FP7-ICT-2013-C TWO!EARS Project 618075

Deliverable 1.1

First Database of Audio-Visual Scenarios

WP 1 *

November 30, 2014

* The TWO!EARS project (`http://www.twoears.eu`) has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 618075.

Project acronym:	TWO!EARS
Project full title:	Reading the world with TWO!EARS

Work package:	1
Document number:	D1.1
Document title:	First database of audio-visual scenarios
Version:	1.0

Delivery date:	30. November 2014
Actual publication date:	01. December 2014
Dissemination level:	Restricted
Nature:	Other

Editor(s):	Sascha Spors
Author(s):	Fiete Winter, Sascha Spors, Hagen Wierstorf, Ivo Trowitzsch, Ning Ma, Tobias May, Alexander Raake
Reviewer(s):	Jonas Braasch, Dorothea Kolossa, Bruno Gas, Klaus Obermayer

Contents

1 Executive summary

The acoustic signals at the ears serve as input for the auditory scene analysis performed by the human auditory system. The same holds for the human visual system where the eyes provide the input. The goal of the Two!Ears project is to develop an intelligent, active computational model of auditory perception and experience in a multi-modal context. The model relies mainly on the auditory sense but also considers the visual sense for multimodal integration. The synthesis of ear signals and eye images is an important basis for the development and evaluation of the model. The synthesis allows to generate reproducible conditions in contrast to the input in a more or less controllable real-world scenario.

The definition of application scenarios is a versatile tool for a goal-oriented development of the model. It allows for a step-wise testing and validation of the models performance against physical and perceptual labels. A series of scenarios has been negotiated amongst partners. It should be noted that this document does not contain the formal definition of the audio-visual scenarios. The scenario structure is specified in D 6.1.1, where a first set of scenarios is provided as well. This reflects the general concept of Two!Ears: Scenarios for operation represent the top-down initialization of the model. The acoustic and visual information provided to the model at the lowest level (WP1) is selected respectively, using information as specified in this Deliverable D 1.1.

A framework for the synthesis of ear-signals has been implemented and integrated into the overall architecture of the project. This is documented in Section 3. It features dynamic rendering of the ear signals for most of the defined scenarios. The framework has been made available to the community as an open source software. A visual simulation framework is work in progress. It is noted that the simulation framework outlined in this D 1.1 only applies to the development system, where synthetic signals are provided to the auditory and visual processing stages of the Two!Ears model. The robotics system development is carried out in WP5 and its current state described in D 5.1.

The synthesis of ear-signals requires data characterizing the environment. A database of measurements and labels has been compiled for this purpose. The Two!Ears database provides seamless access to the data required for the rendering of audio-visual scenarios and also includes physical and perceptual labels for validation. The database itself additionally represents a contribution to the scientific community.

A hybrid infrastructure separating the publishable, open source licensed content from the restricted, project internal data has been chosen. Files of the database can be accessed via a software interface downloading files on demand while executing the model. An XML file format has been designed for the description of audio-visual scenes and provides physical ground-truth labels. Database infrastructure and file format standards are described in Section 4.

The current content of the database is documented in Section 5. While several impulse response data sets from other publicly available databases have been included, impulse responses of virtual sound sources created with spatial audio rendering techniques are also present. A collection of sound files consisting of speech and general sound stimuli has been added. The database also contains recordings of environmental noise. The mentioned general sound stimuli have been annotated with perceptual on- and offset times. Furthermore, results from hearing experiments including spatial and timbral perception for different sound reproduction techniques are part of the database.

2 Introduction

The acoustic signals at the ears serve as input for auditory scene analysis performed by the human auditory system. The same holds for the human visual system where the eyes provide the input. The perceptual model developed in Two!Ears relies mainly on the auditory input but also considers visual input for multimodal applications. The synthesis of ear signals and eye images is an important basis for the development and evaluation of the model. The synthesis allows reproducible conditions to be generated in contrast to the input in a more or less controllable real-world scenario. Figure 2.1 illustrates the integration of the audio-visual simulation framework into the Two!Ears architecture. The blue block at the lower left hand side constitutes the simulation framework. Its audio output is connected to the robot interface which provides the interface to the model. The head-orientation and location of the virtual listener is controlled via the robot interface by the upper stages of the model. The visual simulation framework, in contrast, is connected via the robot operating system (ROS). The ROS itself is connected to the model by the MATLAB bridge and the robot interface, described in more detail in Deliverable D 5.1. It is noted that the audio and visual simulations are seperate components generating virtual environments for the Two!Ears model. The two "worlds" are combined only in the "mind" of the model.

This choice had to be made due to the technical complexity of a unified simulation. However, it well reflects the Two!Ears concept of a modular perceiving model, where the incoming physical information is analyzed in terms of perception and meaning extraction.

The model can also be connected to the microphones and cameras of the robot for a proof of concept in real world scenarios. Here both modalities, audio and vision are interfaced by the ROS, its MATLAB bridge and the robot interface. The model controls the movements of the robot through the robot interface and the MATLAB bridge to the ROS, as described in Deliverable D 5.1.

The next Section describes the audio-visual simulation framework developed in work package 1. The synthesis of ear signals requires data characterizing acoustic environments. The data structures used for this data are detailed in Section 4. The database itself is a growing repository of audio, visual and perceptual data. Its current contents are given in Section 5. In order to facilitate the development process and the systematic evaluation

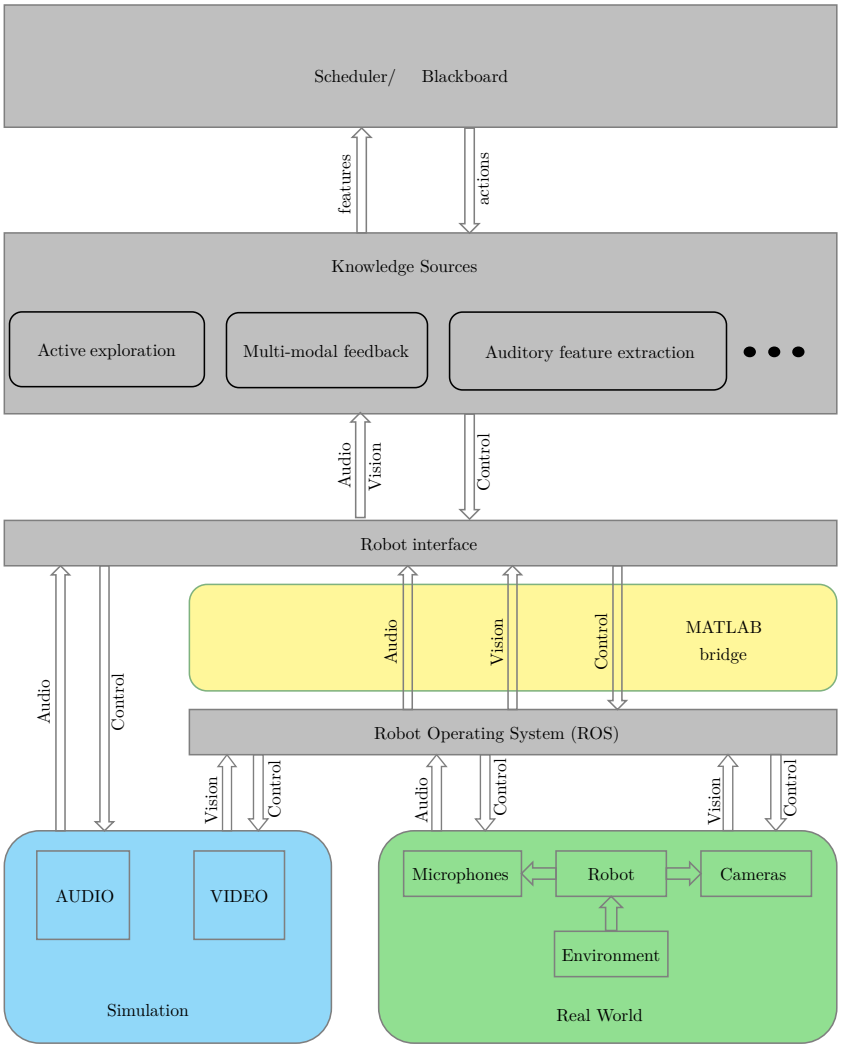


Figure 2.1: Architecture of audio-visual simulation framework.

of the model, a scenario-based approach as well as a first series of scenarios has been specified between all the partners. The current state of these scenarios is documented in Deliverable D 6.1.1.

3 Simulation Framework

The following section provides a brief overview on the methods applied in Two!Ears for synthesizing ear-signals. Each technique has particular benefits and weaknesses which are summarized in Section 3.1.6. Depending on the task, one particular technique or a combination is used for the synthesis of ear signals. Section 3.2 reports on the technical details of the audio simulation framework. The simulation of visual stimuli is reviewed in Section 3.3.

3.1 Techniques for Binaural Synthesis

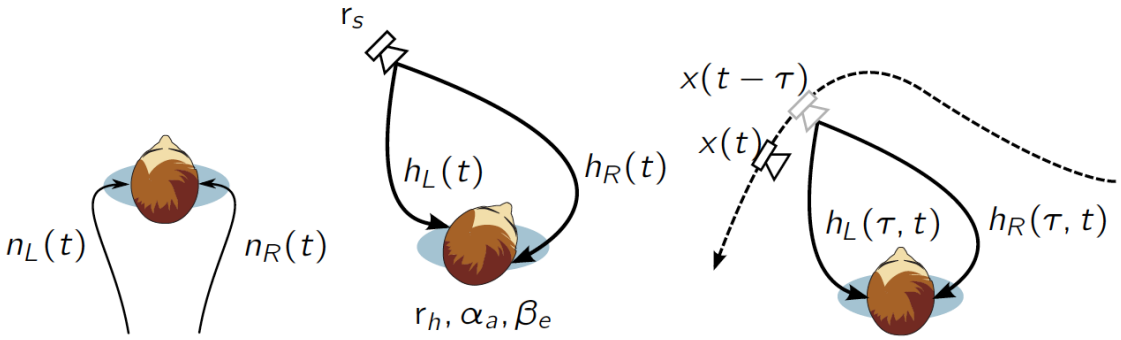
Binaural synthesis refers to the synthesis of the sound pressure at a defined position in the ear-canal. Often, the sound pressure at the blocked entrance of the ear canal is used as reference. This is also the reference assumed in Two!Ears.

3.1.1 Pre-Recorded Binaural Signals

A straightforward approach is to *record* the ear signals. The recording can either be performed by placing small microphones at a defined position in the ear canal of a listener. Intuitively the microphone should be placed close to the eardrum. However due to the medical risks involved in such a procedure, binaural signals are often captured at the blocked entrance of the ear canal. This has proven to work well in practice. As alternative to a human listener such recordings can also be performed by a Head and Torso Simulator (HATS).

The synthesis of pre-recorded binaural signals is then performed by playing back the recorded signals. This is illustrated in Figure 3.1a. A (free-field) compensation of the microphones or HATS frequency response might be required to compensate for their influences.

Any sound field can be captured. This also includes diffuse sound fields and moving sound sources. However, the head orientation is fixed by the recording and cannot be changed during synthesis. The Two!Ears model is a binaural model which includes feedback



(a) pre-recorded binaural signals **(b)** static binaural room impulse responses **(c)** dynamic binaural room impulse responses

Figure 3.1: Techniques for binaural synthesis applied in TWO!EARS. Left/right binaural signals are denoted by $n_{\{L,R\}}(t)$, BRIRs by $h_{\{L,R\}}(t)$, time-variant BRIRs by $h_{\{L,R\}}(\tau, t)$. The source signal is denoted by $x(t)$, the source and receiver position by r_s and r_h, respectively.

mechanisms and active listening. As a consequence, the synthesis of pre-recorded binaural signals is only of interest for diffuse background noise where the head orientations does not play a role. This is summarized in Table 3.1.

WP 1 has reviewed existing databases of pre-recorded binaural signals and collected them in the central database. See Chap. 5 for a listing of the material included so far. The simulation framework for the computation of ear signals allows the use of pre-recorded binaural signals. For a detailed description refer to Section 3.2.

3.1.2 Static Head-Related and Binaural Room Impulse Responses

Under the assumption of time-invariant linear acoustics, the transfer path from a sound source to the ears can be characterized by impulse responses. Under free-field conditions these are termed as Head-Related Impulse Responses (HRIRs) and in reverberant environments as Binaural Room Impulse Responses (BRIRs). HRIRs depend on the source position with respect to the listener and on the head orientation. BRIRs depend additionally on the position and orientation of the source and the listener in the environment. In order to capture these degrees of freedom, databases of HRIR/BRIRs have to be captured. For HRIRs typically the incidence angle of the source for a fixed distance is varied. For BRIRs the head orientation is varied for a fixed source and listener position.

The synthesis of ear-signals is performed by convolving the desired source signal with the appropriate left/right HRIR/BRIR from the database. This is illustrated in Fig-

ure 3.1b. A change in head-orientation can be considered by exchanging the set of HRIR/BRIRs.

The use of HRIR/BRIRs is restricted to compact sound sources which do not move. Diffuse sound fields can be approximated by superposition of many source from different directions. Moving sound sources can be approximated by concatenating a series of static source positions. However, this is typically not possible for BRIRs since this would require a densely captured grid of source positions. A change in head-orientation can be modeled by exchanging the HRIR/BRIRs. Hence, static HRIR/BRIRs are used for the simulation of single sources. The properties of this technique are summarized in Table 3.1.

WP 1 has reviewed existing databases of HRIR/BRIRs, converted and collected them in the central database. See Chap. 5 for a listing of the material included so far. The simulation framework for the computation of ear signals allows the use of HRIR/BRIRs for the simulation of sound sources. For a detailed description refer to Section 3.2.

3.1.3 Dynamic Binaural Room Impulse Responses

The transfer path from a moving sound source to the ears can be characterized by time-variant impulse responses. The identification of such impulse responses requires specific signal processing techniques that cope for the time-variance of the underlying linear system. Time-variant BRIRs can capture the movement/orientation of a sound source on a fixed trajectory for a fixed head-orientation in a reverberant environment. Alternatively, the time-variance of an environment can be captured for a fixed source and head position/orientation. Different head-orientations could be captured if the source trajectory were exactly reproducible. However that is hard to realize in practice.

The synthesis of ear-signals is performed by time-variant convolution of the desired source signal with the time-variant BRIRs, as illustrated in Figure 3.1c. The speed at which the sound source moves along the trajectory can be modified by subsampling or interpolation of the time-variant BRIRs.

The time-variant BRIRs contain the fine structure of the room which is excited by the moving sound source. All effects due to the moving source, e.g. the Doppler effect, are included. In practice, only a fixed head-orientation and fixed trajectory can be considered. This technique is suitable for the accurate synthesis of moving sound sources in rooms. The properties are summarized in Table 3.1.

In the past, time variant system identification techniques have been applied to the measurement of spatially continuous HRIRs [1]. WP 1 has extended this by considering moving sound sources and time-variant acoustic environments. This required an advancement of

the existing techniques [2, 3]. Various simulations and measurements have been performed to evaluate the technique in synthetic and real environments. Figure 3.2 shows time-variant BRIRs for a moving scatterer which obstructs the direct path between the loudspeaker and HATS used for the measurement. This constitutes a time-variant acoustic scenario. The effects caused by the scattering are clearly visible in the BRIRs. Future work includes the optimization of various algorithmic parameters, as well as the inclusion of the method into the simulation framework described in Section 3.2.

3.1.4 Data-Based Binaural Synthesis

The binaural synthesis techniques above cannot cope for translatory movements of the listener in reverberant environments. Small-scale translatory movements are considered to provide additional localization cues. Data-based binaural synthesis allows to compute ear-signals for translated listener positions. It bases on a directional analysis and translation of the reverberant sound field [4, 5]. The sound field captured by a (spherical) microphone array is decomposed into plane waves using (modal) beamforming. With respect to the origin, plane waves can be shifted in space by applying a spatial phase shift. The shifted plane waves are then filtered by the respective HRIRs and summed up for auralization. The

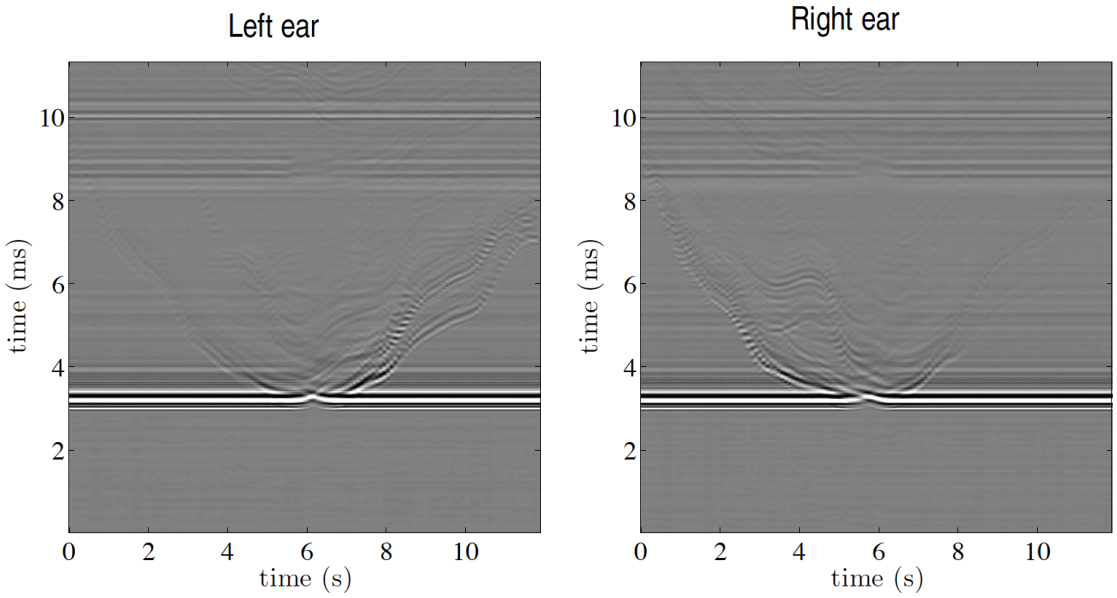


Figure 3.2: Dynamic BRIRs captured in a typical office room using a static loudspeaker and HATS position. The horizontal axis denotes the time t at which a BRIR has been captured. The vertical axis shows the time axis τ of the BRIR. A vertical slice represents a BRIR at a given time. A moving scatterer is obstructing the direct path between loudspeaker and HATS at $t \approx 6$ s.

overall procedure is illustrated in Figure 3.3. The technique also allows individual HRIRs to be used. As alternative to the procedure outlined above, Multi-Channel Room Impulse Responses (MRIRs) from a source to the microphone array can be used as input. The result are the BRIRs for a given listener position which can then be used for auralization using convolution by a source signal.

Data-based binaural synthesis allows to consider small-scale translatory movements of the listener around a fixed location. All physical aspects of a reverberant sound field are included. The accuracy depends, amongst others, on the number of microphones used and their self-noise. The properties of the technique are listed in Table 3.1.

The basic processing for data-based binaural synthesis has already been published by URO. The perceptual properties in terms of localization for various technical parameters have been evaluated in WP 1 and published [6]. The implementation into the simulation framework described in Section 3.2 is work in progress.

3.1.5 Numerical Simulation of Acoustic Environments

The numerical simulation of acoustic environments has been a very active field of research for several decades. Besides the numeric solution of the wave equation, various approximations have been developed. These approximations are based on the assumption that sound propagation can be modeled by rays, which is reasonable if the dimensions of the considered

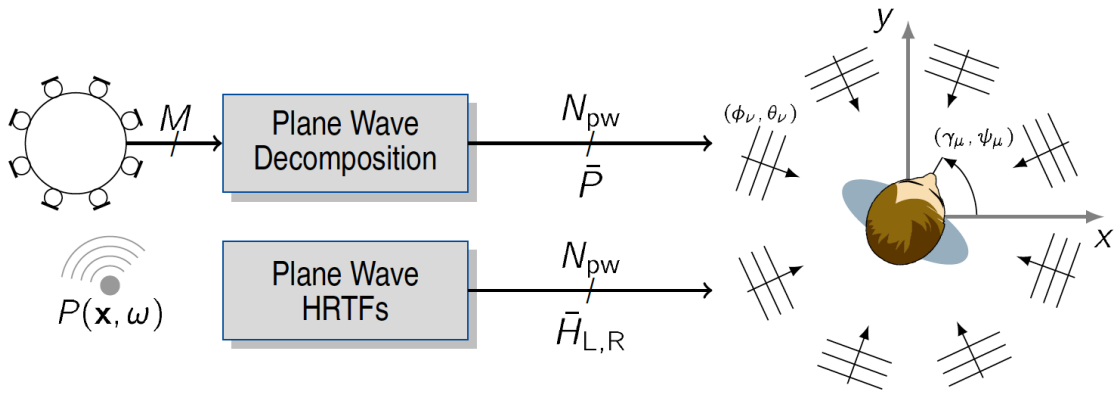


Figure 3.3: Data-based binaural synthesis including small scale translatory movements of the listener. The number of microphones is denoted by M, the number of plane waves by N_{pw}, the coefficients of the plane wave decomposition by $\bar{P}$, the left/right far-field HRTFs by $\bar{H}_{\mathrm{L,R}}$. The incidence angles of the plane waves is denoted by (ϕ_ν, θ_ν), the look directions of the listener by (γ_μ, ψ_μ).

objects are large compared to the wavelength. State-of-the art simulation software combines a mirror-image source model for the direct sound and early-reflections with energetic methods (e.g. ray tracing) for the late reverberation. Hence, diffraction and standing waves are often neglected or only approximated to some limited degree. Another source of inaccuracies is the handling and determination of boundary conditions. For an accurate numerical simulation, the incidence- and frequency-dependent reflection properties of all acoustically relevant materials for a given environment are required. In practice these are only known to a limited degree. Most of the available software packages allow BRIRs to be computed for a given environment.

The numerical simulation of acoustic environments allows BRIRs to be computed for a given head-orientation and listener position. These BRIRs can then be used to compute ear-signals by convolution with a desired source signal. However, the physical accuracy of such simulations is limited due to the underlying approximations and material properties. It is not clear how relevant acoustic cues, e.g. for localization, are preserved. The properties of the technique are listed in Table 3.1.

WP 1 has screened most of the available commercial and non-commercial simulation frameworks for the application in Two!Ears. Besides accuracy, the software-based control of source and listener position/orientation is a major selection criterion, due to the intended exploratory nature of the Two!Ears model. The simulation framework RAVEN of the Institute of Technical Acoustics, RWTH Aachen [7] features a MATLAB interface to control these and other parameters. This allows for a seamless integration into the ear-signal simulation framework. The integration of RAVEN is work in progress.

3.1.6 Comparison of Binaural Synthesis Techniques

Table 3.1 summarizes the properties of the different binaural synthesis techniques discussed above and their application in the Two!Ears simulation framework.

technique	diffuse sound field	moving sources	head-orientation	head-translation	realism	application
pre-recorded binaural signals	yes	yes	**no**	**no**	high	background noise
static HRIR/BRIRs	limited	limited	yes	limited	high	static sound sources
dynamic BRIRs	**no**	yes	**no**	**no**	high	moving sound sources
data-based synthesis	yes	**no**	yes	small-scale	high	static sound sources
numerical simulation	limited	yes	yes	yes	limited	acoustic navigation

Table 3.1: Comparison of binaural synthesis techniques used in Two!Ears.

3.2 Synthesis of Ear Signals

A binaural simulation framework has been implemented in order to appropriately address the defined scenarios and to provide a feasible testbed for an efficient development of the auditory model. It provides synthesized ear signals of virtual acoustic environments and supports the active explorative feedback, as it is defined in the model architecture. For model training and validation purposes, various instances of the same scenario can be realized in a straight-forward manner.
The software architecture of the binaural simulator is depicted in Figure 3.4. The open-source software SoundScape Renderer (SSR) [8] is used as the core of the simulation tool (red box). While the core is written in C++, the rest of the framework is implemented in MATLAB in order to ensure compatibility to the software of the auditory model. A detailed explanation of individual system components is given in the following sections.

3.2.1 SoundScape Renderer

The SSR is a tool for realtime spatial audio reproduction, supporting the (dynamic) binaural synthesis of spatial audio scenes. For free-field scenarios it utilizes a set of a-priori measured HRIRs to recreate the acoustic signals at the ears of the listener. The sound reproduction is implemented by filtering a dry source signal with the appropriate HRIR for the left and the right ear. These HRIRs are selected by the position of the sound source relative to the listener's head. Different source distances are obtained by adapting the source's sound volume and delay. Dynamic scenes and listener movements can be addressed due to the frame-based signal processing framework[9] of the SSR. This allows for cross-fading the HRIRs in each time frame in order to simulate dynamic scenes or head

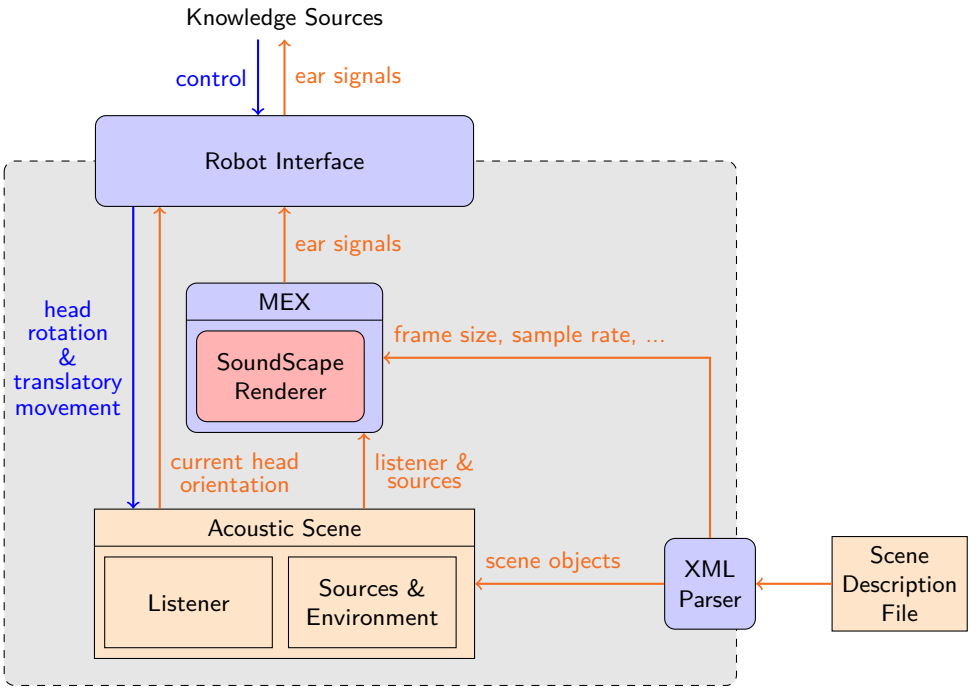


Figure 3.4: Software architecture of the binaural simulator

motion.

The binaural synthesis of reverberant environments is handled using an individual set of BRIRs for each sound source. Since the respective source position and the room acoustics are already encoded in the impulse responses, scene dynamics are limited to the listener's head rotation. In addition, this rendering technique can be used to binaurally synthesize the sound reproduction of loudspeaker systems by taking the BRIRs of each loudspeaker into account.
The SSR can be controlled and configured via a MATLAB executable (MEX) interface (red and blue box, see fig. 3.4). While processing parameters like e.g. frame size and sample rate have to be defined initially and kept constant during a simulation run, properties of the listener's head and sound sources may be changed between each time frame.

3.2.2 Acoustic Scene

The acoustic scene (orange box, see fig. 3.4) is an ensemble of acoustic objects defined in a three-dimensional coordinate system. As a central object the *Listener* represents the human head and acts as an acoustic sensor with two channels for the left and right

ear.

Sound *Sources* are essentially characterized by their position, source signal and source type. Several binaural reproduction techniques (see sec. 3.1) are considered by a respective source type: Sources emitting a spherical or plane wave field in a non-reverberant environment are represented by *point* and *plane wave* sources, respectively. Binaural recordings of environments can be auralized via a two-channel *direct* source, while the plane wave decomposition (PWD) of multi-channel microphone array recordings are addressed by a multi-channel PWD source. The latter is implemented as group of plane wave sources, where each source is oriented and weighted according to the plane wave decomposition. The PWD source also allows for the simulation of spatially expanded noise sources.

Reverberant environments can be realized in two ways: As already shortly described in Section 3.2.1, each source may be endowed with a set of BRIRs. In this case, source properties like e.g. position and source type are not configurable, since all physical information is already provided by the BRIRs. Active exploration by translatory movement of the listener is not possible. As the second alternative, an image source model for rectangular [10] and arbitrary convex [11] rooms is implemented within the binaural simulator. It simulates the specular reflections of sound at each wall and takes multiple reflections up to a user-defined order into account. For this purpose *Wall* objects building a room have to be defined.

3.2.3 Configuration and Interfaces

For the general application case the simulation core and the acoustic scene (see sections 3.2.1 and 3.2.2, respectively) are encapsulated (grey box, see Figure 3.4) and can only be manipulated and controlled via specific interfaces. In the following, these three methods are presented:

Scene Description File: The XML Parsing functionality of the binaural simulation tool allows for the definition of the whole acoustic scene via an XML scene description file. This also includes events which manipulate a scene object at a certain point in time and describe the scene dynamics. For a detailed depiction of the file format the reader is referred to Chap. 4. The XML Parser is based on the built-in W3C Document Object Model (DOM) of MATLAB. Besides the acoustic scene, also processing parameters of the simulation core can be initialized via the description file.

Robot Interface: In order to encapsulate the simulation functionalities and its specifics to the upper auditory model layers, a simple interface has been designed which narrows the possibilities of manipulation down to the control mechanisms of a real

robot (blue arrows, see Figure 3.4). This strategy ensures compatibility between the robotics platform and the simulation. Binaural signals of a desired length can be requested by the upper model stages. Exploratory feedback is realized by head rotation and translatory movement, which implicitly modifies the head object of the acoustic scene.

Direct Configuration: For training and validation purposes of the auditory model, it is necessary to iterate over several parameter combinations and thus generate a large amount of training/validation scenarios. The definition of each scenario using the scene description file is not feasible. The *expert* user therefore has the possibility to circumvent the encapsulation and directly access the acoustic scene and control the simulation core.

3.2.4 Integration and Application

The simulation framework for the synthesis of ear signals described above has been fully integrated into the Two!Ears software framework. It is interfaced with the peripheral processing and feature extraction stage of work package two (WP2). The blackboard system, developed by WP3, acts as the core of the integrated framework and provides additional interfaces to include feedback mechanisms investigated within work package four (WP4) and a robotic interface to communicate with work package five (WP5). The listener position and head orientation is controlled by the robot interface. The integration is described in more detail in Deliverable D 3.2. The simulation framework for the synthesis of ear signals has been published under `https://github.com/TWOEARS/binaural-simulator`.

The simulation framework has been applied successfully in the first year of the project. The full integration of all work packages has already allowed significant scientific contributions in many tasks to be made. These are described for instance in the context of work packages 2.2 and 3.2 (D 2.2, D 3.2).

3.3 Simulation of Visual Stimuli

As the elements of the Two!Ears system that are concerned with feedback will be developed over a time span of almost two years, many of these elements are not immediately available for constructing and testing feedback procedures. This is particularly true for methods that deal with more abstract and/or more complex functions, like active exploration and multi-modal feedback methods. Both of these feedback paths will require the Two!Ears system to be endowed with sophisticated visual processing methods, e.g.,

for visual object detection/recognition, audio-visual speaker identification, or vision-based collision avoidance.

To be able to perform early feedback experiments, and test techniques for visual processing, we set up a virtual test environment (VTE) which mimics the feedback-relevant parts of the Two!Ears system and enables the visualization of simulated environments.

Our project partners can test their own feedback-related ideas in the virtual environment, take advantage of the visual data provided by the VTE and set up cooperation with WP4's feedback routines early. By that strategy, potential issues might be detected and eliminated long before final system assembly. While experimenting with the feedback loops in the virtual environment, environmental variates and labels will be identified that are definitely needed for reliable feedback, thus allowing for algorithmic streamlining.

A first VTE realized in the Two!Ears context is the Bochum Experimental Feedback Testbed (BEFT) [12], on which we report in the following. For completeness, note that sections 3.3.1 and 3.3.2 are directly correlated to excerpts from Deliverable D 4.1, part D.

3.3.1 Simulation Environment

BEFT integrates a virtual 3D visualization environment based on the OGRE 3D rendering engine [13], and hosts a mobile front end – currently a CAD model of the PR2 robot, respectively a personation of a Knowles Electronics Manikin for Acoustic Research (KEMAR) dummy head mounted on a rotational axis. The testbed allows further scene components to be to read in directly from XML files. This way, simulated entities like persons (e.g., victims), walls, terrains, and so on, can easily be added to a scenario.

The entities convey physical parameters, like distance and azimuth (w.r.t. the robot), or percentage of occlusion. Based on these parameters, BEFT simulates category labeling for each entity: "hand-crafted" degradation functions [12] weaken given a-priori knowledge of the entities' true categories in order to emulate, as closely as possible, uncertainty in category estimation caused by sensor noise or algorithmic issues.

According to the estimated parameters and the inferred category label, the virtual mobile front end can be actuated (e.g., via active exploration mechanisms, see Deliverable D 4.1, part D) in order to update and enhance the parameter/label estimates and sharpen the robot's internal world model.

Note that BEFT was intended to operate on the cognitive rather than on the signal level,

allowing for very early feedback testing and multi-modal analysis, by skipping Two!Ears signal processing and pre-segmentation stages that were "under construction" at that time. However, BEFT's 3D display capabilities and its abilities to handle robot control based on the ROS [14] middleware are clearly limited. The first issue might hamper simulation of visually challenging scenarios; the latter problem, however, would cause major re-work of algorithms tested in BEFT in order to port these methods to physical robot devices operating in real-world scenarios.

Figure 3.5: Modular OpenRobots Simulation Engine (MORSE) simulation of a KEMAR

As this is clearly unacceptable on the long run, the MORSE robotics simulator [15] will inherit from BEFT and become the standard VTE for visual simulation in Two!Ears. Note that MORSE is based on the BLENDER 3D modeling/simulation software [16], using **(author?)** [17] to enable physically plausible behavior of the simulated environment.

As MORSE has the ability to operate different types of robotic middleware, e.g., ROS, porting issues can be minimized by enabling the Two!Ears framework to control/read out the virtual robotic front-end (motors, cameras, microphones, etc.) using exactly the same methods that will be employed to control/read out the physical device in later project stages. Further, MORSE comes with multiple pre-defined industrial components (sensors, actuators, controllers, and robotic platforms) which can be assembled into complete robots using straightforward Python™ scripting. For a more detailed overview of MORSE and its integration into the Two!Ears framework, see Deliverable D 4.1, part D, and Deliverable

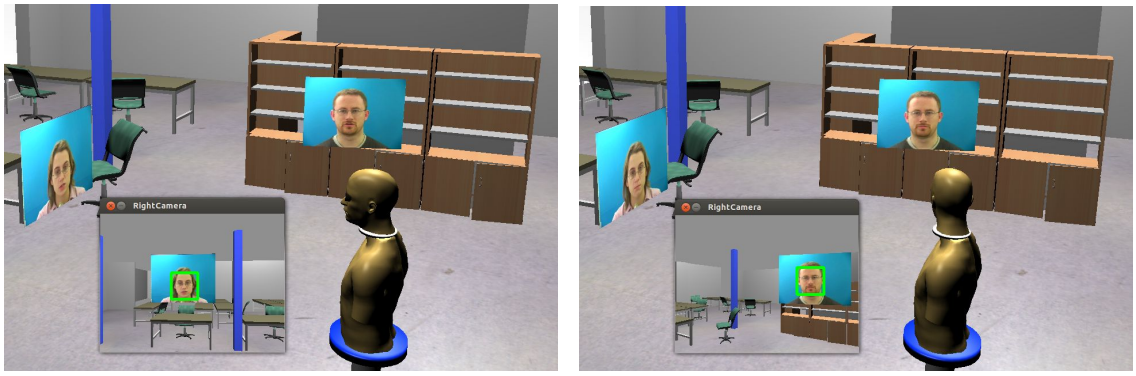
D 5.1.

3.3.2 Progress on Visual Processing

We set up a first proof-of-concept application to demonstrate visual processing capabilities in the MORSE environment: an emulated KEMAR-like head/torso combination is embedded in a standard MORSE scenario, see Figure 3.6. The artificial head is allowed to rotate freely and has two virtual cameras attached. Videos of human speakers are projected into the virtual environment using basic video texturing methods.

Note that the videos are currently chosen from the audio-visual corpus of **(author?)** [18]. One of the robot's cameras is connected to an external ROS node that was created using GenoM3 [19]. This external node is enabled to perform fast image processing based on the OpenCV library [20]: images from the virtual camera are streamed into the node and are then analyzed with OpenCV's face detection engine.

Figure 3.6 shows the results of face detection in MORSE: found faces are marked by green rectangles and the corresponding face regions could be propagated to higher system layers, e.g., to perform audio-visual speaker identification. Note that there are still some issues to be solved with respect to synchronization between the audio and the video data stream, see Deliverable D 4.1, part D for a more detailed discussion. Also, the above example application is not yet fully integrated into Two!Ears. To eventually become consistent with the framework's architecture, the face detection mechanism will have to be encapsulated in a knowledge source and communication with the visual processing ROS node has to be established via Two!Ears's robot interface (see Deliverable D 4.1, part D).



(a) **(b)**

Figure 3.6: Face detection in the MORSE simulator

4 Accessibility and Compatibility of the Database

The synthesis of ear signals requires data characterizing the acoustic environment to be simulated. Physical and perceptual labels are required to systematically evaluate the model performance. This data is collected in a central database for seamless access by the model. The infrastructure of the database and data formats are discussed in the following.

4.1 Infrastructure

4.1.1 Public and Project-Internal Databases

The Two!Ears database represents a compilation of existing data from other databases and recorded/measured data. Several databases are published under an open source license allowing their distribution in the context of Two!Ears. However, for the development, testing and validation of the model additional data is necessary whose access is restricted to the consortium members. With regard to the final public release of the database after 36 months it is plausible to separate both types of data from the beginning. A hybrid approach utilizing two databases is therefore used: All data is stored in a restrictive, project-internal repository, which is integrated in the project management web application Redmine[1]. The public database mirrors the open source licensed part of the data. Both databases can be accessed via a web interfaces or with the version control software git[2]. It allows for recovering old file versions and for an easy synchronization between the two databases. It has been decided amongst the project partners to make the public database available to the scientific community at an early stage of the project[3].

1 see `http://www.redmine.org`
2 see `http://git-scm.com`
3 see `https://dev.qu.tu-berlin.de/projects/twoears-database/repository`

4.1.2 Software Interface

For the dissemination among the scientific community and efficient development of the TWO!EARS model it is necessary to make access to the database as seamless as possible. While the version control support of git is a huge benefit, it is primarily designed for cooperative software development mostly storing text and source code files in a so-called remote repository on a server. Since the user needs all source code files in order to get the software working, git provides a straightforward interface to download the whole repository with one command. For the mentioned application case the file sizes are small and download time is acceptable. The download of a single file from the repository is however more complicated and in some cases not even supported by the server hosting the respective repository.
The TWO!EARS database mostly contains binary files of comparably big size. Even for model applications where only a few files are needed, the whole database would have to be downloaded. This might distract potential users from testing/using the TWO!EARS software framework. In order to overcome these limitations a MATLAB software interface has been designed to access the needed files on demand. It circumvents the necessity of git's single file download functionality and uses the web interface of the databases. Standard MATLAB functions[1] are used to access the remote data via the Hypertext Transfer Protocol (HTTP). The software interface provides a flexible file inclusion mechanism in order to prevent unnecessary downloads for the user. It takes different possible locations (in order of appearance) of the needed file into account:

1. Search the file relatively to the current working directory.
2. Search the file relatively to the root of the file system.
3. Search inside a local copy of the whole repository, which has been downloaded via git. The location of the local copy can be defined by the user using the software interface.
4. Search inside a temporary directory, which caches single files downloaded from the database before in order to prevent repeating downloads. The location of the temporary directory can be defined by the user using the software interface. If desired, the temporary directory can be cleared.
5. Try to download the file from the remote repository. The URL of the remote repository can be defined by the user using the software interface. The downloaded file will be stored inside the temporary directory.

1 `urlwrite`, see `http://www.mathworks.fr/help/matlab/ref/urlwrite.html`

4.2 Data Formats

4.2.1 Impulse Responses

For the auralization and rendering of acoustic scenes a-priori measured impulse responses play an important role. The acoustic transmission from a sound source to both ears of the human head can be described by HRIRs or their temporal Fourier transform called Head-Related Transfer Function (HRTF). The measurement of this transfer functions is usually done with a HATS, which models the human outer ears. While HRTFs imply free field conditions and an anechoic environment, BRIRs include reverberation caused by the obstacles and walls. In order to binaurally auralize a sound source emitting a certain stimulus, the respective impulse response is convolved with an anechoic recording of the stimulus. As a standard format for storing and utilizing such impulse responses, the Spatially Oriented Format for Acoustics (SOFA)[21] has been chosen. It represents the basis of the AES-X212 HRTF file format standardization project[22] and is therefore a suitable format for the Two!Ears project.
Besides HRTF and BRIR datasets, MRIRs and multi-loudspeaker BRIRs are supported by the format. MRIRs can be interpreted as a generalization of BRIRs using a multi-channel microphone array instead of a two-channel HATS. While BRIRs can be measured for different source position by moving the sound source inside the room, multi-loudspeaker BRIRs assume a static setup of loudspeakers and capture the impulse responses for each loudspeaker.

4.2.2 Audio-Visual Scene Description

The definition of a scene description format allows for a formal standardization of audio-visual scenes among all work packages and project partners. It can be interpreted as a physical ground truth and is therefore a useful tool for validating the auditory model against physical labels. It is also used for the configuration of the acoustic rendering tools in order to simulate these scenes appropriately. These tools are described in detail within the deliverable 6.1.1 report.
During the format's conception binary data containers, e.g, the Hierarchical Data Format (HDF) or the Network Common Data Form (NetCDF), have been considered as a possible basis for the format. They are designed to efficiently handle a large amount data in order to provide flexible access to it. However, special software tools are needed to access and modify the data inside these container files. Hence, the Extensible Markup Language (XML) has been selected as the suitable format for the scene description. It supersedes the mentioned alternatives due to its human readability and cross-platform compatibility. Files written in this format can easily be modified with any text editor. Large binary data, e.g. impulse responses (see Section 4.2.1) or audio stimuli, is referenced inside the scene

description file and stored separately.
The structure of the XML files is defined by a so-called XML Schema Definition (XSD), which is utilized for an automatic syntax validity check of the scene description files. An exemplary file is shown in Fig. 4.1. While general rendering parameters are defined as attributes of the `scene` element, scene entities like sound `source`s and `sink`s are characterized within child elements of the `scene`. An audio `buffer` may be added in order to define an input signal for the respective sound source. Basic support of additional visual simulation attributes, e.g. 3D mesh files, is provided. The dynamic part of the scene is described by the `dynamic` element consisting of several `event`s manipulating certain attributes of scene objects. The events' parameters `Start` and `End` can be used to specify a period over which the manipulation is carried out. This is necessary to simulate e.g. continuous motion of a scene object from its current position to the target.

```
<?xml version="1.0" encoding="utf-8"?>
<scene
  BlockSize="4096"
  SampleRate="44100"
  MaximumDelay="0.05"
  PreDelay="0.0"
  LengthOfSimulation="5.0"
  NumberOfThreads="1"
  Renderer="ssr_binaural"
  HRIRs="impulse_responses/qu_kemar_anechoic/QU_KEMAR_anechoic_3m.sofa">
  <source Position="1 2 1.75"
          Type="point"
          Name="Cello"
          MeshFile="cello.mesh"
          Volume="0.4">
    <buffer ChannelMapping="1"
            Type="fifo"
            File="stimuli/anechoic/instruments/anechoic_cello.wav"/>
  </source>
  <source Position="1 -2 1.75"
          Type="point"
          Name="Castanets">
    <buffer ChannelMapping="1"
            Type="fifo"
            File="stimuli/anechoic/instruments/anechoic_castanets.wav"/>
  </source>
  <sink Position="0 0 1.75"
        UnitFront="1 0 0"
        UnitUp="0 0 1"
        Name="Head"/>
  <dynamic>
    <event Name="Castanets"
           Attribute="Position"
           Value="1 2 1.75"
           Start="1.5"
           End="5.0"/>
    <event Name="Cello"
           Attribute="Position"
           Value="1 -2 1.75"
           Start="3.0"/>
  </dynamic>
</scene>
```

Figure 4.1: Exemplary scene description file written in XML

5 Current Content of the Database

The Two!Ears database constitutes a central repository of data used for the synthesis and evaluation of the scenarios described in Deliverable 6.1.1. The database is amended during the project. In the following an overview is given on its current contents.

5.1 Impulse Responses

5.1.1 Head-Related Transfer Functions

Database Entry #1

Title:	Near-field HRTFs from SCUT database of the KEMAR		
Path:	`impulse_responses/scut_kemar_anechoic`		
Short:	SCUT_KEMAR_ANECHOIC	**Reference:**	[23]
License:		**Access:**	Public

The three-dimensional HRTF dataset was measured with a KEMAR dummy head. The HRTFs were measured for ten different distances of 0.2m, 0.25m, 0.3m and 0.4m, 0.5m, ..., 1.0m. The elevation angle varies from -30° to +90° with a stepsize of 15°. The azimuth angle varies from 0° to 360° with a resolution of 5° for elevation angles between ±30°. Above +30° elevation angle the azimuthal resolution is 10°, while for +90° elevation only one measurement per distance was performed.

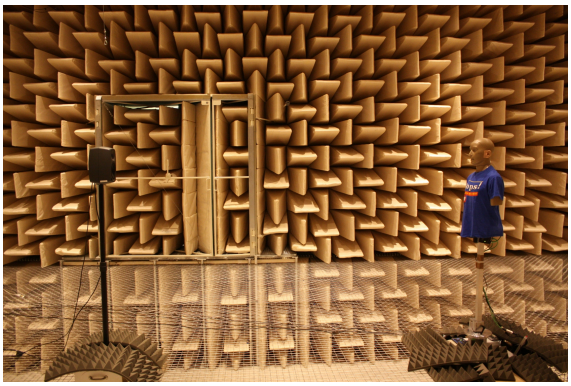
Database Entry #2

Title:	HRTF Measurements of a KEMAR Dummy-Head Microphone		
Path:	`impulse_responses/mit_kemar_anechoic`		
Short:	MIT_KEMAR_ANECHOIC	**Reference:**	[24]
License:	©MIT Media Lab	**Access:**	Public

The three-dimensional HRTF dataset was measured with a KEMAR (type DB-4004) equipped with a large right ear (type DB-065) and a normal-size left ear (type DB-061). A small two-way loudspeaker (Realistic Optimus Pro 7) was used as a sound source. The HRTFs were measured for a distances of 1.4m. The elevation angle varies from -40° (40° below horizontal plane) to +90° (directly overhead) with a stepsize of 10°. The azimuth angle varies from 0° to 360° with an elevation angle dependent resolution. For further details, see `http://sound.media.mit.edu/resources/KEMAR.html`

Database Entry #3

Title:	Anechoic HRTFs from the KEMAR manikin with different distances		
Path:	`impulse_responses/qu_kemar_anechoic`		
Short:	QU_KEMAR_ANECHOIC	**Reference:**	[25]
License:	CC BY-NC-SA 3.0	**Access:**	Public



The HRTFs were measured using a KEMAR (type 45BA) with the corresponding large ears (type KB0065 and KB0066). The manikin was mounted on the turntable of the VariSphear measurement system [26] to be able to rotate it automatically with high mechanical precision. All measurements were performed for a full horizontal rotation (360°) in 1° increments. The distance between the center of the manikin and the center of the loudspeaker was set to 3m, 2m, 1m and 0.5m, respectively. An active two-way loudspeaker (Genelec 8030A) was positioned at ear height of the manikin.
Ear signals were recorded with G.R.A.S. 40AO pressure microphones using a RME QuadMic pre-amplifier and a RME Multiface II audio interface. All data was recorded with a sampling rate of 44.1kHz and stored as single precision floating point values.

Database Entry #4

Title:	Spherical Far Field HRIR Compilation of the Neumann KU100		
Path:	`impulse_responses/fhk_ku100_anechoic`		
Short:	FHK_KU100_ANECHOIC	**Reference:**	[27]
License:	CC BY-SA 3.0	**Access:**	Public

Three-dimensional HRIR datasets were measured with the Neumann KU100 dummy head. An active 3-way louspeaker (Genelec 8260A) was used as a sound source with a constant distance of approximately 3.25m. Different apparent source positions were realized by rotating the dummy head around two axis using the VariSphear measurement system [26]. The impulse responses were captured for different sampling configurations of the source's position:

- horizontal plane with a resolution of 1°
- two different equidistant spherical Lebedev grids [28] with 2354 and 2702 sampling points
- full sphere equiangular 2° Gauss grid with 16020 sampling points

For further details, see `http://www.audiogroup.web.fh-koeln.de/ku100hrir.html`

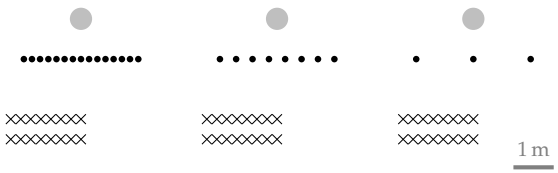
Database Entry #5

Title:	HRIR for simulating loudspeakers in anechoic environment		
Path:	`experiments/aipa_sound_field_synthesis/localization/single_loudspeaker_anechoic`		
Short:	BRS_SIMULATED_ANECHOIC	**Reference:**	[29, 30]
License:	CC BY-SA 3.0	**Access:**	Public

11 different loudspeakers were placed at the same positions as the one measured in room *Calypso*, described in database entry #16. All of them were simulated via inter- and extrapolation from the HRIRs presented in database entry #3.

Database Entry #6

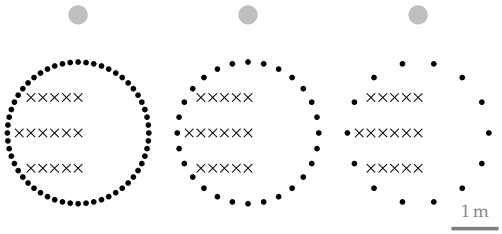
Title:	Point source synthesized with Wave Field Synthesis using a linear loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/wfs_ps_linear`		
Short:	BRS_WFS_PS_LINEAR	**Reference:**	[31, 30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of Binaural Room Synthesis (BRS) files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #26.

Database Entry #7

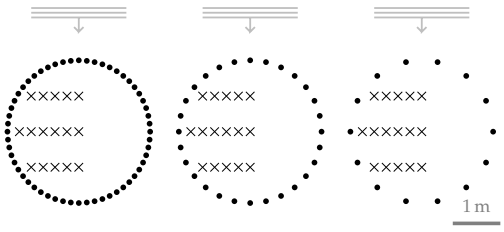
Title:	Point source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/wfs_ps_circular`		
Short:	BRS_WFS_PS_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of BRS files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #27.

Database Entry #8

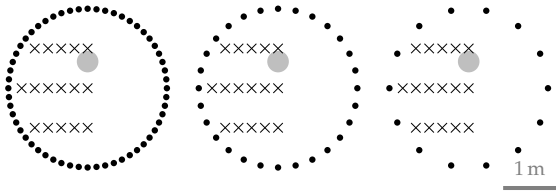
Title:	Plane wave synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/wfs_pw_circular`		
Short:	BRS_WFS_PW_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of BRS files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #28.

Database Entry #9

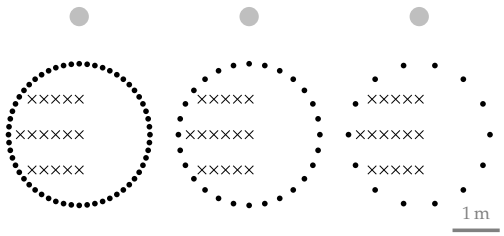
Title:	Focused source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/wfs_fs_circular`		
Short:	BRS_WFS_PW_CIRCULAR	**Reference:**	[30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of BRS files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #29.

Database Entry #10

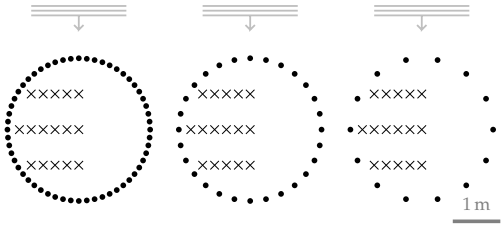
Title:	Point source synthesized with near-field compensated higher order Ambisonics using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/nfchoa_ps_circular`		
Short:	BRS_NFCHOA_PS_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of BRS files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #30.

Database Entry #11

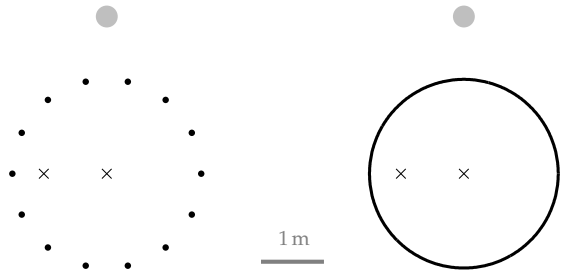
Title:	Plane wave synthesized with near-field compensated higher order Ambisonics using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization/nfchoa_pw_circular`		
Short:	BRS_NFCHOA_PW_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



Data set of BRS files used in a localization experiment. 16 different listening positions and 3 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #31.

Database Entry #12

Title:	Point source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/coloration/wfs_ps_circular`		
Short:	BRS_COLORATION_WFS_PS_CIRCULAR	**Reference:**	[33]
License:	CC BY-SA 3.0	**Access:**	Public

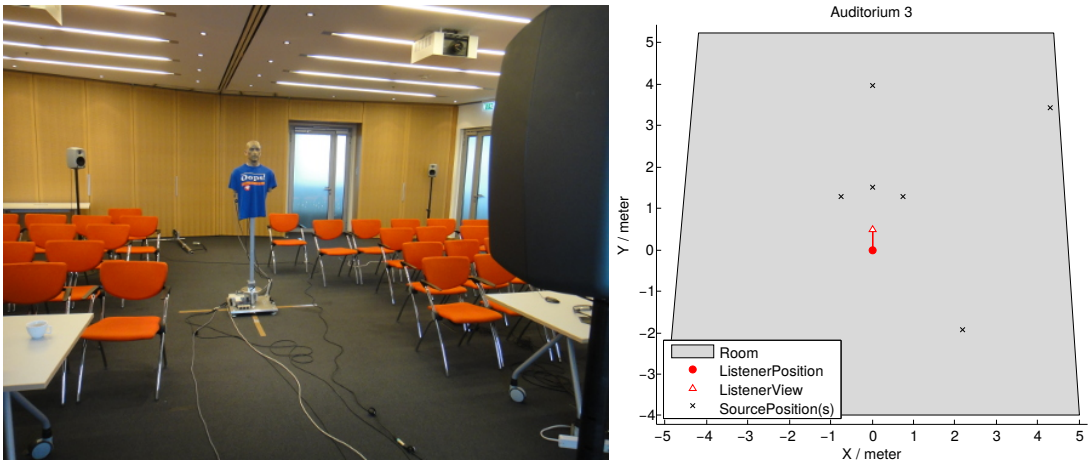


Data set of BRS files used in a coloration experiment. 2 different listening positions and 9 different loudspeaker arrays are binaurally simulated using the HRTFs from dataset #3. The data of the experiment is provided with dataset #32.

5.1.2 Binaural Room Impulse Responses

Database Entry #13

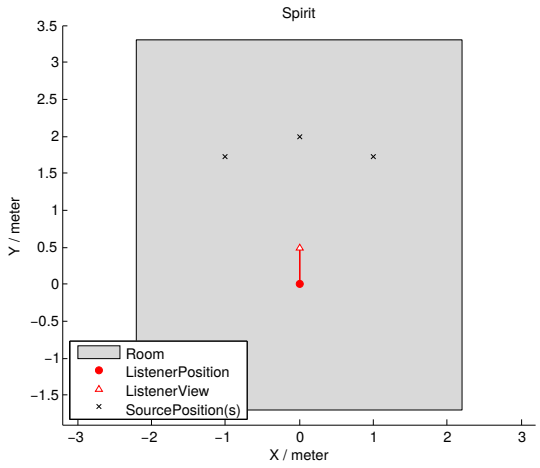
Title:	BRIRs from the KEMAR manikin in room *Auditorium 3* at TU Berlin		
Path:	`impulse_responses/qu_kemar_rooms/auditorium3`		
Short:	QU_KEMAR_AUDITORIUM3	**Reference:**	[34]
License:	CC BY-SA 3.0	**Access:**	Public



The BRIRs are measured with the same hardware involved in database entry #3. The only difference was that another step motor was inserted into the dummy head in order to turn its head. During the measurement the torso of the dummy had always the same orientation, only its head was rotated from -90° to 90° with a resolution of 1°. As sources three loudspeakers were placed at two different positons each, leading a total number of 6 different source positions as illustrated above.

Database Entry #14

Title:	BRIRs from the KEMAR manikin in room *Spirit* at TU Berlin		
Path:	`impulse_responses/qu_kemar_rooms/spirit`		
Short:	QU_KEMAR_SPIRIT	**Reference:**	[34]
License:	CC BY-SA 3.0	**Access:**	Public



The BRIRs are measured with the same hardware involved in database entry #13. The measurements were done with three loudspeakers placed at different positions and head rotations from -90° to 90° with a resolution of 1°.

Database Entry #15

Title:	BRIRs from the Cortex HATS in 4 rooms at University of Surrey		
Path:			
Short:	SURREY_ROOMS	**Reference:**	[35]
License:		**Access:**	Internal

The BRIRs are measured with the Cortex Instruments Mk.2 HATS in four different rooms. A Genelec 8020A active loudspeaker has been used as the sound source. While position and orientation of the HATS is kept constant for one room, the sound souce was placed in the horizontal plane at 1.5m distance with an azimuth between ±90° with an increment of 5°.

Database Entry #16

Title:	BRIRs of loudspeakers placed in studio room Calypso at TU Berlin		
Path:	`experiments/aipa_sound_field_synthesis/localization/single_loudspeaker_room_calypso`		
Short:	BRS_SIMULATED_CALYPSO	**Reference:**	[29, 30]
License:	CC BY-SA 3.0	**Access:**	Public

BRIRs for 11 different loudspeakers placed in room *Calypso* at TU Berlin were measured. The room is a studio listening room. The 11 loudspeakers were placed in a line and were part of a linear loudspeaker array consisting of 19 loudspeakers. The measurement was done with the KEMAR dummy head as described by database entry #3 and Fostex PM0.4 loudspeakers. The dummy head was wearing AKG K601 open headphones during the measurement and was rotated from $-90°$ to $90°$ in $1°$ steps.

5.2 Sound Databases

5.2.1 Speech

Database Entry #17

Title:	GRID audiovisual sentence corpus		
Path:			
Short:	GRID	**Reference:**	[36]
License:	Freely available for research use	**Access:**	Internal

GRID is a large multitalker audiovisual sentence corpus to support joint computational-behavioral studies in speech perception. The corpus consists of high-quality audio and video (facial) recordings of 1000 sentences spoken by each of 34 native British English talkers (18 male, 16 female). Sentences are of the form "put red at G9 now". More details about GRID can be found at `http://spandh.dcs.shef.ac.uk/gridcorpus`

Database Entry #18

Title:	TIMIT Acoustic-Phonetic Continuous Speech Corpus		
Path:			
Short:	TIMIT	**Reference:**	[37]
License:	LDC User Agreement for Non-Members[1]	**Access:**	Internal

TIMIT is a corpus of phonemically and lexically transcribed read speech. It has been designed to provide speech data for the acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems. TIMIT contains broadband recordings of 630 speakers of eight major dialects of American English, each reading ten phonetically rich sentences. The TIMIT corpus includes time-aligned orthographic, phonetic and word transcriptions as well as a 16-bit, 16kHz speech waveform file for each utterance. More details can be found at `https://catalog.ldc.upenn.edu/LDC93S1`

1 see `https://catalog.ldc.upenn.edu/license/ldc-non-members-agreement.pdf`

5.2.2 Environmental Noise

Database Entry #19

Title:	IEEE AASP CASA Challenge - Scene Classification Dataset		
Path:	`sound_databases/IEEE_AASP/scenes`		
Short:	IEEE_AASP_SCENES	**Reference:**	[38]
License:	CC BY 2.0 UK	**Access:**	Public

Binaural recordings of acoustic environmental scenes. The following acoustic scenes are included

- bus
- busystreet (busy street with heavy traffic)
- office
- openairmarket (open-air or semi-open market)
- park
- quietstreet (quiet street with mild traffic)
- restaurant
- supermarket
- tube (train in the Transport for London, Underground and Overground, train networks)
- tubestation (tube station in the Transport for London, Underground and Overground, train networks, either subterranean or supraterranean)

Each class contains 10 recordings. Each recording is 30sec long.

Database Entry #20

Title:	ICRA Noise database		
Path:			
Short:	ICRA	**Reference:**	[39]
License:	*No copyright is claimed* [...] *can freely be copied.*	**Access:**	Internal

The ICRA database contains speech-shaped and speech-modulated signals with well-defined spectral and temporal properties.

1. Speech-shaped, stationary Gaussian noise (1 male, normal effort)
2. Speech-shaped, stationary Gaussian noise (1 male, raised effort)
3. Speech-shaped, stationary Gaussian noise (1 male, loud effort)
4. Speech-shaped, speech-modulated Gaussian noise (1 female, normal effort)
5. Speech-shaped, speech-modulated Gaussian noise (1 male, normal effort)
6. Speech-shaped, speech-modulated Gaussian noise (1 female & 1 male, normal effort)
7. Speech-shaped, speech-modulated Gaussian noise (6 person babble, normal effort)
8. Speech-shaped, speech-modulated Gaussian noise (6 person babble, raised effort)
9. Speech-shaped, speech-modulated Gaussian noise (6 person babble, loud effort)

5.2.3 General Sounds

Database Entry #21

Title:	TWO!EARS internal general sounds development database		
Path:			
Short:	NIGENS	**Reference:**	
License:	Commercial. Only for use in consortium.	**Access:**	Internal

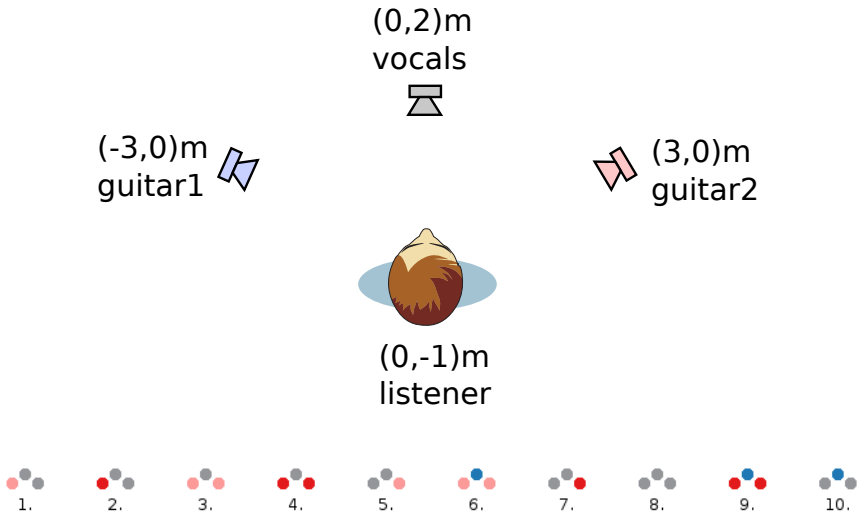
The TWO!EARS internal general sounds development database was compiled from the Stockmusic[1] sounds archive as a training database particularly for the modeling work in work package three. 834 isolated high quality sound files were selected for the following classes: alarm, crying baby, crash, barking dog, running engine, female speech, burning fire, footsteps, knocking on door, ringing phone, piano, and a general "anything else" class with as much variety as possible. Care has been taken to select sound classes representing different features, like noise-like or pronounced, discrete or continuous, while using classes that are possibly related to the later demonstration scenarios of TWO!EARS. The database in total comprises 4 hours and 34 minutes of sound material, recorded with 32-bit precision and 44100 Hz sampling rate.
For sound type classification training, information additional to the sound class, namely the exact onset and offset times of the sound events, is needed. These have been created in a perceptual manner using the self-developed "sound event labeling tool" (see #23)

1 Stockmusic is the online redistributor of the famous Sound Idea's sounds library, offering individual files rather than precompiled collections. See `http://www.stockmusic.com/sound_effects`

5.2.4 Listening Tests

Database Entry #22

Title:	Music scene reproduced with Wave Field Synthesis		
Path:	`experiments/aipa_sound_field_synthesis/scene_quality/stimuli`		
Short:	QUALITY_WFS_SCENE_SOUNDS	**Reference:**	[40, 41]
License:	CC BY-SA 3.0	**Access:**	Public



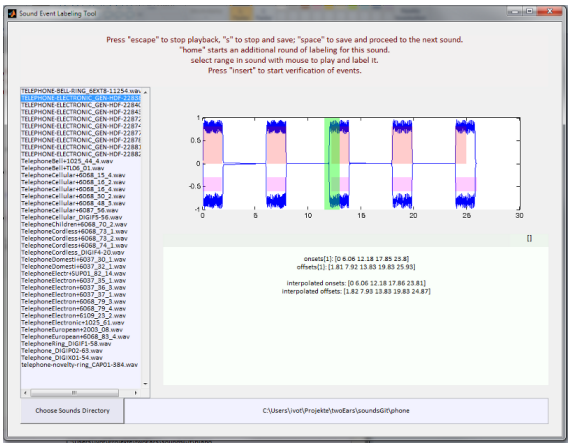
This experiment investigated the influence of the amount of degradation on different parts of a music scene. The music scene consists of two guitars placed at the sides of the listener and a singer placed at the front of the listener. The degradations were introduced by using different Wave Field Synthesis setups synthesizing a point or a focused source, whereby different setups were used for the single parts of the scene in order to vary the degradations. For the guitars two different kinds of degradations were introduced and one for the vocal. The different combinations are shown in the figure, gray indicates now degradation, red and blue show different kind of degradations. The stimuli are stored as binaural signals, where the different Wave Field Synthesis setups were simulated with the HRIRs from database entry #3. The corresponding rating results from a paired comparison test of the stimuli are provided as data set #33.

5.3 Perceptual Labels

5.3.1 Sound Event Labels

Database Entry #23

Title:	On- and Offset times for the TWO!EARS internal general sounds development database		
Path:			
Short:	NIGENS_EVENTS	**Reference:**	
License:		**Access:**	Internal



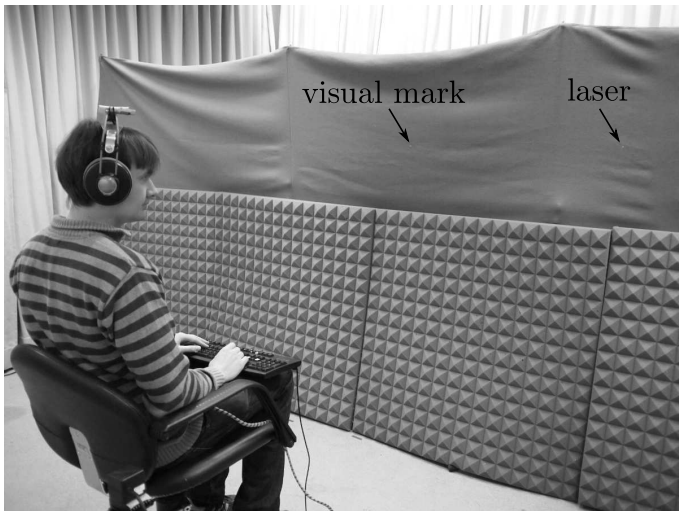
In order to effectively train models that identify sound events of particular classes, the sound files of the general sound database (#21) have been annotated by time stamps indicating perceptual on- and offsets of occurring sound events.
A sound event labeling tool (see figure) has been designed to speed up the manual labeling process by automatically selecting and (aurally) presenting extracts of the sound files and letting the user label via a simple and streamlined user interface. This work flow maintains a high precision of the resulting on- and offset times compared to the perceptually "true" ones (error within 50 milliseconds).
Since each sound file may contain an arbitrary number of instances of the same event class, a list of on- and offset times for each sound file is stored in a separate text file:

```
0.002268 1.814059
6.061224 7.922902
12.176871 13.827664
...
```

5.3.2 Head movements during the rating of different audio presentation techniques

Database Entry #24

Title:	Head movements during localization for real and simulated loudspeakers		
Path:	`experiments/aipa_sound_field_synthesis/localization/human_label_head_movements_real_vs_simulated_loudspeaker/`		
Short:	HEADMOVEMENTS_REAL_VS_SIMULATED	**Reference:**	[29]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 11 different loudspeakers placed in room *Calypso*. All of them were placed there directly or simulated with the corresponding measured BRIRs or via extrapolation from the HRIRs presented in database entry #3. The corresponding BRS used in the experiment are presented in database #5 and #16. The head movement results include the time and orientation of the head of the listener made before localizing the sound.

5.3.3 Spatial perception for different spatial audio presentation techniques

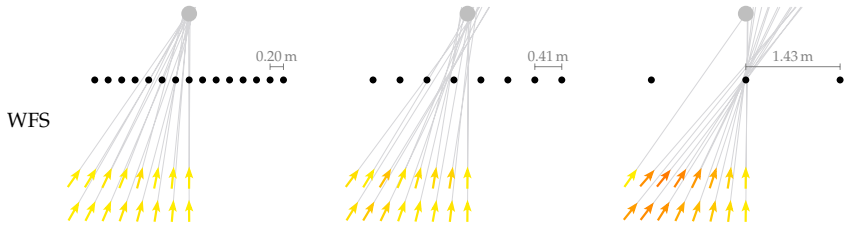
Database Entry #25

Title:	Localization and locatedness for real and simulated loudspeakers		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_REAL_VS_SIMULATED	**Reference:**	[29, 30]
License:	CC BY-SA 3.0	**Access:**	Public

The experiment was run for 11 different loudspeakers placed in room *Calypso*. All of them were placed there directly or simulated with the corresponding measured BRIRs, or via extrapolation from the HRIRs presented in database #3. The corresponding BRS are presented in database #5 and #16. The results include mean and standard deviation of the perceived direction in the horizontal plane.

Database Entry #26

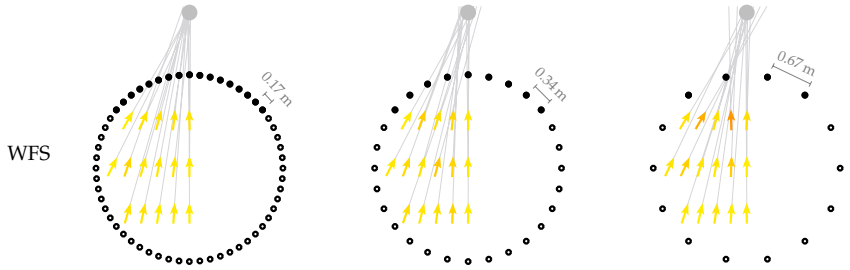
Title:	Localization and locatedness for a point source synthesized with Wave Field Synthesis using a linear loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_WFS_PS_LINEAR	**Reference:**	[31, 30]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 16 different listener positions and 3 different loudspeaker array geometries. The corresponding BRS data set is provided as dataset #6.

Database Entry #27

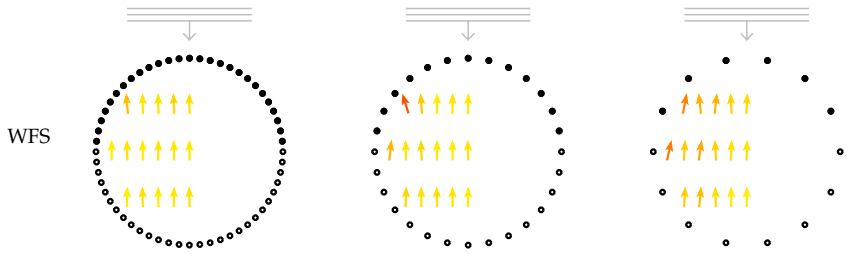
Title:	Localization and locatedness for a point source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_WFS_PS_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 16 different listener positions and 3 different loudspeaker array geometries. The corresponding BRS data set is provided as dataset #7.

Database Entry #28

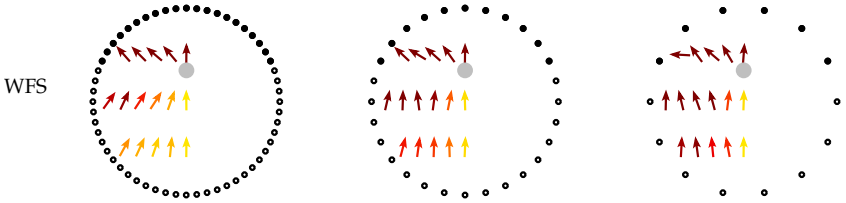
Title:	Localization and locatedness for a plane wave synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_WFS_PW_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 16 different listener positions and 3 different loudspeaker array geometries. The corresponding BRS data set is provided as dataset #8.

Database Entry #29

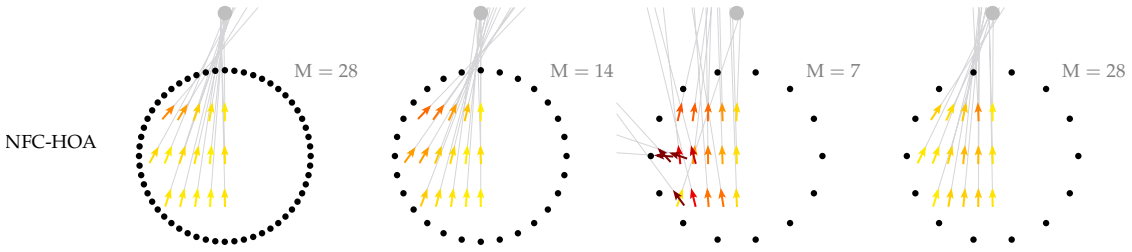
Title:	Localization and locatedness for a focused source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_WFS_FS_CIRCULAR	**Reference:**	[30]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 16 different listener positions and 3 different loudspeaker array geometries. The corresponding BRS data set is provided as dataset #9.

Database Entry #30

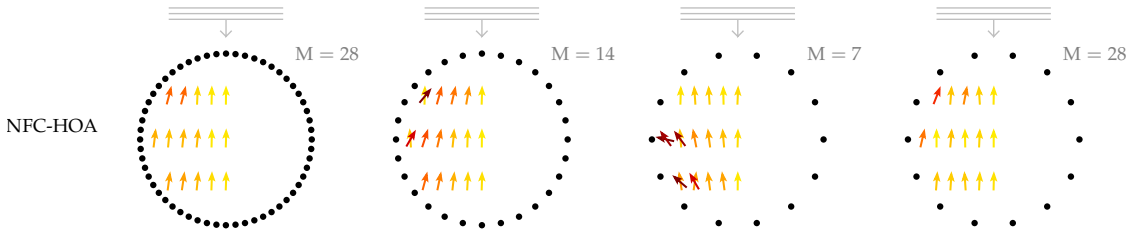
Title:	Localization and locatedness for a point source synthesized with near-field compensated higher order Ambisonics and a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_NFCHOA_PS_CIRCULAR	**Reference:**	[32, 30]
License:	CC BY-SA 3.0	**Access:**	Public



The experiment was run for 16 different listener positions, 3 different loudspeaker array geometries, and 4 different ambisonic orders. The corresponding BRS data set is provided as dataset #10.

Database Entry #31

Title:	Localization and locatedness for a plane wave synthesized with near-field compensated higher order Ambisonics and a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/localization`		
Short:	SPATIAL_NFCHOA_PW_CIRCULAR	**Reference:**	[32]
License:	CC BY-SA 3.0	**Access:**	Public

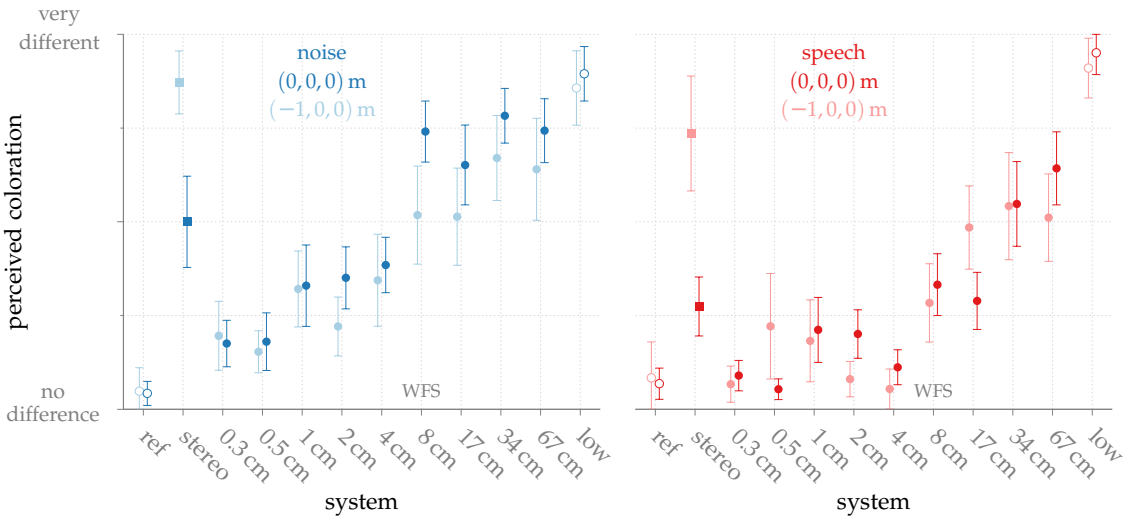


The experiment was run for 16 different listener positions, 3 different loudspeaker array geometries, and 4 different ambisonic orders. The corresponding BRS data set is provided as dataset #11.

5.3.4 Timbral perception for different spatial audio presentation techniques

Database Entry #32

Title:	Coloration for a point source synthesized with Wave Field Synthesis using a circular loudspeaker array		
Path:	`experiments/aipa_sound_field_synthesis/coloration`		
Short:	TIMBRAL_WFS_PS_CIRCULAR	**Reference:**	[33, 30]
License:	CC BY-SA 3.0	**Access:**	Public

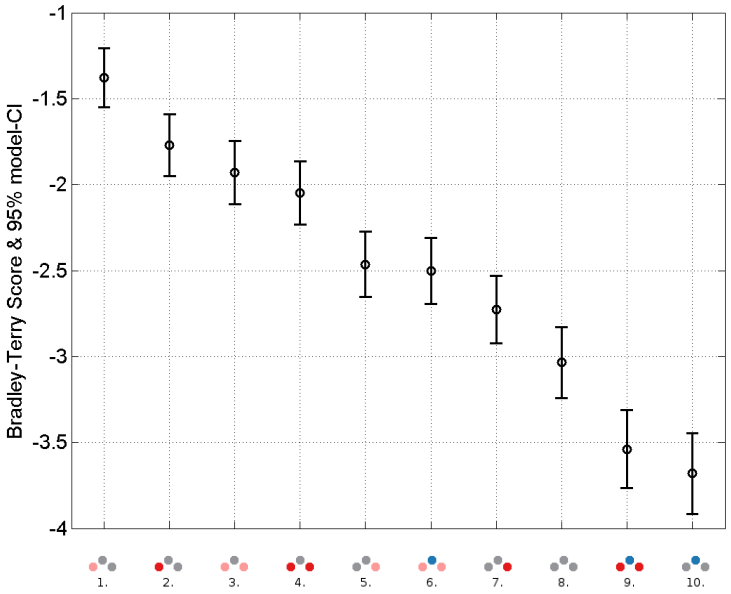


The experiment was run for 2 different listener positions and 9 different loudspeaker array geometries. Three different source materials (speech, music, pink noise) were used. The corresponding BRS data set is provided as dataset #12.

5.3.5 Quality ratings for different spatial audio presentation techniques

Database Entry #33

Title:	Quality ratings for a music scene in Wave Field Synthesis		
Path:	`experiments/aipa_sound_field_synthesis/scene_quality/results`		
Short:	QUALITY_WFS_SCENE	**Reference:**	[40, 41]
License:	CC BY-SA 3.0	**Access:**	Public



Ratings from a paired comparison test asking. The listener were asked to indicate which of the two presented sounds had the better quality. The 10 different stimuli of the experiment are provided and described in database entry #22.

6 Conclusions and Outlook

The synthesis of ear and eye signals plays an important role in the development and evaluation of the Two!Ears model. A first implementation of the simulation framework is already fully integrated into the model. This served as a proof of concept for the ear signal simulation and the architecture of the model. The capabilities of the simulation framework are extended in the second year of the project. For the ear signals this includes the use of data-based binaural synthesis and dynamic BRIRs. The visual simulation framework is extended and integrated into the model.

During the first year of the Two!Ears project, a database of considerable size has been established. Among others, it includes several utility data to render different acoustic environments and scenarios. In order to evaluate the behavior of the auditory model against human performance in such environments a variety of perceptual labels has been added to the database. They cover aspects of the both application cases addressed in Two!Ears, namely Dynamic Auditory Scene Analysis (WP 6.1) and Quality of Experience evaluation (WP 6.2).

Recently a multi-loudspeaker BRIR data set for a 64-channel loudspeaker setup at the audio laboratory of Universität Rostock has been measured. The data will be published in the near future. Additional BRIRs of a 5.0 stereophony system will be acquired at Technische Universität Berlin. In conjunction with these measurements, listening experiments for assessing the quality of 5.0 reproduction methods are planned.

Acronyms

BEFT Bochum Experimental Feedback Testbed 17, 18

BRIR Binaural Room Impulse Response 8–15, 23, 34–36, 44, 45, 51

BRS Binaural Room Synthesis 30–33, 44

CC BY 2.0 UK Creative Commons Attribution 2.0 UK: England & Wales, see `http://creativecommons.org/licenses/by/2.0/uk/legalcode` 38

CC BY-NC-SA 3.0 Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported, see `http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode` 28

CC BY-SA 3.0 Creative Commons Attribution-ShareAlike 3.0 Unported, see `http://creativecommons.org/licenses/by-sa/3.0/legalcode` 29–36, 41, 44–50

HATS Head and Torso Simulator 7, 23, 35

HDF Hierarchical Data Format 23

HRIR Head-Related Impulse Response 8–11, 13, 23, 29, 41, 44, 45

HRTF Head-Related Transfer Function 23, 27, 28

HTTP Hypertext Transfer Protocol 22

KEMAR Knowles Electronics Manikin for Acoustic Research 17–19, 27, 28, 34, 35

MEX MATLAB executable 14

MORSE Modular OpenRobots Simulation Engine 18, 19

MRIR Multi-Channel Room Impulse Response 11, 23

NetCDF Network Common Data Form 23

PWD plane wave decomposition 15

ROS robot operating system 5, 18, 19

SOFA Spatially Oriented Format for Acoustics 23

SSR SoundScape Renderer 13, 14

VTE virtual test environment 17, 18

XML Extensible Markup Language 23–25, 54

XSD XML Schema Definition 24

Bibliography

[1] G. Enzner, "Analysis and optimal control of LMS-type adaptive filtering for continuous-azimuth acquisition of head related impulse responses," Las Vegas, NV, April 2008. (Cited on page 9)

[2] N. Hahn and S. Spors, "Measurement of time-variant binaural room impulse responses for data-based synthesis of dynamic auditory scenes," in *German Annual Conference on Acoustics (DAGA)*, March 2014. (Cited on page 10)

[3] ——, "Identification of dynamic acoustic systems by orthogonal expansion of time-variant impulse responses," in *IEEE-EURASIP International Symposium on Control, Communications, and Signal Processing*, May 2014. (Cited on page 10)

[4] S. Spors, H. Wierstorf, and M. Geier, "Comparison of modal versus delay-and-sum beamforming in the context of data-based binaural synthesis," in *132nd Convention of the Audio Engineering Society*, April 2012. (Cited on page 10)

[5] F. Schultz and S. Spors, "Data-based binaural synthesis including rotational and translatory head-movements," in *52nd Conference on Sound Field Control - Engineering and Perception, Audio Engineering Society*, September 2013. (Cited on page 10)

[6] F. Winter, F. Schutz, and S. Spors, "Localization properties of data-based binaural synthesis including translatory head-movements," in *Forum Acousticum*, September 2014, submitted. (Cited on page 11)

[7] T. Lentz, D. Schröder, M. Vorländer, and I. Assenmacher, "Virtual reality system with integrated sound field simulation and reproduction," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 2007, article ID 70540. (Cited on page 12)

[8] M. Geier and S. Spors, "Spatial Audio Reproduction with the SoundScape Renderer," in *27th Tonmeistertagung - VDT International Convention*, Cologne, Germany, Nov. 2012. [Online]. Available: http://www.int.uni-rostock.de/fileadmin/user_upload/publications/spors/2012/Geier_TMT2012_SSR.pdf (Cited on page 13)

[9] M. Geier, T. Hohn, and S. Spors, "An Open Source C++ Framework for Multithreaded Realtime Multichannel Audio Applications," in *Linux Audio Conference*, Stanford, USA, Apr. 2012. [Online]. Available: http://lac.linuxaudio.org/2012/download/

lac2012_proceedings.pdf (Cited on page 13)

[10] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, 1979. (Cited on page 15)

[11] J. Borish, "Extension of the image model to arbitrary polyhedra," *The Journal of the Acoustical Society of America*, vol. 75, no. 6, 1984. (Cited on page 15)

[12] T. Walther and B. Cohen-L'hyver, "Multimodal feedback in auditory-based active scene exploration," in *Proc. Forum Acusticum*, 2014. (Cited on page 17)

[13] OGRE, "Ogre - open source 3d graphics engine," 2014. [Online]. Available: http://www.ogre3d.org/ (Cited on page 17)

[14] ROS, "The robot operating system," 2014. [Online]. Available: http://www.ros.org/ (Cited on page 18)

[15] MORSE, "Morse, the modular openrobots simulation engine," 2014. [Online]. Available: https://www.openrobots.org/wiki/morse/ (Cited on page 18)

[16] Blender Foundation, "Blender - 3d open source animation suite," 2014. [Online]. Available: http://www.blender.org/ (Cited on page 18)

[17] BULLET, "The bullet physics engine," 2014. [Online]. Available: http://bulletphysics.org/wordpress/ (Cited on page 18)

[18] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, p. 2421, November 2006. (Cited on page 19)

[19] A. Mallet and M. Herrb, "Recent developments of the genom robotic component generator," in *6th National Conference on Control Architectures of Robots*. Grenoble, France: INRIA Grenoble Rhône-Alpes, May 2011. (Cited on page 19)

[20] WillowGarage, "Open source computer vision library," 2014. [Online]. Available: http://sourceforge.net/projects/opencvlibrary/ (Cited on page 19)

[21] P. Majdak, Y. Iwaya, T. Carpentier, R. Nicol, M. Parmentier, A. Roginska, Y. Suzuki, K. Watanabe, H. Wierstorf, H. Ziegelwanger, and M. Noisternig, "Spatially Oriented Format for Acoustics: A Data Exchange Format Representing Head-Related Transfer Functions," in *134th Convention of the Audio Engineering Society*, 2013. (Cited on page 23)

[22] M. Parmentier. (2012, Oct.) Project AES-X212, HRTF file format. [Online].

Available: http://www.aes.org/standards/meetings/init-projects/aes-x212-init.cfm (Cited on page 23)

[23] B.-s. Xie, *Head-Related Transfer Function and Virtual Auditory Display.* J Ross, 2013. (Cited on page 27)

[24] B. Gardner, K. Martin *et al.*, "HRTF measurements of a KEMAR dummy-head microphone," Massachusetts Institute of Technology, Tech. Rep. 280, 1994. (Cited on page 27)

[25] H. Wierstorf, M. Geier, and S. Spors, "A free database of head related impulse response measurements in the horizontal plane with multiple distances," in *Audio Engineering Society Convention 130*, May 2011. (Cited on page 28)

[26] B. Bernschütz, C. Pörschmann, S. Spors, and S. Weinzierl, "Entwurf und Aufbau eines variablen sphärischen Mikrofonarrays für Forschungsanwendungen in Raumakustik und virtual Audio," in *German Annual Conference on Acoustics (DAGA)*, Berlin, Germany, 2010. (Cited on pages 28 and 29)

[27] B. Bernschütz, "A spherical far field HRIR/HRTF compilation of the neumann KU 100," in *Proceedings of the 40th Italian (AIA) Annual Conference on Acoustics and the 39th German Annual Conference on Acoustics (DAGA) Conference on Acoustics*, 2013. (Cited on page 29)

[28] V. Lebedev and D. Laikov, "A quadrature formula for the sphere of the 131st algebraic order of accuracy," in *Doklady. Mathematics*, vol. 59, no. 3. MAIK Nauka/Interperiodica, 1999, pp. 477–481. (Cited on page 29)

[29] H. Wierstorf, S. Spors, and A. Raake, "Perception and evaluation of sound fields," in *59th Open Seminar on Acoustics*, 2012. (Cited on pages 30, 36, 44, and 45)

[30] H. Wierstorf, "Perceptual assessment of sound field synthesis," Ph.D. dissertation, Technische Universität Berlin, 2014, to appear. (Cited on pages 30, 31, 32, 33, 36, 45, 46, 47, and 49)

[31] H. Wierstorf, A. Raake, and S. Spors, "Binaural assessment of multi-channel reproduction," in *The technology of binaural listening*, J. Blauert, Ed. New York: Springer, 2013, pp. 255–78. (Cited on pages 30 and 45)

[32] ——, "Localization in wave field synthesis and higher order ambisonics at different positions within the listening area," in *Proceedings of the 40th Italian (AIA) Annual Conference on Acoustics and the 39th German Annual Conference on Acoustics (DAGA) Conference on Acoustics*, 2013. (Cited on pages 31, 32, 33, 46, 47, and 48)

[33] H. Wierstorf, C. Hohnerlein, S. Spors, and A. Raake, "Coloration in wave field

synthesis," in *Audio Engineering Society Conference on Spatial Audio 55*, August 2014. (Cited on pages 33 and 49)

[34] N. Ma, T. May, H. Wierstorf, and G. Brown, "A machine-hearing system exploiting head movements for binaural sound localization in reverberant conditions," in *ICASSP*, 2015, submitted. (Cited on pages 34 and 35)

[35] C. Hummersone, R. Mason, and T. Brookes, "Dynamic precedence effect modeling for source separation in reverberant environments," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 7, pp. 1867–1871, Sept 2010. (Cited on page 35)

[36] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–4, 2006. (Cited on page 37)

[37] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic-phonetic continuous speech corpus CD-ROM," *National Inst. Standards and Technol. (NIST)*, 1993. (Cited on page 37)

[38] D. Giannoulis, E. Benetos, D. Stowell, and M. D. Plumbley, "IEEE AASP challenge on detection and classification of acoustic scenes and events - public dataset for scene classification task," 2012. [Online]. Available: http://c4dm.eecs.qmul.ac.uk/rdr/handle/123456789/29 (Cited on page 38)

[39] W. A. Dreschler, H. Verschuure, C. Ludvigsen, and S. Westermann, "Icra noises: Artificial noise signals with speech-like spectral and temporal properties for hearing instrument assessment," *International Journal of Audiology*, vol. 40, no. 3, pp. 148–157, 2001. (Cited on page 39)

[40] A. Raake, H. Wierstorf, and J. Blauert, "A case for Two!Ears in audio quality assessment," in *Forum Acusticum*, 2014. (Cited on pages 41 and 50)

[41] J. Dierkes, "Qualität räumlicher Audiowiedergabe: Ist es szenenspezifisch oder objektspezifisch?" 2014, Bachelor Thesis. (Cited on pages 41 and 50)