest (2. Teil)

6.2.2. Konvertieren der Daten in das Koordinatensystem der WFS-Anlage

Bevor die Ergebnisse ausgewertet werden können, müssen die Daten des Trackingsystems in das Koordinatensystem der WFS-Anlage konvertiert werden. Sie lassen sich so leichter mit den Positionsdaten der fokussierten Tonquelle, welche im Hörtest verwendet wurde, vergleichen. Für die Umwandlung der Daten wird, wie bereits für das zuhörerbasierte Rendering, die Software MoWeC verwendet [Nog12]. Sie stellt bereits die notwendigen Funktionen für eine

Konvertierung zur Verfügung. Auf dieser Basis kann sie für den Hörtest um ein paar Klassen erweitert werden, welche die bereits implementierten Funktionen nutzen. Ein Objekt der Klasse ConverterBox wird zusammengestellt, welches mehrere Aktionen auf den Daten ausübt (an den Achsen verschieben, spiegeln und rotieren) und sie so in die Form des neuen Koordinatensystems bringt. Nur Objekte der Klasse Source können in einer ConverterBox konvertiert werden, weshalb eine Methode implementiert wird, die aus den Daten des Hörtests Sources erstellt. Dabei werden nicht alle Daten, welche in den Screenshots zu sehen sind, verwendet. Aus den Daten, welche im ersten Teil des Hörtests gespeichert wurden, werden nur die x- und y-Koordinate und die Rotation um die z-Achse ausgewählt, da hier die Blickrichtung der Personen erfasst wurde. Von den Daten des zweiten Teils werden nur die x- und y-Koordinaten verwendet, da sich die Personen hier auf die Positionen der fokussierten Tonquelle stellen mussten und die Blickrichtung dabei nicht von Relevanz ist. Aus diesen Daten werden Source-Objekte erstellt, welche mit der ConverterBox umgewandelt werden. Das Ergebnis sind wieder Source-Objekte, welche die konvertierten Daten enthalten. Diese können nun mit den Positionsdaten der fokussierten Tonquelle aus dem Hörtest verglichen werden.

6.2.3. Auswertung der Ergebnisse

Für die Auswertung des ersten Teils werden zunächst die Soll-Winkel berechnet. Ein Soll-Winkel ist der Winkel zwischen der Testperson und der fokussierten Tonquelle. Er entspricht dem gewünschten Wert, welcher erreicht wird, wenn die Person exakt auf die Position der fokussierten Tonquelle schaut. Die Position einer Testperson war stets vorgegeben (erst Mittelpunkt, dann linker oberer Bereich), es treten jedoch Schwankungen der Positionen im Zentimeterbereich auf. Bei Drehung des Kopfes in Richtung der fokussierten Tonquelle kann sich beispielsweise die Position des Kopfes ändern. Zusätzlich haben die Testpersonen zwar ungefähr, jedoch nicht exakt an derselben Stelle gestanden. Deshalb wird kein fester Wert als Soll-Winkel verwendet, sondern pro Screenshot, also pro Konstellation von Testperson und fokussierter Tonquelle, wird dieser neu berechnet. Dazu werden die Positionsdaten (x- und y-Koordinate) der Testperson und der fokussierten Tonquelle verwendet. Mit Hilfe der Java-Methode `Math.atan2(double y, double x)` kann der Winkel zwischen den beiden Punkten berechnet werden. Nach Berechnung des Soll-Winkels wird dieser vom tatsächlichen Winkel abgezogen. Der tatsächliche Winkel ist die z-Rotation des Targets, welche vom Trackingsystem erfasst wurde und auf dem Screenshot zu sehen ist. Die Differenz zwischen dem tatsächlichen Winkel und dem Soll-Winkel entspricht dem Lokalisationsfehler. Der Lokalisationsfehler gibt an, um wie viel Grad der Soll-Winkel verfehlt wurde. Aus allen Differenzwerten wird ein

Diagramm erstellt, welches in Abbildung 6.6 dargestellt ist. Der Lokalisationsfehler ist in Grad angegeben.

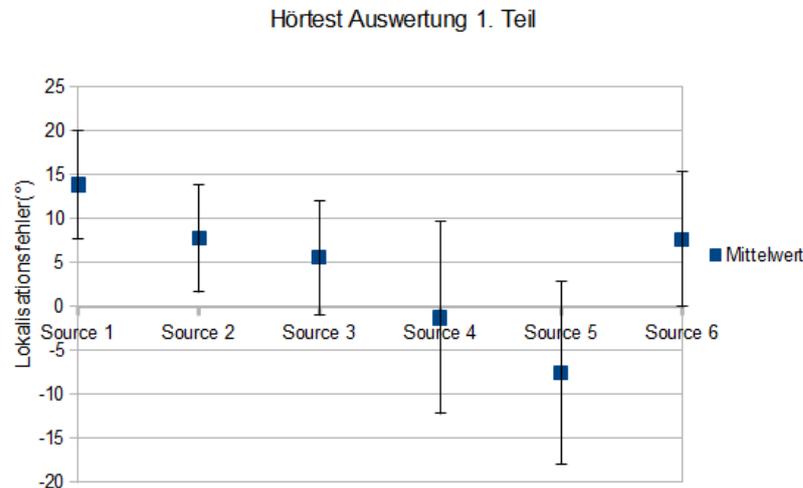


Abbildung 6.6.: Auswertung des Hörtests (1. Teil)

Die x-Achse zeigt die sechs verschiedenen Positionen der fokussierten Tonquelle. In dem Diagramm wird der Mittelwert, welcher sich aus den Lokalisationsfehlern aller Testpersonen ergibt, angezeigt. Die vertikalen Linien (Fehlerindikatoren) zeigen die Standardabweichungen der Werte. In dem Hörtest waren die ersten drei Positionen der fokussierten Tonquelle (Source 1-3) identisch zu den nachfolgenden drei Positionen (Source 4-6). Die Position der Testperson hat sich im zweiten Durchlauf jedoch geändert. Bei Source 1-3 stand die Testperson im Mittelpunkt der Fläche, bei Source 4-6 im linken oberen Bereich der Fläche.

Die Mittelwerte zeigen, um wie viel Grad die fokussierte Tonquelle im Durchschnitt verfehlt wurden. Source 1, 2 und 3 wurden im Durchschnitt in positiver Drehrichtung, Source 4 und 5 in negativer Drehrichtung verfehlt. Source 6 wurde im Durchschnitt wieder in positiver Drehrichtung verfehlt und fällt aus dem Schema heraus. Da die Blickrichtung der Testpersonen bei Source 1, 2 und 3 entgegengesetzt zu der Blickrichtung bei Source 4, 5 und 6 war, kann also behauptet werden, dass die Testpersonen die fokussierte Tonquelle an einer versetzten Position wahrgenommen haben. Diese Behauptung wird an Source 2 und 5 erläutert, da sie hier am ehesten zutrifft. Source 2 wurde im Durchschnitt um ca. 7 Grad und Source 5 um ca. -7 Grad verfehlt. Würde bei Source 2 eine Linie vom Mittelpunkt aus gezogen werden, welche die fokussierte Tonquelle um 7 Grad verfehlt, und bei Source 5 eine Linie von der linken oberen Ecke aus, welche die fokussierte Tonquelle um -7 Grad verfehlt, würden sich diese

beiden Linien in einem Punkt schneiden. Dieser Punkt wurde im Durchschnitt als Position der fokussierten Tonquelle interpretiert, obwohl sie sich in Wirklichkeit etwas versetzt dazu befand.

Ein Teil der Testpersonen empfand das Lokalisieren von der Ecke heraus schwieriger, was sich auch in der Auswertung des Hörtests zeigt. Die Standardabweichungen von Source 4 und 5 sind am größten, das heißt die Streuung der Lokalisationsfehler ist hier am größten.

Zur Auswertung des zweiten Teils werden nur die Positionsdaten von Testperson und fokussierter Tonquelle benötigt. Da sich die Testperson in diesem Teil des Tests auf die Position der fokussierten Tonquelle stellen sollte, ist die Soll-Position die Position der fokussierte Tonquelle. Die Position der Testperson, welche durch einen Screenshot festgehalten wurde, und die Position der fokussierten Tonquelle müssen verglichen werden. Dazu wird die Differenz der x-Koordinaten beider Positionen sowie die Differenz der y-Koordinaten beider Positionen berechnet. Die Differenzen entsprechen den Lokalisationsfehlern. Aus den Werten ergibt sich wieder ein Diagramm, welches in Abbildung 6.7 dargestellt ist.

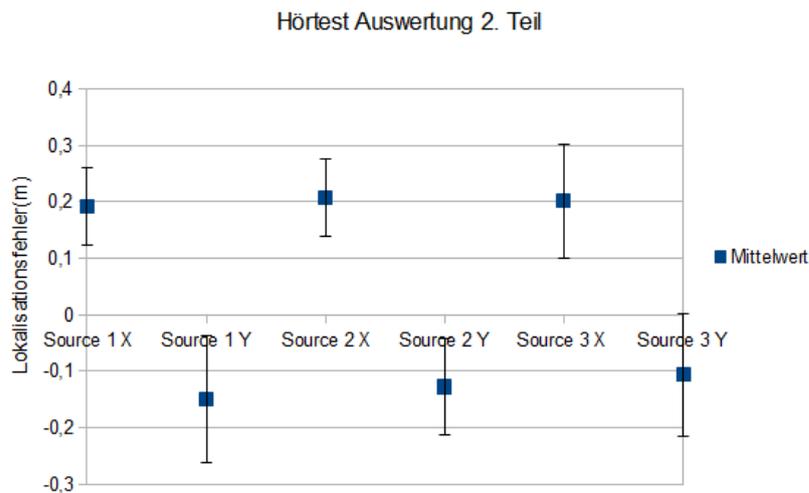


Abbildung 6.7.: Auswertung des Hörtests (2. Teil)

In diesem Teil des Hörtests wurde die fokussierte Tonquelle an drei verschiedene Stellen platziert. Pro Position der fokussierten Tonquelle sind zwei Mittelwerte im Diagramm abgebildet. Source 1 X zeigt beispielsweise den durchschnittlichen Lokalisationsfehler entlang der x-Achse, Source 1 Y den durchschnittlichen Lokalisationsfehler entlang der y-Achse; beide Mittelwerte beziehen sich jedoch auf die erste Position der fokussierten Tonquelle. Der Lokalisationsfehler ist in Metern angegeben. Die Fehlerindikatoren zeigen die Standardabweichungen.

Die Lokalisationsfehler entlang der x-Achse unterscheiden sich nur geringfügig voneinander, ebenso die Lokalisationsfehler entlang der y-Achse. Daraus lässt sich schließen, dass alle drei Positionen der fokussierten Tonquelle gleich gut getroffen wurden. Auffällig ist jedoch, dass die Testpersonen stets eine zur Originalposition versetzte Position gewählt haben. Im Durchschnitt wurde eine um 20 cm in x-Richtung, sowie um -10 bis -15 cm in y-Richtung verschobene Position als exakte Position der fokussierten Tonquelle wahrgenommen. Da dies bei allen drei Positionen der fokussierten Tonquelle der Fall ist, könnte eine ungenaue Kalibrierung des Brillentargets eine mögliche Ursache dafür sein.

Bei diesem Hörtest haben sowohl Testpersonen mitgemacht, die die Beschallung durch die WFS-Anlage bereits kannten, als auch Testpersonen, die die Anlage noch nicht kannten. Auch das Prinzip der Wellenfeldsynthese, virtuelle Tonquellen zu bilden, wurde vor dem Test nicht erklärt. Dennoch sind die Ergebnisse bei allen Teilnehmern gut. Die Positionen der fokussierten Tonquelle wurden in beiden Teilen des Tests bis auf geringfügige Abweichungen stets gefunden. Es gab keine Testperson, die die Positionen gar nicht gefunden hat.

7. Schluss

Der praktische Teil der Arbeit wird in diesem Kapitel zusammengefasst und ein Fazit gezogen. Anschließend wird ein Ausblick auf mögliche Erweiterungen gegeben.

7.1. Zusammenfassung und Fazit

Die Problemstellung, welche sich beim Rendern fokussierter Tonquellen in der Wellenfeldsynthese ergibt, wurde zu Beginn der Arbeit erläutert. In Anbetracht dieses Problems wurde im Labor der HAW Hamburg eine Lösung in Form einer Modifikation der verwendeten Software entwickelt. Nachdem die Vorgehensweise dargestellt und die Software genauer beschrieben wurde, wurde die schrittweise Implementierung erläutert.

Eine Herausforderung bestand darin, einen Überblick über die Software zu gewinnen und die Stellen zu finden, die erweitert oder modifiziert werden mussten. Da das Starten der Software nur in Verbindung mit einer WFS-Anlage Sinn ergibt, konnten die einzelnen Implementierungsschritte nur im Labor getestet werden. Mehrere Schritte und Konsolenbefehle sind nötig, um eine neue Version auf dem Rechner im Labor zu kompilieren und zu installieren, was das Vorankommen erschwert hat. Im Rahmen der Arbeit wurden Festplattenabbilder der Systemumgebung im Labor (des WFS-Servers und der beiden Nodes) erstellt, welche das Kompilieren der sWONDER Software auf dem eigenen Rechner möglich machen. Diese erleichtern den Prozess der Entwicklung, können jedoch nicht das reale System mit der WFS-Anlage ersetzen. Die Qualität der entwickelten Lösung wurde mit Hilfe von zwei Tests, einer Lautstärkemessung und einem Hörtest, überprüft. Die Ergebnisse der Lautstärkemessung waren nicht eindeutig, zeigten jedoch, dass es Schwankungen in der Lautstärke gibt. Der Hörtest hat bestätigt, dass eine Lokalisierbarkeit von fokussierten Tonquellen mit dem neuen Rendering-Verfahren möglich ist.

7.2. Ausblick

Bei der Entwicklung des neuen Verfahrens wurde eine mögliche Erweiterung auf mehrere Zuhörer berücksichtigt. Die Beschränkung auf einen Zuhörer ist von Nachteil, wenn eine

akustische Umgebung mit fokussierten Tonquellen für eine Gruppe von Personen erzeugt werden soll. Erfasst das Trackingsystem die Position einer der Personen aus der Gruppe, dürfen sich die anderen Personen für eine korrekte Lokalisation der fokussierten Tonquellen nicht weit von dieser entfernen. Ein Rendering für mehrere Personen ist zwar nicht uneingeschränkt möglich, eine Umsetzung könnte jedoch darin bestehen, den bestmöglichen Kompromiss zu finden.

Eine Einbeziehung der Konfigurationsdateien kann die Bedienung der modifizierten Software für den Benutzer leichter machen. Gibt es eine Erweiterung auf mehrere Zuhörer, könnte in einer Konfigurationsdatei beispielsweise angegeben werden, wie viele Zuhörer es insgesamt gibt.

Des Weiteren kann eine auf das neue Rendering angepasste visuelle Darstellung hilfreich sein. Dazu kann die Oberfläche von *xwonder* erweitert oder eine neue grafische Darstellung entwickelt werden. In dieser Arbeit wurde eine virtuelle Tonquelle als Hilfsmittel zur Darstellung der Zuhörerposition verwendet, um ein visuelles Feedback zu erhalten. Eine Darstellung der Zuhörerposition durch ein neues Symbol, welches sich von den virtuellen Tonquellen unterscheidet, würde zu einer besseren Übersicht beitragen, vor allem bei einer Erweiterung auf mehrere Zuhörer.

Bei der Durchführung des Lokalisationstests wurde von mehreren Personen ein Springen der Musik zwischen gegenüberliegenden Lautsprechern bemerkt, wenn sie versucht haben, in dem Punkt der fokussierten Tonquelle stehen zu bleiben. Das Springen zwischen den Lautsprechern liegt an dem implementierten Algorithmus, sollte jedoch nicht vom Zuhörer selbst wahrgenommen werden. Um dies zu vermeiden, könnte eine Lösung gefunden werden, bei der eine langsame Überblendung auf die gegenüberliegenden Lautsprecher erreicht wird.

Glossar

ARTTRACK System Das Trackingsystem der Firma Advanced Realtime Tracking GmbH. 17, 21, 25, 34, 48, 49

ConverterBox Eine Komponente der Software MoWeC zur Übertragung von Daten von einem Koordinatensystem in ein anderes. 42

cwonder Die Kontrolleinheit der Software sWONDER. 12, 18, 19, 24–27, 29–34

dtrack Die Software des ARTTRACK Systems. 21, 22, 50

Elementarwelle Mehrere Elementarwellen synthetisieren eine Wellenfront. 5, 6, 12, 16, 20, 26, 28, 49

Gestensteuerung Steuerung eines Systems durch Gesten. 3, 10

Linearquelle Eine virtuelle Tonquelle, deren Wellenfront eine Gerade bildet. 15, 26, 27, 32, 50

Listener Repräsentation des Zuhörers in der Software sWONDER. 31, 32

ListenerArray Repräsentation mehrerer Zuhörer in der Software sWONDER. 31, 32

Marker Ein Objekt, dessen Position vom ARTTRACK System erkannt wird. 21, 49

MoWeC Eine Software, die die Schnittstelle zwischen WFS-Anlage und ARTTRACK System bildet. 25, 32–34, 41, 48, 49

Open Sound Control Ein Protokoll, welches in Sound- und Multimedia-Installationen zur Kommunikation über das Netzwerk verwendet wird. 14, 48

OSC-Nachricht Eine Nachricht, die auf dem Open Sound Control Protokoll basiert. 14, 15, 18, 24–27, 29–34, 39, 49

- PersonTracker** Komponente der Software MoWeC zur Übertragung der Zuhörerposition an sWONDER. 33
- Punktquelle** Eine virtuelle Tonquelle, deren Wellenfront gekrümmt ist. 15, 26, 27, 32
- Render Stream** Datenstrom innerhalb der Software sWONDER zur Übertragung von OSC-Nachrichten, welche Informationen für eine visuelle Darstellung enthalten. 26, 30
- Rendering** Das Erzeugen von Elementarwellen durch mehrere Lautsprecher, um Wellenfronten zu synthetisieren. 16, 18–20, 23–29, 31–36, 39, 41, 46, 47, 49, 50
- Rendering-Verfahren** Algorithmus zur Auswahl der aktiven Lautsprecher beim Rendering nach bestimmten Kriterien. 3, 23, 24, 27, 28
- sWONDER** Eine Software zur Realisierung der Wellenfeldsynthese. 2–4, 12, 14, 15, 17–19, 23, 25, 29, 32–34, 46, 48–50
- Target** Ein Objekt, dessen Position und Ausrichtung vom ARTTRACK System erkannt wird und aus mehreren Markern besteht. 21, 22, 25, 32, 33, 36, 39, 42
- Tonquelle** Eine virtuelle Tonquelle, wie sie bei der Wellenfeldsynthese erzeugt wird. 1–16, 18–21, 23–29, 32–34, 36, 37, 39–50, 54
- Trackingsystem** Ein System zur Erfassung der Position, Ausrichtung oder Bewegung von Objekten im dreidimensionalen Raum. 2, 4, 10, 11, 17, 21, 25, 29, 32–34, 39–42, 47, 48
- twonder** Die Rendering Einheit der Software sWONDER. 12, 14, 16, 18, 19, 24–32, 34
- Visual Stream** Datenstrom innerhalb der Software sWONDER zur Übertragung von OSC-Nachrichten, welche Informationen für das Rendering enthalten. 26, 30
- Wellenfeldsynthese** Ein räumliches Audiowiedergabeverfahren. 1–12, 45, 46, 49, 50
- WFS-Anlage** Eine Anlage zur Umsetzung der Wellenfeldsynthese. 1, 3–5, 8, 9, 17, 25, 31–34, 41, 45, 46, 48, 54
- xwonder** Die grafische Oberfläche und Benutzerschnittstelle von sWONDER. 13–15, 18, 19, 26, 30, 33, 34, 39, 40, 47, 50, 54

Abbildungsverzeichnis

2.1.	Prinzip der Wellenfeldsynthese - Synthese einer Wellenfront	6
2.2.	Wellenfronten einer fokussierten Tonquelle	7
3.1.	Die grafische Oberfläche der Komponente xwonder	13
3.2.	Wellenfronten einer Punkt- und Linearquelle	15
4.1.	3D-Modell der Systeme im Labor [Nog12]	18
4.2.	Verteilung der Komponenten von sWONDER im Labor [Fou]	19
4.3.	Visualisierung des Wertes „focus limit“	20
4.4.	Position des Zuhörers für das korrekte und nicht korrekte Hören einer fokussierten Tonquelle im Labor	21
4.5.	Grafische Oberfläche der Software dtrack	22
5.1.	Darstellung der aktiven Lautsprecher für unterschiedliche Zuhörerpositionen bei dem neuen Rendering-Verfahren	29
5.2.	Das Brillentarget	33
5.3.	Diagramm zum Ablauf des Zuhörerbasierten Renderings	35
6.1.	Positionen der fokussierten Tonquelle und des Mikrofons bei der Lautstärkemessung	37
6.2.	Auswertung der Lautstärkemessung (Mitte)	38
6.3.	Auswertung der Lautstärkemessung (Ecke)	38
6.4.	Positionen der fokussierten Tonquelle im Hörtest (1. Teil)	40
6.5.	Positionen der fokussierten Tonquelle im Hörtest (2. Teil)	41
6.6.	Auswertung des Hörtests (1. Teil)	43
6.7.	Auswertung des Hörtests (2. Teil)	44

Literaturverzeichnis

- [A⁺13] ANDRÉ, Cédric u. a.: A New Validated Method for Improving the Audiovisual Spatial Congruence in the Case of Stereoscopic-3D Video and Wave Field Synthesis. In: *International Conference on 3D Imaging (IC3D)*, 2013
- [Adv14] ADVANCED REALTIME TRACKING GMBH: *Technology - ART Advanced Realtime Tracking*. <http://www.ar-tracking.com/technology/>. Version: 2014. – Zugriff: 09.05.2014
- [AS08] AHRENS, Jens ; SPORS, Sascha: Notes on rendering focused directional virtual sound sources in wave field synthesis. In: *Tagung der Deutschen Akustischen Gesellschaft*. Dresden, 2008
- [Baa04] BAALMAN, M. A. J.: Application of Wave Field Synthesis in electronic music and sound installations. In: *Proceedings of the 2nd International Linux Audio Conference*. Karlsruhe, 2004
- [Baa08] BAALMAN, M. A. J.: *On Wave Field Synthesis and Electro-Acoustic Music*, Diss., 2008
- [BP04] BAALMAN, M. A. J. ; PLEWE, Daniel: WONDER - a software interface for the application of Wave Field Synthesis in electronic music and interactive sound installations. In: *Proceedings of the International Computer Music Conference (ICMC)*. Miami, USA, 2004
- [DBB03] DE BRUJIN, Werner P. J. ; BOONE, Marinus M.: Application of Wave Field Synthesis in life-size videoconferencing. In: *Audio Engineering Society Convention 114*, 2003
- [DV⁺08] DE VRIES, Diemer u. a.: User-Dependent Optimization of Wave Field Synthesis Reproduction for Directive Sound Fields. In: *Audio Engineering Society Convention 124*, 2008
- [FN13] FOHL, Wolfgang ; NOGALSKI, Malte: A gesture control interface for a Wave Field Synthesis system. In: *International Conference on New Interfaces for Musical Expression*, 2013

- [Fou] FOUR AUDIO GMBH: *Kurze Einführung: Wellenfeldsynthese-Software und -Cluster*
- [Fra08] FRAIETTA, Angelo: Open Sound Control: Constraints and Limitations. In: *8th International Conference on New Interfaces for Musical Expression NIME08*, 2008, S. 19
- [GS12] GEIER, Matthias ; SPORS, Sascha: *Spatial Audio with the SoundScape Renderer*, 2012
- [Jan97] JANSEN, G.: *Focused wavefields and moving virtual sources by wavefield synthesis*, Technical University of Delft, Diplomarbeit, 1997
- [K⁺04] KERBER, Stefan u. a.: Experimental investigations into the distance perception of nearby sound sources: Real vs. WFS virtual nearby sources. In: *Proceedings of the 7th Congrès Français d'Acoustique/30th Deutsche Jahrestagung für Akustik (CFA/DAGA 04)*, 2004, S. 1041–1042
- [L⁺10a] LESLIE, Grace u. a.: Grainstick: A collaborative, interactive sound installation. In: *Proceedings of the International Computer Music Conference (ICMC)*, 2010
- [L⁺10b] LOPEZ, Jose J. u. a.: Wave Field Synthesis for next generation videoconferencing. In: *Proceedings of the 4th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, IEEE, 2010, S. 1–4
- [M⁺14] MÜLLER, Joerg u. a.: *The BoomRoom: Mid-air Direct Interaction with Virtual Sound Sources*, 2014
- [MLDV05] MELCHIOR, Frank ; LAUBACH, Tobias ; DE VRIES, Diemer: Authoring and user interaction for the production of wave field synthesis content in an augmented reality system. In: *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality* Computer Society, IEEE, 2005, S. 48–51
- [N⁺08] NAOE, Munenori u. a.: Performance evaluation of 3d sound field reproduction system using a few loudspeakers and wave field synthesis. In: *Second International Symposium on Universal Communication (ISUC)*, IEEE, 2008, S. 36–41
- [Nog12] NOGALSKI, Malte: *Gestengesteuerte Positionierung von Klangquellen einer Wellenfeldsynthese-Anlage mit Hilfe eines kamerabasierten 3D-Tracking-Systems*, Diplomarbeit, 2012

- [Spo04] SPORER, Thomas: Wave field synthesis - Generation and reproduction of natural sound environments. In: *7th International Conference on Digital Audio Effects (DAFx-04)*, 2004
- [TW04] THEILE, Günther ; WITTEK, Helmut: Wave field synthesis: A promising spatial audio rendering concept. In: *Acoustical science and technology* 25 (2004), Nr. 6, S. 393–399
- [WFM03] WRIGHT, Matthew ; FREED, Adrian ; MOMENI, Ali: Open sound control: State of the art 2003. In: *Proceedings of the 2003 conference on New interfaces for musical expression*, 2003, S. 153–160
- [WT06] WEINZIERL, Stefan ; TAZELAAR, Kees: Raumsimulation und Klangkunst. In: *Sonambiente Berlin*, 2006, S. 350–365

A. Informationen zu den Tests des zuhörerbasierten Renderings

A.1. Einzeldaten der Lautstärkemessung

Die nachfolgenden Tabellen zeigen die ermittelten Werte bei der Lautstärkemessung um fokussierte Tonquellen (Kapitel 6.1) sowie die daraus errechneten Mittelwerte und Standardabweichungen. Die Zahlen entsprechen der gemessenen Lautstärke in dezibel. Die Ausrichtung des Mikrofons zu der Position der virtuellen Tonquelle ist durch Himmelsrichtungen¹ beschrieben. Eine grafische Darstellung der Mikrofonpositionen ist in Abbildung 6.1 zu sehen.

<i>Tonquellenposition: Mitte</i>	80 cm	100 cm	120 cm
N	-6,6	-7,0	-7,9
NO	-7,2	-7,5	-7,3
O	-7,4	-8,8	-9,5
SO	-7,8	-8,7	-10,5
S	-7,3	-8,1	-8,9
SW	-6,0	-6,4	-8,8
W	-8,0	-9,9	-10,7
NW	-6,0	-7,6	-9,2
Mittelwert	-7,04	-8,00	-9,10
Standardabw.	0,71	1,04	1,09

¹Die Angabe der Himmelsrichtung bezieht sich auf die grafische Darstellung der WFS-Anlage in xwonder und entspricht nicht der tatsächlichen Himmelsrichtung.

A. Informationen zu den Tests des zuhörerbasierten Renderings

Tonquellenposition: Ecke	80 cm	100 cm	120 cm
N	-7,5	-7,0	-9,1
NO	-7,4	-7,8	-6,9
O	-7,4	-7,3	-8,1
SO	-7,7	-9,0	-10,7
S	-6,9	-7,1	-7,9
SW	-5,8	-7,3	-7,8
W	-5,4	-6,3	-7,3
NW	-6,1	-6,8	-7,8
Mittelwert	-6,78	-7,33	-8,20
Standardabw.	0,88	0,80	1,19

A.2. SuperCollider Skript des Hörtests

Das folgende Skript wurde zur Positionierung der fokussierten Tonquelle im Hörtest verwendet.

```
1 // WFS Hörtest
2
3 // 1. Teil: Person schaut in die Richtung der Quelle
4
5 // Zuhörer steht in der Mitte & in der linken oberen Ecke
6
7 NetAddr("192.168.3.1", 58100).
8 sendMsg("/WONDER/source/position", 0, -0.69, 1.92);
9 NetAddr("192.168.3.1", 58100).
10 sendMsg("/WONDER/source/position", 0, -0.35, 0.31);
11 NetAddr("192.168.3.1", 58100).
12 sendMsg("/WONDER/source/position", 0, -1.75, 2.33);
13
14 // 2. Teil: Person bewegt sich auf die Position der Quelle
15
16 NetAddr("192.168.3.1", 58100).
17 sendMsg("/WONDER/source/position", 0, 1.51, 1.4);
18 NetAddr("192.168.3.1", 58100).
19 sendMsg("/WONDER/source/position", 0, -0.19, 3.7);
20 NetAddr("192.168.3.1", 58100).
21 sendMsg("/WONDER/source/position", 0, -1.48, 2.16);
```

A.3. Einzeldaten des Hörtests

Die nachfolgenden Tabellen zeigen die Einzeldaten aller Probanden des Hörtests (Kapitel 6.2) sowie die daraus errechneten Mittelwerte und Standardabweichungen. Die Zahlen entsprechen den Lokalisationsfehlern in Grad (erste Tabelle) bzw. in Metern (zweite Tabelle).

<i>Hörtest (1. Teil)</i>	Source 1	Source 2	Source 3	Source 4	Source 5	Source 6
Proband 1	11,01	6,73	4,53	-21,40	-17,74	3,10
Proband 2	18,98	14,35	16,57	-21,29	-10,24	3,60
Proband 3	10,99	5,09	7,25	18,75	-15,23	11,43
Proband 4	11,03	6,52	-2,10	-9,77	-21,48	-5,52
Proband 5	15,59	3,99	3,39	9,87	-1,65	12,47
Proband 6	-1,84	2,40	-2,27	-4,41	1,26	-1,27
Proband 7	28,92	17,40	8,44	6,66	-14,49	16,99
Proband 8	13,53	10,66	11,72	9,78	-19,34	10,13
Proband 9	12,72	9,50	-1,35	1,56	-7,54	21,41
Proband 10	14,69	-5,46	N/A	-12,72	-5,44	-1,54
Proband 11	16,19	13,15	14,22	5,89	4,29	N/A
Proband 12	16,82	8,69	5,35	-3,32	13,85	4,56
Proband 13	9,35	4,47	6,53	7,75	-3,32	8,34
Proband 14	20,03	10,05	15,59	-1,13	-1,80	6,20
Proband 15	6,11	9,35	9,65	-4,52	-9,06	2,17
Proband 16	14,61	1,61	N/A	N/A	-20,29	13,54
Proband 17	12,66	13,48	1,49	-11,42	-21,16	-0,65
Proband 18	11,83	16,03	6,94	14,12	-11,70	22,68
Proband 19	21,18	10,83	-8,39	-3,66	11,71	12,53
Proband 20	8,82	-4,06	2,47	-6,93	1,64	3,13
Proband 21	18,00	N/A	5,53	0,76	-10,53	8,77
Mittelwert	13,87	7,74	5,56	-1,27	-7,54	7,60
Standardabw.	6,18	6,12	6,50	10,92	10,36	7,65

A. Informationen zu den Tests des zuhörerbasierten Renderings

Hörtest (2. Teil)	Source 1 X	Source 1 Y	Source 2 X	Source 2 Y	Source 3 X	Source 3 Y
Proband 1	0,2482	-0,1209	0,2845	-0,0450	0,2290	-0,1279
Proband 2	0,1266	-0,2275	0,2160	-0,1149	0,1243	-0,1521
Proband 3	0,0977	-0,1867	0,1792	-0,1424	0,3600	-0,0105
Proband 4	0,2001	-0,0879	0,1817	-0,2851	0,1116	-0,2227
Proband 5	0,3205	0,1399	0,2409	-0,1573	0,3624	0,1939
Proband 6	0,1913	-0,1108	0,2055	-0,1135	0,1877	-0,1324
Proband 7	0,2233	-0,0782	0,1993	-0,1512	0,4610	0,0217
Proband 8	0,0355	-0,1292	0,1935	-0,2588	0,1848	-0,1212
Proband 9	0,2590	-0,2938	0,2031	-0,2793	0,2072	-0,1528
Proband 10	0,0978	-0,1658	0,0600	-0,1858	0,0083	-0,0631
Proband 11	0,2618	-0,2629	0,1003	-0,0672	0,0745	-0,1285
Proband 12	0,1954	-0,2870	0,1830	0,0622	0,1103	0,0378
Proband 13	0,1687	-0,3913	0,1874	-0,1342	0,1819	-0,1547
Proband 14	0,2007	-0,0082	0,2027	-0,0913	0,1916	-0,1456
Proband 15	0,1977	-0,0730	0,4164	-0,0104	0,2694	-0,1499
Proband 16	0,2616	-0,1256	0,1953	-0,0757	0,2332	-0,2387
Proband 17	0,1831	-0,1090	0,1860	-0,1980	0,1803	-0,0118
Proband 18	0,2009	-0,1004	0,2245	-0,0728	0,2104	-0,1873
Proband 19	0,2102	-0,2297	0,2456	-0,1053	0,1938	-0,1578
Proband 20	0,0967	-0,1371	0,2508	-0,1385	0,1657	-0,2875
Proband 21	0,2343	-0,1522	0,1959	-0,1167	0,1835	-0,0478
Mittelwert	0,19	-0,15	0,21	-0,13	0,20	-0,11
Standardabw.	0,068	0,11	0,07	0,08	0,10	0,11

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 13.05.2014

Carola Christoffel