Master's Thesis

Evaluation of Soundfield Rotation Methods in the Context of Dynamic Binaural Rendering of Higher Order Ambisonics

Maximilian Neumayer

[redacted]

University Professor:	Prof. Dr. Stefan Weinzierl, TU Berlin
Supervisor:	Jan Plogsties, Fraunhofer IIS
	Jorgos Estrella, Fraunhofer IIS

Technische Universität Berlin, Audio Communication Group

Erlangen, October 2017

Fakultät |
Geisteswissenschaften

Prüfungsausschüsse

Studiendekan und Vorsitzender

Prof. Dr. Friedrich Steinle

Eidesstattliche Erklärung

Ist jeder an der TU Berlin verfassten schriftlichen Arbeit eigenhändig unterzeichnet beizufügen!

Hiermit erkläre ich an Eides statt gegenüber der Fakultät I der Technischen Universität Berlin, dass die vorliegende, dieser Erklärung angefügte Arbeit selbstständig und nur unter Zuhilfenahme der im Literaturverzeichnis genannten Quellen und Hilfsmittel angefertigt wurde. Alle Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinn nach entnommen wurden, sind kenntlich gemacht. Ich reiche die Arbeit erstmals als Prüfungsleistung ein. Ich versichere, dass diese Arbeit oder wesentliche Teile dieser Arbeit nicht bereits dem Leistungserwerb in einer anderen Lehrveranstaltung zugrunde lagen.

Titel der schriftlichen Arbeit

VerfasserIn/VerfasserInnen*

Name | Vorname | Matr.-Nr.

Betreuende/r DozentIn

Name | Vorname

Mit meiner Unterschrift bestätige ich, dass ich über fachübliche Zitierregeln unterrichtet worden bin und verstanden habe. Die im betroffenen Fachgebiet üblichen Zitiervorschriften sind eingehalten worden.

Eine Überprüfung der Arbeit auf Plagiate mithilfe elektronischer Hilfsmittel darf vorgenommen werden.

Ort, Datum | Unterschrift**

*Bei Gruppenarbeiten sind die Unterschriften aller VerfasserInnen erforderlich.
**Durch die Unterschrift bürgen Sie für den vollumfänglichen Inhalt der Endversion dieser schriftlichen Arbeit.

Contents

Abbreviations

VBAP	Vector Base Amplitude Panning
FOA	First Order Ambisonics
TOA	Third Order Ambisonics
VST	Virtual Studio Technology
GUI	Graphical User Interface
IQR	Interquartile Range
ANOVA	Analysis of Variance
SD	Standard Deviation
M	Mean
ACN	Ambisonic Channel Number
HRTF	Head Related Transfer Function
HRIR	Head Related Impulse Response
ITD	Interaural Time Difference
ILD	Interaural Level Difference
FUMA	Furse-Malham Higher-Order Format
OSC	Open Sound Control
MIDI	Musical Instrument Digital Interface
LSB	Least Significant Bit
MSB	Most Significant Bit

Abstract

With affordable Virtual Reality devices entering the mass market the demand for high quality spatial audio increases. To present the user a realistic auditory scene in an immersive environment it is essential to track the user's head position and rotate the sound scene accordingly to achieve a stable and realistic perception of the scene. So far this was mostly achieved by interchanging HRTF filters but the Ambisonics surround sound technology offers an alternative way to rotate soundfields, bringing this technology back into the focus of developers. Another way for rotation of a sound scene is the use of phantom sources within a given loudspeaker layout by using the VBAP algorithm. In this thesis a listening test was designed, implemented and conducted to assess the perceptual quality of those different sound scene rotation methods. The results show that Ambisonics rotation of third order source material and the rotation which uses the VBAP algorithm revealed the best ratings. Ambisonics domain rotation using first order source material was rated lower. The rotation which relies on the interchange of HRTF filters received the lowest ratings.

1 Introduction

1.1 Motivation

To present the user a realistic and immersive auditory scene in Virtual Reality technologies it is essential to track the user's head movements and rotate the sound scene accordingly to achieve a stable and realistic perception of the scene. The traditional solution is the interchange of HRTF filters corresponding to the user's head orientation. The advent of Ambisonics concepts for spatial audio content opens a new option to rotate a sound scene. Instead of adapting convolution filters according to the user's head orientation it is now possible to use the headtracker data to rotate the soundfield in Ambisonics domain by using matrix operations. When rotating soundfields in Ambisonics domain no time variant HRTF filters have to be used which avoids the real-time switching and interpolation of those filters. Less memory space is needed because no large database of HRTF filters covering the whole sphere is needed. Currently there is no information available concerning the perceptual quality of this method. This begs the following two research questions:

1. What is the optimal way to rotate Ambisonics audio material dependent on the user's head orientation for binaural reproduction over headphones?

2. What is the best trade-off taking into account perceptual quality and computational complexity?

1.2 Theoretical Background

This chapter presents the theoretical basics of the methods used in this thesis. The fundamentals of binaural hearing (Chapter 1.2.1) and dynamic binaural rendering (Chapter 1.2.4) will be presented as well as the concept of Ambisonics (Chapter 1.2.2) and the method of rotation in the Ambisonics domain. Chapter 1.2.3 sums up the VBAP panning algorithm.

1.2.1 Binaural Hearing

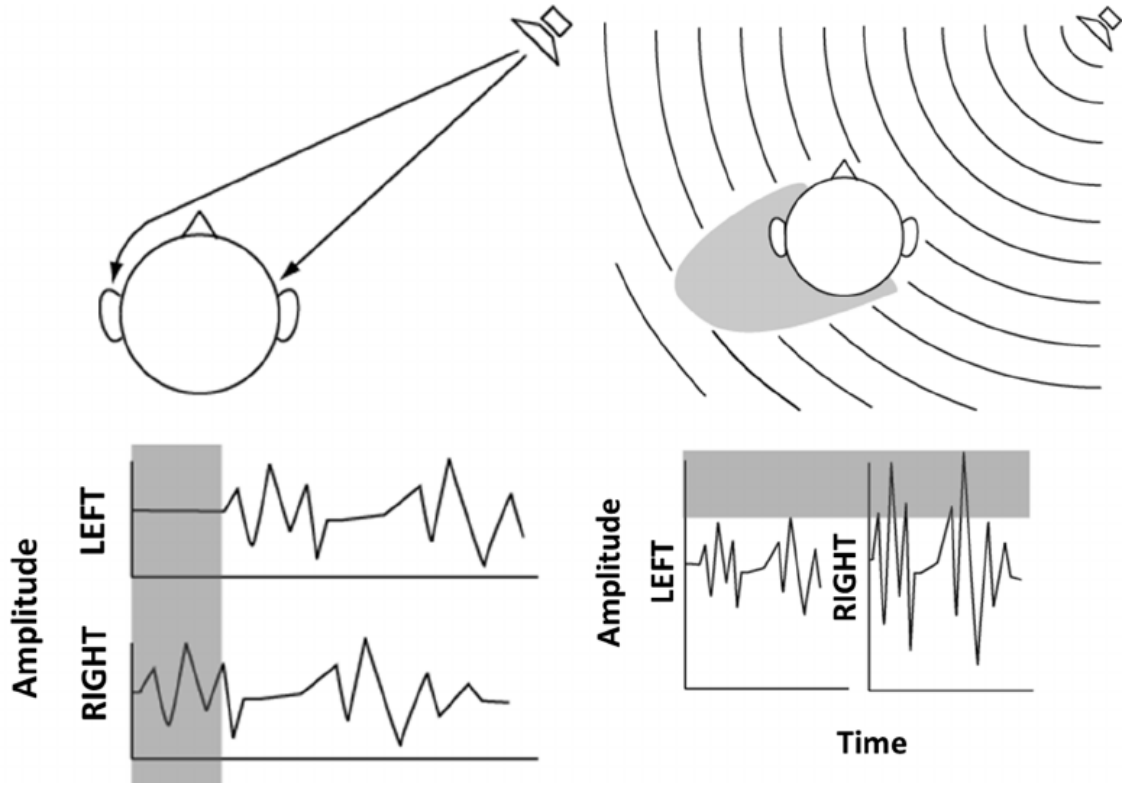

Figure 1.1: ITD (left) and ILD (right), taken from Zhong et al. (2015)

Human's auditory system is able to detect the incident angle of an auditory event with impressive accuracy. There are monaural cues involved in the localization process of an auditory event as well as binaural ones which are a result of comparison of the signals received at both ears:

The *Interaural Time Difference* (ITD) is a result of a signal propagation delay between the two ears and has the main influence on the localization of low frequencies.

The *Interaural Level Difference* (ILD) is the difference in loudness and frequency distribution between the two ears. The shape of head, torso, and pinna is altering the arriving sound depending on the direction and frequency. It has predominant influence on the localisation of higher frequencies because with decreasing wavelength the influence of those structures increases.

Head Related Impulse Responses (HRIR) in the time domain as well as their counterpart Head Related Transfer Functions (HRTF) in the spectral domain are capable to contain those temporal and spectral cues. Convolution of a sound source with a HRIR produces the listener's perception of sound coming from a particular direction.

1.2.2 Ambisonics

Unlike other surround sound formats the Ambisonics channels do not carry any loudspeaker signals. Instead of that the channels contain a representation of the soundfield and need to be decoded before playback. This gives the advantage that Ambisonics signals can be used for nearly any arbitrary loudspeaker setup.
First order Ambisonics encoding is easy to understand. The 0th channel, also known as the W channel contains a pure mono signal which is independent from the angles of incident. Spatial characteristics is recorded by the three other channels 1, 2, and 3, also knwon as Y, Z, and X which are containing signals recorded by figure-of-eight style microphones aligned at their respective axes. With increasing order those spatial characteristics are becoming more complex but the fundamental idea remains the same. Figure 1.2 gives an overview about the shape of the spherical harmonics up to 5th order. The term Higher Order Ambisonics, often abbreviated as HOA, denotes Ambisonics orders greater than the first one. Increasing the order enhances the spatial resolution as well as the size of the sweet spot, the area of good reproduction of the soundfield with the disadvantage of increasing the numbers of channels necessary.

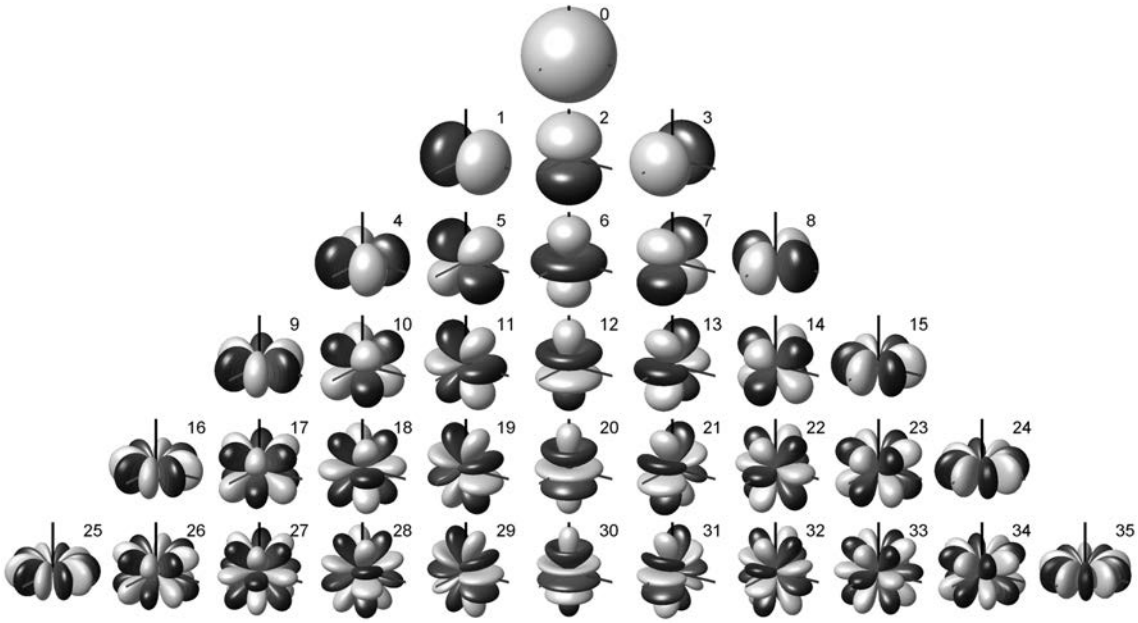

Figure 1.2: Spherical Harmonics up to the 5th Order (Zotter Franz, 2017).

Coming back to the stage of decoding it is important to note that accurate reconstruction becomes impossible if the number of loudspeakers used is fewer than the number of Ambisonics channels (Lopez et al., 2014, p. 2395).
The playback of Ambisonics via a real or virtual loudspeaker setup requires a decoder matrix which can be computed as follows:
Assuming $\boldsymbol{B}$ is a column vector containing the samples of the Ambisonics channels, $\boldsymbol{p}$ the loudspeaker feeds for a given loudspeaker setup, and $\boldsymbol{C}$ a re-encoding matrix. The decoding function

$$\boldsymbol{B} = \boldsymbol{C} * \boldsymbol{p} \tag{1.1}$$

can be transposed to reveal the loudspeaker signals

$$\boldsymbol{p} = \boldsymbol{C}^{-1} * \boldsymbol{B}. \tag{1.2}$$

The inverse of a matrix can be computed easily when the matrix is square which means that the number of Ambisonics channels is equal to the number of loudspeakers used. If the matrix is not square, i.e. the number of loudspeakers is smaller or greater the number of Ambisonics channels the *Moore-Penrose pseudoinverse* will be the

method of choice to compute the inverse:

$$\boldsymbol{C}^{-1} = \boldsymbol{C}^T(\boldsymbol{C} * \boldsymbol{C}^T)^{-1} = pinv(\boldsymbol{C}). \tag{1.3}$$

Combining equations 1.2 and 1.3 results in

$$\boldsymbol{p} = \boldsymbol{C}^T(\boldsymbol{C} * \boldsymbol{C}^T)^{-1} * \boldsymbol{B} = pinv(\boldsymbol{C}) * \boldsymbol{B} \tag{1.4}$$

which contains the desired loudspeaker signals.

1.2.2.1 Principles and Conventions

No standardized exchange format has been established yet when dealing with Ambisonics source material and a number of different conventions are existing. Those conflicting conventions affect channel order as well as weighting within the channels. In this thesis, unless stated otherwise, the Ambix format will be used (Nachbar et al., 2011). It is a proposed standard which uses the *Ambisonic Channel Number* (ACN) for the component ordering and SN3D normalization. Table 1.1 provides an overview of the ACN channel ordering up to the third order.
ACN channel ordering has the advantage that the Ambisonics channel number can be computed easily from the order and degree of the spherical harmonic which is widely used in mathematics:

$$ACN = l^2 + l + m \tag{1.5}$$

Where l denotes the order and m the degree of the spherical harmonic. The Schmidt semi-normalization (SN3D) has the advantage that it avoids clipping "because the peak amplitude of single point sources will never exceed the level of the 0th order signal" (Nachbar et al., 2011, p. 3). It is computed as follows, where l still denotes the order and m the degree of the spherical harmonic:

$$N_{l,m}^{SN3D} = \sqrt{(2 - \delta_m)\frac{(l - |m|)!}{(l + |m|)!}}. \tag{1.6}$$

(l,m)	ACN	Letter Code
(0,0)	0	W
(1,-1)	1	Y
(1,0)	2	Z
(1,1)	3	X
(2,-2)	4	V
(2,-1)	5	T
(2,0)	6	R
(2,1)	7	S
(2,2)	8	U
(3,-3)	9	Q
(3,-2)	10	O
(3,-1)	11	M
(3,0)	12	K
(3,1)	13	L
(3,2)	14	N
(3,3)	15	P

Table 1.1: Ambisonics Channels sorted by Order l and Degree m

δ is the Kronecker Delta which is defined as follows:

$$\delta_m = \begin{cases} 1 & \text{for m=0} \\ 0 & \text{otherwise} \end{cases} \tag{1.7}$$

1.2.2.2 Rotation

Different methods exist to rotate HOA Ambisonics source material around the three room axes. Zotter (2009) solves this problem by successive rotations around the Z and the Y axis and Blanco et al. (1997) uses an even different approach to rotate real spherical harmonics.

In this thesis the algorithm derived by Ivanic and Ruedenberg (1996, p. 6346) with the corrections from Ivanic and Ruedenberg (1998) will be implemented. The algorithm is capable of computing rotation matrices for an arbitrary order by using recurrence relations from the 3×3 first order rotation matrix. The computation of the elements $R^l_{mm'}$ of a rotation matrix $\boldsymbol{R}^l$ with order l is given as follows:

$$R^l_{mm'} = u^l_{mm'} U^l_{mm'} + v^l_{mm'} V^l_{mm'} + w^l_{mm'} W^l_{mm'}. \tag{1.8}$$

The numerical coefficients $u^l_{mm'}$, $v^l_{mm'}$, and $w^l_{mm'}$ are computed as a function of the order l and degree m with the Kronecker Delta δ_m which is defined in Equation 1.7:

$$\text{if } |m'| < l \Rightarrow \begin{cases} u^l_{mm'} = \left[\dfrac{(l+m)(l-m)}{(l+m')(l-m')}\right]^{\frac{1}{2}} \\ v^l_{mm'} = \dfrac{1}{2}\left[\dfrac{(1+\delta_{m0})(l+|m|-1)(l+|m|)}{(l+m')(l-m')}\right]^{\frac{1}{2}}(1-2\delta_{m0}) \\ w^l_{mm'} = -\dfrac{1}{2}\left[\dfrac{(l-|m|-1)(l-|m|)}{(l+m')(l-m')}\right]^{\frac{1}{2}}(1-\delta_{m0}) \end{cases} \tag{1.9}$$

$$\text{if } |m'| = l \Rightarrow \begin{cases} u^l_{mm'} = \left[\dfrac{(l+m)(l-m)}{(2l)(2l-1)}\right]^{\frac{1}{2}} \\ v^l_{mm'} = \dfrac{1}{2}\left[\dfrac{(1+\delta_{m0})(l+|m|-1)(l+|m|)}{(2l)(2l-1)}\right]^{\frac{1}{2}}(1-2\delta_{m0}) \\ w^l_{mm'} = -\dfrac{1}{2}\left[\dfrac{(l-|m|-1)(l-|m|)}{(2l)(2l-1)}\right]^{\frac{1}{2}}(1-\delta_{m0}) \end{cases} \tag{1.10}$$

.

The functions $U^l_{mm'}$, $V^l_{mm'}$, and $W^l_{mm'}$ occurring in equation 1.8 are defined by the following expressions:

$$\text{if } m = 0 \Rightarrow \begin{cases} U^l_{mm'} = {}_0P^l_{mm'} \\ V^l_{mm'} = {}_1P^l_{lm'} + {}_{-1}P^l_{-1m'} \\ W^l_{mm'} = - \end{cases} \tag{1.11}$$

$$\text{if } m > 0 \Rightarrow \begin{cases} U^l_{mm'} = {}_0P^l_{mm'} \\ V^l_{mm'} = {}_1P^l_{m-1,m'}(1+\delta_{m1})^{\frac{1}{2}} - {}_{-1}P^l_{-m+1,m'}(1-\delta_{m1}) \\ W^l_{mm'} = {}_1P^l_{m+1,m'} + {}_{-1}P^l_{-m-1,m'} \end{cases} \tag{1.12}$$

$$\text{if } m < 0 \Rightarrow \begin{cases} U^l_{mm'} = {}_0P^l_{mm'} \\ V^l_{mm'} = {}_1P^l_{m+1,m'}(1-\delta_{m,-1}) + {}_{-1}P^l_{-m-1,m'}(1+\delta_{m,-1})^{\frac{1}{2}} \\ W^l_{mm'} = {}_1P^l_{m-1,m'} - {}_{-1}P^l_{-m+1,m'} \end{cases} \tag{1.13}$$

.

The functions ${}_iP^l_{\mu m'}$ are given from the matrix elements R_{ij} and $R^{l-1}_{\mu,m'}$ as follows:

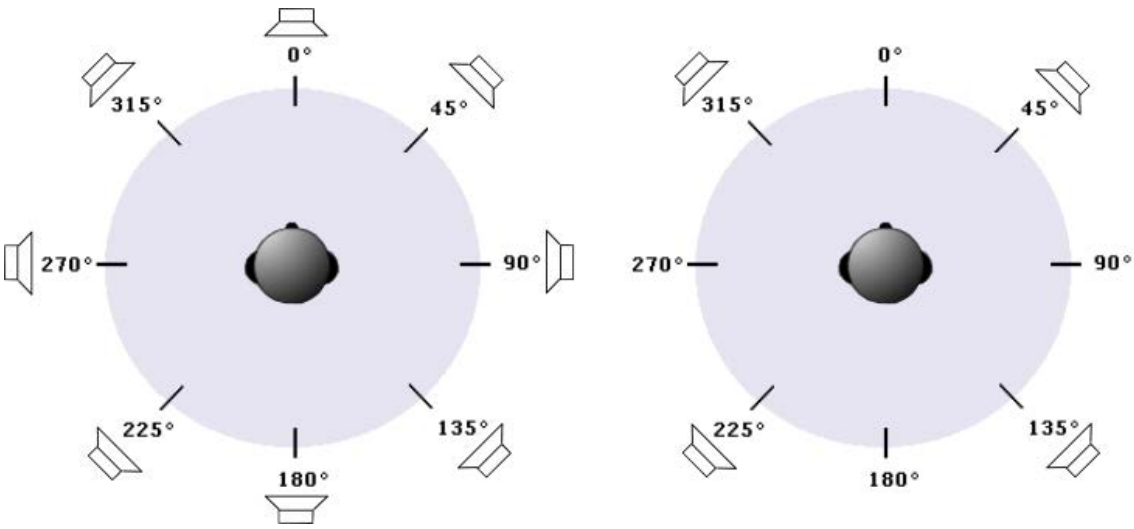

Figure 1.3: Loudspeaker Setup Used. The Left Side Depicts the Loudspeakers in the Horizontal Plane, the Right Side the ±45° Elevated Loudspeakers

$$
{}_iP^l_{\mu m'} = \begin{cases} R_{i,0}R^{l-1}_{\mu,m'} & \text{if } |m'| < l \\ R_{i,1}R^{l-1}_{\mu,l-1} - R_{i,-1}R^{l-1}_{\mu,-l+1} & \text{if } m' = l \\ R_{i,1}R^{l-1}_{\mu,m'} + R_{i,-1}R^{l-1}_{\mu,l-1} & \text{if } m' = l \end{cases} \tag{1.14}
$$

The implementation of this algorithm in C++ is described in Chapter 4.2.1.

1.2.3 VBAP

VBAP is a approach which allows virtual sound source positioning in an arbitrary loudspeaker setup by using phantom sources. The three dimensional case will be explained here. In VBAP the position of the phantom source depends on gain factors and therefore of the amplitude of the emanating signals. The sound power C is calculated from the gain factors of the three loudspeakers involved as in the 3D case

$$
g_1^2 + g_2^2 + g_3^2 = C. \tag{1.15}
$$

Those gain factors are applied to place a virtual source on the surface of a three-dimensional sphere with the radius which is defined by the distance of the listener to the loudspeakers.

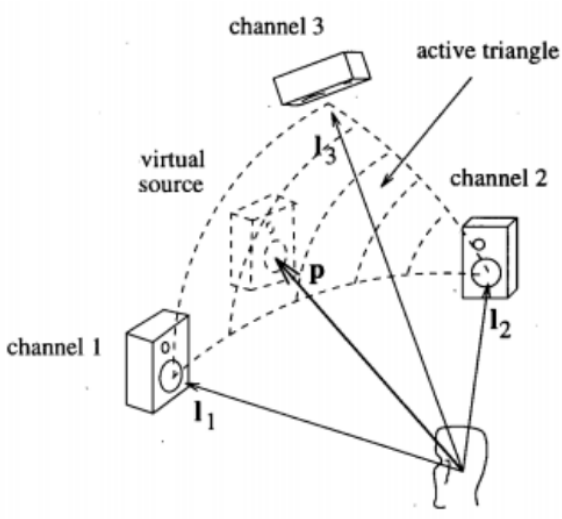

Figure 1.4: The VBAP Panning Algorithm, taken from Pulkki (1997, p. 460)

Unit vector which points to the loudspeakers is defined as

$$l_n = [l_{n1}\ l_{n2}\ l_{n3}]^T \tag{1.16}$$

where $n = [1,2,3]$ defines the directions of the loudspeakers. The direction of the virtual source is

$$\boldsymbol{p} = [p_1\ p_2\ p_3]^T \tag{1.17}$$

expressing $\boldsymbol{p}$ as a linear combination of the three loudspeaker vectors $\boldsymbol{l}_1$, $\boldsymbol{l}_2$, and $\boldsymbol{l}_3$ and the three gain factors results in

$$\boldsymbol{p} = g_1\boldsymbol{l}_1 + g_2\boldsymbol{l}_2 + g_3\boldsymbol{l}_3 \tag{1.18}$$

and

$$\boldsymbol{p}^T = \boldsymbol{g}\boldsymbol{L}_{123}. \tag{1.19}$$

With $\boldsymbol{g} = [g_1\ g_2\ g_3]$ and $\boldsymbol{L}_{123} = [\boldsymbol{l}_1\ \boldsymbol{l}_2\ \boldsymbol{l}_3]$ there is a solution for $\boldsymbol{g}$

$$\boldsymbol{g} = \boldsymbol{p}^T\boldsymbol{L}_{123}^{-1} = [p_1 p_2 p_3]\begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix}^{-1}. \tag{1.20}$$

Finally scaling reveals the gain factors which are

$$\boldsymbol{g}^{scaled} = \frac{\sqrt{C}\boldsymbol{g}}{\sqrt{g_1^2 + g_2^2 + g_3^2}}. \tag{1.21}$$

VBAP allows rotation of an arbitrary loudspeaker layout by placing virtual sources within the active triangles, as depicted in Figure 1.5.

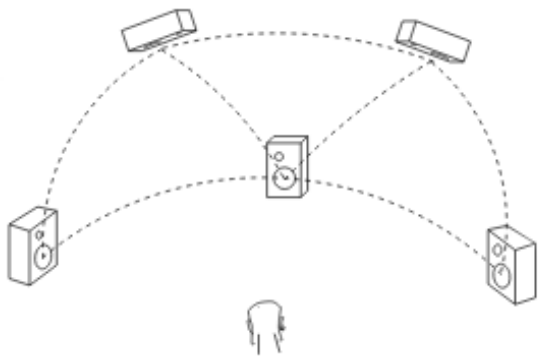

Figure 1.5: Segmentation of the Loudspeaker Setup into Triangles, Extracted from (Pulkki and Lokki, 1998)

1.2.4 Dynamic Binaural Synthesis

A system for dynamic binaural synthesis must be capable of switching HRTF filters or their counterpart in time domain, the HRIR filters, according to the user's head orientation. Hence every time the head orientation changes the system must look up for the matching set of a pair of filters in a database where all the filters are stored. Then it has to interchange those filters for the audio signal processing following. Computation time, especially when looking for the the new set of filters is crucial and should not exceed a certain value to avoid spatial instability. Audible impairments can occur in the moment of the interchange of the filters and therefore a cross-fade is often applied between the new and the old filters.

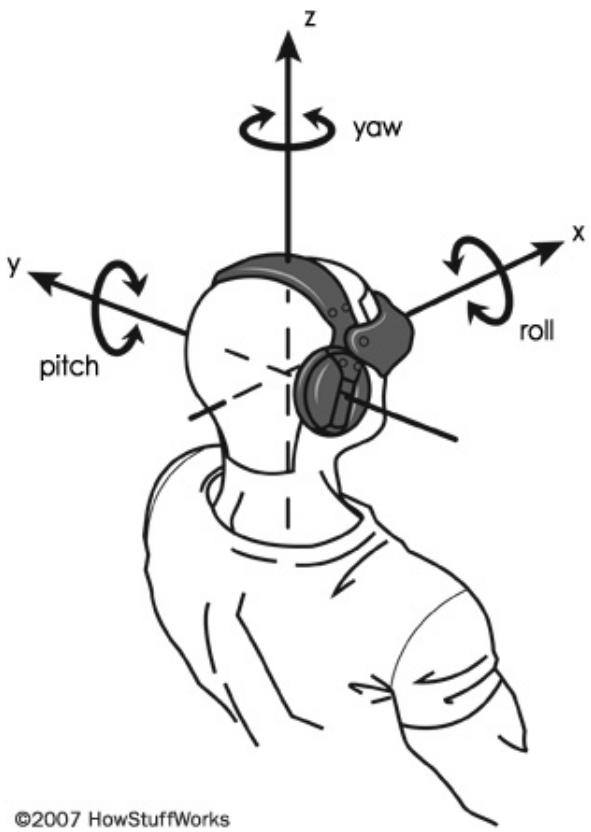

Figure 1.6: The Coordinate System Used in this Thesis (Strickland, 2017)

2 Related Work

Although the perception quality of different sound scene rotation methods when dealing with Ambisonics source material has not been investigated yet, there have been lots of investigations carried out in the past dealing with important subfields of this topic.

A listening test to compare seven different configurations of vector-based and Ambisonics amplitude panning methods in a hemispherical listening environment was carried out by Marentakis et al. (2014). The seven configurations constituted of either 12 or 24 loudspeakers and were presented to 30 participants. The participants had to rate the perceptual attributes of envelopment, spatial clarity, sound quality, and stability while listening to musical material from three different musical genres (popular, classical, and contemporary spatial music). In terms of sound quality this test revealed that the configuration which used VBAP with 12 and 24 loudspeaker was rated equal to the third order Ambisonics decoded to 12 loudspeakers.

Berge and Barrett (2010) conducted a MUSHRA test containing first, third and fifth order Ambisonics decoded to virtual loudspeakers without headtracking. A monophonic sound binaurally rendered served as reference. In this experiment the means of the participant's ratings for the first and third order stimulus were "poor" while the fifth order stimulus was rated as "good".

A pairwise comparison of different loudspeaker setups was done by Bertet et al. (2009). They used Ambisonics source material from first to fourth order with two different loudspeaker configurations each. 25 participants were asked to do a pairwise

comparison of their global perception on a scale from identical to different. The experiment showed that the participtans were able to distinguish between first and higher order Ambisonics as well as between different loudspeaker configurations. This shows that Ambisonics reproduction relies not only on the order but also on the reproduction system. When coming to the reproduction systems (Solvang, 2008) condsidered the number of loudspeakers used in an Ambisonics system. He discovered that "for HOA the number of loudspeakers [N] is a tradeoff between the reproduction error for $kr < N$ and spectral impairments for $kr > N$", showing that impairments are frequency dependent.

Kearney et al. (2012) investigated distance perception by comparing binaural synthesis using Ambisonics to the playback of a real loudspeaker. In this study participants were blindfolded and had to indicate the perceived distance while listening to a real loudspeaker and a binaurally synthesised, first, second and third order Ambisonics. It is interesting that they conclude that it "is noteworthy that for each order, there is no significant difference in the perception of the source location when compared to real-world sources."

When considering the resolution of headtracking systems Laitinen et al. (2012) had a close look at the temporal resolution of headtracking systems for binaural synthesis. He conducted a listening test with different update rates which yields that an update rate of $18\,Hz$ (50 ms) is still distinguishable from an update rate which was set to $100\,Hz$ (10 ms). Investigating the perception of latency in binaural synthesis Lindau (2009) revealed that in only a few cases latencies below $64\,ms$ were detectable.

3 Method and Test Design

As outlined in Chapter 2 the perceptual quality of different rotation methods when dealing with Ambisonics audio material hasn't been investigated yet. Hence it seems necessary to conduct a listening test. Before conducting a listening test it is essential to consider what research questions should be answered by the test and how the test should be designed to achieve this goal. Fortunately Bech and Zacharov (2007) present comprehensive work about audio listening tests and how they are designed. In the following the most relevant steps of the process of designing the test will be discussed based on the process diagram depicted in Figure 3.1.

Two research questions were stated which are the core of this investigation:

1. What is the optimal way to rotate Ambisonics audio material dependent on the user's head orientation for binaural reproduction over headphones?
2. What is the best trade-off taking into account perceptual quality and computational complexity?

Out of those research questions the following hypotheses are stated which the listening test should reject or accept.

Hypothesis I There is no audible difference between the rotation methods.

Hypothesis II The subjective impression of defects in FOA rotation is equal to the defects in TOA rotation.

Because formal listening tests are a time consuming and therefore an expensive task the experimenter should first consider whether a predictive model exists. Predictive

models have been developed as an alternative to listening tests with real subjects. Those models comprise a model of the human auditory and cognitive system to estimate subject's scoring. Because such a model was not available a listening test was the method of choice. It is also noteworthy that the development of such a model is a very complicated and time consuming task and therefore just worth of the work if it will be used often. It is important to note that the use of a model can provide misleading results when it is used besides its usually particular domain.

A number of standards have been developed to assess perceptual audio quality. When designing the listening test it was discussed whether to use the ITU-R Recommendation BS.1534-3 (2015), also known as MUSHRA (Multi-Stimulus Test with Hidden Reference and Anchor) test for intermediate audio quality or the ITU-R Recommendation BS.1116-3 (2015), also known as ABCHR (ABC Hidden Reference) test for small audio impairments. Both tests rely strongly on a reference signal which is also used as a hidden reference. Several options were discussed which kind of stimulus could be used as the reference signal. The decision was made not to stick to the two standards because of the absence of an appropriate reference signal. Instead of that the test was designed on a MUSHRA like bases where participants rate the subjective impression of impairments on a scale from 0 to 100 when comparing different methods against each other. No reference or anchors were used.

Furthermore it was decided not to run an attribute oriented (e.g. spatial impression, stability, spatial clarity ...) test but rather ask for global perception differences between the rotation methods because the magnitudes of the perceptual differences were estimated to be small. Making excessive demands on the participants should be avoided.

The number of participants who should take part in the experiment was set to 20 because the MUSHRA recommendation states out that 20 participants are a sufficient number for drawing appropriate conclusions. Furthermore all stimuli and rotation methods should be presented in randomised order.

Finally it was also decided that a pilot test with four participants will be conducted before conduction the main listening. This pilot test serves a number of purposes,

such as checking the experimental setup, the choice of stimuli and finally if the experimental hypotheses can be tested.

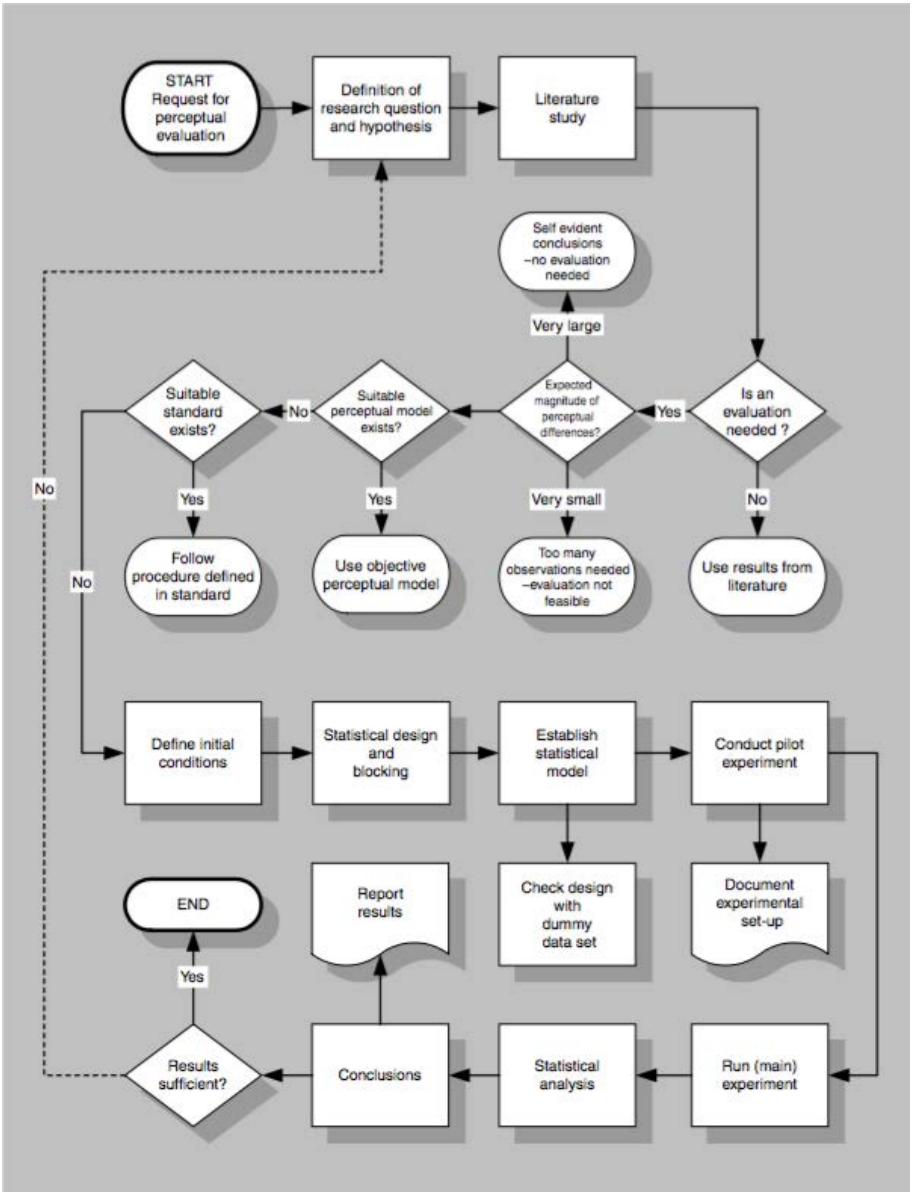

Figure 3.1: Process Diagram for the Preparation of a Listening Test, taken from Bech and Zacharov (2007, p. 31)

4 Implementation

4.1 Matlab

An Ambisonics rotator and decoder was implemented. Both are processing Ambisonics source material with FUMA channel ordering and are presented in the two following chapters.
In the beginning it was not clear which channel order format will be used and therefore a converter was implemented which resorts a given Ambisonics source file with ACN channel ordering to the FUMA convention and applies appropriate normalization factors. This converter works up to the third order.

```
function [out_fuma] = acn2fuma(in_acn)
```

Where the parameter **in_acn** denotes the Ambisonics source data with ACN channel ordering and SN3D normalization and the parameter **out_fuma** the resorted output data with FUMA channel order.

4.1.1 Ambisonics Rotator

A function was implemented to rotate Ambisonics source material up to the third order.

```
function [ out ] = rotateTOA( in,roll,pitch,yaw )
```

The parameters are:

in The third order Ambisonics source which will be rotated with FUMA ordering

roll, pitch, yaw The rotation angles in degrees

This function creates rotation matrices for yaw, pitch, and roll for each order seperatly which results in nine rotation matrices. For each order the three matrices for yaw, pitch, and roll then are multiplied to obtain 3 matrices for the order components. Those are applied to the channels representing the orders, i.e. the first order rotation matrix is applied to the channels two to four, the second order rotation matrices to channels five to nine, and the third order rotation matrices to the channels ten to sixteen. The first channel is not affected when rotating the field, thus no alteration of the signal happens there. The rotation matrices used were extracted from Zmoelnig (2002, p. 34–36).

4.1.2 Ambisonics Decoder

To decode first and third order Ambisonics source data a decoder was implemented in Matlab. It reads the samples from a four or sixteen channel Ambisonics audio file and converts it to the FUMA channel order if necessary by using the converter described in chapter 4.1. Roll, pitch, and yaw can be set if a rotation is desired. Then it takes the positions of the loudspeakers to compute the decoder matrix by using the Moore-Penrose pseudoinverse with Matlab's **pinv()** function as described in Heller et al. (2008, p. 23). This decoder matrix is multiplied with the matrix containing the samples of the source file which produces the signal for each of the loudspeakers. Finally those signals are convoluted with the matching HRIRs, depending on the position of the loudspeaker, normalized and exported as a stereo audio file.

4.2 VST

Because the listening test requires real-time sound processing Matlab was not the means of choice. A VST plug-in which runs within a digital audio workstation is more suitable for the scope of the test. Therefore a VST plug-in was implemented which rotates third order Ambisonics source data around three axis. The implementation is described in the following chapter.

4.2.1 Ambisonics Rotator

The Ambisonics rotator was implemented using the Juce C++ application framework (Juce Framework, 2017).

The VST receives the three rotation angles as parameters ranging from 0 to 1, where 0 denotes a rotation by 0° and 1 a rotation of 360°. Those parameters are then translated to values between −180° and 180° inside the VST. They are then transmitted to the function

```
void RotatorPlugInAudioProcessor::calcArrayTOA(float x, float y, float
    z)
```

where the computation of the rotation matrices starts. The function

```
void RotatorPlugInAudioProcessor::calcArrayFOA(float x, float y, float
    z)
```

is called first to obtain the 3×3 rotation matrix for the first order. There the values are converted into radians and then 3 matrices are computed for the rotation around the yaw, pitch, and roll axis.

```
        x = x * 3.1415 / 180;
        y = y * 3.1415 / 180;
```

```
        z = z * 3.1415 / 180;

        Eigen::Matrix3d rotRoll, rotPitch, rotYaw;
        rotRoll, rotPitch, rotYaw = Eigen::Matrix3d::Zero(3, 3);

        rotRoll.setZero();
        rotPitch.setZero();
        rotYaw.setZero();

        rotRoll(0, 0) = rotRoll(1, 1) = cos(x);
        rotRoll(0, 1) = -sin(x);
        rotRoll(1, 0) = sin(x);
        rotRoll(2, 2) = 1.f;

        rotPitch(0, 0) = 1.f;
        rotPitch(1, 1) = rotPitch(2, 2) = cos(y);
        rotPitch(1, 2) = -sin(y);
        rotPitch(2, 1) = sin(y);

        rotYaw(0, 0) = rotYaw(2, 2) = cos(z);
        rotYaw(0, 2) = sin(z);
        rotYaw(1, 1) = 1.f;
        rotYaw(2, 0) = -rotYaw(0, 2);

        rotArray = rotRoll * rotPitch * rotYaw;
```

The resulting 3×3 rotation matrix is then stored in the variable *rotArray* which is used to calculate the higher order rotation matrix according to the equations 1.8 to 1.14 in Chapter 1.2.2.2 by using the following functions

```
double calcU(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
double calcV(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
double calcW(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
```

```
double P(int i, int l, int mu, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_lm1)
```

The result is a rotation matrix with a size depending on the Ambisonics order. For example a third order rotation matrix is a 15×15 matrix because the first channel is not affected by rotation and therefore no alteration happens there. Every time a new audio buffer will be processed the function

```
void RotatorPlugInAudioProcessor::processBlock (AudioSampleBuffer&
    buffer, MidiBuffer& midiMessages)
```

is called which contains the main loop which applies the final rotation matrix to the 16-channel audio buffer by using two for-loops iterating trough all input and output buffer channels except the first channel

```
for (int out = 1; out < channels; out++)
        {
                for (int in = 1; in < channels; in++)
                {
                        outputBuffer.addFrom(out, 0, buffer.
                            getReadPointer(in), numSamples, (float)
                            rotMatrix(out, in));
                }
        }
```

Therefore the outputBuffer.addFrom function provided by the Juce API is used with the following parameters:

out The destination channel where the samples are added

0 The start sample within the buffer's channel. This is always 0 because the processing of a buffer starts always at the beginning of a complete buffer.

buffer.getReadPointer(in) A read pointer to the source data to use from the input buffer.

numSamples The number of samples to process.

(float)rotMatrix(out, in) The rotation matrix containing the gains which are applied to the source data.

4.3 Software Testing

When writing complex software code it is essential to test its correct function. Even small errors in computer code like typing errors can inhibit correct execution. Hence the code had to be tested. In the first stage it is obvious to listen to the results when dealing with audio signal processing but a more reliable test had to be conducted. This was achieved by testing whether the rotation matrices produced by the computer code match the properties of proper rotation matrices. Those properties are (taken from Altmann (1986, p. 65))

$$\det \boldsymbol{R} = 1 \tag{4.1}$$

and

$$\boldsymbol{R}^{-1} - \boldsymbol{R}^{T} = 0. \tag{4.2}$$

If condition 4.1 is met the rotation matrix is a proper rotation matrix. If condition from equation 4.2 is met the matrix is a orthogonal matrix, containing real entries which means that the row and column vectors are orthogonal unit vectors. To test the rotation matrices calculated by the computer code randomly generated matrices were tested if they meet those two conditions. Except for some small values due to numerical computation the matrices met those conditions. The perceptual test revealed correct results, too.

4.4 Max MSP

4.4.1 Audio Signal Processing

Various options for the flow of the signal were considered when designing the system. The first idea was to split the chain into four subsequent flows where every flow computes one rotation method. This idea was discarded soon because the computational effort to compute four subsequent signal chains introduced strong artefacts into the output signal. Finally the flow of the signal was designed that the signal always runs through one single signal chain where just the modules which introduce a rotation to the field are activated. Figure 4.5 shows the flowchart of the system. In the following the numbered parts of the singal chain will be explained:

1. All the seven stimuli used in the test were stored in one folder. Every time the test software was set up for a new participant a random sequence of the numbers from one to seven was generated and stored, determining the order of appearance of the stimuli. Based on this number the inherent audio file was red and stored in four 16-channel audio buffers. One buffer for each rotation method. When playback is started the actual sample count was transmitted in between all the four buffers to ensure simultaneous playback of all buffers. Omitting this synchronisation could have introduced artefacts like clicks when changing the audio source. Buffers not in use were muted. All the Ambisonics source files were in ACN channel order with SN3D normalization.

2. The VST plug-in receives the 16 channels from the Ambisonics playback as well as the headtracker data. The soundfield is then rotated using the implementation described in Chapters 1.2.2.2 and 4.2.1. An advantage of this implementation is that if the VST receives no headtracker data it just multiplies the samples of each channel with 1. Hence no degradation of the signal should happen in this stage if headtracking is disabled.

3. In this stage the Ambisonics source data is decoded to reveal the loudspeaker feeds. There are two decoders with one decoding matrix each to pocess first

and third order Ambisonics. The decoders are VST plugins taken from (ambiX v0.2.7 - Ambisonic plug-in suite, 2017).

4. The VBAP Rotator receives the loudspeaker feeds as well as the headtracking data. The system knows about the loudspeaker configuration in use (see Figure 4.1) and computes the gain factor of every loudspeaker and channel with respect to the received headtracker data. The VBAP Rotator was implemented by using the VBAP Max external by (Pulkki, 2000).

```
define_loudspeakers 3 0 0 45 0 90 0 135 0 180 0 -135 0 -90 0 -45 0 45 45 135 45 -135 45 -45 45 45 -45 135 -45 -135 -45 -45 -45
```

Figure 4.1: Loudspeaker Setup Definition used by the VBAP External

5. This is the stage of the dynamic binaural synthesis. It receives the loudspeaker feeds and processes them to obtain the two headphone signals. If headtracking is enabled for this stage it will look for the matching set of HRTFs from the database with respect to the headtracker data received. If headtracking is disabled it will use constant HRTFs with respect to the fixed position of the virtual loudspeakers. The system was implemented by using the SPAT software (IRCAM Spat software suite, 2017) with HRTFs recorded from a KEMAR mannequin from the CIPIC Database (Algazi et al., 2001) containing HRTFs for 1250 different directions (Figure 4.2). The fixed loudspeaker setup in Spat is depicted in Figure 4.3.

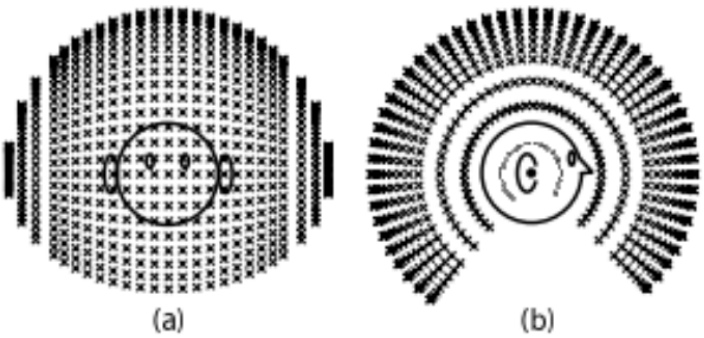

Figure 4.2: Location of data points at the (a) front and (b) side, extracted from Algazi et al. (2001)

6. The system receives the tracking data from the headtracker by using the MIDI

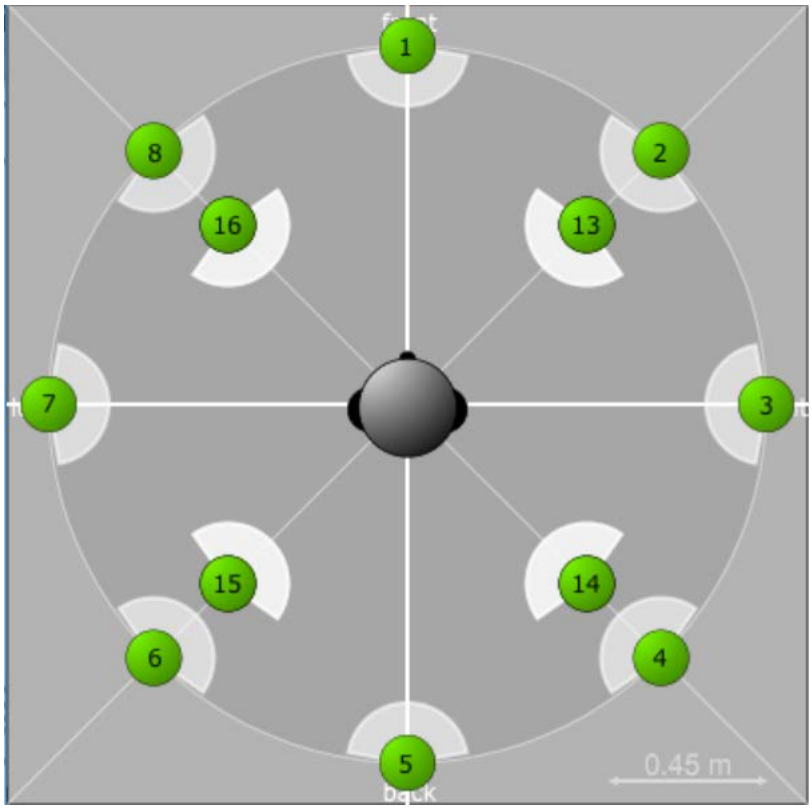

Figure 4.3: Loudspeaker Setup in Spat

protocol. A detailed description of the headtracker can be found in Chapter A.3.5.

7. The Rotation Method Selector receives the filename of the audio file which is played back at the moment. The filename contains the rotation method which shall be used. It extracts this information and passes the headtracker data to the correct rotation method. No headtracker data is send to the other methods not in use. It also controls the FOA/TOA toggle which routes the signal to the correct decoder depending on the Ambisonics order in use. The Max patch is depicted in Figure 4.4.

Figure 4.4: Rotation Method Selector

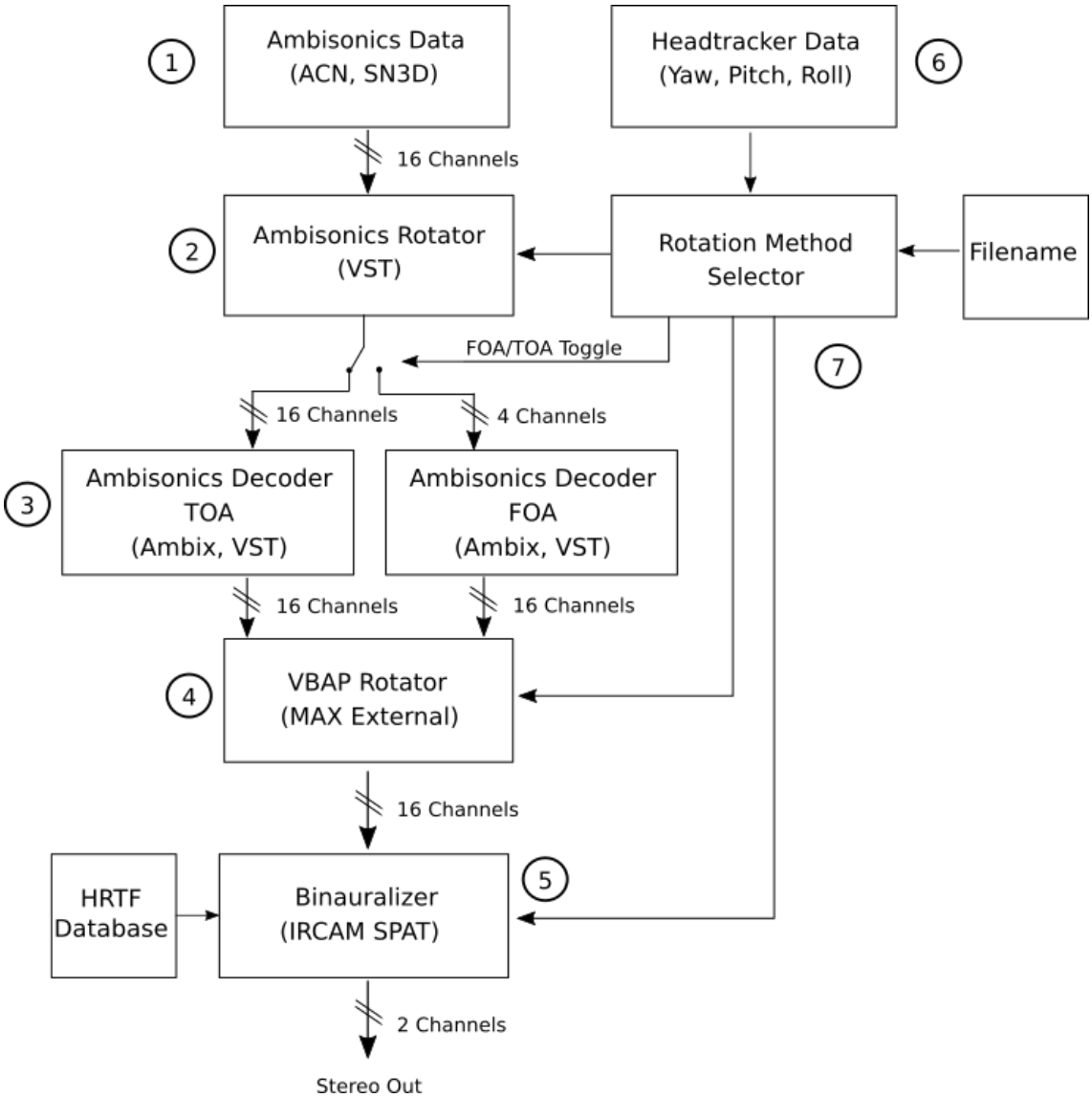

Figure 4.5: Flowchart of the Audio Signal Processing

4.4.2 Listening Test GUI

As basis for the listening test a Max patcher from the Institute of Sound Recording, University of Surrey (2017) was taken and modified. Some changes had to be applied to the audio signal processing chain to enable the processing of 16 channels as well as to the graphical appearance of the test. Figure 4.6 shows the listening test's graphical user interface. The sliders to give the ratings are located on the left hand side and the right hand side contains some instructions.

4.5 Headtracker

Different models of headtrackers were considered when building up the test setup. In the beginning some attempts were made with the InteriaCube[2] headtracker which is depicted in Figure 4.7. According to its datasheet it provides an angular resolution of $0.01°$ RMS with an update rate of $180\,Hz$ with three degrees of freedom (yaw, pitch, and roll). The headtracker is connected via the computer's serial port. It provides the option to set it up to send its headtracker data via the OSC protocol which can be received by Max MSP. Another option is to integrate the communication with the headtracker directly in the VST plug-in but unfortunately both configurations proved as not to be stable. Because of this instability and because of the use of the outdated serial port other options had to be considered.

Two other headtrackers were soldered and assembled. One of them is the *Hedrot* hedtracker. It transmits its three degrees of freedom headtracker data via the OSC protocol when connected to a computer using the USB port. The developers report an overall latency of $25 - 45$ ms. The other one is a headtracker called *Razor AHRS*. It consits of a *FTDI Breakout* board and a nine degrees of freedom *Razor IMU* headtracking unit. Unfortunately experiments revealed that those two headtrackers introduced an audible latency to the system. Therefore those headtrackers were not the means of choice, too. Finally a fourth headtracker model was chosen which revealed good results concerning latency and handling. *MrHeadTracker* which was developed by Romanov et al. (2017) is based on the Arduino platform and the BNO055

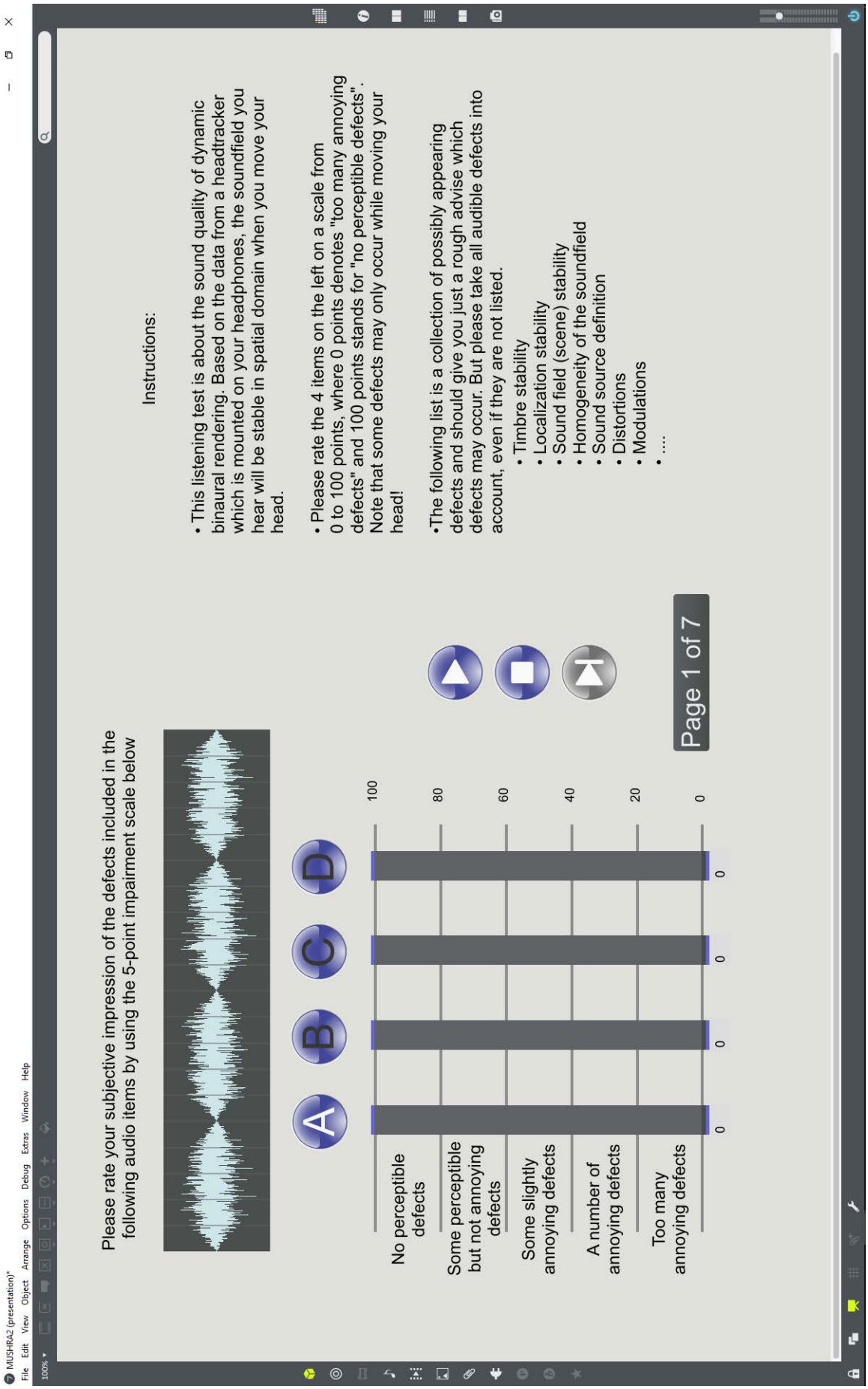

Figure 4.6: The Listening Test GUI

Figure 4.7: InertiaCube^2 Headtracker, extracted from the InertiaCube^2 Datasheet (2017)

sensor by Bosch Sensortec (2017). It uses the MIDI communications protocol to transfer its data which can be interpreted by almost every DAW including Max MSP. But because the standard MIDI resolution of 7 bit just allows an angular resolution of $\frac{360°}{2^7} = 2.81°$ it uses the 14 bit MIDI technique instead which transfers the LSB and the MSB in two separate 7 bit values which doubles its bit resolution which increases the angular resolution to $\frac{360°}{2^{14}} = 0.02°$. The developers state that this headtracker is "able to track rotations very fast and does not exceed a maximum delay of $\Delta time = 26ms$. Its response time of about $10ms$ and angular standard deviation between 0.5° and 2.50° serves the requirements remarkably well" (Romanov et al., 2017, p. 5).

4.6 Computational Complexity

When considering computational complexity the resources in terms of memory space needed as well as the computation time come into focus. The interchange of HRTF filters requires a fast search algorithm which looks up for the matching pair of filters in a usually huge database. Because the spatial resolution of the filters stored in the database is always limited and therefore the exact matching set for a certain

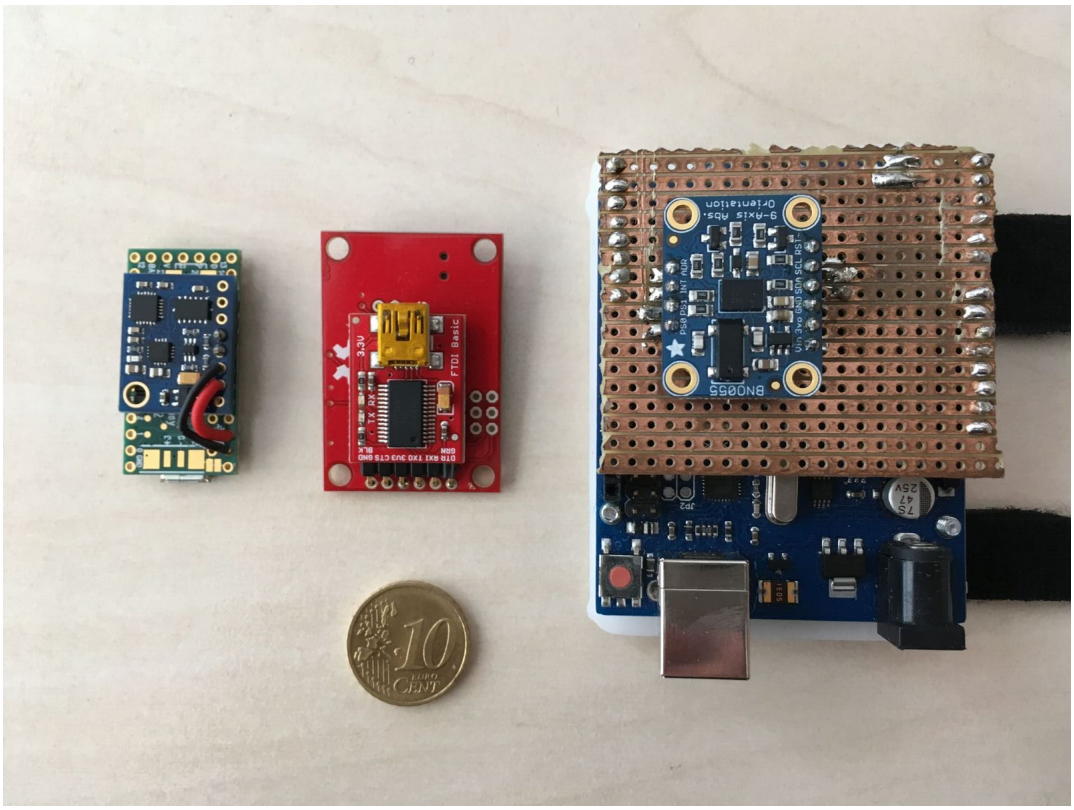

Figure 4.8: Proposed Headtracker models. From left to right: Hedrot (Baskind et al., 2017), Razor-AHRS (Guse et al., 2017) and MrHeadTracker (Romanov et al., 2017).

direction is rarely found, different methods of interpolation can be applied. After that the old filters used by a convolution engine have to be replaced with the new ones. The database needs memory space and the searching, computation, and replacing of the filter requires computation time. When rotating a sound scene using the VBAP panning algorithm this huge database is no longer needed because only as many pairs of HRTFs have to be stored as virtual loudspeakers are in the setup. Interpolation and replacement of the filters are no longer required, however the computation of the vectors and gains as outlined in Chapter 1.2.3 still has to be done with the cost of computational load. The system which does soundfield rotation in the Ambisonics domain also benefits from the advantage of time invariant filters. But the calculation of the rotation matrix by recurrence relations needs a lot of mathematical operations to calculate the higher order rotation matrices beginning from the first order rotation matrix. Many loops iterating through matrix elements have to be computed and this computation must be conducted every time a rotation angle changes. Pre-calculation of rotation matrices for many different rotation angles seems not as an alternative, bringing back the disadvantage of the huge database and the search algorithm.

5 Listening Test

A listening test was conducted in order to find an answer to the research question. First a small pilot test was set up to check the experimental system. The system, procedure, and its results are outlined in Chapter 5.1. A detailed description of the main listening test can be found in Chapter 5.2.

5.1 Pilot Test

Before conducting the main listening test a pilot test with four participants was conducted. This pilot test serves a number of purposes, such as checking the experimental setup, the choice and number of stimuli and finally a last check whether the experimental hypothesis can be tested.
Six different audio items were chosen which are described in Chapter 5.2.2:

1. Pink Noise
2. Pulsed Noise
3. Stadium Crowd
4. Guitar
5. Aurelia
6. Male speech

A desk was placed inside a regular office room on which a monitor, keyboard, and mouse was placed. The monitor was used to display the experiment software. A

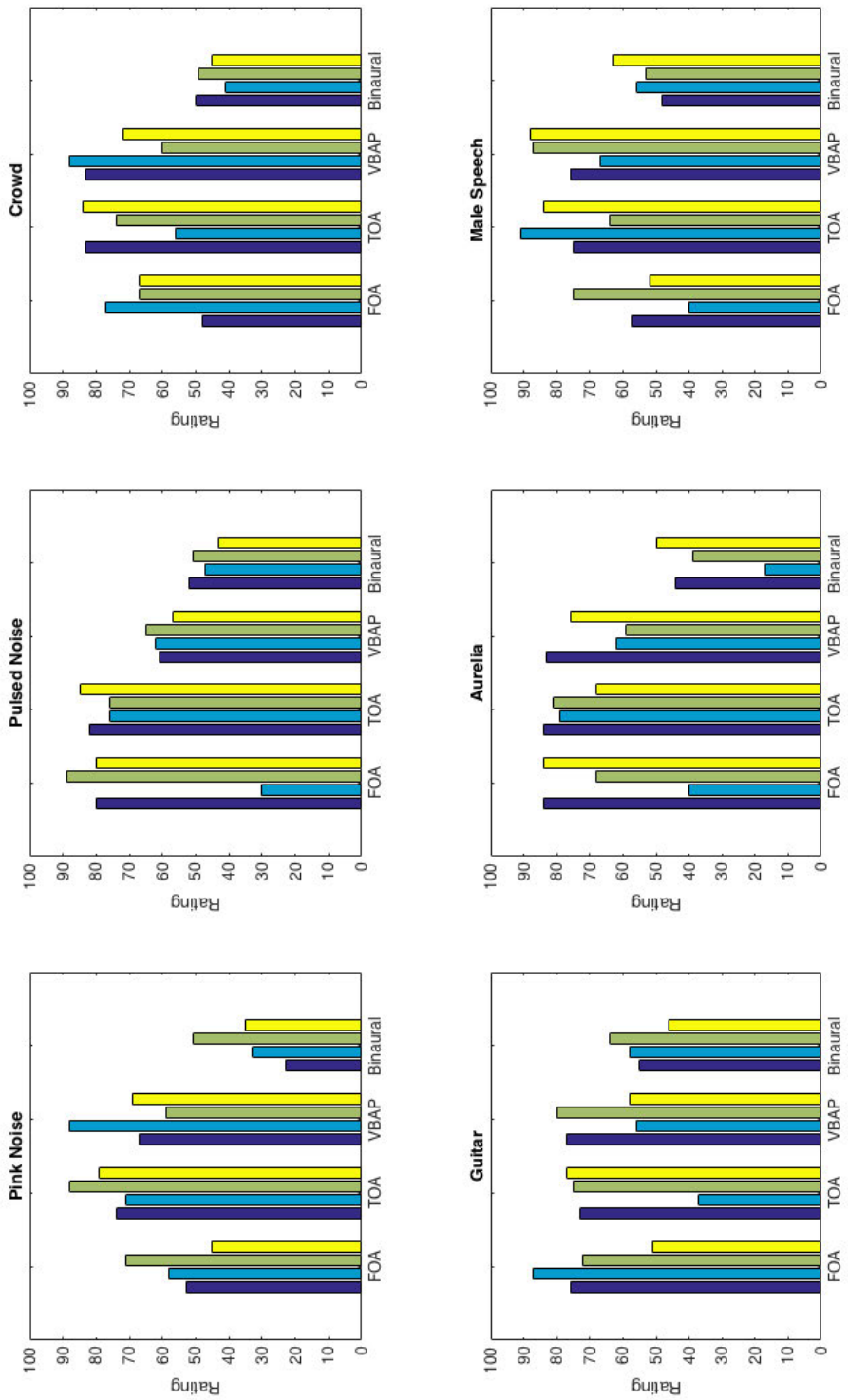

Figure 5.1: Results of the Pretest

RME Babyface audio interface and Beyerdynamic DT-990 250Ω Headphones with the headtracker attached were used for audio playback.

Four participants from the institute were chosen. They were verbally advised to rate the overall quality of the four different rotation methods under test by taking into account every kind of impairment. No written instructions were presented.
None of the participants needed more than 25 minutes to conduct this test.

The results of the pilot test are depicted in Figure 5.1. It can be seen that the binaural rotation method mainly revealed the lowest ratings, followed by the FOA stimulus.

Since the test did not take more than 25 minutes, the decision was made to included a 7th stimulus into the main test.

5.2 Listening Test

The listening test took part on three consecutive days and is described in the following chapters. The participants and the results of the questionnaire are described in Chapter 5.2.1. Chapter 5.2.2 gives an overview about the Stimuli used in the listening test and the hardware used is described in Chapter 5.2.3. In Chapter 5.2.4 the procedure of the listening test is described in detail.

5.2.1 Participants

Twenty participants (2 female, 18 male) volunteered to participate in the experiment. They were all employees of the Institute. They were between 20 and 52 years old (M = 32.60, SD = 8.60) and nineteen of them reported that they already conducted at least one listening test before. Ten participants reported that they already participated in a dynamic binaural listening test before and ten out of twenty stated out that they are specialist in spatial audio. Seven reported to be experts in judging timbre.

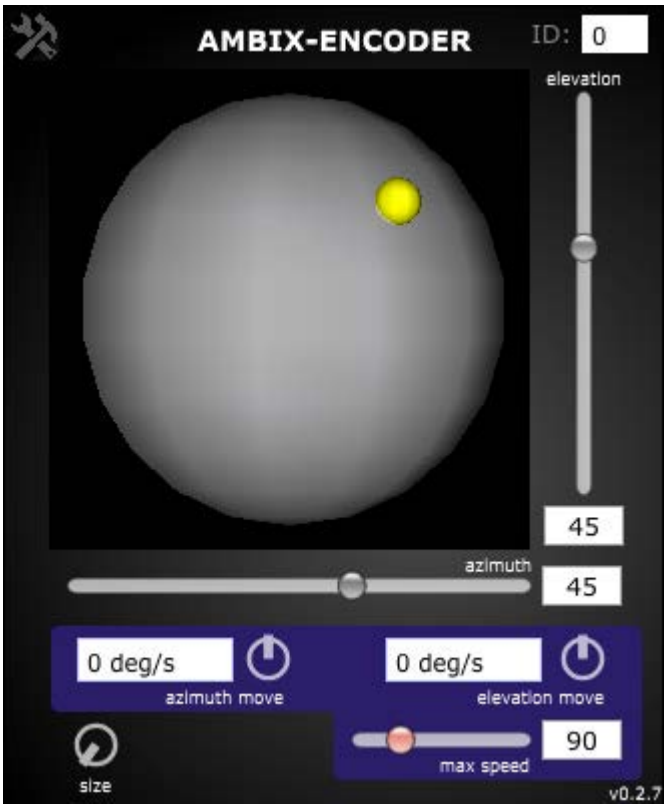

Figure 5.2: Encoding a Mono Source Placed on the Surface of a Sphere

5.2.2 Stimuli

The seven audio stimuli were chosen to cover an area as wide as possible. They include spatial real world recordings as well as synthetical generated signals. Some of the synthetical signals were generated using the ambiX v0.2.7 - Ambisonic plug-in suite (2017) plug in which allows the positioning of mono sources at an arbitrary position on a sphere (see Figure 5.2) and encoding to third order Ambisonics. Others were mixed by sound engineers and were provided by the Institute. The real world recordings were provided by the Institute, too and are recorded by using the *Eigenmike* soundfield microphone. The audio items had a length between fourteen and twenty seconds and were encoded as third order Ambisonics in the Ambix format (Nachbar et al., 2011) with a sampling frequency of 44100 Hz. The following list will give a detailed overview about the stimuli.

White Noise Bursts An excerpt with a duration of 20 seconds containing continuous white noise bursts with a duration of 75ms each

and $75ms$ of silence in between the bursts. An in and out crossfade was applied to each burst.

Pink Noise A twenty seconds long excerpt containing four bursts of pink noise with a duration of five seconds each. One second before and after each burst a one second long crossfade was applied. The mono audio source was panned to the front direction with no elevation using the Ambix encoder.

Stadium Crowd A yelling and hand clapping crowd in a sports stadium. This recording was made by using an Eigenmike soundfield microphone and has a duration of nineteen seconds. This stimulus was provided by the Institute.

Helicopter scene Recorded by using the Eigenmike soundfield microphone. It contains the sound of a helicopter flying from the left to the right above the listener. This stimulus was provided by the Institute.

Aurelia A music recording mixed by a professional mixing engineer containing a male singer with instruments. This stimulus was provided by the Institute.

Male speech A male speaker reading a text in English. The mono audio source which was provided by the institute was panned to the front direction with no elevation using the Ambix encoder.

Guitar A guitar playing mixed by a professional mixing engineer and provided by the Institute.

5.2.3 Materials and Apparatus

The test software was running on a HP Elitebook notebook with 16GB of RAM and a Intel Core i7-6500 dual-core 2.5 GHz processor. The buffer size of the audio processing was set to 1024 samples with a sampling frequency of $f_s = 44100\,Hz$. A *RME Madiface Pro* served as audio interface which was connected to the *STAX*

Studio Monitor amplifier with diffuse field compensation switched on. *STAX Lambda Pro New* headphones were used. The participants were free to adjust the loudness.

5.2.4 Procedure

The participants had to accomplish one test session in total. This session took part in a small office room containing the test equipment. They sat in front of a computer monitor that ran the experiment software. First they were asked to fill out a short questionnaire to retrieve informations about their gender, age, if they have conducted a listening test before, and a self-rating of their listening test expertise. After answering the questionnaire they were informed that this is an experiment about spatial audio and that a headtracker is attached to their headphones. They were also told that they can rotate their head freely and that the soundfield should stay stable when they move their head. After that they put on the headphones and the experimenter made sure that they wear the headphones with the correct orientation. They were asked to look straight forward at the computer monitor so that the experimenter could reset the orientation of the headtracker and the experimenter left the room. From now on all instructions were presented only by the experiment software and all interaction with the test only happend inside the experiment software. The listening test which is depicted in Figure 4.6 was presented to the participant containing the follwing question:

> Please rate your subjective impression of the defects included in the following audio items by using the 5-point impairment scale below

100-80	No perceptible defects
80-60	Some perceptible but not annoying defects
60-40	Some slightly annoying defects
40-20	A number of annoying defects
20-0	Too many annoying defects

On the right hand side of the experiment software additional instructions were presented:

Instructions:

- This listening test is about the sound quality of dynamic binaural rendering. Based on the data from a headtracker which is mounted on your headphones, the soundfield you hear will be stable in spatial domain when you move your head.
- Please rate the 4 items on the left on a scale from 0 to 100 points, where 0 points denotes "too many annoying defects" and 100 points stands for "no perceptible defects". Note that some defects may only occur while moving your head!
- The following list is a collection of possibly appearing defects and should give you just a rough advise which defects may occur. But please take all audible defects into account, even if they are not listed.
  - Timbre stability
  - Localization stability
  - Sound field (scene) stability
  - Homogeneity of the soundfield

- Sound source definition
- Distortions
- Modulations

The four sliders for rating the four rotation methods under test were located on the left hand side of the experiment software as well as an illustration of the waveform of the audio item. The sliders were named with the letters A, B, C, and D and the assignment of the different rotation methods to the sliders were randomized every time a new audio item was loaded. The participants were able to start and stop the audio playback and to give their ratings using the computer mouse or keys on the keyboard instead. Each audio item could be played back as often as desired. The participants rated the four rotation methods of one audio item and could then proceed to the next item if every slider was moved at least once. Every time the participant proceeded to the next audio item, which were also presented in a randomized order a pop-up window appeared. On this pop-up window a countdown was counting from ten to zero and the participant was asked to watch this countdown. During this time the headtracker heading was re-setted. The procedure was repeated until the participant rated all seven audio items.

6 Results and Statistics

A total of 20 participants took part in the listening test. The ratings from one participant had to be discarded because after the test the participant stated out that he was overchallenged by the listening test. Hence the results of 19 participants, each one listening to seven audio items with four different rotation methods, respectively, were evaluated. This leads to a sum of 133 observations for each rotation method and 532 observations in total.

All the evaluation shown in this chapter was done using *MATLAB* and its *Statistics and Machine Learning Toolbox*.

First a boxplot was created, containing the ratings of all participants for the seven audio items on the four rotation methods being tested. The plot is depicted in Figure 6.3. Additionally a boxplot was created, containing the cumulative ratings for the four rotation methods, which is depicted in Figure 6.1.

In the boxplots the central mark indicates the median, which separates the data samples into a lower and an upper half. The bottom and top edges of the surrounding box show the 25th and 75th percentiles. The distance between those two percentiles is the *Interquartile Range* (*IQR*). The lines extending vertically from the boxes are the so called *Whiskers*, which extend to a maximum of 1.5 times the *IQR*. Red crosses are indicating *Outliners*, data points which are outside the reach of the *Whiskers*.

Since many statistical methods assume normal distributed data, quantile-quantile plots (*Q-Q Plots*) were rendered as a visual method to assess a normal distribution. The plots display the quantiles of the sample data versus theoretical values from a normal distribution. If the requirement of normal distribution is met, the plotted points will be spotted along on the red marked line. The *Q-Q Plots* of the cumultative

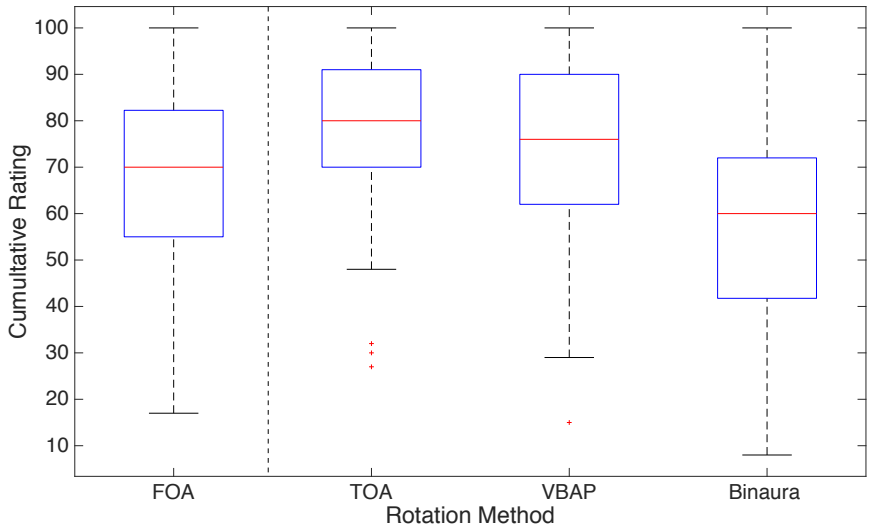

Figure 6.1: Boxplot with the Cumultative Ratings of all 4 Methods

ratings of the rotation methods are depicted in Figure 6.2.

Before conducting an Analysis of Variance (ANOVA) the homogeneity of the variances in the sample data should be analysed, because an ANOVA assumes equality of variances. Therefore a *Levene's Test* was conducted, which checks for equality of variances between the groups. This test was highly significant (p=0.0008) which means that the variances were inequal. But because of the equality of the sample sizes the ANOVA-Test should be robust against inequalities of the variances (Rogan and Keselman, 1977).
The one-way ANOVA showed a statistically significant difference between the groups (F(3,528) = 23.24, $p < .01$).

To check among which groups a statistically significant difference exists a Welch's t-test, which is robust against unequal variances was chosen. Before conducting this test, a Bonferroni correction had to be applied to the significance level to counteract the problem of multiple comparisons. Six observations are made in total, lowering the significance level of α which was set to 0.05 to a new value of $\alpha = 0.05/6 = 0.0083$. The t-test showed a significant difference in the ratings for the TOA rotation method (M=78.22, SD=15.56) and the binaural rotation method (M=58.95,

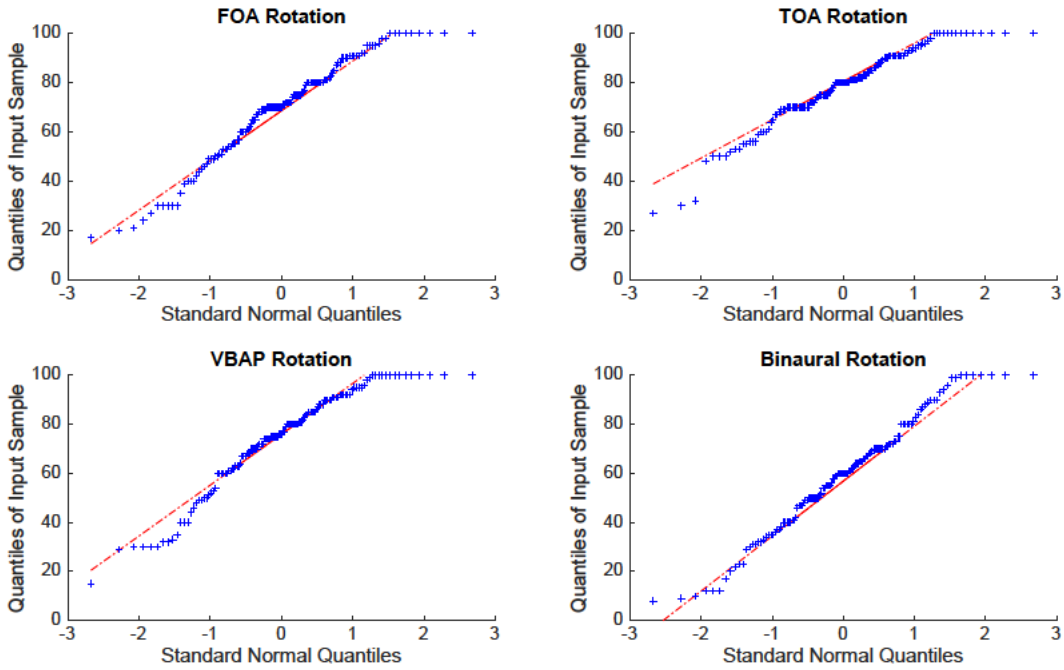

Figure 6.2: *Q-Q Plots* of the Cumultative Ratings for the Rotation Methods

SD=22.74); t(232)=8.08, p<.001. There was also a significant difference in the ratings for the FOA rotation method (M=69.15, SD=20.38) and the binaural rotation method (M=58.95, SD=22.74); t(260)=3.85, p<.001. Comparing the ratings for the VBAP rotation method (M=73.95, SD=20.00) with the binaural rotation method (M=58.95, SD=22.74) showed also a significant difference; t(259)=5.71, p<.001. No significant difference was found comparing the FOA with the VBAP rotation method; t(263)=-1.93, p=.054, as well as the comparison of the TOA with the VBAP rotation method; t(248)=1.95, p=.052. Furthermore a significant difference was found between the FOA and TOA rotation method; t(246)=-4.09, p<.001.

To assess the strength of the significant effects *Cohen's d* was calculated. *Cohen's d* is a measurement expressing score distances in units of variability (Cohen, 1988, p. 20). According to S. Sawilowsky (2009) a *Cohen's d* of 0.2 is considered a small effect, a value of 0.5 a medium effect, 0.8 as a large effect, and 1.2 a very large effect. *Cohen's d* indicates a large effect ($d = 1$) for the comparison of the TOA rotation method with the binaural rotation method. Medium effects are observed in between the two Ambisonics rotation methods ($d = .5$), between the FOA and

binaural rotation methods ($d = .5$), as well as for the observation of the VBAP with the binaural rotation method ($d = .7$).

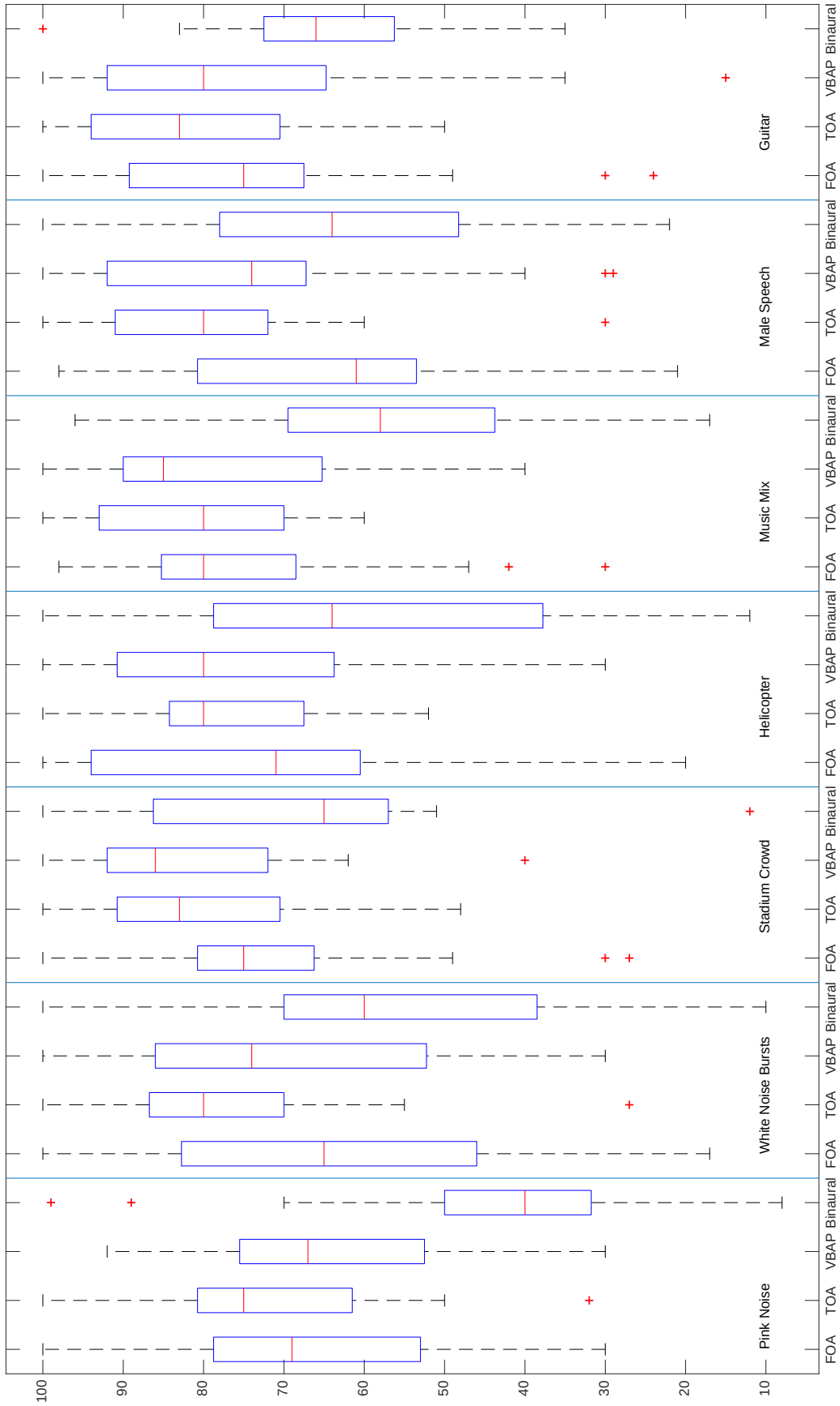

Figure 6.3: Results of all Rotation Methods and Stimuli Depicted as Boxplots

7 Discussion

The experiment presented was designed to investigate audible differences between three different soundfield rotation methods when using Ambisonics source material. The analysis of the experiment data revealed differences between the methods. The results are showing that soundfield rotation in the Ambisonics domain obtained the best ratings by the participants, followed by the the rotation method which uses the VBAP panning algorithm. The soundfield rotation method which rotates the field by interchanging HRTF filters according to the user's head orientation obtained the lowest ratings.
The participants were also able to distinguish between first and third order Ambisonics stimuli.

The test results are showing large variances within the given ratings. Minimizing those variances could be obtained with some improvements to the test design.
First of all it seems advisable to include a training session in the listening test. Some of the participants reported that it was difficult to listen to the small impairments while moving their head. A training session which sensitises the participants to impairments occurring could improve the results.
Another option to improve the results could be achieved by presenting a bimodal environment to the listeners. Larsson et al. (2002) revealed that by adding a visual stimulus to listening tests conducted in virtual environments the participants were more focused on the situation and it improves their task performance. Nevertheless choosing a suitable bimodal virtual environment is not a trivial task since the combination of unsuitable visual and acoustic stimuli could disturb the performance of the test.

When looking at the VBAP panning algorithm Pulkki and Lokki (1998) figured out that if the number of virtual loudspeakers is increased better localisation and less colouration will occur. It is also known that the spread of amplitude panned virtual sources depends on the panning direction (Pulkki, 1999).

To conclude, the rotation of the rendering loudspeaker layout affects perceptual quality since switching HRTF filters causes audible artifacts. Soundfield rotation using the VBAP panning algorithm avoids this problem, but there is still the drawback of being dependent on the loudspeaker layout. Taking all this into account the optimal way to rotate Ambisonics audio material is the rotation in the Ambisonics domain.

8 Summary

In this thesis the perceptual quality of three different sound scene rotation methods were investigated. While traditional dynamic binaural synthesis relies on a large database of HRTFs covering the sphere, the use of the Ambisonics format allows soundfield rotation according to a user's head orientation by using matrix operations which reduces the amount of HRTF filters needed and avoids the switching of filters in real time. As a third method the VBAP panning algorithm was investigated which rotates a sound scene by creating phantom sources within a given loudspeaker layout.

Because little is known about the perceptual quality of those three methods the two research questions arise:

1. What is the optimal way to rotate Ambisonics audio material dependent on the user's head orientation for binaural reproduction over headphones?
2. What is the best trade-off taking into account perceptual quality and computational complexity?

To find an answer to those questions a listening test was desinged to accept or reject the following hypothesis:

Hypothesis I There is no audible difference between the rotation methods.

Hypothesis II The subjective impression of defects in FOA rotation is equal to the defects in the TOA rotation.

The listening test was based on the concept of a MUSHRA test with participants judging global perception differences.

Such a listening test setup requires real-time audio processing and a fast switching between the three different rotation methods. An Ambisonics rotation VST plug-in was implemented which computes and applies rotation matrices depending on the user's head position using a recurrence approach. The rotation which uses the VBAP panning algorithm to create phantom sources within a given loudspeaker layout was realized by using a generic panning tool which is implemented in Max MSP by Pulkki (2000). Dynamic binaural synthesis was also used as a rotation method, realised by using the SPAT software which uses the Kemar HRTFs from the CIPIC Database (Algazi et al., 2001).

20 participants took part in the listening test. They rated impairments occurring between the three rotation methods of third order Ambisonics source files. To evaluate if the participants can distinguish between first and third order Ambisonics a forth stimulus was included with roation in the Ambisonics domain using first order source material.

The results of the test show that the rotation of the third order stimulus in the Ambisonics domain and the rotation using the VBAP algorithm received the highest ratings. Dynamic binaural synthesis was rated significantly lower which rejects the first hypothesis. The second hypothesis was rejected as well because the first order stimulus was rated significantly lower than the third order stimulus.

List of Figures

List of Tables

Bibliography

Algazi, V. R., Duda, R. O., Thompson, D. M., and Avendano, C. (2001). The CIPIC HRTF database. In *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, pages 99–102.

Altmann, S. L. (1986). *Rotations, Quaternions and Double Groups*. Oxford University Press.

ambiX v0.2.7 - Ambisonic plug-in suite (2017). `http://www.matthiaskronlachner.com/?p=2015`. [Online; accessed on 11/5/2017].

Baskind, A., Messonnier, J.-C., Lyzwa, J.-M., and Aussal, M. (2017). Hedrot – Open-Source Head Tracker. `https://abaskind.github.io/hedrot/`. [Online; accessed on 25/9/2017].

Bech, S. and Zacharov, N. (2007). *Perceptual Audio Evaluation - Theory, Method and Application*. John Wiley & Sons, 2007.

Berge, S. and Barrett, N. (2010). A new method for B-format to binaural transcoding. In *40th AES International conference, Tokyo, Japan*, pages 8–10.

Bertet, S., Daniel, J., Parizet, E., and Warusfel, O. (2009). Influence of microphone and loudspeaker setup on perceived higher order ambisonics reproduced sound field. In *1st Ambisonics Symposium, Gratz, Austria*.

Blanco, M. A., Florez, M., and Bermejo, M. (1997). Evaluation of the rotation matrices in the basis of real spherical harmonics. *Journal of Molecular Structure: THEOCHEM*, 419(1):19 – 27.

Bosch Sensortec (2017). https://www.bosch-sensortec.com/bst/products/all_products/bno055. [Online; accessed on 12/10/2017].

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates.

Guse, D., Wierstorf, H., and Bartz, P. (2017). Razor AHRS. https://github.com/Razor-AHRS. [Online; accessed on 25/9/2017].

Heller, A., Lee, R., and Benjamin, E. M. (2008). Is My Decoder Ambisonic? In *AES 125th Convention, San Francisco*, pages 1–21.

InertiaCube^2 Datasheet (2017). https://snebulos.mit.edu/projects/voila/docs/datasheets/InertiaCube2.pdf. [Online; accessed on 5/10/2017].

Institute of Sound Recording, University of Surrey (2017). https://github.com/IoSR-Surrey/MUSHRA-MaxMSP. Max MSP patcher, [Online; accessed on 7/6/2017].

IRCAM Spat software suite (2017). http://forumnet.ircam.fr/product/spat-en/. [Online; accessed on 8/6/2017].

ITU-R Recommendation BS.1116-3 (2015). http://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1116-3-201502-I!!PDF-E.pdf. Methods for the subjective assessment of small impairments in audio systems.

ITU-R Recommendation BS.1534-3 (2015). https://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1534-3-201510-I!!PDF-E.pdf. Method for the subjective assessment of intermediate quality level of audio systems.

Ivanic, J. and Ruedenberg, K. (1996). Rotation Matrices for Real Spherical Harmonics. Direct Determination by Recursion. *The Journal of Physical Chemistry*, 100(15):6342–6347.

Ivanic, J. and Ruedenberg, K. (1998). Rotation Matrices for Real Spherical Harmonics. Direct Determination by Recursion. Additions and Corrections. *The Journal of Physical Chemistry*, 102(45):9099–9100.

Juce Framework (2017). `https://www.juce.com/`. [Online; accessed on 21/8/2017].

Kearney, G., Gorzel, M., Rice, H., and Boland, F. (2012). Distance Perception in Interactive Virtual Acoustic Environments using First and Higher Order Ambisonic Sound Fields. *Acta Acoustica united with Acoustica*, 98:61–71.

Laitinen, M.-V., Pihlajamäki, T., Lösler, S., and Pulkki, V. (2012). Influence of Resolution of Head Tracking in Synthesis of Binaural Audio. In *Audio Engineering Society Convention 132*.

Larsson, P., Vastfjall, D., and Kleiner, M. (2002). Better Presence and Performance in Virtual Environments by Improved Binaural Sound Rendering. In *Audio Engineering Society Conference: 22nd International Conference: Virtual, Synthetic, and Entertainment Audio*.

Lindau, A. (2009). The perception of system latency in dynamic binaural synthesis. In *NAG/DAGA 2009 International Conference on Acoustics, Rotterdam*.

Lopez, J. A. T., Okamoto, T., Iwaya, Y., Sakamoto, S., and iti Suzuki, Y. (2014). Evaluation of a high-order Ambisonics decoder for irregular loudspeaker arrays through reproduced field measurements. *The Journal of the Acoustical Society of America*, 135(4):2394–2394.

Marentakis, G., Zotter, F., and Frank, M. (2014). Vector-Base and Ambisonic Amplitude Panning: A Comparison Using Pop, Classical, and Contemporary Spatial Music. *Acta Acustica united with Acustica*, 100(5):945–955.

Nachbar, C., Zotter, F., Deleflie, E., and Sontacchi, A. (2011). Ambix – Suggesting an Ambisonics Format. In *3rd International Symposium on Ambisonics and Spherical Acoustics*.

Pulkki, V. (1997). Virtual Sound Source Positioning Using Vector Base Amplitude Panning. *J. Audio Eng. Soc*, 45(6):456–466.

Pulkki, V. (1999). Uniform spreading of amplitude panned virtual sources. In *Applications of Signal Processing to Audio and Acoustics, 1999 IEEE Workshop*, pages 187–190.

Pulkki, V. (2000). Generic Panning Tools for MAX/MSP. In *Proceedings of International Computer Music Conference*, pages 304–307.

Pulkki, V. and Lokki, T. (1998). Creating Auditory Displays with Multiple Loudspeakers Using VBAP: A Case Study with DIVA Project. In *Proceedings of the 1998 International Conference on Auditory Display*, ICAD'98, pages 23–23, Swindon, UK. BCS Learning & Development Ltd.

Rogan, J. C. and Keselman, H. J. (1977). Is the ANOVA F-Test Robust to Variance Heterogeneity When Sample Sizes are Equal?: An Investigation via a Coefficient of Variation. *American Educational Research Journal*, 14(4):493–498.

Romanov, M., Berghold, P., Frank, M., Rudrich, D., Zaunschirm, M., and Zotter, F. (2017). Implementation and Evaluation of a Low-Cost Headtracker for Binaural Synthesis. In *Audio Engineering Society Convention 142, Berlin*.

S. Sawilowsky, S. (2009). New Effect Size Rules of Thumb. *Journal of Modern Applied Statistical Methods*, 8:597–599.

Solvang, A. (2008). Spectral Impairment of Two-Dimensional Higher Order Ambisonics. *J. Audio Eng. Soc*, 56(4):267–279.

Strickland, J. (2017). How Virtual Reality Gear Works. `http://electronics.howstuffworks.com/gadgets/other-gadgets/VR-gear6.htm`. [Online; accessed on 11/9/2017].

Zhong, X., Yost, W., and Sun, L. (2015). Dynamic binaural sound source localization with ITD cues: Human listeners. *The Journal of the Acoustical Society of America*, 137(4):2376–2376.

Zmoelnig, J. (2002). Entwurf und Implementierung einer Mehrkanal-Beschallungsanlage. Diploma thesis, University of Music and Performing Arts Graz, Austria.

Zotter, F. (2009). Analysis and Synthesis of Sound-Radiation with Spherical Arrays. Phd thesis, University of Music and Performing Arts Graz, Austria.

Zotter Franz (2017). Ambisonics – Wikipedia, The Free Encyclopedia. 2017, CC BY-SA 3.0. `https://en.wikipedia.org/wiki/Ambisonic_data_exchange_formats#/media/File:Spherical_Harmonics_deg5.png`. [Online; accessed on 14/9/2017].

A Zusammenfassung auf Deutsch

A.1 Abstract

Immer mehr erschwingliche Geräte für Virtual Reality drängen in den Massenmarkt und der Bedarf an hochqualitativem Raumklang dafür wächst stetig. Für eine realistische Abbildung einer Klangszene und damit der Erzeugung einer Immersion ist es vonnöten die Kopfbewegungen des Nutzers zu erfassen und das Klangfeld entsprechend anzupassen. Dies wurde bisher hauptsächlich durch den Austausch von HRTF Filtern gemacht, jedoch bietet das Ambisonicsformat eine alternative dazu. Eine weitere Alternative dazu ist der Erzeugung von Phantomschallquellen mittels des VBAP Algorithmus. In dieser Arbeit wurde ein Hörversuch entworfen, implementiert und durchgeführt um perzeptive Unterschiede zwischen diesen Methoden zu untersuchen. Die Resultate des Versuches zeigen, dass die Schallfelddrehung von Ambisonics dritter Ordnung im Ambisonicsraum, sowie die Drehung mittels Phantomschallquellen von den Probanden am besten bewertet wurde. Danach folgt die Ambisonicsdrehung von Ambisonicsmaterial erster Ordnung. Die Drehung mittels des Austausches von HRTF Filtern erzielte die schlechtesten Bewertungen.

A.2 Methode und Testdesign

Da die perzeptiven Unterschiede von Schallfelddrehungen von im Ambisonicsraum codiertem Audiomaterial noch nicht untersucht wurden, war es nötig einen Hörversuch durchzuführen, um Antworten auf folgende Forschungsfragen zu finden:

1. Was ist die optimale Methode um im Ambisonicsraum codiertes Audiomaterial entsprechend der Kopfbewegung eines Nutzers für die Wiedergabe mittels Kopfhörern zu drehen?
2. Welches ist die beste Methode für diese Drehung im Hinblick auf perzeptive Qualität und Rechenaufwand?

Aus diesen Forschungsfragen wurden Hypothesen generiert:

Hypothese I Es gibt keinen Hörbaren Unterschied zwischen diesen Methoden.

Hypothese II Die wahrgenommenen Unterschiede der Rotation von Ambisonics erster Ordnung sind gleich derer bei der Rotation von Ambisonics dritter Ordnung.

Etliche schon vorhandene Standards kamen für einen solchen Hörversuch in frage. Es wurde diskutiert, einen sogenannten MUSHRA Test durchzuführen. Dieser ist entwickelt worden, um Qualitätsunterschiede mittleren Ausmaßes zu erfassen. Auch ein sogenannter ABCHR-test wurde diskutiert. Beide Konzepte beruhen auf der Idee einer versteckten Referenz, welche die bestmöglichste Qualität bietet und vom Probanden im besten Fall auch am höchsten bewertet wird. Da bei diesem Test die Darbietung eines Stimulus mit bestmöglichster Qualität nicht möglich ist, kamen beide Konzepte nicht in Frage. Der Test sollte sich an das MUSHRA Konzept anlehnen, ohne es komplett zu übernehmen. Außerdem wurde entschieden, nicht einzelne Attribute abzufragen, sondern nur die allgemeine Wahrnehmung von Unterschieden bewerten zu lassen, um die Probanden nicht zu überfordern. Die Anzahl der Probanden wurde auf 20 festgelegt. Abschließend wurde noch entschieden, einen kleinen Vortest durchzuführen. Dieser soll der Überprüfung der Testsoftware und der verwendeten Stimuli dienen.

A.3 Implementierung

A.3.1 VST

Für die technische Umsetzung des Tests war es nötig ein VST Plug-In zu programmieren mit welchem im Ambisonicsformat aufgenommene Schallfelder möglichst schnell und effizient um alle drei Raumachsen gedreht werden können. Für die Implementierung wurde das Juce Framework verwendet (Juce Framework, 2017). Die Berechnung der Rotationsmatrix findet in folgender Funktion statt:

```
void RotatorPlugInAudioProcessor::calcArrayTOA(float x, float y, float
    z)
```

Als Parameter bekommt die Funktion die Winkel, um welche das Schallfeld gedreht werden soll, übergeben und diese ruft zunächst nachfolgende Funktion, ebenfalls mit den Winkeln als Parameter, auf:

```
void RotatorPlugInAudioProcessor::calcArrayFOA(float x, float y, float
    z)
```

Dort werden die 3×3 Rotationsmatrizen für die erste Ordnung berechnet:

```
        x = x * 3.1415 / 180;
        y = y * 3.1415 / 180;
        z = z * 3.1415 / 180;

        Eigen::Matrix3d rotRoll, rotPitch, rotYaw;
        rotRoll, rotPitch, rotYaw = Eigen::Matrix3d::Zero(3, 3);

        rotRoll.setZero();
        rotPitch.setZero();
        rotYaw.setZero();

        rotRoll(0, 0) = rotRoll(1, 1) = cos(x);
        rotRoll(0, 1) = -sin(x);
        rotRoll(1, 0) = sin(x);
        rotRoll(2, 2) = 1.f;
```

```
        rotPitch(0, 0) = 1.f;
        rotPitch(1, 1) = rotPitch(2, 2) = cos(y);
        rotPitch(1, 2) = -sin(y);
        rotPitch(2, 1) = sin(y);

        rotYaw(0, 0) = rotYaw(2, 2) = cos(z);
        rotYaw(0, 2) = sin(z);
        rotYaw(1, 1) = 1.f;
        rotYaw(2, 0) = -rotYaw(0, 2);

        rotArray = rotRoll * rotPitch * rotYaw;
```

Diese Rotationsmatrix wird in der Variablen *rotArray* für die weitere Verwendung gespeichert. Dann beginnt die rekursive Berechnung für die höheren Ordnungen mit folgenden Funktionen

```
double calcU(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
double calcV(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
double calcW(int l, int m, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_l_m1)
double P(int i, int l, int mu, int m_, Eigen::Matrix3d& R_1, Eigen::
    MatrixXd& R_lm1)
```

Die resultierende Matrix wird dann mittels zweier for-Schleifen auf das zu drehende Schallfeld angewendet

```
for (int out = 1; out < channels; out++)
        {
                for (int in = 1; in < channels; in++)
                {
                        outputBuffer.addFrom(out, 0, buffer.
                            getReadPointer(in), numSamples, (float)
                            rotMatrix(out, in));
                }
        }
```

A.3.2 Softwaretest

Selbst kleine Tippfehler können in der Softwareprogrammierung zu großen Fehlern führen. Daher musste die korrekte Berechnung der Matrizen überprüft werden. Dazu wurde einerseits perzeptiv das Ergebnis evaluiert um eventuelle Fehler herauszuhören, andererseits überprüft, ob die Matrizen die Eigenschaften von Rotationmatrizen überhaupt erfüllen (Altmann, 1986, p. 65).

$$\det \boldsymbol{R} = 1 \tag{A.1}$$

and

$$\boldsymbol{R}^{-1} - \boldsymbol{R}^{T} = 0. \tag{A.2}$$

Beide Überprüfungen, sowie zu perzeptive Prüfung vielen positiv aus.

A.3.3 Audiodatensignalverarbeitung

Bei dem Design der Audiodatensignalverarbeitung wurde Wert darauf gelegt, dass die Audiosignale unabhängig von der aktuellen Rotationsmethode den gleichen Weg durch das System zurücklegen. Dieser Signalfluss ist in Abbildung A.1 abgebildet.

A.3.4 Listening Test GUI

Die Grundlage für den Hörversuch bildete ein Max Patcher der Universität Surrey ((Institute of Sound Recording, University of Surrey, 2017)) der modifiziert wurde. Die grafische Oberfläche des Hörversuchs ist in Abbildung A.2 zu sehen.

A.3.5 Headtracker

Verschiedene Modelle von Headtrackern wurden ausprobiert, jedoch lieferte nur ein Headtrackermodell namens *MrHeadTracker* ((Romanov et al., 2017)) zufriedenstellende Ergebnisse und wurde daher ausgewählt.

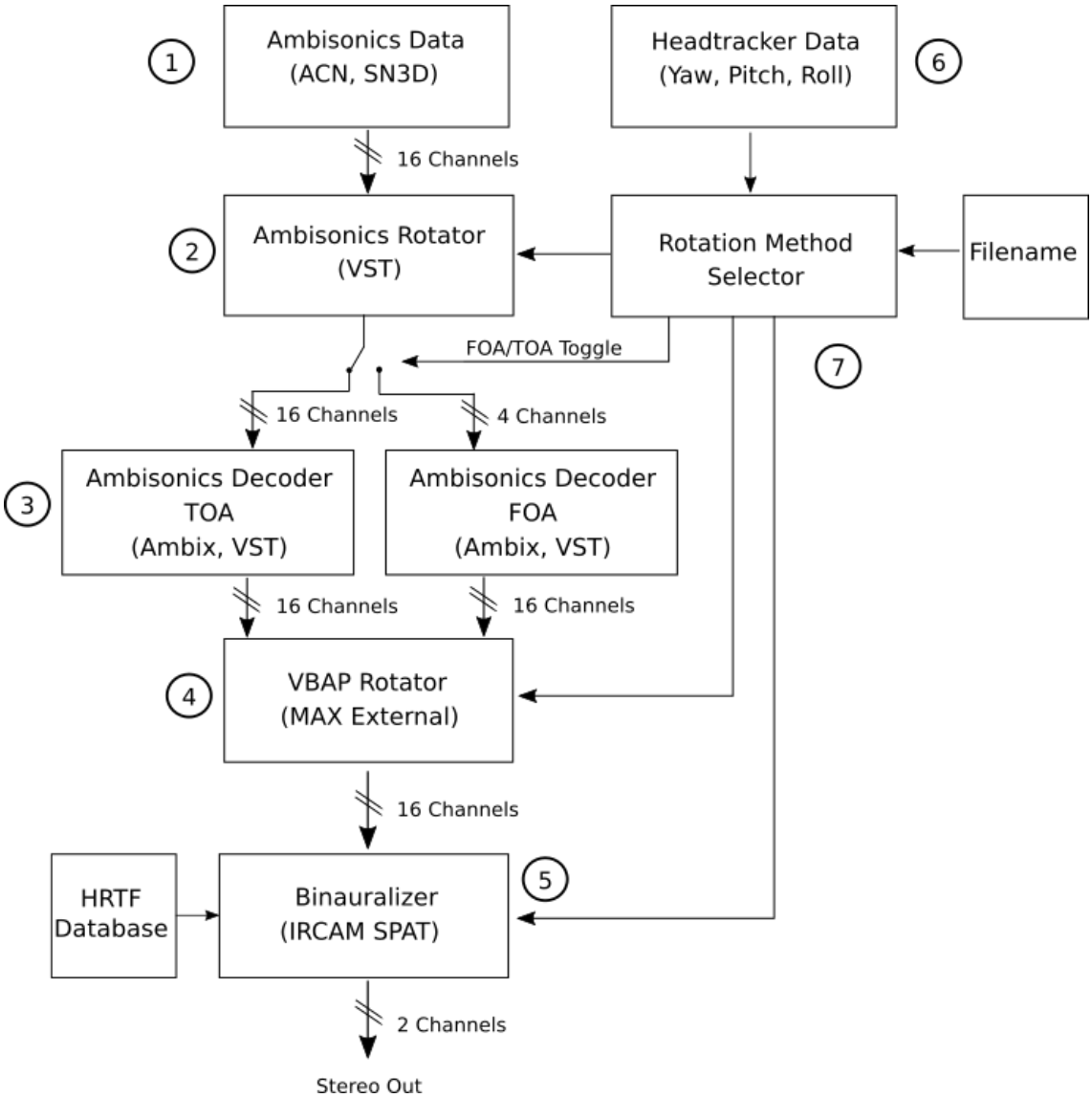

Abbildung A.1: Flussdiagramm der Audiodatensignalverarbeitung

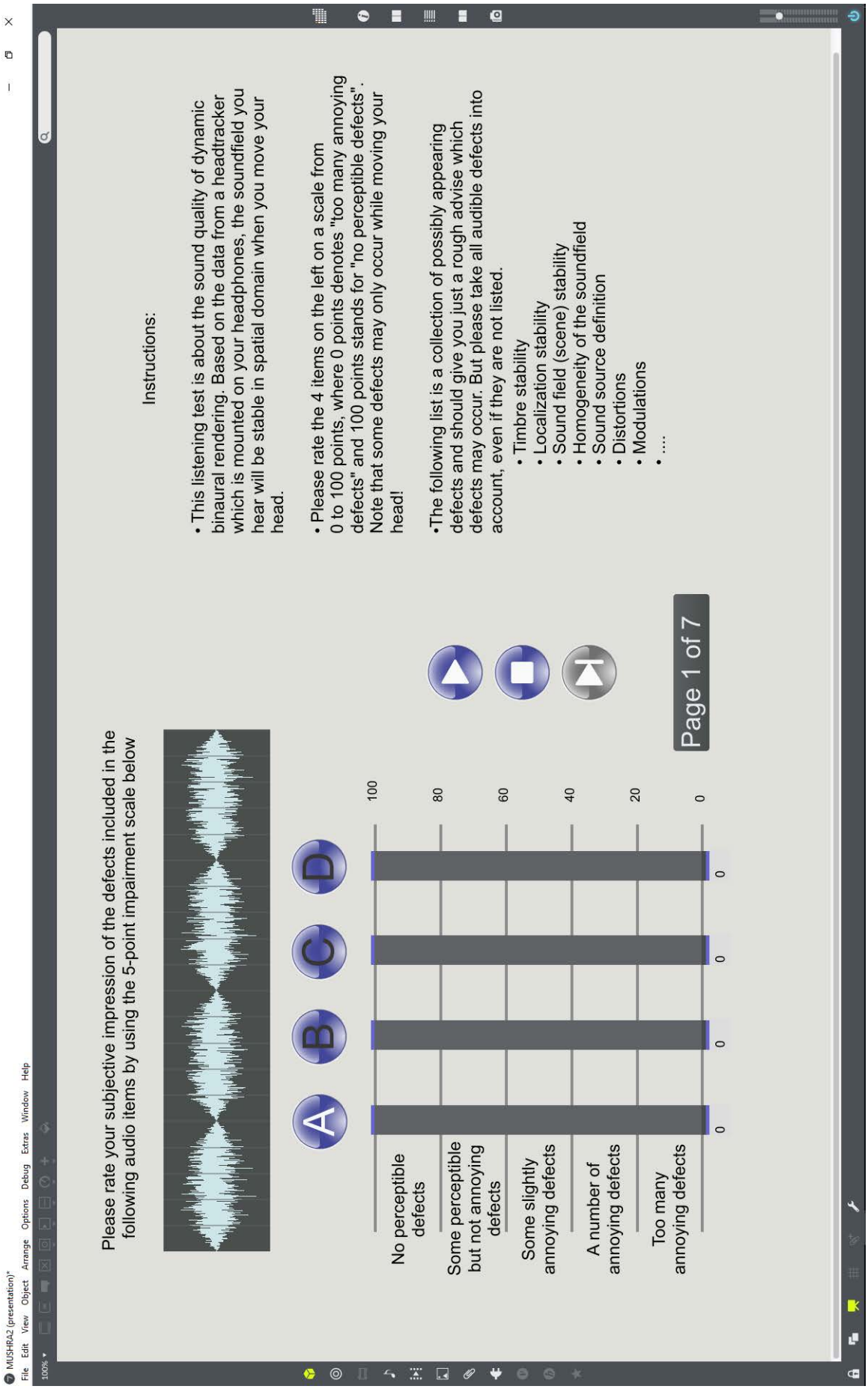

Abbildung A.2: Grafische Oberfläche des Hörversuchs

A.4 Hörversuch

Um Antworten auf die Forschungsfragen zu finden wurde ein Hörversuch durchgeführt. Der Hörversuch fand an drei aufeinanderfolgenden Tagen in einem Büroraum des Instituts statt. Insgesamt haben 20 Personen (2 davon weiblich, 18 männlich) im Alter von 20 bis 52 Jahren an dem Hörversuch teilgenommen, davon gaben 19 Probanden an, dass dies nicht ihr erster Hörversuch war. 10 der Probanden stuften sich selbst als Spezialisten für Raumklang ein.

Folgende Stimuli wurden in dem Hörversuch verwendet, die alle eine Länge zwischen 14 und 20 Sekunden hatten und im Ambix format (Nachbar et al., 2011) enkodiert waren:

Gepulstes weißes Rauschen Weißes Rauschimpulse mit einer Länge von $75ms$ und jeweils $75ms$ stille dazwischen. Zwischen den Pulsen und der Stille wurde sanft ein- und ausgeblendet.

Rosa Rauschen Vier pulse mit rosa Rauschen mit einer jeweiligen Länge von fünf Sekunden. Am Anfang und Ende eines jeden Pulses wurde jeweils ein- und ausgeblendet.

Stadion Eine reale Aufnahme aus einem Stadion welches mit dem Eigenmike Schallfeld Mikrofon aufgenommen und für diesen Test vom Institut zur Verfügung gestellt wurde.

Helikopter Ein Helikopter welcher über dem Probanden von links nach rechts fliegt. Ebenfalls mit dem Eigenmike Mikrofon aufgenommen und vom Institut zur Verfügung gestellt.

Aurelia Ein männlicher Sänger mit Instrumentalbegleitung. Von einem professionellem Toningenieur abgemischt.

Männlicher Sprecher Ein Text vorgetragen von einem männlichen Sprecher auf Englisch.

Gitarre Ein Gitarrenstück, ebenfalls von einem professionellem Toningenieur abgemischt.

Für den Hörversuch wurde ein HP Elitebook Notebook mit 16GB RAM und einem Intel Core i7-6500 dual-core 2.5 GHz Prozessor verwendet. Als Audiointerface diente ein *RME Madiface Pro*, welches mit einem *STAX Studio Monitor* Verstärker verbunden war an dem die Diffusfeldentzerrung aktiviert war. *STAX Lambda Pro New* dienten als Kopfhörer.

Bei dem Hörversuch saß der Proband an einem Schreibtisch auf dem sich Tastatur, Maus und ein Computerbildschirm, auf dem die Testsoftware lief, befanden. Der Proband wurde gebeten einen kurzen Fragebogen auszufüllen um einige persönliche Daten abzufragen. Danach wurde ihm erläutert, dass es sich um einen Hörversuch mit Raumklang handelt und seine Kopfbewegungen ausgeglichen werden - das wahrgenommene Klangfeld sich also nicht mitbewegt. Dem Probanden wurden die Kopfhörer angezogen, der Headtracker kalibriert und der Experimentator verließ den Raum. Ab diesem Moment hat der Proband nur noch mit der Testsoftware interagiert. Dort wurden ihm auf der rechten Seite einige Anweisungen eingeblendet. Auf der linken Seite befanden sich vier Schieberegler mit denen der Proband eine Bewertung abgeben konnte. Die Reihenfolge der Stimuli war randomisiert, ebenso wie Zuweisung der vier verschiedenen Rotationsmethoden zu den einzelnen Schiebereglern. Der Proband konnte selbstständig die Audiowiedergabe starten und pausieren und seine Bewertungen entweder mit der Maus oder der Tastatur abgeben. Erst wenn für jede Rotationsmethode eine Bewertung abgegeben wurde, konnte der Proband zum nächsten Stimulus wechseln. Würden alle sieben Stimuli bewertet war der Versuch abgeschlossen.

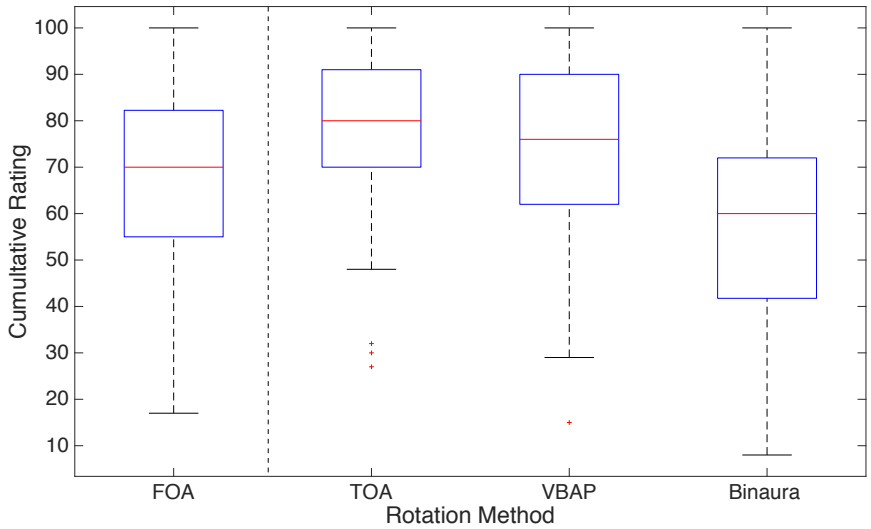

Abbildung A.3: Boxplot mit allen Bewertungen für die 4 Rotationsmethoden

A.5 Ergebnisse und Statistik

Insgesamt nahmen 20 Probanden an dem Hörtest teil, wobei die Ergebnisse eines Probanden nicht in die Auswertung eingeflossen sind, da dieser nach eigener Aussage von dem Hörtest überfordert war.

Zur schnellen Visualisierung wurde ein Boxplot A.3 erstellt. In einem Boxplot markiert jede Box den Bereich in dem 50% der Daten liegen. Die fettgedruckte Linie innerhalb jeder Box repräsentiert den Median. Die Linien, ausgehend von der Box markieren die so genannten *Whiskers* die eine maximale Länge von dem 1,5-fachen der Box haben.

Um zu überprüfen ob die gewonnenen Daten normalverteilt sind wurden *Quantil-Quantil-Diagramme* erstellt. Befinden sich die Datenpunkte entlang der rot markierten Linie, liegt Normalverteilung vor. Die *Quantil-Quantil-Diagramme* sind in Abbildung A.4 dargestellt.

Bevor eine Varianzanalyse durchgeführt wird, empfiehlt es sich die Varianzen der erhobenen Daten zu betrachten, da die Varianzanalyse gleiche Varianzen zwischen den Gruppen voraussetzt. Dazu wurde ein *Levene Test* durchgeführt, der die Gleichheit der Varianzen nicht bestätigte.

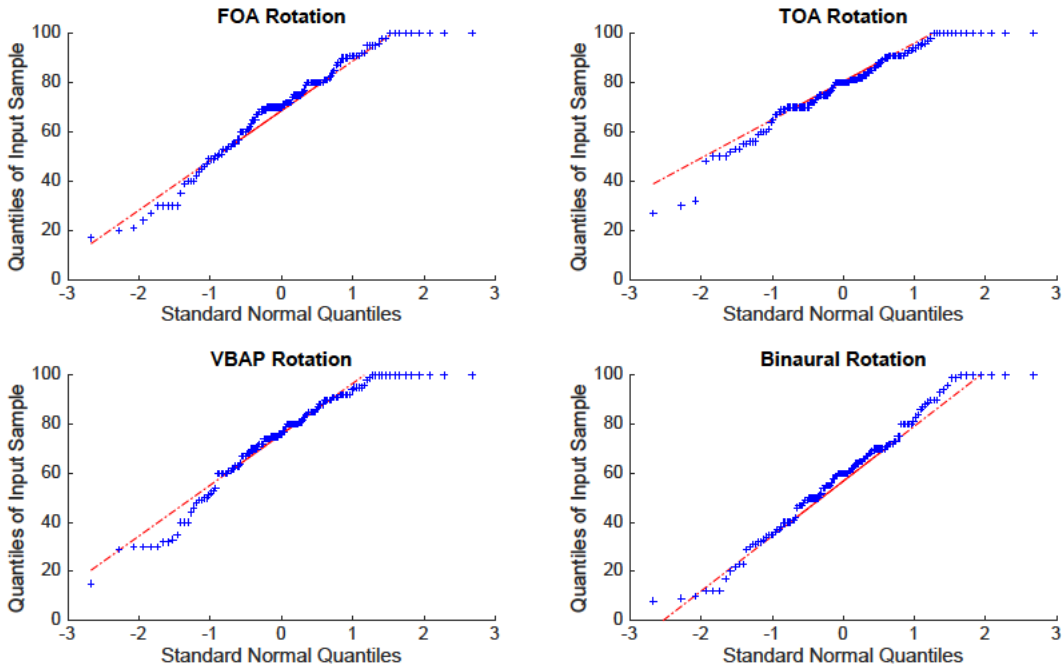

Abbildung A.4: Quantil-Quantil-Diagramme der Ergebnisse

Da jedoch die Anzahl der Datenpunkte in diesem Experiment je Gruppe gleich ist kann, sollte die Varianzanalyse robust gegen Ungleichheiten der Varianzen sein (Rogan and Keselman, 1977). Die einfaktorielle ANOVA zeigte statistisch signifikante Unterschiede zwischen den Gruppen ($F(3,528) = 23.24$, $p < .01$).

Um zu überprüfen zwischen welchen Gruppen signifikante Unterschiede bestehen wurden t-tests durchgeführt. Eine Bonferronikorektur senkte das Signifikanzniveau α welches ursprünglich auf 0,05 gesetzt wurde auf einen neuen Wert von $\alpha = 0{,}05/6 = 0{,}0083$.

Um auch die Effektstärke zu quantifizieren wurde *Cohen's d* berechnet (Cohen, 1988, p. 20). Nach (S. Sawilowsky, 2009) steht dabei ein Wert von 0,2 für eine kleine Effektgröße, ein Wert von 0,5 für einen mittleren Effekt, 0,8 ein großer Effekt und 1,2 für einen sehr großen Effekt. Die Effektgröße für den Vergleich der TOA Rotationsmethode mit der binauralen Methode betrug $d = 1$ und hat damit einen großen Effekt. Mittlere Effekte konnten zwischen den beiden Ambisonics Methoden, zwischen der FOA und der binauralen Methode, sowie zwischen der VBAP und binauralen Methode beobachtet werden.

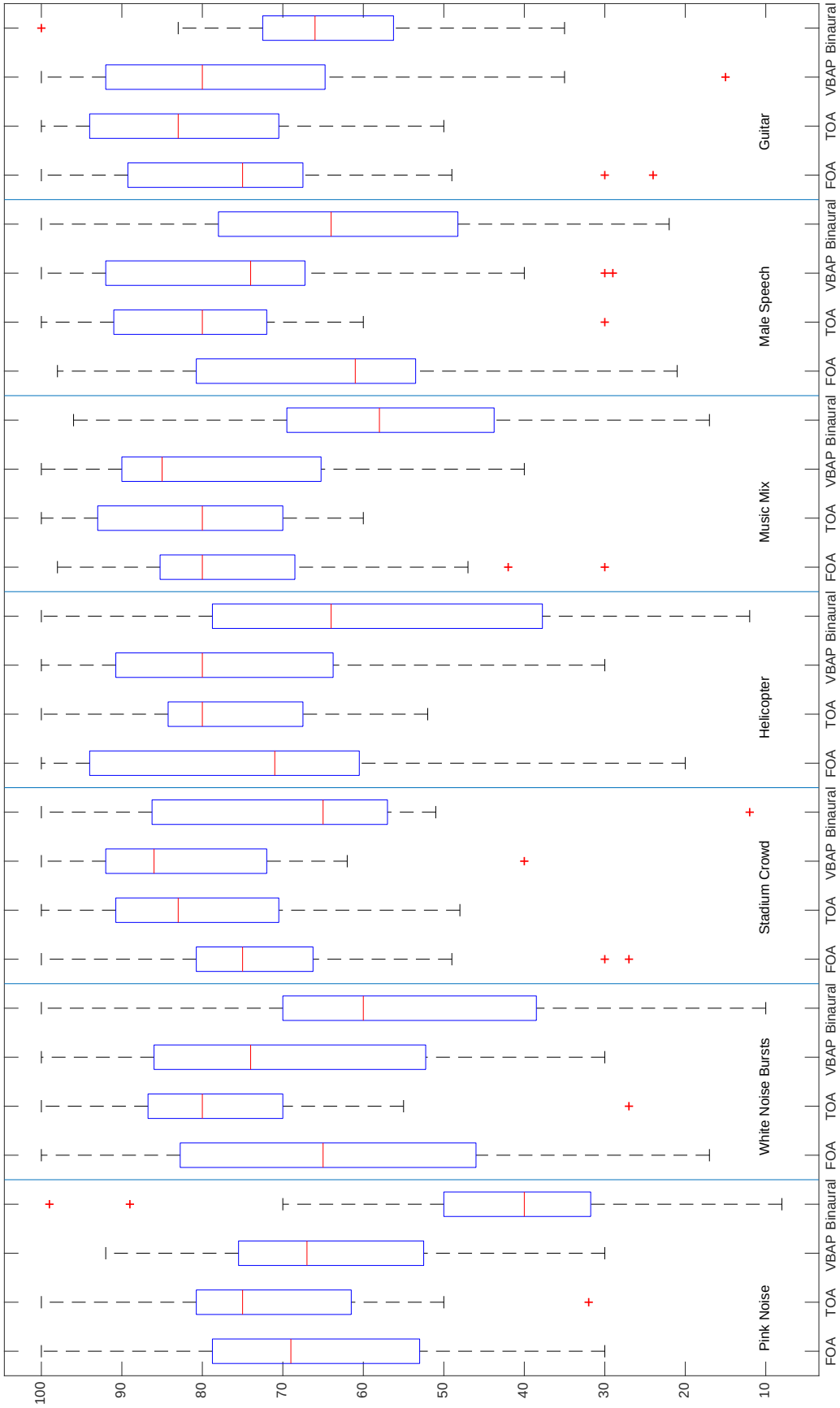

Abbildung A.5: Boxplots über alle Rotationsmethoden und Stimuli

A.6 Diskussion

Das durchgeführte Experiment diente dazu, hörbare Unterschiede zwischen verschiedenen Methoden zur Schallfelddrehung zu untersuchen. Die Resultate zeigen, dass die Drehung welche direkt im Ambisonicsraum durchgeführt wird, die besten Ergebnisse liefert. Die binaurale Methode, die auf dem Austausch von HRTF-filtern beruht wurde am schlechtesten bewertet. Die Probanden waren auch in der Lage zwischen Ambisonicsquellen erster und dritter Ordnung zu unterscheiden.

Die Erhobenen Daten weißen hohe Varianzen auf. Die Varianzen könnten durch Verbesserungen am Testdesign eventuell verkleinert werden.
Manche Probanden berichteten nach dem Test, dass es ihnen schwer fiel, Fehler durch die Schallfelddrehung herauszuhören. Daher wäre es zum Beispiel ratsam die Probanden vor der Durchführung des eigentlichen Versuches zu trainieren. Dies könnte dadurch geschehen, dass man ihnen eventuell auftretende Fehler der Klangfelddrehung vorspielt um sie so zu sensibilisieren.
Eine andere Möglichkeit wäre es den Probanden ein bimodale Umgebung zu präsentieren. Larsson et al. (2002) fand heraus, dass Probanden mehr fokussiert sind und bessere Ergebnisse liefern, wenn ihnen nicht nur der akustische Reiz, sondern auch ein passender visueller Reiz präsentiert wird. Hierbei ist jedoch Vorsicht geboten, da ein unpassender visueller Reiz die Ergebnisse negativ beeinflussen könnte.

Betrachtet man den VBAP Algorithmus, so hat Pulkki and Lokki (1998) herausgefunden, dass die Lokalisation von Schallquellen verbessert wird und Klangverfärbungen abnehmen, wenn man die Anzahl der Lautsprecher erhöht. Es ist auch bekannt, dass die wahrgenommene Breite einer Schallquelle von ihren Abstand zu den Lautsprechern abhängt (Pulkki, 1999).

Zusammenfassend lässt sich festhalten, dass die Schallfeldrotation durch den Austausch von HRTF-filtern zu hörbaren Artefakten führt. Rotiert man ein Schallfeld

mittels des VBAP Algorithmus, so treten weniger Artefakte auf. Jedoch ist die Leistung des VBAP Algorithmus abhängig von Anzahl und Position der Lautsprecher. Der optimale Weg um in Ambisonics codierte Schallfelder zu drehen ist die Drehung direkt im Ambisonicsraum.