Technische Universität München
Lehrstuhl für Datenverarbeitung

Localization, Tracking, and Separation of Sound Sources for Cognitive Robots

Marko Đurković

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzende(r): Univ.-Prof. P. Lugli, Ph.D.

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. K. Diepold
2. Univ.-Prof. G. Cheng, Ph.D.

Die Dissertation wurde am 19. Januar 2012 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 30. Oktober 2012 angenommen.

Marko Đurković. *Localization, Tracking, and Separation of Sound Sources for Cognitive Robots.* Dissertation, Technische Universität München, Munich, Germany, 2012.

Institute for Data Processing, Technische Universität München, 80290 München, Germany, http://www.ldv.ei.tum.de.

Abstract

An artificial auditory system plays an important role for the perception system of cognitive robots, as it complements a robot's other senses as a source of information. Hearing is essential to detect events that are purely acoustic in nature, because these events cannot be perceived by any of the other senses. Furthermore, sound propagates in all directions and hearing a sound does not require a direct line of sight between a sensor and an acoustic event. A cognitive robot's hardware and field of application impose specific requirements and constraints that existing algorithms for artificial auditory systems generally do not meet.

This thesis investigates individual components of robot auditory systems, particularly focussing on the data processing modules for sound source localization, tracking, and separation. The localization module determines the positions of active sound sources, while the tracking module filters the localized positions and follows the source movement over time. For robots in unconstrained dynamical environments, both modules have to be able to process multiple simultaneously active sources from observations that are subject to reverberation and noise. The separation module estimates the original source signals from the recorded observations, which contain convolved mixtures of the original signals. Since robots must be able to quickly react to acoustic events, only a small algorithmic latency is acceptable for all three modules. Additionally, the complexity of the algorithms must be kept low, as the computational resources on robots are generally limited.

This thesis presents algorithms for the three mentioned auditory system modules that are explicitly tailored to the requirements of cognitive robots. The localization algorithm operates in the time-frequency domain and exploits signal sparseness to estimate the positions of multiple sources. The tracking algorithm uses particle filters with bimodal observation probability densities to post-process localization results. The separation algorithm is based on binary masking and shares its computations with the localization module to keep its complexity low. The performance of the algorithms is evaluated and compared to state-of-the-art techniques in elaborate real-world experiments. The evaluations reveal that the presented algorithms perform better than state-of-the-art techniques, while simultaneously operating inside the requirements and constraints of robotic systems.

Contents

1 Introduction

In nature the sense of hearing is an extraordinary ability. It converts air pressure changes into nerve impulses and thus enables humans and animals to perceive sounds with their ears. However, what makes it truly remarkable is the processing by the auditory system. It enables, for example, the Barn Owl to hunt for food in complete darkness or humans to follow a conversation in a crowded room. For humans hearing complements the other senses and is an important source of information for understanding and interpreting the world.

1.1 Auditory systems for robots

In the recent years robotic research has moved in the direction of building cognitive systems. Researchers in the field of embodied cognition [20] believe that only an embodied and situated autonomous agent will be able to achieve true artificial intelligence. In their view, a cognitive system has to be equipped with sensors to be able to perceive its dynamic environment and also to have motor skills to interact with it. There are many reasons why hearing is important for a robot's perception system. Unlike vision, hearing is omni-directional and works equally well in the dark. Some events, like the ringing of a door bell, are purely acoustic in nature and cannot be detected by other senses. Additionally, speech plays a major role in human-human interaction and its significance for human-robot interaction will rise in the future. Being able to better understand the world through hearing will improve the cognitive capabilities of robotic systems.

1.1.1 Modules of an auditory system

A robot auditory system has to master several skills that are dependent on one another to a varying degree. The auditory system can be split into multiple subproblems, which in turn can be solved by separate modules. Low-level modules process the raw sound signals while their results serve as inputs to higher level modules. In such a processing chain, a module hierarchy emerges and its layout depends on the actual algorithms used for each task.

Figure 1.1 shows an overview of some important auditory system modules. Despite the subdivision of the complete system into smaller tasks, each module still implements a complex functional-

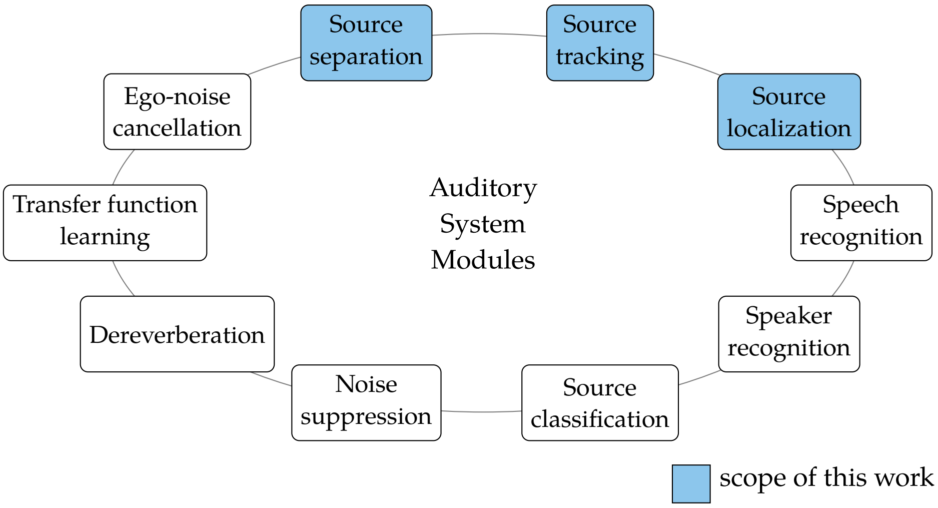


Figure 1.1: An overview of the modules of a robot auditory system. This work investigates the modules for sound source localization, tracking, and separation.

ity. The scope of this work is therefore limited to the investigation of three modules, namely source localization, tracking, and separation. These are somewhat related in that they are situated at a low level of the hierarchy and their solutions are often dependent on each other.

Sound source localization

The sound source localization module determines the relative position of a sound source in respect to the observer purely from acoustic cues. In a spherical coordinate system, this position is defined by the direction and the distance of the source. Humans are very good at estimating the source direction, but their ability to determine the distance of a source is limited [70].

Sound source tracking

While the localization module determines the position of a source at a certain point in time, sound source tracking observes source movement over time. Tracking takes previous sound source positions and knowledge about possible source movement patterns into account. It filters the position estimates of the localization module and predicts source positions in the near future.

Sound source separation

When multiple sound sources are active at the same time, each of the robot's microphones will record a mixture of all sources. Sound source separation retrieves the original separate source signals from the mixed observations. This step is necessary, as subsequent processing modules often require sound streams with only one active sound source.

1.1.2 The perfect auditory system

The perfect artificial auditory system would enable a robot to detect and understand events in its environment just by hearing the associated sound. Like humans, robots would be able to determine what has happened, which objects were involved in the event, and where in the environment the event took place. The communication between robots and humans would improve, as the robot would be able to recognize a speaker by his voice and to engage in a natural conversation. In the presence of multiple sound sources the system would be able to interpret all sounds simultaneously and focus its attention to the sources of greatest interest. The perfect artificial auditory system would be robust to changes in the environment, noise, and reverberation.

The perfect localization module would match the human performance in determining the source direction and also reliably estimate its distance. In the presence of multiple sources the localization algorithm should be able to calculate all positions simultaneously.

The ideal tracking algorithm would adapt to the movement statistics of each source. Sources at stationary positions, as well as quickly moving sources, should be tracked with the same accuracy. If a source signal is not observable for a short time the tracking has to be able to predict the most likely position of the source.

The perfect sound separation algorithm is characterized by its ability to completely suppress the signals of interfering sound sources without distorting the source signal of interest. Separated sound streams will ideally only incorporate information from one sound source and the associated reverberation.

1.2 State of the art

For the three auditory system modules of interest adequate algorithms are necessary. In literature algorithms for localization, tracking, and separation of sound sources have been studied extensively. Many of the existing algorithms were inspired by the knowledge about hearing in nature and try to mimic some aspects of human or animal auditory processing.

For this work, existing research for each module can be divided into two categories. On the one hand, there is research that investigates the mentioned audition problems explicitly in the context of robotic systems. On the other hand, there are algorithms that were developed as general solutions or targeted at a different application. Results from the first category are highly relevant for this work, while the results from the second often do not meet the requirements and constraints that are given on robotic systems.

1.2.1 Requirements and constraints

Robot auditory systems have several requirements that processing algorithms have to fulfill. If one of the requirements cannot be met, the practical usability of an algorithm is at least questionable. The following requirements apply to algorithms for robot auditory system modules:

Robustness towards reverberation In a real environment sound waves are reflected by all surfaces and reach the microphones from multiple directions. Due to this multipath wave propagation, each microphone records the original source signal and its reflections. The amount of reverberation depends highly on the environment, but under realistic conditions at least some amount of reverberation will always be present. The modules of an auditory system have to tolerate reverberation without loosing accuracy.

Many existing algorithms were developed under the assumption of free-field conditions and do not account for reverberation at all. Making these algorithms robust towards reverberation is a hard and often impossible task.

Robustness towards noise Recorded observations of the environment are subject to noise. Firstly, there is sensor noise, which is introduced by the microphones. Secondly, there can be unwanted sound sources in the environment that also corrupt the recordings. The auditory system has to be robust towards both types of noise.

The design and evaluation of existing algorithms often does not account for noise. In the presence of noise the accuracy of such algorithms usually suffers significantly.

General applicability Some algorithms are designed to process the sound of some particular types of sources and the quality of their results deteriorates if a source with different signal characteristics is present. In a dynamic environment the occurrence of arbitrary sources cannot be prevented and the auditory processing has to function properly even if sources of unknown type are present.

Number of sources This applies specifically to the localization and tracking of sound sources. In dynamic environments the number of sound sources cannot be predicted or controlled. Both approaches have to be able to process observations where the number of active sources is unknown and possibly greater than one. The localization has to determine the positions of all sources simultaneously and the tracking has to be able to follow multiple sources. Both modules have to be able to identify the number of active sources.

Some existing techniques can localize and track exactly one sound source. The usefulness of such approaches for robot auditory systems is limited, as the assumption of only one active sound source does not hold in most environments.

Number of dimensions Sound sources can be located anywhere in three dimensional space and a robot auditory system should at least be able to detect the direction of the sound source. This means that localization and tracking should be able to determine the position of the source in at least two dimensions.

Many existing techniques assume that two parameters of the source position are known a priori and they limit the localization to one dimension.

The former requirements arise from the audition problem. The limited resources available on a robot impose additional constraints on the signal processing. Due to size and energy restrictions, the processing power of mobile robots is limited and causes the following requirements:

Low computational complexity The computational complexity of each auditory system module has to be kept low. Algorithms have to work on-line, meaning that they have to be able to process data at the same rate as it arrives.

Low latency Each auditory system module could add latency to the complete processing chain. While this is sometimes unavoidable, each module should keep its algorithmic latency at a minimum, as latency accumulates in the processing chain and the end-to-end delay can become quite large. A low total latency is mandatory for applications where responsiveness is important.

Scalability It is hard to predict how much computational resources will be available for auditory processing on a robot. Therefore, the computational complexity of the localization module should be statically and possibly dynamically scalable. Scalability always implies a trade-off between different optimization parameters. Lower computational complexity is often bought with lower accuracy or a higher latency. With static scalability the final algorithm can be tuned for each particular robot model to deliver the best results with the available resources. Dynamic scalability would make it possible to adjust the complexity of the algorithm during runtime, thus lowering the processor load when needed or enabling higher accuracy in certain situations.

1.2.2 Limitations of existing algorithms

Many existing techniques for the localization, tracking, and separation of sound sources are not applicable to robot auditory systems due to the following limitations:

- An algorithm was not developed with the requirements of robot auditory systems in mind. As a consequence, the algorithm lacks some features that are vital to the applicability on robots.
- The performance of an algorithm does not hold up in real-world evaluations, as algorithms are often tested exclusively in simulations. Only testing under realistic conditions provides reliable information about the strengths and weaknesses of an algorithm.

For example, research from the field of computational auditory scene analysis (CASA) [29, 117] tries to solve audition problems, but is often not directed at robots and is therefore usually eliminated by point one. On the contrary, techniques that were explicitly created for robot audition do not necessarily meet all requirements. There is research on robot audition that produced algorithms, which fulfill only a subset of all features, as the research focuses on special applications. To my knowledge, there is no artificial auditory system that performs localization, tracking, and separation while unconditionally fulfilling all necessary requirements.

It is much harder to decide if an algorithm is subject to the second limitation. There is no universally accepted framework for the evaluation of algorithms for auditory systems. Additionally, authors often evaluate only a few properties of their algorithm to illustrate one particular feature. A comparison and ranking of algorithms for audition is practically impossible by just using the reported results.

In summary, there are no algorithms that perfectly match the requirements and constraints of robot auditory systems. It is hard to identify which of the existing algorithms would be the best candidate for adoption to a robot.

1.3 Formulation of the research problem

In this thesis I investigate the components of a robot auditory system. Solutions in literature lack features that would make them generally applicable to any robot audition system. I want to derive and properly evaluate algorithms that meet all requirements and constraints for robot audition for the following three modules:

- A localization module that analyzes observed sound signals and determines the positions of all currently active sound sources.
- A tracking module that follows the movement of the sources that are found by the localization.
- A separation module that creates individual sound streams for every active source from the observations.

Each of these modules implements a complex functionality and therefore other components of a robot auditory system are beyond the scope of this work. Nevertheless, the algorithm design will take into account that the considered modules will have to interoperate well with other components. This interaction happens mostly in the form of data exchange and the algorithms should be able to share final and intermediate results with other modules.

1.3.1 Algorithms

Inspired by human and animal auditory systems this work will explore binaural approaches. The localization algorithm will process raw sound data that is recorded by two microphones. For the localization some important problems have to be addressed:

- Localization cues are hidden in the binaural observations. The localization algorithm has to detect these cues and exploit them to calculate the positions of the active sources. In the presence of multiple sources the localization cues will likely contradict and the localization algorithm needs to group cues that belong to the same source.
- The localization algorithm requires a method to reliably determine the number of active sound sources.

The tracking algorithm is closely related to the localization and processes its results. The design of the tracking algorithm will eventually have to account for the properties of the localization algorithm and for the following points:

- Some approaches in literature differentiate between the activity and observability of a sound source. This means that an active sound source is not necessarily always observable. Therefore, the tracking algorithm itself should determine which sources it considers active.
- Existing tracking algorithms sometimes assume a certain source movement or source signal characteristic. While this can improve the tracking for certain applications, the assumptions often do not hold in the general case. To be applicable to a wide range of applications, the tracking algorithm should keep the number of assumptions about the signal and movement of the sources as low as possible.

The biggest challenge for the separation algorithm is the limited number of sensors that makes the separation of more than two sound sources underdetermined. Separation does not have to be performed blindly, as the results of the localization and tracking module can provide information about the observed sound scene. The following points should be considered in regards to the separation:

- The separation can completely rely on the number of sources that was estimated by the tracking.
- If possible, the separation algorithm should reuse intermediate results from the localization or tracking to keep its computational complexity low.

1.3.2 Evaluation

The created algorithms have to be implemented and appropriately evaluated. A test data set for the evaluation of the properties of algorithms for auditory systems does not exist and has to be created. Appropriate test conditions have to be defined for the testing of the algorithm properties and comparing of different algorithms. The test data should include a representative number of recorded observations that can be used to test the average performance of an algorithm in the real world.

For the testing of individual algorithm properties a large amount of test data will be necessary. If it is practically unfeasible to record all this data, a valid simulation approach has to be identified and implemented. A method is necessary to quantify the difference between results created from simulated and recorded data.

Existing state-of-the-art methods that try to solve the same robot audition problems have to be identified. The overall performance of the developed algorithms should be compared to those techniques.

1.4 Contributions

In this thesis I define all requirements and constraints that have to be met by algorithms for a robot auditory system. I analyze the related research in the field of robot audition to identify state-of-the-art methods that meet the defined requirements as closely as possible.

My investigation concentrates on sound source localization, tracking, and separation modules. For each of these modules I derive a new algorithm that meets the requirements better than the state of the art. The algorithms are based on a binaural approach and require knowledge about the transfer functions between observer and sound source.

For the evaluation of the algorithms and their comparison to the state of the art, I define appropriate real-world *experiments* and record extensive *test data sets*.

The *localization algorithm* estimates the positions of multiple sources simultaneously by exploiting signal sparseness in some transform domain. The localization algorithm can also output a measure for the certainty of an estimated position, and it can determine how many sources are observable at a certain point in time. Detailed evaluations reveal that the localization algorithm has a better accuracy than state-of-the-art techniques. The testing of the algorithm properties shows the behavior of the localization in regards to interfering sources, noise, and reverberation.

My proposed *tracking algorithm* processes the positions that are determined by the localization. The algorithm can properly assign observations to the tracked sources and filters the localized positions using a Sequential Monte Carlo simulation. The localization of moving sources is prone to front-back confusions, where sources in the front or back hemisphere are detected in the opposite hemisphere. This can be interpreted as a non-linear measurement error, for which the tracking accounts with bimodal probability distributions. Source movement is modeled with a Langevin process and the tracking can handle stationary and quickly moving sources. Unlike other tracking approaches, it does not make assumptions about movements statistics. The functional capability of the tracking system is verified with adequate real-world tests.

The sound source *separation algorithm* performs binaural masking to segregate the individual source signals. The necessary masks are not created directly from the observations. Instead, the separation is tightly integrated with the localization algorithm and reuses its internal knowledge about the activity of each source in different parts of the signal spectrum. This integration enables an extremely low complexity for the separation of sound sources. With the information about the source position from the tracking, the separation algorithm can also decide which sensor is closer to the source and use the better observation for the segregation. The separation quality of the proposed algorithm is superior to state-of-the-art methods for robot audition and also compares well to state-of-the-art blind source separation systems.

My results show that algorithms for auditory systems can be designed to better meet the requirements and constraints relevant for robot audition. The performance of such algorithms matches and even surpasses the performance of existing methods that are not restricted in the same way.

1.5 Overview

The remainder of this thesis is organized as follows. In Chapter 2 I describe the basic approaches that can be used to localize, track, and separate sound sources. Chapter 3 shows an overview of the major existing robot auditory systems and compares their performance. I choose the three systems that yield the best performance to later compare them against my own algorithms.

In Chapter 4 I derive my localization algorithm and introduce the signal properties it exploits. For the evaluation of the localization algorithm a test data set is necessary. Chapter 5 describes the experiment setup and recording of the test data, while Chapter 6 uses this data to evaluate the properties of the localization algorithm and to compare it against competing algorithms.

In Chapter 7 I derive a tracking approach that can be well integrated with the localization algorithm. A short evaluation of the tracking algorithm is also given. Chapter 8 presents the separation algorithm, which is tightly coupled with the localization. The performance of the separation algorithm is evaluated and compared to the competing algorithms. Chapter 9 gives a short summary and concludes this thesis.

2 Fundamentals of auditory processing

This chapter formulates the problems that sound source localization, tracking, and separation try to solve. It introduces the fundamental approaches that existing auditory systems use to solve these problems.

2.1 Coordinate systems

Sound localization algorithms calculate the relative position of a source in respect to the observer. Localization algorithms are better at determining the direction from which a sound is coming than estimating the distance of the sound source. When interpreting localization results the use of a spherical coordinate is more descriptive.

Depending on the context, this thesis uses the Cartesian or the spherical coordinate system shown in Figure 2.1. The observer is located in the origin and looks in the direction of the y-axis towards the point (0/1/0). The poles of the spherical coordinate system are above and below the observer at $(0/0/\pm 1)$. Elevation and azimuth are zero for a point directly in front of the observer. The elevation angle θ is in the range $[-90°, \dots, 90°]$ and is positive for points with a positive z-coordinate. The azimuth angle φ is in the range $[-180°, \dots, 180°]$ and is positive for points with a negative x-coordinate.

2.2 Mixing models

In a real-world environment there can be any number of sound sources at any given time. When an artificial auditory system observes the world with one or multiple microphones, each microphone will record a mixture of all active sound sources. Based on the physics of sound wave propagation, existing algorithms for sound source localization or separation assume a more or less complicated mixing model for the sound sources. The two most frequently assumed models are linear mixing and convolutive mixing.

In the following sections the variable M will denote the number of microphones and the variable N the number of active sound sources.

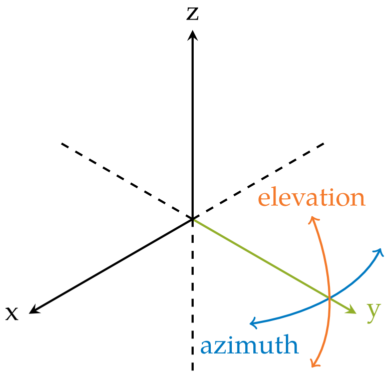


Figure 2.1: Cartesian or spherical coordinate systems used in this thesis. The observer is located in the origin and looks in the direction of the y-axis towards the point (0/1/0). In spherical coordinates the direction of a sound source is given by the elevation and azimuth angle. Elevation and azimuth are zero for a point directly in front of the observer.

2.2.1 Linear mixing

This model takes into account that sound waves travel at a finite speed and their intensity diminishes with the traveled distance. The observation $x_j(t)$ of a sound signal $s_i(t)$ at microphone j is then given by

$$x_j(t) = a_{ji} \cdot s_i(t - \tau_{ji}), \tag{2.1}$$

where a_{ji} is the attenuation of the sound source, and τ_{ji} is the time needed to travel from sound source i to microphone j. Both parameters depend solely on the distance between source and sensor. An illustration of the mixing is depicted in Figure 2.2. The observation of a complete sound scene is then

$$x_j(t) = \sum_{i=0}^{N} a_{ji} \cdot s_i(t - \tau_{ji}),\ j \in [1..M], \tag{2.2}$$

which is a linear superposition of multiple sound sources. One variant of linear mixing only considers gain and assumes that all delays are zero. As each sensor records an instantaneous mixture of all active sources, it is referred to as instantaneous mixing.

2.2.2 Convolutive mixing

Linear mixing only models the sound waves that travel from the source directly to the microphones. However, in real environments, sound is reflected, refracted and diffracted by surfaces and objects. Therefore, in addition to the direct component, time-delayed reflections of all sound signals will also reach each sensor. The convolutive mixing model takes these reverberation effects into account. Figure 2.3 shows a simple example of the observation of a sound source with reverberation. Reverberation depends on the positions of source and sensor in the environment and can be represented

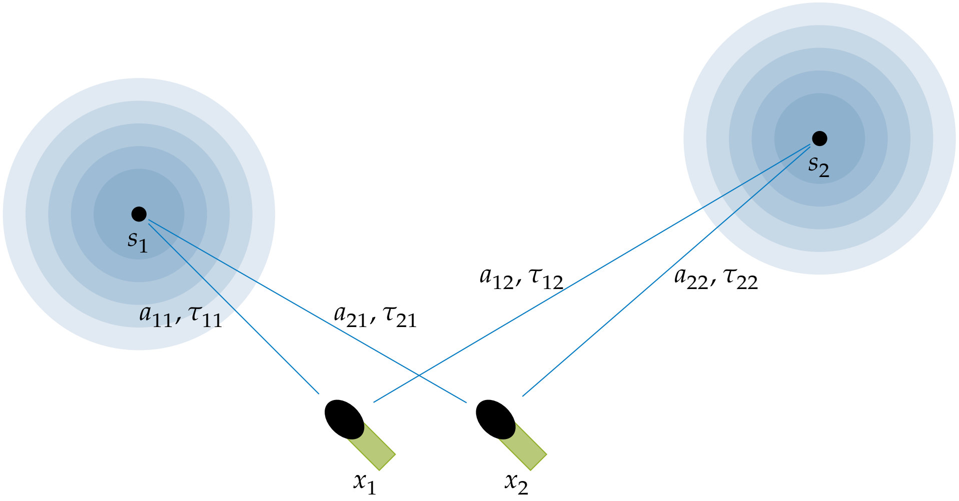


Figure 2.2: The linear mixing model. A time-delayed and attenuated version of each source signal is received by each microphone.

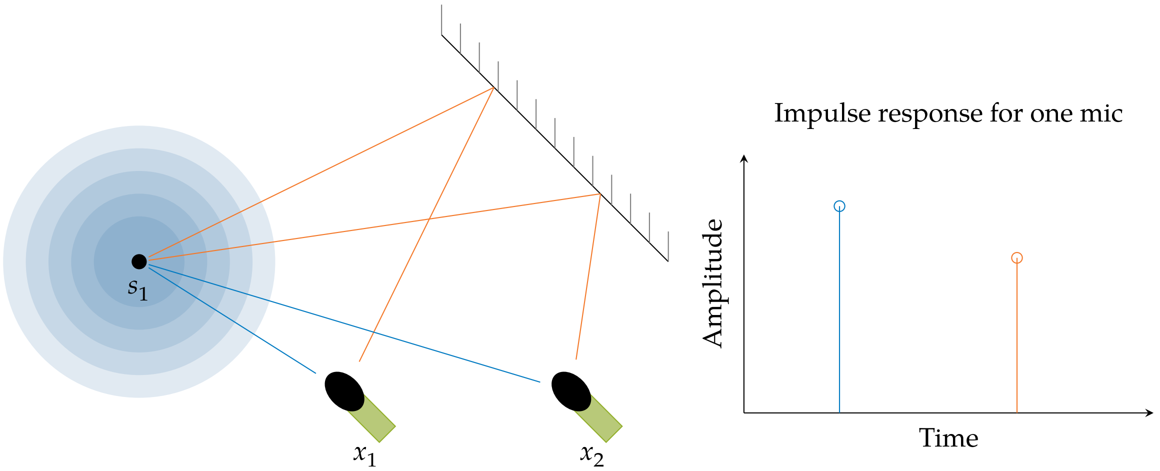


Figure 2.3: The convolutive mixing model. In addition to the direct path, reflections of a sound source are also observed by the microphones. Reverberation can be modeled by filtering with an appropriate transfer function. For this simplistic multipath scenario with one reflection the corresponding filter impulse response has two peaks. The first peak corresponds to the direct sound and occurs earlier than the reflection, which has also a lower amplitude. Each source-sensor position pair has its own transfer function and in reality the impulse responses are continuous signals.

by a position dependent transfer function or its time-domain equivalent impulse response $h_{ji}(t)$. The observation of one sound source in the reverberant model is

$$x_j(t) = h_{ji}(t) * s_i(t), \tag{2.3}$$

where $(*)$ denotes the convolution operator. Equally to the linear mixing, the observation of a sound scene is acquired by the superposition of multiple single sources

$$x_j(t) = \sum_{i=1}^{N} h_{ji}(t) * s_i(t),\ j \in [1..M]. \tag{2.4}$$

As convolution in the time-domain corresponds to a multiplication in the frequency domain the mixing can also be written as

$$X_j(f) = \sum_{i=1}^{N} H_{ji}(f) \cdot S_i(f),\ j \in [1..M],\ f \in [1..F], \tag{2.5}$$

with $X_j(f)$, $H_{ji}(f)$ and $S_i(f)$ being the frequency domain equivalents of $x_j(t)$, $h_{ji}(t)$ and $s_i(t)$. F is the number of frequency bins obtained by the discrete Fourier transformation of the time signals. For the binaural case this can be written in matrix notation as

$$\begin{bmatrix} X_1(f) \\ X_2(f) \end{bmatrix} = \begin{bmatrix} H_{11}(f) & H_{12}(f) \\ H_{21}(f) & H_{22}(f) \end{bmatrix} \cdot \begin{bmatrix} S_1(f) \\ S_2(f) \end{bmatrix},\ f \in [1..F] \tag{2.6}$$

if two sources are active and

$$\begin{bmatrix} X_1(f) \\ X_2(f) \end{bmatrix} = \begin{bmatrix} H_{11}(f) & H_{12}(f) & H_{13}(f) \\ H_{21}(f) & H_{22}(f) & H_{23}(f) \end{bmatrix} \cdot \begin{bmatrix} S_1(f) \\ S_2(f) \\ S_3(f) \end{bmatrix},\ f \in [1..F] \tag{2.7}$$

for three active sources.

Reverberation is caused by many factors like the walls and floor of a room or objects in the environment. Even the sensors of the auditory system cause reverberation. Some localization and separation algorithms assume that only some of these effects are dominantly present and they accordingly use impulse responses that only model these effects.

2.3 Sound source localization

Most existing sound source localization algorithms that have been successfully implemented on robots fall in one of the following four major categories:

- Exploiting binaural cues
- Exploiting spectral effects
- Steered beamforming
- Subspace methods

Each of these approaches is briefly introduced in the following sections.

2.3.1 Exploiting binaural cues

Most auditory systems in nature have two acoustic sensors and are therefore called binaural systems. Binaural cues occur when comparing the signals of spatially separated sensors. Assuming the linear mixing model from Section 2.2.1 and only a single sound source, one microphone signal will be a time-delayed and amplitude-scaled version of the other microphone signal. The delay and the scaling are both binaural cues and are called interaural time difference and interaural level difference respectively. Lord Rayleigh [97] observed these effects studying the hearing of human subjects. He proposed the duplex theory, which states that the human hearing system uses time differences to localize low frequency sounds and level differences to localize high frequency sounds.

Interaural time difference

The speed of sound c is 343 m/s in dry air at a temperature of 20°C. If the sensors are not equidistant from the sound source, the time delay of arrival (TDOA) of the sound wave at each sensor will be different. This interaural time difference (ITD) is zero for sources in the median plane between the two sensors and reaches its maximum for sources on the axis that goes through both microphones. For humans the ITD is maximal for sources on the left or right side of the head and reaches a value of approximately 0.6 ms [16].

Figure 2.4 shows how the angle of an arriving wavefront affects the time delay of arrival between two microphones. The ITD can be used to estimate this angle of arrival in the horizontal plane. The time difference between the two observed signals $x_1(t)$ and $x_2(t)$ is extracted by

$$\Delta t = \arg\max_{\tau} (x_1 \star x_2)(\tau) = \arg\max_{\tau} \int_{-\infty}^{\infty} x_1(t) x_2(t+\tau) dt, \tag{2.8}$$

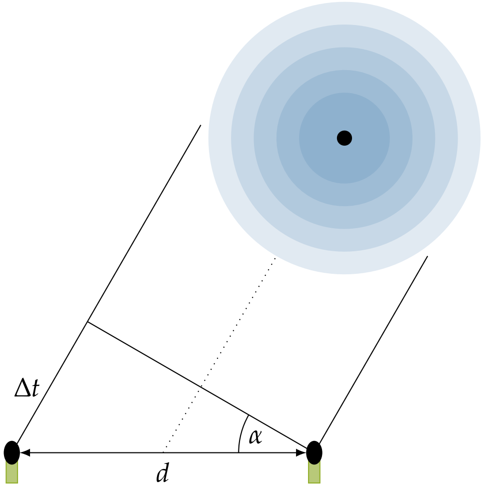


Figure 2.4: The time difference of arrival Δt of a sound depends on the azimuth angle α of the source. The wavefronts of the sound source are assumed to be planar.

where $\star$ denotes the cross-correlation operator. From the time delay the azimuth can be calculated by

$$\alpha = \arcsin\left(\frac{\Delta t \cdot c}{d}\right), \tag{2.9}$$

where d is the distance between the two microphones. This estimation is only possible in the far field where planar wave fronts can be assumed.

The cross-correlation can also be calculated in the frequency domain using

$$(x_1 \star x_2)(\tau) = \int_{-\infty}^{\infty} X_1(f) X_2^*(f) e^{i2\pi f \tau} df, \tag{2.10}$$

where the $(\cdot)^*$ operator denotes the conjugated complex. Knapp and Carter proposed a generalized cross-correlation method (GCC) with the phase transform (PHAT) weighting function [59], which normalizes the magnitude at each frequency to 1 and thus only takes phase difference into account. The time delay between two signals is then calculated by

$$\Delta t = \arg\max_{\tau} \int_{-\infty}^{\infty} \frac{X_1(f) X_2^*(f)}{|X_1(f) X_2^*(f)|} e^{i2\pi f \tau} df. \tag{2.11}$$

GCC-PHAT creates better discriminable peaks in the cross-correlated results and is more robust towards reverberation than classical cross-correlation.

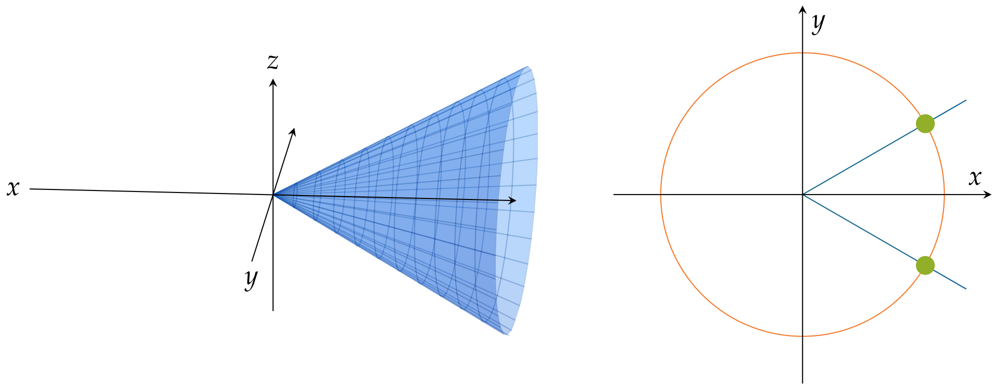


Figure 2.5: The cone of confusion. ITD and ILD based localization yields ambiguous results that lie on the cone of confusion (left). The two microphones are positioned equidistantly from the origin on the x-axis. Some algorithms assume that sources lie on the horizontal plane (z=0) and that they have a fixed distance to the origin. The two possible source positions then lie on the intersection of a circle with the cone of confusion (right). One of the green positions is the true source position and the other is its front-back confusion.

Interaural level difference

The intensity of a sound wave diminishes with traveled distance and observations of sound sources not equidistant to the sensors will have an interaural level difference (ILD). When a sound source is on one side of the head of a human, the observation at the contralateral ear will be additionally attenuated by the head shadow. Lord Rayleigh observed [97] that head shadow effects depend on the frequency of the sound and are dominant at higher frequencies, whereas sound with larger wavelengths passes through the obstructing object and attenuation is not as pronounced. Depending on the hardware design of an artificial hearing system the head shadow will also play a role during acoustic observation. Similarly to the ITD, the ILD is also often used to determine the azimuth of the sound source in the horizontal plane.

Limits of ITD and ILD localization

Interaural level difference and interaural time difference can only be calculated from sound signals with one active source. In the presence of a second sound source these difference estimations become unreliable and will not produce meaningful results.

The 3D position of a sound source is defined by three parameters. Interaural difference based localization can only estimate one of those parameters, namely the lateral position of the source. It cannot provide meaningful information about the vertical angle or distance of a sound source. Since only one of three parameters can be determined, the localization does not yield one unique point as the localization result, but a set of unlimited possible source locations that lie on a 2D surface embedded in 3D space. The surface is a cone (see Figure 2.5) that lies on the axis connecting the two sensors and is called the *cone of confusion*.

To remedy these ambiguities, some localization algorithms assume that all sources lie on the horizontal plane and have a fixed distance to the observer. This means that the algorithm expects all possible sources to lie on a circle around the observer. This assumption reduces the number of ambiguous points to two, which are in the intersecting set of the cone and the circle. The two points are at the same lateral angle, one in front of the observer, the other behind it. A sound source at any of those two points will create identical ITD and ILD sensations to the observer. Since those two points are still indistinguishable with ITD and ILD, one point is called the front-back confusion of the other. The human auditory system solves front-back confusions by small unconscious head movements.

2.3.2 Exploiting spectral effects

Sound source localization based on ITD and ILD cues is limited to one dimension. For the localization of the other two parameters, additional cues are necessary. The human auditory system relies on spectral effects, especially on higher frequencies [37], to determine the vertical angle of a sound source.

The timbre of a sound signal changes when it is perceived from different directions [104]. The outer ear, head and torso of an observer cause reflection, refraction and diffraction of the sound wave and modify the signal spectrum. These modifications depend mostly on the shape of the pinna and are unique for each source direction. They can be represented as direction dependent filters in form of transfer functions of their respective impulse responses. In the context of human or animal auditory studies these filters are called head-related transfer functions (HRTFs) or head-related impulse responses (HRIRs). Head-related effects can be modeled by convolving a sound signal with the HRTF filters corresponding to the source directions.

The HRTFs for each direction are stored in a database and for localization they have to be known a priori. HRTF databases are usually obtained by measurement of the ear canal response to stimuli from different directions. Custom databases can also be calculated from existing ones with regression models that match the observers anthropometric parameters to the characteristics of the transfer functions [100]. The observations of a sound source from an unknown direction are distorted by one HRTF filter pair from the database. The localization algorithm can analyze these frequency dependent distortions to identify which filter is active in the observations.

The spectral effects can also be seen as interaural phase differences (IPDs) and interaural intensity differences (IIDs), which describe the time delay and level difference between the two observations at each frequency.

2.3.3 Steered beamforming

Beamforming is a technique for directional signal transmission or receiving. It is used in communications engineering with antenna arrays to improve the receive/transmit gain. It combines signals from multiple sensors, so that the signal from a desired direction is enhanced by constructive interference, while all other directions are suppressed by destructive interference. In acoustics, beamforming enables a microphone array to adjust its directivity and to listen to a particular source direction.

The most simple approach to achieve spatial selectivity is the delay-and-sum beamformer. Assuming a linear mixing model, the observations at each sensor will be the superposition of the time-delayed signals of all sources. The delay-and-sum beamformer delays the individual observations, so that the sound coming from one direction is aligned in time, and it adds the signals. The addition reinforces the signal originating at the desired position and reduces interference and noise. The filter-and-sum beamformer applies a frequency dependent weighting function to the observations before addition.

Localization with a microphone array is mostly performed analyzing the steered response power (SRP) [18, 17]. If the phase transform (Section 2.3.1) is used as the weighting function in a filter-and-sum beamformer, the localization approach is referred to as SRP-PHAT. The localization algorithm first defines a spatial search grid for all potential sound source positions in the room. It then steers the directivity of the microphone array to each potential source position and calculates the power of the beamformer output. The source positions yielding the highest output powers are most likely to inhabit an actual sound source.

2.3.4 Subspace methods

Methods based on the signal and noise subspaces exploit the Cross-Sensor Covariance Matrix (CSCM) of the microphone array signals to estimate a set of constant parameters upon which the observations depend. Two well-known algorithms from this family that can be used for high-resolution direction-of-arrival (DOA) estimation are MUltiple SIgnal Classification (MUSIC) [103] and ESPRIT [101].

MUSIC assumes that the received signals at each array element are linear combinations of the sound sources and noise. It calculates the covariance matrix of the microphone array observations and performs a principal component analysis on this matrix to separate the disjoint signal and noise subspaces. The eigenvectors belonging to the largest eigenvalues span the signal subspace, whereas the remaining eigenvectors span the noise subspace. Based on the array geometry MUSIC estimates the distance of the array's steering vectors for all possible directions to the noise subspace and picks the most likely sound source directions. MUSIC works well with large arrays and needs at least N sensors to localize $N-1$ sources, but it is computationally expensive due to the necessary eigenvalue decomposition. The ESPRIT algorithm exploits the data model in a similar fashion and adds requirements to the geometry of the sensor setup. ESPRIT has a lower computational complexity compared to MUSIC and is more robust towards lower SNRs, but requires twice as many microphones to localize the same number of simultaneous sound sources.

2.4 Sound source tracking

Sound source tracking can be used for filtering localization results in order to make them more robust or to follow the position of a moving sound source. Existing tracking approaches are mostly from the following two categories:

Artificial neural networks A neural network consists of interconnected neurons, which are organized in different layers. Sound localization cues can be fed into the input layer and filtered tracking results that are calculated by the network become available at the output neurons. Neural networks have been successfully used for source tracking [46, 36, 41, 116] or for the integration of multimodal cues [1].

Bayesian filtering The Kalman filter [50] was successfully employed to smooth sound source localization results [85, 95, 58, 106]. As Kalman filters are restricted to linear state transitions and purely Gaussian noises, several authors propose using Sequential Monte Carlo simulations, also known as particle filters [35, 47], for the tracking of moving sources [112, 119, 118, 60, 72, 13]. Bayesian filtering predicts the next position of a sound source with a process model using previous states of the source. A measurement model updates the prediction with the current sound localization results.

The biggest drawback of neural networks is the need to train the networks with a large representative set of real-world data. Therefore, their usability for robots in unrestricted dynamic environments is limited.

The biggest challenge in source tracking is not so much the filtering of source movements, but the problem of assigning observations correctly to tracked sound sources and accounting for the possibility of new sources becoming active or existing sources being silent. Valin et al. suggest a solution for this assignment problem based on a particle filter [108].

2.5 Sound source separation

The human auditory system excels at separating individual sources from an observed mixture of sound signals. Humans can focus on a single speaker in spite of multiple simultaneous conversations and noises in the background. This ability is referred to as the cocktail party effect. Numerous different sound source separation algorithms have been proposed to implement this ability on artificial auditory systems. The approaches can be grouped into four major categories:

- Blind source separation
- Steered beamforming
- Inverse filtering
- Binary masking

The following sections will briefly discuss each approach.

2.5.1 Blind source separation

Most blind source separation (BSS) techniques rely on the independent component analysis (ICA) to separate the sources signals from multiple observed mixtures. ICA assumes that the observations are linear combinations of the source signals and that the sources are pairwise statistically independent. In matrix notation the mixing can be described as a multiplication of the source signal vector with a mixing matrix. In blind source separation the mixing matrix is unknown and ICA estimates the inverse matrix except for a permutation and scaling ambiguity. Since the classical ICA cannot demix signals that are observed with different time delays at different sensors, the best approach is separation in the frequency domain. As seen in Section 2.2.2 the convolutive mixing model equates to instantaneous mixing at each frequency. The blind source separation is then performed in Fourier domain at each frequency bin individually using complex ICA. Since the results between neighboring frequency bins are possibly permuted and scaled, the demixed results of all bins have to be aligned and scaled in order to obtain the source signals.

The main drawbacks of blind source separation are its high computational costs, mostly due to the separate demixing of each frequency bin. Since ICA is a statistical approach, it has to operate on blocks of observed data that are large enough to capture the statistics of each source. This block operation mode limits the minimal possible latency of the sound source separation.

In the context of artificial auditory systems, as they are considered in this work, much information about the sources and their locations in particular is already known. Blind source separation takes neither this knowledge nor the available information about the sensory setup of the auditory

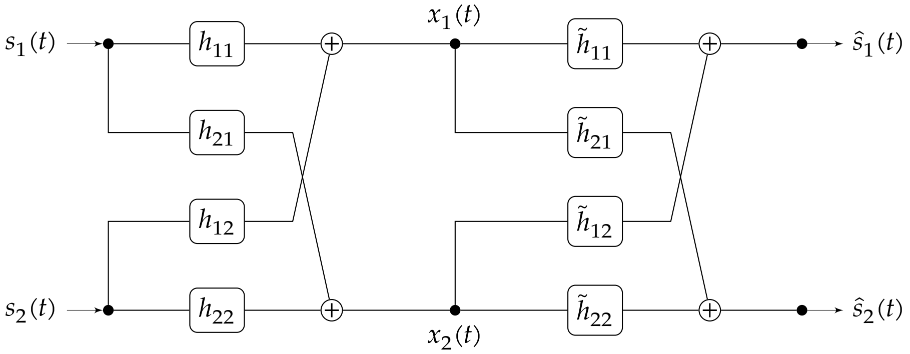


Figure 2.6: A crosstalk cancellation network used for sound source separation. The left half of the Figure illustrates the mixing process. The filters on the right side can demix the sources perfectly if they are chosen appropriately.

system into account. Therefore, the applicability of BSS approaches for simultaneous sound source localization and separation is limited.

2.5.2 Steered beamforming

As stated in Section 2.3.3 the sensitivity of a microphone array can be focused to any point in the environment. The array is able to actively listen to one particular position in the environment. The steering vector has to be known a priori and is usually acquired by previous localization of sound sources with the microphone array.

There are some extensions, e.g. the generalized sidelobe canceller (GSC), that try to further minimize the crosstalk from interfering sources in a delay-and-sum or filter-and-sum beamformer.

2.5.3 Inverse filtering

As shown in Section 2.2.2 the convolutive mixing model corresponds to instantaneous mixing at each frequency in Fourier domain. Binaural sound source localization using spectral effects usually tries to identify the HRTF filters that where used for the convolutive mixing. Once those filters are identified, the mixing is completely known to the separation process and can be inverted. Two sources can be demixed by

$$\begin{bmatrix} S_1(f) \\ S_2(f) \end{bmatrix} = \begin{bmatrix} H_{11}(f) & H_{12}(f) \\ H_{21}(f) & H_{22}(f) \end{bmatrix}^{-1} \cdot \begin{bmatrix} X_1(f) \\ X_2(f) \end{bmatrix}, f \in [1..F] \tag{2.12}$$

at each frequency individually. This kind of inverse filtering is also known in literature as crosstalk cancellation. Figure 2.6 shows the cancellation network for two sources and two observations. The entries of cancellation filters are the entries of the inverse mixing matrix

$$\begin{bmatrix} \tilde{H}_{11}(f) & \tilde{H}_{12}(f) \\ \tilde{H}_{21}(f) & \tilde{H}_{22}(f) \end{bmatrix} = \begin{bmatrix} H_{11}(f) & H_{12}(f) \\ H_{21}(f) & H_{22}(f) \end{bmatrix}^{-1} . \tag{2.13}$$

This approach works in environments with slight reverberation. However, as the amount of reverberation increases, the inversion of the impulse responses becomes numerically unstable and additional regularization steps become necessary.

The same separation process can also be applied in the underdetermined case, when more than two active sources are present. Let the mixing process in the three-source case at each frequency in a compact notation be defined by

$$\boldsymbol{x} = \boldsymbol{H} \cdot \boldsymbol{s},\ \boldsymbol{x} \in \mathbb{C}^2,\ \boldsymbol{H} \in \mathbb{C}^{2\times 3},\ \boldsymbol{s} \in \mathbb{C}^3. \tag{2.14}$$

Since the mixing matrix is singular and cannot be inverted directly, there is no unique solution to this problem. The most obvious approach would be to use the Moore-Penrose pseudoinverse $\boldsymbol{H}^+ = \boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$ for demixing, which is equivalent to the least square solution of the optimization problem

$$\min ||\boldsymbol{H} \cdot \boldsymbol{s} - \boldsymbol{x}||_2. \tag{2.15}$$

However, this solution does not yield satisfying results and another constraint has to be added. Sound signals like speech are known to be sparse and one possibility to solve the inversion is to assume sparse source signals. In this case the desired solution $\boldsymbol{s}$ is the one that lies in the solution space of the mixing problem and has minimal ℓ_1-norm at the same time. The vector $\boldsymbol{s}$ that solves the optimization problem

$$\min ||\boldsymbol{s}||_1 \ \text{ s.t. } \boldsymbol{H} \cdot \boldsymbol{s} = \boldsymbol{x} \tag{2.16}$$

fulfills this condition. In the two-source case the demixing depends only on information from the HRTF filters. With three sources the demixing process is tuned to choose the sparsest solution that could have created the mixture.

2.5.4 Binary masking

Binary masking assumes that different parts of an observation's spectrum are dominated by only one sound source. If a sound source localization algorithm can determine which parts of the spectrum are dominated by which sound source, binary masking can be used to segregate the sources. For each sound source the algorithm creates a binary matrix that indicates what parts of the spectrum are dominated by the respective sound source. Applying these binary matrices as masks to the observations, the spectra of all sound sources can be approximated. Masking out interfering sources does not estimate the source signal at its point of origin, but it recovers the observation of each sound source at each sensor without interference. One of the most prominent sound source separation algorithms that perform binary masking is the Degenerate Unmixing Estimation Technique (DUET) [98].

Masking is not limited to Fourier domain and can be performed in any other transform domain, as long as the signal representations in the respective domain are sparse [94]. Binary masking assumes sparseness of the signal spectra and its performance deteriorates if the source spectra overlap. The spectrum of a segregated source signal is zero at all time-frequency points, at which one of the other sources was dominant. Masking cannot determine the correct values of these holes and their presence can cause audible distortions in a signal. Some binary masking algorithms post-process the masks or segregated signals to minimize these effects [4].

3 Related work

This chapter presents the previous work done in the field of robotic hearing. The first section gives an overview of existing robotic hearing systems and the following section compares these systems in terms of their capabilities.

3.1 Survey of robot auditory systems

One of the first robots to implement advanced sound source localization was the Cog [19]. The localization system proposed by Irie [46] extracts eight different ITD and ILD cues from binaural recordings in the time domain and in the frequency domain. It feeds the cues to a neural network, which estimates the source position. These estimations are compared to visual feedback obtained by the Cog's cameras and the resulting error signal is propagated back into the neural network. The neural network is trained with a training data set and afterwards tested with a validation data set. The localization performance on the validation data is poor, which according to the author is most likely caused by overfitting of the network to the training set. The localization system is limited to one active sound source by its design.

Hashimoto et al. [40] implement one-dimensional sound source localization based on IPD on the Hadaly robot. Research on the ROBITA robot by Matsusaka et al. [69, 68] enable the robot to follow a conversation between multiple people. Their research concentrates on multi-modal integration and detailed information on the sound processing is not available.

Huang et al. create a robot that navigates towards an acoustic beacon using sound localization [45]. The robot is equipped with three microphones arranged in an equilateral triangle. The system models the precedence effect in order to avoid localizing echoes [44, 43]. According to the authors its onset detection is robust in respect to stationary noise. The actual localization is performed by TDOA estimation between pairs of microphones in one frequency band. The robot localizes sources in the horizontal plane. Experiments reveal that the robot is able to navigate to sound sources that are occluded by other objects. The system shows good accuracy when localizing narrowband signals and hand-clapping, but according to the authors it is not able to deal with gentle-slope-onset sound like speech.

Nakadai et al. implement an active audition system [78, 79] for the upper-torso humanoid robot SIG [57]. Based on auditory epipolar geometry the authors propose a binaural localization system. It is able to localize two sound sources in one dimension and change the position of the SIG's cameras and microphones by motor control. The system also includes two additional microphones, which are used for the canceling of motor noise. The localization system is tested in experiments with pure tones as excitation signals. Nakadai et al. extend this approach in [84] to speech sources with a localization error ±15° and ±30° for the first and second sound source respectively. One drawback of this technique according to the authors is its susceptibility to front-back confusions. Nakadai et al. combine their sound localization with the SIG's cameras into a multi-modal speaker tracking system [75, 76, 91, 85]. The authors report a good performance with a tracking delay of 200 ms. Okuno et al. [91, 92] integrate this tracking into a simple human-robot interaction scenario. Nakadai et al. present a sound separation system built on active direction-pass filtering in [77, 86]. It shows a 6-10 dB SNR improvement when separating two active speech sources that are more than 30° apart in azimuth direction. Furthermore, they improve the sound localization and separation for sources in front of the robot [80] and implement a speech recognition system for simultaneously talking speakers [87]. The system is trained for three distinct speakers and three source locations in front of the robot. The speech recognition system is then able to recognize simultaneous utterances by the same speakers from the training set.

Choi et al. [27] implement a speech enhancement system for service robots. They process sound with a robust adaptive beamformer, which is a modified version of an adaptive generalized sidelobe canceler. The main feature of their approach is the connection of the adaptive canceling filters and adaptive blocking filters in feedback loops. It allows them to reduce the number of required filter taps compared to other source separation techniques. They test their method on a service robot with a circular array of eight microphones in a reverberant room.

The Jijo-2 robot uses a delay-and-sum beamformer for sound source localization and separation in a dialog system build by Matsui et al. [67]. Asano et al. [8, 9] implement a more advanced localization technique based on the MUSIC algorithm and use a minimum-variance beamformer for the sound separation in the near field.

Young and Scanlon equip an iRobot ATRV-2 with an eight microphone circular array for military applications [121]. The task of the robot is to detect and localize shots fired from sniper rifles in an urban environment. The eight microphones are divided into 56 different microphone triples and each triple estimates the azimuth and elevation of the sound source using TDOA. These intermediate results are median filtered to eliminate outliers and are subsequently used to steer the attention of the robot's cameras to the target. According to the authors, sound localization has to be suspended during pan-tilt unit movements, as the system is not robust to self-noise.

Nishiura et al. [90, 89] present a talker tracking system for a mobile robot that is equipped with a microphone array. The 16 microphones are arranged in a circle with a diameter of 60 cm. The algorithm localizes multiple sound sources by calculating the GCC between pairs of microphones. This approach is error-prone to front-back confusions and crosstalk effects between sound sources. To overcome these problems, the authors combine the results from multiple microphone pairs to get a robust localization estimation on the horizontal plane. The estimated source positions steer a delay-and-sum beamformer to obtain separated sound source signals. Additionally, the system is able to determine if a separated signal is a speech source. The authors test their approach in simulations with different parameters. They create a virtual auditory space with one speech and one non-speech signal at fixed positions. An autonomous mobile robot moves in this virtual space and successfully navigates towards the detected speech source.

Li and Levinson [61] implement binaural sound localization on an unspecified robot. Their algorithms measure the TDOA by cross-correlating the binaural inputs and unwrapping the phase of the result. The algorithm tries to find high energy segments in the signal spectrum with reliable phase information by using clustering ideas from pattern recognition. Under the assumption of a continuous spectrum, the algorithm uses this initial slope information to unwrap the phase in the whole spectrum and to calculate the final slope of the resulting signal. The TDOA estimate is used for the calculation of the azimuth angle and shows an accuracy of $\pm 10^\circ$ in real-world experiments.

Andersson et al. [3] enable a robot to navigate towards an acoustic source using a localization algorithm [38] presented by Handzel and Krishnaprasad. The robot's head is a hard spherical shell with two microphones mounted at antipodal points. Given this head model the theoretical IPDs and ILDs for all potential sound source directions can be calculated. For each recorded sound segment the localization algorithm computes the observed IPDs and ILDs in each frequency bin. It calculates the distance between the observed and theoretical values for all directions using an appropriate metric. The source direction is the one that yields the minimum distance. The authors restrict the algorithm to work on the horizontal plane. Due to the symmetry of the head, this approach suffers from front-back confusions and the authors remedy this problem by exploiting robot movement. The approach has a localization error of $\pm 2^\circ$ when localizing broadband noise signals. The authors also successfully conduct tests where the robot navigates acoustically towards a sound source while avoiding obstacles.

For the audio processing system of the humanoid HRP-2 robot [51] Hara et al. [39] implement a speaker tracking system based on audio and video fusion. They use arrays of eight microphones and the MUSIC algorithm for acoustic localization. A Bayesian network [10, 11] fuses estimated sound source locations with information obtained by a face tracker. The simultaneous occurrence of an audio and a video event in the same region of the environment is classified as a speech event.

Sound separation is performed by a maximum likelihood beamformer that is updated with the locations obtained by the information fusion. In experiments the authors show that the system is able to track and separate one standing and one moving speaker. Asoh et al. [12, 13] replace the Bayesian network with a more elaborate particle filter from Checka et al. [23] to track speech activity. The authors conduct an experiment with an HRP-2 head that is tracking one speech and one non-speech source simultaneously and report a speech detection error rate of 15%.

Martinson and Schultz [66] enable an iRobot B21R robot to create a map of active sound sources in its environment. They call this technique the auditory evidence grid. The robot is equipped with a rectangular shaped array of four microphones and uses a steered response power localization approach. The authors observe that localization errors are often concentrated along the axis going through the sound source and microphone array. In other words the estimation of the azimuth angle is more reliable than the estimation of the distance of the sound source. The purpose of the proposed algorithm is creating a map that indicates how probably a position in the environment is inhabited by a sound source. Initially, all possible positions in the room are initialized to a fixed probability. Sound sources are assumed to be continuously active and located at fixed positions in the room during mapping. The algorithm localizes sound sources with a steered response power (SRP) approach. It interprets the resulting power values as probabilities for source presence and updates its internal auditory map using a Bayesian approach. These localization and update steps are repeated several times from different positions in the room. With each iteration the map is refined and in the final step the algorithm extracts the positions of the active sound sources from the map. In real-world experiments the authors show that their scheme is able to successfully map up to two active sources. However, in trials with more sources the approach fails to detect at least one source. Trials with only two microphones show similar mapping capabilities as with the full array.

Murase et al. implement a multiple moving speaker tracking system [72] for the SIG2 robot. Their approach incorporates an eight microphone SRP beamforming algorithm [109] and a set of multiple Kalman filters that are used for tracking the source position. The authors suggest using multiple Kalman filters to solve problems with nonlinearities that occur in the real world. The state of their Kalman filters is represented by a history of past source positions and the filter update is performed using this movement history. The authors use a set of multiple filters with different history lengths to model sources with different movement characteristics. A longer history is better suited for continuous movement, whereas shorter histories can better model drastic velocity changes. The method is successfully tested on the SIG2 robot with up to three simultaneous sources.

Valin et al. present a robust sound source localization and tracking method [110, 107, 108]. They successfully apply it to stand-alone microphone arrays and different robot platforms (ActivMedia Pioneer 2, Spartacus and SIG2). Their approach uses a steered response beamformer with the phase transform weighting to perform the initial localization. The PHAT produces sharper cross-correlation peaks and therefore more reliable localization results. However, it has the drawback that all frequency bins of the spectrum contribute equally to the localization result, even if the observed signal has a bad SNR in some frequency bins. Therefore, the authors suggest an additional weighting step depending on the expected signal reliability at a specific frequency. The authors use the Minima-Controlled Recursive Average (MCRA) algorithm [28] to estimate the background noise. Additionally, they predict the amount of reverberation in one frequency bin by modeling the precedence effect [43]. The reliability of a specific frequency is then given by the a priori SNR, which Valin et al. calculate from the background noise and reverberation using the decision-directed approach presented in [32]. The beamformer calculates the energy for all points on its spherical search grid and extracts the locations of the first four dominant sources. In the second stage of the algorithm a particle filter tracks the sound sources. The salient feature of the proposed approach is its ability to solve the problem of assigning observations to sources and to dynamically add or remove sources from the tracking process. The authors test their approach in an experiment with two different array configurations and reported low error rates of $< 2°$. In [111] Valin et al. present a source separation and post-filtering algorithm that can be used to track sources.

Nakadai et al. [81, 82] present a method to localize sound sources by using a robot microphone array and room microphone array simultaneously. In the first step both arrays localize all sound sources individually. The room array has 64 microphones and uses weighted delay-and-sum beamforming [83] to estimate directivity patterns and locations of sound sources. The robot array has 8 microphones and localizes sources using the MUSIC-based approach presented in [39]. The authors implement a particle filter to combine the results from both arrays and to track the sources over time. They conduct experiments with a Honda ASIMO head serving as the mobile robot and test localizing up to two sources. The authors report improved tracking robustness and slightly reduced localization errors compared to localization with the room array only.

Keyrouz et al. present HRTF-based sound source localization and separation [56, 52, 55, 54, 53] aimed at mobile robots for telepresence applications [22]. The algorithm processes the binaural inputs in the STFT domain and constructs feature vectors containing IPD and IID information in each frequency bin. The algorithm clusters this data over time with the self-splitting competitive learning technique [122] and extracts one cluster center per source in each frequency bin. In its next step the algorithm has to assign each cluster center to its corresponding sound source. The authors solve this permutation problem by assuming smoothness between feature vectors of neighboring frequency bins. The algorithm iterates through all bins starting at the lowest frequency and chooses the permutation that minimizes the pairwise Euclidean distance between the aligned vectors of the current and previous bin. The algorithm compares the feature vectors of each sound source to all filter pairs in the HRTF database. The filter pair, which matches a source best, indicates the position of the source. The sound separation is performed by inversion of the mixing matrix in the frequency domain once the HRTF filters are known. The method is tested in simulations and the authors report good localization and separation results.

Murray et al. [74] present a method for tracking a single speech source using neural networks. Their algorithm determines the azimuth of the source by calculating the TDOA between two of its microphones. The tracking is performed by a recurrent neural network that takes azimuth estimations in the range of ±90° as inputs and outputs the tracked location of the sound source. The neural network learns the trajectory of the sound source by using previous and current source positions and is able to predict future locations of the source. To teach the network the temporal differences of sound sources moving at different angular speeds the authors provide the neural network with artificial training sets. These consist of input activation patterns and the desired output activations. Real-world experiments were conducted on an ActivMedia PeopleBot robot and the authors reported localization errors of ±1.5° to ±7.5° depending on the position of the source. Murray and Erwin also present an elevation estimation algorithm [73] based on monaural notch classification by a neural network. Their approach relies on spectral effects introduced by a pinna and they train a feed-forward neural network to detect elevation specific notches in the observation. The authors plan to incorporate this algorithm with a binaural localization and tracking system.

Berglund et al. [14, 15] use a parameter-less self-organizing map (PLSOM) and reinforcement learning to build an audition system that is able to learn sound localization in an autonomous fashion. A self-organizing map is a special kind of neural network that produces low-dimensional representations from high-dimensional data sets while preserving its topological properties. The map is created in an unsupervised learning process using input examples. The parameter-less variant of the SOM eliminates the need for manual setting of crucial learning parameters. The inputs for the PLSOM are feature vectors from the binaural sound signals, which are constructed mainly from ITD, IPD and IID. During localization, feature vectors are presented to the PLSOM and it outputs the position of the winning node, which corresponds to the direction of the sound source. This information is fed into a reinforcement learning system that can produce motor commands for the robot head and is trained to move the robot head to look at the sound source. Based on the feature vector the system is only able to detect sources in the horizontal plane and estimation of the elevation is performed by tilting the robot head and repeating the localization. The authors test this system on a Sony Aibo ERS-210 robot in real-world experiments and report localization results with error rates around ±5° and a latency of 0.5 s.

Chisaki et al. [26, 25, 24] present a localization system based on the frequency domain binaural model (FDBM). Nakashima et al. [88] introduce FDBM as a preprocessor for a speech recognition system or sound segregation system for humanoid robotics. FDBM builds a map of IPD and ILD values for all filter pairs from a HRTF database. For sound source localization FDBM calculates the IPDs and ILDs from the input signals and uses them to calculate scores for each filter in the HRTF database. The scores of each frequency bin are weighted with the signal energy to give more importance to bins with better SNR. Finally, the sound sources are localized by identifying the filters with the highest scores. FDBM separates the sound sources by building binary masks from the scores of the filters corresponding to the detected source directions. The authors test FDBM in simulations and report successful localization of two concurrent sound sources in azimuth and elevation direction with high accuracy.

3.2 Comparison of existing systems

This section tries to compare the existing robotic auditory systems with regard to the requirements and constraints defined in Section 1.2.1. Such a comparison proves to be difficult for several reasons.

Most of the algorithms were developed and optimized for different, sometimes completely custom robots. The comparability of the experiments performed with different robots is arguable, especially since every algorithm was tested under different conditions. Often important parameters of an experiment, e.g. the reverberation time of an environment, are unknown. Therefore, measurements of the localization accuracy cannot be compared between two different approaches. Since there is no standardized evaluation method for algorithms that implement auditory processing, the reported experiments often consist of ad hoc tests to demonstrate one particular property of an algorithm. For most algorithms only a subset of the properties necessary for an in-depth comparison is reported by the authors.

One property that is rarely mentioned is the latency of the algorithms. It could theoretically be determined from the algorithm description, but in practice this is mostly not possible due to unmentioned implementation details. If an algorithm is a frequency domain based approach, then the algorithmic latency is a function of the used FFT block size and depends on the actual implementation parameters.

Since an analysis with regard to all properties of interest is not possible, the algorithms are compared with regard to three objectively comparable properties. Table 3.1 lists the values of these properties for each algorithm. The first column denotes the number of microphones used for localization. As some approaches support a varying number of microphones, the given value corresponds to the number of sensors the authors used in their experiments. It can be expected that this number represents the necessary minimum for good localization results. The second column

Algorithm	# of microphones	# of sources	# of dimensions
Irie	2	1	1 (f)
Hashimoto et al.	2	1	1 (f)
Huang et al.	3	1	1
Nakadai et al. [78]	2	2 / 2	1 (f)
Choi et al.	8	0 / 3	2
Asano et al.	8	2	1
Young and Scanlon	8	1	2
Nishiura et al.	16	2	1
Li and Levinson	2	1	1 (f)
Andersson et al.	2	1	1
Asoh et al.	8	2	1
Martinson and Schultz	4	2 / 0	2
Murase et al.	8	3	2
Valin et al.	8	4	2
Nakadai et al. [81]	72	2 / 0	2-3
Keyrouz et al.	2	3	2
Murray et al.	2	1	1 (f)
Berglund et al.	2	1	1 (f)
Chisaki et al.	2	2	2

Table 3.1: Comparison of existing robot hearing systems by some features. The first column lists the number of microphones used for localization and separation. The number of sources the algorithm can handle are given in the second column. If the number of possible sources differs for localization and separation, they are given separately. The last column states in how many dimensions a source can be successfully localized. The suffix (f) denotes that the approach cannot distinguish between front and back. The highlighted algorithms have no apparent flaws that contradict any of the requirements and constraints defined in 1.2.1.

lists the number of sources that the algorithm was able to localize and separate successfully in experiments conducted by the corresponding authors. If the number of possible sources differs for localization and separation, they are given separately. The position of a sound source in 3D space is defined by three parameters and the third column denotes how many of these parameters each algorithm determines. Some sound localization algorithms that determine only the lateral angle via ITD and ILD cues assume that sources are in front of the robot and cannot resolve front-back confusions. These algorithms are marked with the suffix (f) in the last column.

This list can be filtered on the basis of two essential requirements of a sound source localization system, namely the ability to localize multiple sources and the ability to determine at least two position parameters. Six algorithms from the list fulfill these two conditions and are discussed in the following.

The algorithm from Martinson and Schultz creates an acoustic map of the environment in a process that involves observing the sound scene from different positions in the room. This approach has a high latency by design, as the algorithm has to gather sound information from several points in space before it can produce reliable results. The sound source positions are also assumed to be stationary during the whole observation process.

Nakadai et al. [81] report promising results, but their approach requires a huge room microphone array in addition to the robot audition system and is therefore not evaluated in this work.

The remaining four algorithms have no apparent flaws that contradict any of the requirements and constraints. The approaches are highlighted in Table 3.1. The algorithms presented by Murase et al. and Valin et al. are very similar in their design and have been partly developed by the same researchers. They both use the same steered beamformer with subsequent Bayesian filtering. Murase et al. use multiple Kalman filters to track a constant number of moving sound sources. Valin et al. implement a more elaborate particle filter that is able to track multiple sources and account for the appearance and disappearance of individual sources. Due to its advantages only the algorithm proposed by Valin et al. will be analyzed in this work. The remaining two algorithms by Keyrouz et al. and Chisaki et al. are binaural approaches and will both also be considered during the evaluation.

4 Localization of multiple sound sources

This chapter derives a localization algorithm respecting the requirements and constraints necessary for robot audition. Figure 4.1 shows the module in the scope of the robot auditory system.

4.1 Analysis of the cross-convolution localization algorithm

The design of the new localization algorithm is based on a single source localization approach, namely the cross-convolution localization (CCL) algorithm [106]. The inputs of the algorithm are observations from the two microphones and the database of transfer function (TF) pairs. CCL works in block mode and requires several seconds of input data at once to calculate the direction of the sound source reliably. It outputs the estimated elevation and azimuth angle of the source for the whole input block.

4.1.1 The CCL algorithm

CCL is build on the assumption that the perceived observations have been filtered with one particular TF pair η_0 from the database. The TFs are denoted by $h_{j,\eta}(t)$, where $j \in \{1, 2\}$ denotes the left ($j = 1$) or right ($j = 2$) microphone and $\eta \in \{1, \dots, N_h\}$ is the index of the filter pair in the database. Each of the N_h filter pairs belongs to one source direction (θ, φ) and the source directions can be

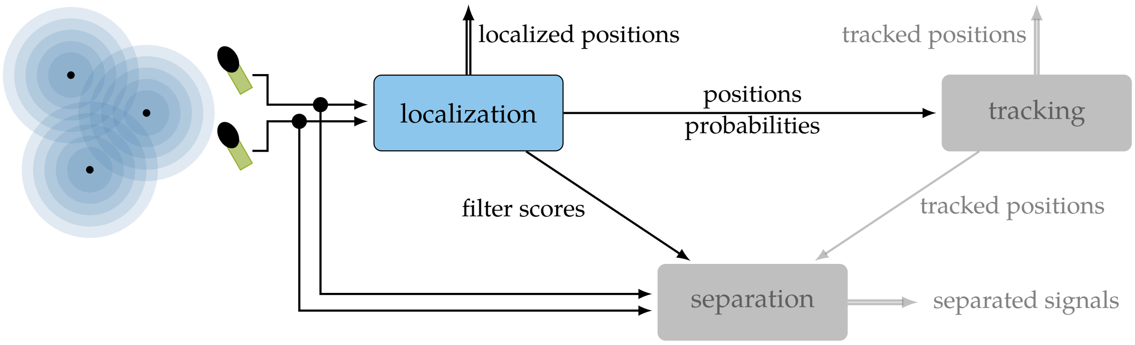


Figure 4.1: Signal flow diagram for the complete auditory system. The localization module calculates the positions and probabilities of all sources and provides them to other modules.

denoted with the indices of their corresponding filters. Localization of a sound source is equivalent to determining the particular filter pair η_0, which can be found by deconvolving the observations $x_j(t),\ j \in \{1,2\}$ with the filter pair of each direction η. The resulting signals $\hat{s}_{1,\eta}(t)$ and $\hat{s}_{2,\eta}(t)$ will be equal when the correct filter pair from the database is used for the deconvolution. This equality criterion can be expressed as

$$\hat{s}_{1,\eta}(t) = \hat{s}_{2,\eta}(t) = s(t), \iff \eta = \eta_0. \tag{4.1}$$

Since deconvolution is usually performed by dividing the spectra of two signals in the frequency domain, it can become numerically unstable if any of the filter entries are close to zero. The CCL algorithm introduces a trick to avoid this potentially problematic step. Each observation is convolved with the TF of the opposite microphone and the signals

$$\begin{aligned} \tilde{s}_{1,\eta}(t) &= h_{2,\eta}(t) * x_1(t) \\ \tilde{s}_{2,\eta}(t) &= h_{1,\eta}(t) * x_2(t) \end{aligned} \tag{4.2}$$

are obtained. Due to the associativity of the convolution operator the resulting signals will again be equal if the correct filter pair is selected:

$$\begin{aligned} \tilde{s}_{1,\eta}(t) &= h_{2,\eta}(t) * x_1(t) \\ &= h_{2,\eta}(t) * h_{1,\eta_0}(t) * s(t) \\ &= h_{1,\eta}(t) * h_{2,\eta_0}(t) * s(t) \\ &= h_{1,\eta}(t) * x_2(t) \\ &= \tilde{s}_{2,\eta}(t) \iff \eta = \eta_0. \end{aligned} \tag{4.3}$$

Real recordings $x_j(t)$ contain noise and additional distortions due to the recording environment and audio equipment. Also limitations of the numerical precision can cause the cross-convolution of the observations with the correct filter pair to generate slightly different signals. Therefore, the equality criterion in the previous equation must be replaced by the maximization of a similarity measure over all possible η. The similarity of the cross-convolved signals can be measured with a cross-correlation at time delay zero

$$(\tilde{s}_{1,\eta}(t) \star \tilde{s}_{2,\eta}(t)) \overset{\text{def}}{=} \int_{-\infty}^{\infty} \tilde{s}_{1,\eta}(\tau)\tilde{s}_{2,\eta}(\tau)d\tau. \tag{4.4}$$

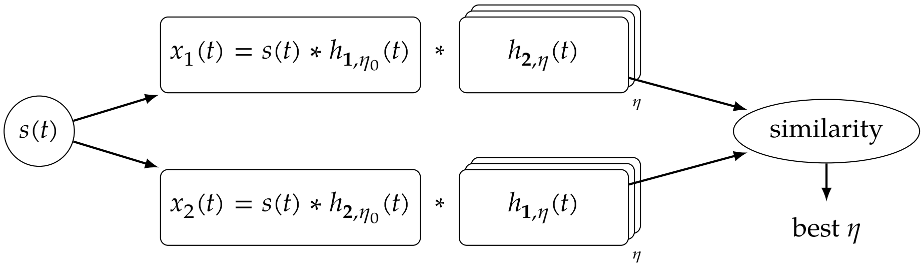


Figure 4.2: Schematic view of the cross-convolution localization algorithm.

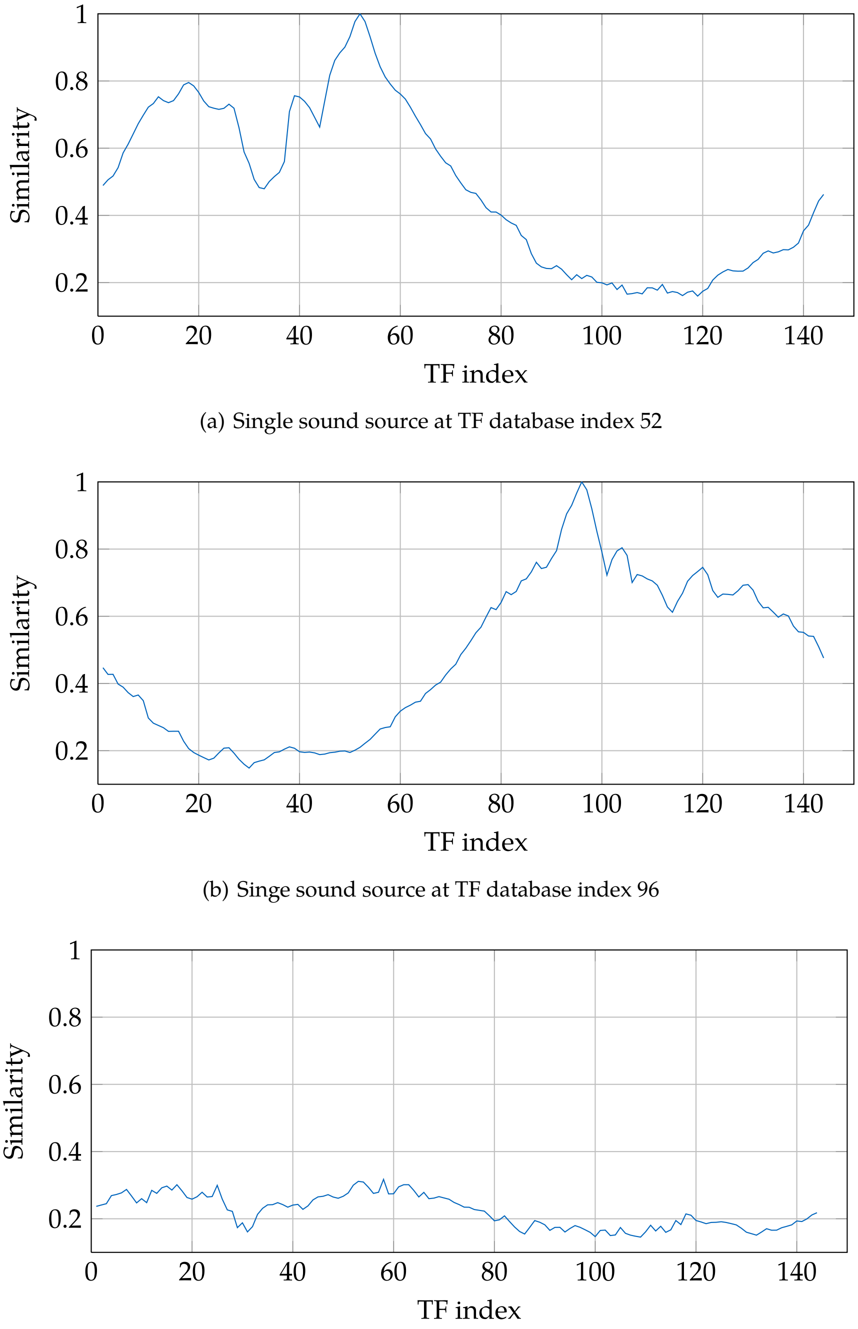
and the localization criterion can be rewritten as

$$\hat{\eta} = \arg\max_{\eta} \tilde{s}_{1,\eta}(t) \star \tilde{s}_{2,\eta}(t) \tag{4.5}$$

A schematic view of the algorithm can be seen in Figure 4.2. It should be noted that the first part of the algorithm involves a brute force search over the whole TF database.

4.1.2 Analysis of some CCL properties

As stated in the algorithm description, CCL works with large block sizes, which cause a considerable algorithmic latency. Depending on the properties of the environment -- mainly reverberation and noise -- the required block size can vary between a few seconds up to even minutes to achieve reliable results. Additionally, the algorithm assumes that the source's relative position is fixed during this time and the possibility of source movement is not considered. CCL is designed to estimate the direction of exactly one sound source. In the presence of noise or another sound source its localization performance drops significantly. Figure 4.3 shows what happens to the CCL similarity measurements in the presence of two active sources. The tested sound scenes have one or two active signals at the database indices 52 and 96. The used database contains TFs of the horizontal plane sampled every 2.5° in azimuth direction. Figures 4.3(a) and 4.3(b) show the resulting similarity values when the sounds are played separately. Peaks at index 52 and 96 indicate that CCL determines the source positions correctly from the separate sound signals. In both plots, the second peaks have a value approximately 0.8 what makes them 20% lower than the dominant peaks. This indicates that the algorithm is able to clearly identify the correct TF filter pair in the case of single sources. When both sound sources are active at the same time, the CCL's similarity values in Figure 4.3(c) do not show a clear winner. The first two peaks have approximately the same value but are much lower than before. The highest peak at index 58 does not correspond to a sound source, while the second highest peak is located correctly at index 52. There is a local maximum at index 92 indicat-



(a) Single sound source at TF database index 52

(b) Singe sound source at TF database index 96

(c) Sound sources at TF database index 52 and index 96 simultaneously active

Figure 4.3: CCL similarity measurements for sound scenes with one or two sources. CCL detects single sound sources (a) and (b) correctly. In the case of two active sources (c) localization does not work.

ing the presence of source two, but it is considerably lower than a number of other maxima which have formed in neighboring parts of the plot. CCL was never designed to find multiple sources in the first place, but it is interesting to see how the presence of a second sound source breaks the ability to find even one source reliably. This is of course a problem for real-world situations where noise and crosstalk from other sources cannot be controlled.

In summary, CCL is a good algorithm for localizing single sources under controlled conditions, but it has several properties and problems that make it unsuitable for the use on a cognitive robot. Since CCL is build on a solid theory, this thesis will develop a new algorithm design based on the same ideas as CCL.

4.2 The challenge of multiple sound sources

The standard CCL algorithm cannot localize multiple simultaneously active sound sources. This limitation is caused by the overlap of the source signals in the binaural observations.

The localization problem can be solved by assuming that the original sound source signals have a sparse representation in some transform domain. This assumption is supported by recent studies of the auditory cortex of animals [42]. The authors present auditive stimuli to animals and measure the neural responses in the auditory cortex. They report that only a small subset (<5%) of all neurons in the auditory cortex is firing at a high rate when audio signals are processed by the brain. Which neurons are participating in a representation depends on the stimulus that is presented.

Sparseness assumptions have successfully been employed for other sound processing techniques like audio compression or blind source separation [102, 94]. In this thesis, the assumption of signal sparseness is the basis for the extension of the cross-convolution approach to multiple sources.

4.2.1 Sparse representations

Any sampled discrete time signal $\boldsymbol{s} \in \mathbb{R}^T$ defined in the time interval $[0, T-1]$ can be decomposed into a weighted sum of F basis vectors $\boldsymbol{\beta_f} \in \mathbb{C}^T$. The sum can be written as

$$\boldsymbol{s} = \sum_{f=0}^{F-1} \boldsymbol{\beta_f} \cdot S_f + \boldsymbol{e} \tag{4.6}$$

where S_f are the weighting coefficients or supports for the basis vectors and $\boldsymbol{e} \in \mathbb{R}^T$ is the residual. The basis vectors form the basis $\boldsymbol{B} = [\boldsymbol{\beta}_0, \dots, \boldsymbol{\beta}_{F-1}] \in \mathbb{C}^{TxF}$ and the supports can be combined to a vector $\boldsymbol{S} = [S_0, \dots, S_{F-1}]^T \in \mathbb{C}^F$. Equation (4.6) can be rewritten to

$$\boldsymbol{s} = \boldsymbol{B} \cdot \boldsymbol{S} + \boldsymbol{e} \tag{4.7}$$

If the basis vectors are chosen appropriately, the energy of the residual $||\boldsymbol{e}||_2^2$ becomes negligible. If $\boldsymbol{s}$ is sparse in respect to the basis the signal is modeled by only a few entries of $\boldsymbol{S}$ and most of the other entries are zero. In reality those entries will probably not be exactly zero but small compared to the few larger entries that contribute most to the energy of the signal.

The vector of supports $\boldsymbol{S}$ can be seen as a representation of the discrete time signal in a different discrete domain. The basis vectors which define a data domain can be defined almost arbitrarily, but are usually chosen so that certain properties of the signal are modeled well.

In statistical signal processing, for example, Principal Component Analysis [93] is used to find a basis that models the variance of the source data well. K-SVD [2] finds a dictionary from sampled data in which the data representation is sparse. Methods like PCA or K-SVD can create representations that model data efficiently, but they require samples from the data to calculate a basis. In sound source localization only sampled data from the observations is available and the underlying sources are unknown. These methods therefore cannot be applied to find representations in which the sources are sparse.

Audio signals can be represented in the frequency domain using the discrete Fourier transform which has exponential functions as basis vectors. Other meaningful transforms in audio processing are the discrete wavelet transform (DWT) and the discrete cosine transform (DCT). Several studies have been performed to identify the basis in which audio is maximally sparse. Rickard and Fallon [99] report that speech is slightly sparser in Fourier domain than in wavelet domain. More importantly, they find that not only the basis, but also the length of the transformation window is important for sparseness. The authors of [96] use a union of modified DCT (MDCT) bases for audio compression and come to the conclusion that a union of MDCT bases needs fewer basis vectors than a single MDCT basis to reach the same SNR.

4.2.2 W-disjoint orthogonality

As shown in Section 4.1.2, CCL has problems localizing two simultaneously active signals. These problems arise from the fact that the information from both sources overlaps in the time domain. Signal sparseness can help to remedy this problem, because overlapping signals in one domain can have completely non-overlapping representations in another domain.

The property of non-overlapping frequency spectra for a set of time signals $\boldsymbol{s}_n, n \in \mathbb{N}^+$ is given by

$$|F^W(\boldsymbol{s}_k)|^T \cdot |F^W(\boldsymbol{s}_l)| = 0 \;\forall k \neq l. \tag{4.8}$$

The operator $F^W(\cdot)$ denotes the windowed discrete Fourier transform. The authors of [49] call this property W-disjoint orthogonality and define it explicitly for windowed Fourier transforms.

Disjointness can also be defined more generally with any other signal transformation $T(\cdot)$. Additionally, in reality pairs of signals will seldom be exactly orthogonal due to noise. A more lenient definition of disjointness would require that for each basis vector at most one signal has a dominant entry. The criterion from Equation (4.8) can be refined to

$$\left|\frac{T(s_k)}{||T(s_k)||_\infty}\right|^T \cdot \left|\frac{T(s_l)}{||T(s_l)||_\infty}\right| < \epsilon \; \forall k \neq l, \tag{4.9}$$

where ϵ is a threshold value. The normalization of the signals accounts for possible level differences between the sound sources. Equation 4.9 measures whether a set of signals has mostly disjoint representations in respect to a basis. It can also be applied to time domain signals when the transformation operator is set to the identity function. Disjointness of a set of signals depends highly on the sparseness of the signal representations and therefore also on the choice of the basis vectors.

4.3 The COMPaSS algorithm

This section presents the details of the COMPaSS (loCalization Of MultiPle Sound Sources) algorithm [30]. The algorithm shares some basic principles with CCL and exploits sparseness and signal disjointness to calculate the positions of multiple sources simultaneously.

4.3.1 Processing of sound frames

Sound localization algorithms usually do not process individual sound samples, but they group a number of consecutive sound samples into sound frames and work on these blocks of data. In practical implementations, the maximal block length has to be finite. Each algorithm has a minimal frame size for which it can still produce accurate localization results. In most cases this minimal frame size is equivalent to the algorithmic latency and should be kept as small as possible.

COMPaSS operates on sound frames and estimates the positions of the active sources for each frame individually. Therefore, it cuts the incoming sound streams into overlapping frames of size N_{fr} with a shift of N_{frs} samples between two consecutive frames.

Usually, sound capturing hardware also operates with sound frames and queues recorded sound samples in local buffers of size N_{hw}. The data does not become available for processing until a buffer is completely filled. In the context of sound frames, the latencies of subsequent processing steps do not add up, but the highest individual latency defines the latency of the whole system. As hardware latency is unavoidable, the localization algorithm can use up to the same frame size as the hardware without adding latency to the system.

In its default setting COMPaSS uses a frame size N_{fr} of 1024 samples and a frame shift N_{frs} of 512.

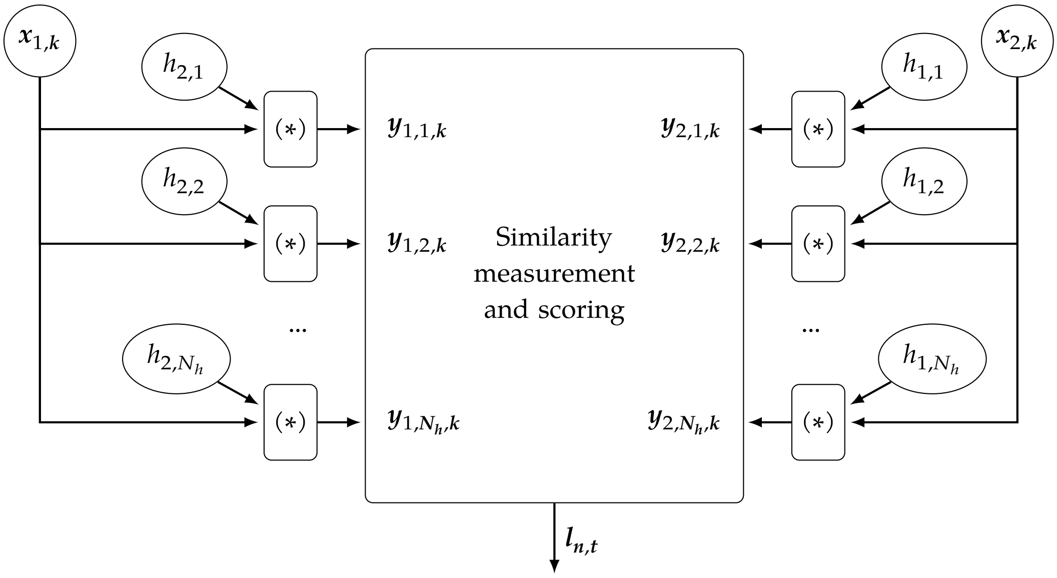


Figure 4.4: A look into the first stage of the algorithm. The input streams x_1 and x_2 are cross-convolved with each of the N_h filter pairs. The new $2N_h$ streams form the inputs for the next stages of the algorithm.

Let the observations for one sound frame be denoted by $\boldsymbol{x}_{j,k} \in \mathbb{R}^{N_{fr}}$. The index $j \in \{1, 2\}$ indicates the left ($j = 1$) and right ($j = 2$) observation and k is the current frame number.

4.3.2 Cross-convolution stage

COMPaSS uses one of the central ideas from the CCL algorithm, namely it tries to identify a number of TFs which correspond to the positions of the sound sources. To this end, the first stage of processing consists of cross-convolving both input streams with each TF filter pair from the database. Figure 4.4 shows a schematic view of the this stage. The input signal from the left microphone is on the far left of the image and its information is passed to the convolution boxes, which are denoted by $(*)$. Second input to each convolution box is a TF from the right microphone. The filters are denoted by $h_{j,\eta}$ where the index $j \in \{1, 2\}$ distinguishes between left ($j = 1$) and right ($j = 2$) microphone responses and the index $\eta \in \{1, \dots, N_h\}$ indicates the index of the filter pair in the database.

The output of the first stage are the $2N_h$ cross-convolved observations $\boldsymbol{y}_{j,\eta,k}$. This initial processing remains unchanged from CCL. The next stages are responsible for identifying the TF filters corresponding to the active sources at each time instant.

4.3.3 Similarity measurement

In its second step COMPaSS has to measure the similarity between the cross-convolved signal pairs $\boldsymbol{y}_{j,\eta,k}$, $j \in \{1, 2\}$. For clarity the subscripts η and k are omitted in this section.

COMPaSS subdivides each signal $\boldsymbol{y}_j$ into N_s smaller overlapping frames. Afterwards, it windows the subframes and transforms them into another signal domain where the source signals are expected to be W-disjoint orthogonal. In this domain the subframes are represented by an N_f-element vector and stored as columns of the matrix $\boldsymbol{Y}_j \in \mathbb{C}^{N_f \times N_s}$. Each entry $\boldsymbol{Y}_j(f, l)$ of the matrix is the support of the f-th frequency bin of the l-th subframe.

Let $\gamma_{j,f}$ denote the f-th row of $\boldsymbol{Y}_j$. The similarity value $\boldsymbol{c}(f)$ is then calculated by

$$\boldsymbol{c}(f) = \left(\frac{|\gamma_{1,f} \cdot \gamma_{2,f}{}^H|}{||\gamma_{1,f}||_2 \cdot ||\gamma_{2,f}||_2} \right)^2 , \tag{4.10}$$

which measures the linear dependence between two corresponding frequency bins over time. All entries of the vector $\boldsymbol{c}$ are in the interval $[0, 1]$, where a higher value indicates higher similarity at the corresponding frequency.

The similarity is clearly defined up to the signal transform operation. This thesis introduces and evaluates two different variants of COMPaSS, one using the Fourier transform and one using the DCT for the similarity measurement, since the different domains might pronounce signal disjointness in a different way.

Magnitude squared coherence

One possible signal transform for the similarity measurement is the Fourier transform. The signals $\boldsymbol{Y}_j$ then correspond to the short-time Fourier transform (STFT) of the cross-convolved signals. The similarity measurement in STFT domain is equivalent to calculating the magnitude squared coherence between the two signals.

The properties of the coherence measurement in regards to cross-convolved inputs can be evaluated in an experiment. The experiment uses a mono speech recording and a TF database with 144 filter pairs. All TFs are on the horizontal plane and the distance between neighboring pairs is $\Delta\varphi = 2.5°$. The TFs are sorted by azimuth starting at $-177.5°$ to $180°$. The TFs at indices 1 and 144 are direct neighbors and are only 2.5° apart.

In the first simulation one single speaker recording is spatialized with the TF filter 54 and, subsequently, the first stages of the algorithm are applied to the generated data. Figure 4.5 shows the coherence values from one frame of the sound stream. The coherence was calculated for all 144 cross-convolved streams at 21 frequency bins. Frequency bins spread equidistantly from 0 to

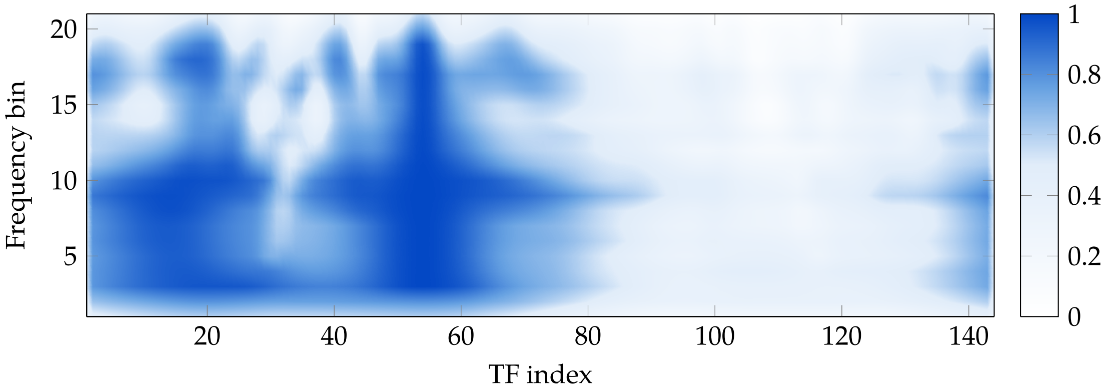
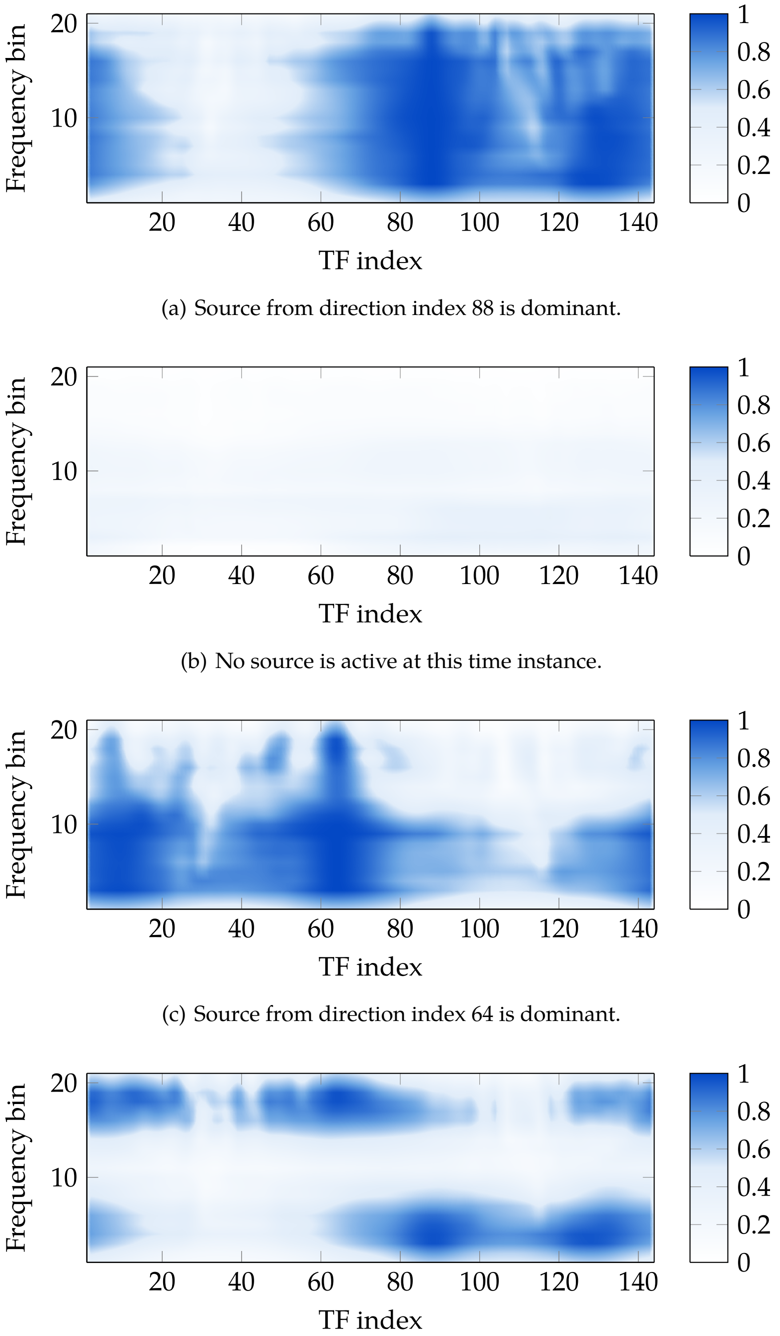


Figure 4.5: Coherence values of one frame when localizing a source at the position corresponding to database index 54. The TF index corresponds to a position on the zero elevation plane and the 21 frequency bin spread equidistantly from 0 to 8 kHz. The area around index 54 has the highest coherence and indicates that the algorithm is able to find the source. The second area of high coherence around 18 is the front-back confusion of the actual source.

8 kHz. The TF filter from index 54 correctly yields the highest coherence with values up to 0.999. It is interesting to note that the coherence in the area around index 54 is generally elevated and decays slowly in azimuth direction. The decay seems to be slowest in the mid frequency range and is much steeper in the lower and higher frequency end. TF filters 54 ± 4 still yield overall coherence values higher than 0.8. This region corresponds to ±10° around the actual source position. Despite only one source being active, the fraction of the area with coherence values lower than 0.2 is small. There is a second area of high coherence around index 18 in the plot. Compared to the first area, the similarity values are lower and decay faster in azimuth direction. The second area is caused by front-back confusions, which manifest themselves as symmetries in the coherence plot. The symmetry axes are always at azimuth ±90°, which corresponds to the TF indices 36 and 108 in the current example.

In summary, the coherence measure does identify the correct TF filter. The neighborhood of the match has comparably high coherence values. This effect could have the negative implication that one of the neighbors might be picked over the correct match. Additionally, this could prove to be a problem when two active sources are close to each other and the two high coherence areas merge. Front-back confusions are a known property of TF-based localization approaches and the scale of the mirror image indicates that the coherence measure could be prone to it.

The next simulation uses two simultaneously active speaker signals, which are spatialized with TFs 64 and 88. The speech signals are not part of a dialog, where most of the time only one speaker is active, but are completely unrelated and both speakers are simultaneously talking most of the time. Figure 4.6 shows the coherence values of the localizer from four handpicked frames. In Figure



(a) Source from direction index 88 is dominant.

(b) No source is active at this time instance.

(c) Source from direction index 64 is dominant.

(d) Both sources have high coherence values in different frequency regions.

Figure 4.6: Coherence values when localizing two active sound sources for different sound frames. The actual sources are at index 88 and 64. In the images either none, one or both sources are active and detected by the coherence measure.

4.6(a) there is one active area in the vicinity of HRTF 88 and the remaining activity in the image can be attributed to the front-back confusion effects. The other source is invisible at this time instance. There is almost no activity in Figure 4.6(b) which means that both sources are silent at the same time. In Figure 4.6(c) there is again only one source visible but this time its the one at index 64. In the plots where only one source is visible the other source is not necessarily completely silent, but the visible source has a much higher energy contribution to the final signal. The fourth image 4.6(d) illustrates a time instance where signal disjointness is pronounced. Both signals are visibly active but their activity is dominant in different frequency regions. The source at index 88 is detected in the lower frequencies up to 4 kHz and the other source is detected in the upper half of the frequency spectrum.

MDCT domain

Audio signals are often also processed in the DCT domain. State-of-the-art audio codecs like MP3 or AAC use a variation of the DCT type IV, the so called MDCT. In contrast to the regular DCT the modified version uses overlapping frames as inputs and applies a windowing function to the input data prior to the transform. MDCT is fully invertible and the original signal can perfectly be reconstructed if the windowing functions are chosen appropriately. The most prominent choices are sine windows or Kaiser-Bessel windows.

The second variant of COMPaSS uses a sine window and the type IV DCT for the similarity measurement. The sparseness of an audio signal in MDCT domain depends highly on the analysis window length. Thus, the audio codec presented in [96] decomposes signals over a union of MDCT bases of different lengths to achieve a higher audio quality at very low bit rates. The reasoning behind this approach is that a signal can contain long stationary components and very short events at the same time. The former can be modeled with longer windows, while the latter are better captured with shorter windows. Audio codecs try to find the sparsest representation with the lowest number of MDCT bases as possible, but this restriction does not apply for sound localization. Therefore, COMPaSS does not have to perform a decomposition of one subframe over all bases simultaneously, but it can calculate an overcomplete signal representation in regards to multiple independent MDCT bases.

Figure 4.7 shows a plot of the DCT similarity measure for the localization of one sound source. The source is at index 64 and its presence is indicated by the region of highest probability. The front-back confusion effects are present as with the coherence measure. The most notable difference to coherence is the much smaller width of the area where a similarity is detected.

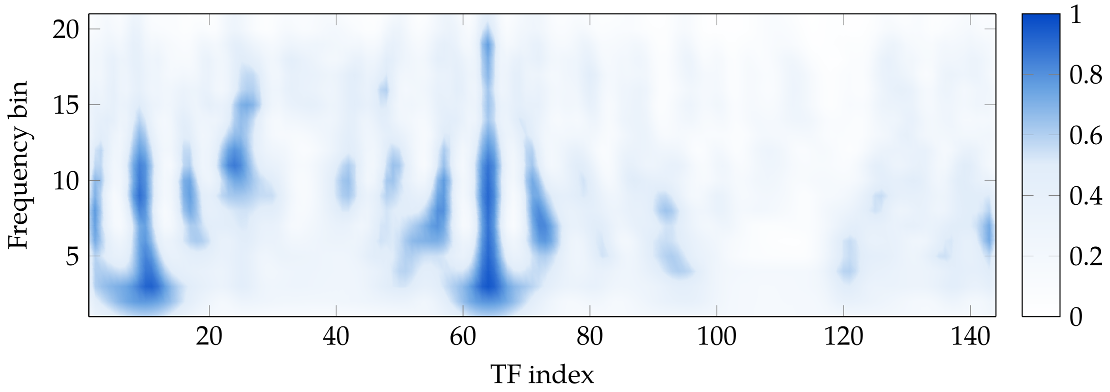


Figure 4.7: The DCT-based similarity measure produces narrower regions of high similarity compared to the spectral coherence. The source is positioned at index 64.

Comparison

COMPaSS has two different similarity measurement methods at its disposal and each has its own set of strengths and weaknesses. The choice between those methods depends highly on the use case.

Magnitude squared coherence produces wide areas of high similarity. This fact will impair the accuracy in the case when two sound sources are close together. The areas of high similarity will overlap, and the algorithm will not be able to extract the correct peaks. However, there are also use cases where wide peaks are desirable, for example, when a TF database is sampled on a coarse spatial grid. In this case the average distance between random source positions and their nearest grid points is longer. When a source lies between grid points, the coherence similarly measure will still assign high scores to the nearest grid points and indicate the presence of an active source. Opposed to this, the DCT similarity measure with its narrower peaks might fail to localize the source correctly in this case.

The strength of the measurement in DCT domain is a better accuracy when densely sampled TF databases are used. The similarity values decline sharply around the correct position. The computational complexity per filter is comparable to the previous method but the overall complexity is higher due to the increased density of the database.

4.3.4 Filter score calculation

COMPaSS stores the similarity values for each η and k of the K last frames as columns of the similarity matrix $\boldsymbol{C}_\eta \in \mathbb{R}^{N_f \times K}$. The entry $\boldsymbol{C}_\eta(f,k)$ is the similarity value achieved by filter pair $h_{j,\eta}$ for the k-th frame at the f-th frequency bin.

COMPaSS has to evaluate these measurements automatically and estimate the filters that were most likely active. In literature, a very similar problem arises also in underdetermined sound source separation, where histogram peak picking has proven to be a good solution [49, 115, 71]. It has also been successfully applied to localization algorithms [24]. Alternatively, clustering algorithms like k-means could be used to find regions with high values in the similarity data. Existing sound source separation or localization algorithms like [102, 6] use clustering to group estimated interaural time and level differences.

Using a winner-takes-it-all approach, COMPaSS obtains the indices of the filters yielding the highest similarity values at one time-frequency point and stores them in the matrix $\boldsymbol{P} \in \mathbb{N}^{N_f \times K}$ with

$$\boldsymbol{P}(f,k) = \arg\max_{\eta} \boldsymbol{C}_{\boldsymbol{\eta}}(f,k). \tag{4.11}$$

Let $\boldsymbol{B}_{\boldsymbol{\eta}} \in \mathbb{N}^{N_f \times K}$ be a binary matrix indicating if filter η has the highest similarity in a specific bin and be obtained by

$$\boldsymbol{B}_{\boldsymbol{\eta}}(f,k) = \begin{cases} 1 & \text{if } \boldsymbol{P}(f,k) = \eta \\ 0 & \text{otherwise} \end{cases}. \tag{4.12}$$

How much a time-frequency bin contributes to the final score of each filter is signal dependent. Chisaki et al. [24] suggest to give more importance to bins with higher signal energy, since a higher SNR can be expected for those. COMPaSS uses a similar weighting of the frequency bins based on signal energy and the achieved similarity values. The filter- and signal-dependent weighting matrices $\boldsymbol{A}_{\boldsymbol{\eta}} \in \mathbb{R}^{N_f \times K}$ are defined as

$$\boldsymbol{A}_{\boldsymbol{\eta}}(f,k) = \left(\frac{|\boldsymbol{X}_1(f,k)| + |\boldsymbol{X}_2(f,k)|}{2} \right)^{\alpha} \cdot \left(\boldsymbol{C}_{\boldsymbol{\eta}}(f,k) \right)^{\beta}, \tag{4.13}$$

where $\boldsymbol{X}_i$ are the observations $x_i(t)$ represented in the same signal domain as the similarity values. The parameters α and β set the influence of the signal energy and the similarity value on the final weight $\boldsymbol{A}_{\boldsymbol{\eta}}$. The choice of $\alpha = 1$ and $\beta = 1$ achieves good results in real-world experiments. If the signal energies of two active sources differ significantly, the influence of the energy term should be lowered by tuning α. Finally, COMPaSS calculates the weighted histogram $\boldsymbol{p} \in \mathbb{R}^{N_h}$, whose entries

$$\boldsymbol{p}(\eta) = tr(\boldsymbol{B}_{\boldsymbol{\eta}} \cdot \boldsymbol{A}_{\boldsymbol{\eta}}{}^{T}) \tag{4.14}$$

are proportional to the probability that a filter pair was active in the observations.

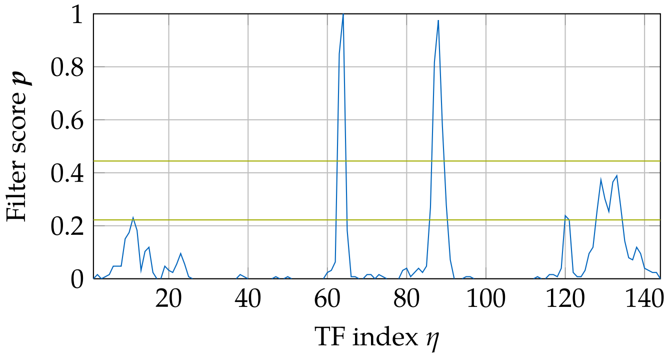


Figure 4.8: Filter scores while localizing a mixture of two signals. The overall scores are low compared to the two pronounced peaks, which correspond to the two sources. The green horizontal lines are thresholds for tracking the source activity and are updated depending on the mean score value of all filters.

4.3.5 Filter identification

At this stage the localization algorithm extracts the most likely positions from the histogram iteratively. The histogram is initialized to $\boldsymbol{p}_1 = \boldsymbol{p}$ and modified in each iteration. The location of the n-th source is extracted with

$$\tilde{\eta}_n = \arg\max_{\eta} \boldsymbol{p}_n(\eta), \tag{4.15}$$

where $\tilde{\eta}_n$ denotes the index of the corresponding transfer function. Let the operator $D(\eta_1, \eta_2)$ calculate the distance between the two source locations corresponding to η_1 and η_2. The histogram is then updated using the following rule

$$\boldsymbol{p}_n(\eta) = \begin{cases} \boldsymbol{p}_{n-1}(\eta) & \text{if } D(\tilde{\eta}_{n-1}, \eta) > \delta_{min} \\ 0 & \text{otherwise} \end{cases}, \tag{4.16}$$

where δ_{min} enforces a minimal distance between two localized sources. By updating the histogram the algorithm ensures that each location is extracted only once and sources with lower signal energy are not obscured.

4.3.6 Signal activity tracking

In any given environment there are numerous potential sound sources, but they do not necessarily have to be active all the time. The filter score histograms allow for very simple source activity tracking, since the peaks of the active sources are very pronounced compared to all other filters.

Therefore, COMPaSS defines two score thresholds in the histogram which are chosen depending on the mean value over all filters

$$\sigma_{on} = \lambda_{on} * \frac{1}{N_h} \sum_{\eta=1}^{N_h} p_t(\eta) \tag{4.17}$$

$$\sigma_{off} = \lambda_{off} * \frac{1}{N_h} \sum_{\eta=1}^{N_h} p_t(\eta) \tag{4.18}$$

where the two scalar values $\lambda_{on} > \lambda_{off}$ are chosen to accommodate for the amount of noise in the histograms. Both values can be seen in Figure 4.8 as the two horizontal lines.

Signal activity tracking is performed by comparing the actual filter score with the thresholds and additionally taking the previous state of a source into account. The binary vector $\boldsymbol{a_k} \in \mathbb{R}^{N_h}$ indicates if a source is active at a given time and is updated using

$$\boldsymbol{a_k}(\eta) = \begin{cases} 0 & \text{if } \boldsymbol{p_t}(\eta) < \sigma_{off} \\ \boldsymbol{a_{k-1}}(\eta) & \text{if } \sigma_{off} \leq \boldsymbol{p_t}(\eta) \leq \sigma_{on} \\ 1 & \boldsymbol{p_t}(\eta) > \sigma_{on} \end{cases} \tag{4.19}$$

where the initial state for all possible source positions is set to $\boldsymbol{a}_0(\eta) = 0$, meaning inactive.

5 Experiments

Experiments are conducted to evaluate COMPaSS's properties and compare its performance to other algorithms. This chapter presents the experimental setup and procedure.

5.1 Considerations

The most realistic experimental setup would implement COMPaSS on a robot and expose it to different sound scenarios while reviewing the algorithm's perception of its environment. The complexity of this experiment is mostly driven by three dominating factors:

Types of sources Sound sources can have completely different characteristics in their temporal behavior or frequency response. These properties are important for the assumption of W-disjoint orthogonality in the COMPaSS algorithm.

Source locations As TFs can have more or less pronounced unique features for different source locations, the performance of the algorithm will partly be dependent on the positions of the sources. Since the algorithm should be able to distinguish even physically close sources, the number of possible locations that have to be tested is high.

Number of simultaneous sources In a multi-source scenario, the active sound sources can be of any type and be located at any position. The number of possible type and location combinations depends highly on the number of simultaneous sources.

The number of permutations resulting from the three mentioned factors is very large. This most realistic evaluation approach is practically not feasible. The necessary simplifications and their implications on the validity of the conclusions in the evaluation are discussed in the following sections.

5.1.1 Extensive simulation and real-world validation

Since a complete real-world evaluation of the algorithm is practically not feasible, the experiments are split into two parts. The first part consists of extensive simulations, which test the properties and

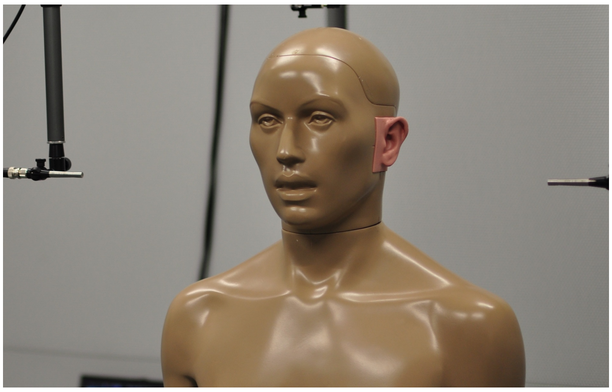


Figure 5.1: The Knowles Electronic Manikin for Acoustic Research (KEMAR) dummy head is an industry standard measurement tool. It has interchangeable ears and microphones in the ear canals.

limitations of COMPaSS. Simulations are cheap in terms of time and effort and allow for complete control over all parameters, e.g. noise, which cannot be controlled in real scenarios. Simulations are helpful especially when configuration parameters of an algorithm have to be determined.

Of course, not all effects from the real world can be simulated with adequate accuracy and the results from simulations do not necessarily represent the real-world case. For example, some sound source localization algorithms perform well in simulations but fail completely in the presence of measurement noise. When it comes to judging the applicability of a sound source localization algorithm in reality, the usefulness of pure simulations is limited. This information gap has to be bridged with additional real-world tests which form the second part of the experiments. Besides verifying the validity of the simulation results, these experiments also measure the unavoidable performance drop between simulation and reality.

5.1.2 Dummy head

In the experiments, a Knowles Electronic Manikin for Acoustic Research (KEMAR) dummy head [21] stands in for an actual robot. The KEMAR was originally developed for measuring and reporting the performance of hearing aids and today the KEMAR is an industry standard tool in acoustic research. Figure 5.1 shows a photograph of the KEMAR dummy head used in the experiments.

The physics of sound wave propagation apply to the KEMAR exactly in the same way as they apply to a robot or any other object. Since the algorithm has no component acting in or reacting to the environment and is purely perceiving the environment, replacing the robot with a dummy head does not simplify the problem that COMPaSS is solving.

5.1.3 Prerecording sound scenes

Repeatability of an experiment is necessary when several algorithms with different configuration parameters have to be tested with the same sound scene. This is not a problem for the first part of the experiments, since a sound scene simulation can be restarted as often as needed.

The second part of the experiments consists of a KEMAR dummy head perceiving its environment with different algorithms. Parameters like noise cannot be controlled in the real world and can change at any time. For the analysis and comparison of different algorithms identical test conditions have to be guaranteed. The easiest method to accomplish this requirement is to prerecord the required sound scenes with the KEMAR microphones and feed the recordings subsequently to the different algorithms. This two step approach is possible, because sound source localization techniques have no active component, which could change the state and subsequently the perception of the environment.

One advantage of this approach is that each sound scene has to be presented to the KEMAR only once instead of multiple times. Additionally, even algorithms which are not capable of online processing can be tested, since the data does not need to be fed to the algorithm in real-time. Prerecording has no disadvantages and does not affect the results of the evaluation in any way.

5.2 The simulations

5.2.1 Requirements for a realistic simulation

The simulations should generate sound signals that resemble the KEMAR's ear response of a specific sound scene as closely as possible. A realistic simulation has to take all imperfections that have an effect in an actual real-world experiment into account:

- Noise can have an effect on every single simulator operation and has to be modeled accordingly.
- Binaural simulations require a TF database for the spatializing of the individual sound sources. In a realistic scenario, the localization algorithm does not have access to exact filter pairs of the mixing process.
- In the presence of multiple sound sources any source can be louder or quieter than the others. The sound pressure level of each individual source has to be adjustable in the simulator.
- The simulation should be able to create dynamic scenarios where the sound sources are able to change their position over time.

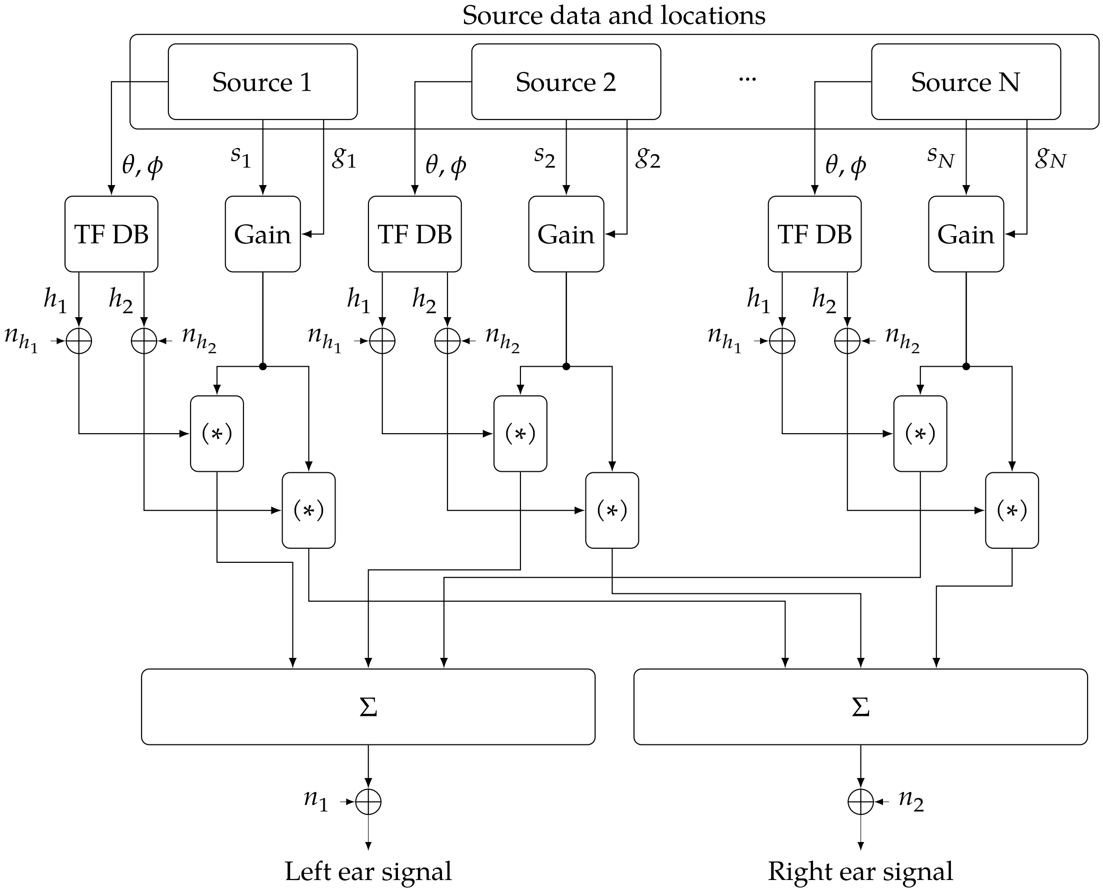


Figure 5.2: Binaural sound scene simulator. The simulator mixes any number of sources into a stereo stream. In the first stage a custom gain is applied to each source and the source signal is subsequently convolved with the possibly noisy TFs, which correspond to the source location. The left and right ear signals of the spatialized sources are summed independently. In the last step noise can be added to the final results.

5.2.2 Binaural sound scene simulator

Figure 5.2 shows the schematic view of the simulator created for this work. The purpose of the simulator is to mix any number of sound sources realistically into a stereo stream. Each source is defined by a number of parameters:

Sound data These are the waveforms of the sound sources that are being mixed. The simulator zero-pads sound signals of shorter duration at the and to obtain input streams of equal length.

Source gain This parameter controls the gain that is applied to each source individually.

Source location The simulator spatializes each source to a certain location in the virtual environment. This can be a fixed position or a trajectory which the virtual source will follow over time.

Using this information the simulator first processes each source individually. In the real world, some sources are louder than others, which is modeled by the gain stages of the simulator. The data stream of each source is passed to the gain control instance, which normalizes the source and then applies the gain. Normalization is necessary to ensure that all sources have the same level before the gain is applied. The gain parameter controls the sound pressure level of each source without introducing spectral changes. The simulator changes the level of the sources only at this early stage explicitly. The interaural level difference does not need to be applied in the simulator as it is implicitly modeled by the TF filter pair of each source location.

For each source the simulator fetches a filter pair from the TF database. Since the database is sampled at discrete locations, the simulator uses the closest available filter pair in terms of Euclidean distance when no pair is available for the exact source position. In a real scenario the TFs for localization are known from some kind of measurement or learning [31]. These measurements will never be perfect as they can be subject to measurement noise and other errors. The simulator accounts for this effect by adding a noise term to the filter impulse responses. This effect can alternatively be modeled by adding noise to the TFs during localization.

The simulator convolves the left and right impulse response with the source signal and yields the left and right signals of the virtually positioned source. For computational performance reasons the convolution is performed in the frequency domain using the overlap-add method. If sources are moving, the filter pair needs to be replaced every few samples. In this case the real-time partitioned convolution algorithm [105] can perform the required low-delay processing.

The result of the convolution operation is one stereo stream per source and the simulator mixes all left and all right channels together yielding one stereo stream of all active sources. In the last step of the simulation noise is added to the stereo stream and finally the left and right microphone signals are obtained. The noise term can be used to model measurement noise or a global noise source which has no associated source position.

The simulator models all imperfections from the real world as closely as possible. There is an additional, maybe not instantly obvious, possibility for adding noise to the final signal. The simulator can model noise sources with directional characteristics by treating them exactly like normal sound sources. Any source signal, e.g. white noise or a computer fan recording, can be placed at any position in the virtual room. The gain parameter of the noise source adjusts the signal-to-noise ratio.

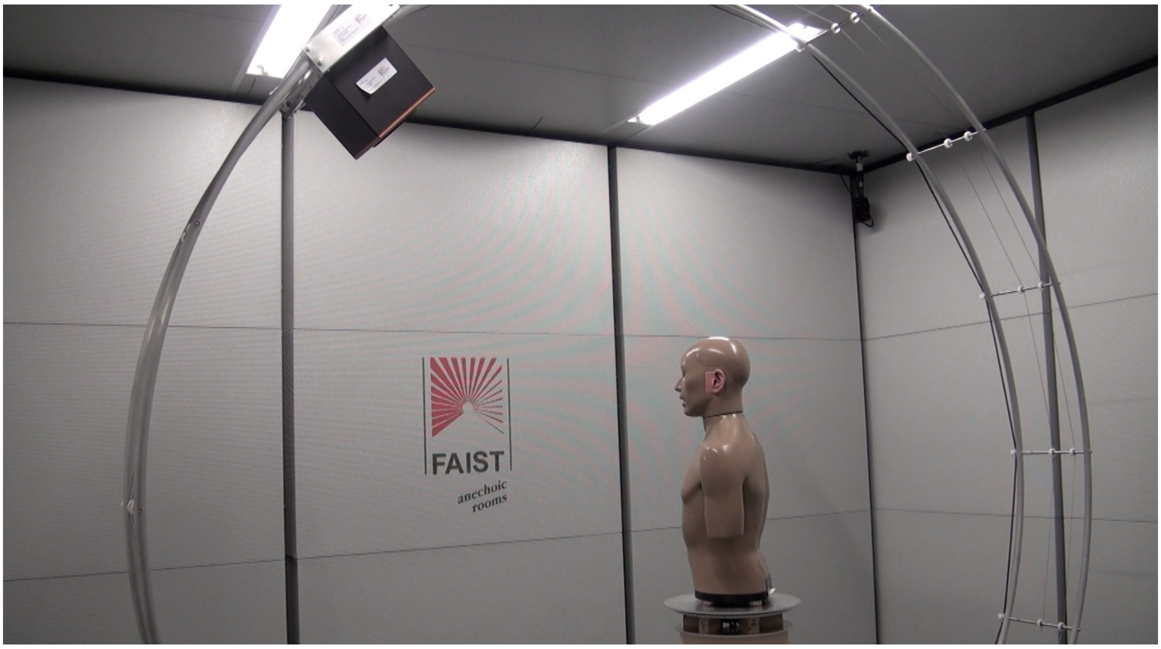


Figure 5.3: TF measurement in an anechoic chamber. The KEMAR is positioned on a turntable and a loudspeaker presenting MLS sequences is moving on the arc. The measurement is controlled by a computer and is fully automated.

5.2.3 TF database

The simulator requires a TF database for the spatialization of the sound sources. Spectral effects change continuously as a function of the source position and therefore a good simulation requires a very densely sampled TF database. The well known MIT or CIPIC databases are sampled at 710 or 1250 grid points respectively which is too coarse for a precise simulation. I measured a dense TF database for the KEMAR dummy head in an anechoic chamber.

The measurement setup can be seen in Figure 5.3. The information about the equipment is listed in Table 5.1. A tracking system tracks all objects using passive markers in the anechoic chamber. The tracking accuracy of under one millimeter allows a very precise positioning and alignment of the objects in the room. A turntable is positioned in the center of the room, its rotational axis pointing exactly upwards. The turntable has a built-in encoder and can be rotated accurately with an error of less than 0.038°. The KEMAR dummy is standing in the center of the turntable, such that the rotational axis cuts the connecting line between the two ear microphones in half. The intersection of those two lines defines the center of the KEMAR's head and is defined to be the origin of the used coordinate system. The arc is a lightweight aluminum structure that can hold a loudspeaker. The center of the arc lies in the origin of the coordinate system and the loudspeaker can be moved

Sound source	KS digital C5 tiny
KEMAR Manikin	GRAS Type 45BA
Microphones	GRAS Type 46AE
Microphone amplifiers	MFA IV81 IEPE
Sound card	RME Multiface II
Lab dimensions	4.7 m x 3.7 m x 2.84 m
Lab noise level	<30 dBA
Lab reverberation time t_{60}	0.08 s
Tracking system	A.R.T. GmbH ARTtrack2
Turntable positioning control	Nanotec PD6-N
Arc positioning control	Nanotec SMCI47-S

Table 5.1: Information about the equipment used in the TF measurement.

on a circle around the origin. The positioning control allows for loudspeaker movements with an error of less than 0.0034°.

Regarding the direction of incidence the native coordinate system of this hardware setup is the spherical coordinate system with vertical poles as defined in Section 2.1. Elevation and azimuth are controlled independently, the former with the loudspeaker position on the arc and the latter with the KEMAR rotation. Due to the physical constraints of the arc the elevation angle cannot get lower than $-45°$, which is the only restriction on possible source directions in respect to the KEMAR.

Special attention is paid to calibrating the home position of the turntable rotation. At azimuth $\varphi = 0°$ the loudspeaker should be in the median plane. As the time delay of arrival for sources in the median plane is equal to zero, the home position is found by examining the cross-correlation between the ear signals at time lag zero.

The calibrated setup measures TF filters with ML sequences using the same technique as the MIT database [34]. The resulting database consists of 7920 filters measured at a dense grid with a grid point resolution of 2.5° in elevation and azimuth direction.

5.2.4 The test signals

Due to the W-disjoint orthogonality assumption, COMPaSS will perform better or worse with different signal configurations. The simulation uses the following types of signals:

Speech These signals consist of sentences uttered by five different speakers (4 male, 1 female). They have the typical activity distribution of speech signals with parts of high energy and almost silent speech pauses.

Music Three music tracks with different signal characteristics. The first one is a classical piece with strings and brass. It has a large dynamic range with alternating louder and quieter parts. The second track is an excerpt from a movie score and has the typical characteristics of a loudness enhanced track, where the dynamic range is reduced in order to make softer parts sound louder. As a result this track has almost no parts with low signal energy. The third track contains male vocals accompanied by a guitar. The first half of the track is dominated by the voice, thus resembling mostly a speech signal, while second part is dominated by the guitar.

Artificial noise Artificial tracks are useful for testing algorithms under extreme conditions. The broadband and narrowband noise signals can be used to target specific frequency bands and analyze how well the algorithms can localize in these regions.

5.2.5 Simulated scenarios

Two different kinds of scenarios are simulated for later evaluation:

Random scenarios The goal of this simulation is to predict how well a localization algorithm will perform at the task it was designed for. It measures the algorithm performance for a large number of randomly generated sound scenes. Since the localization performance depends partly on the analyzed scenario, averaging over many random trials gives a good estimate of the overall results.

Selected scenarios Random testing measures how good an algorithm works overall, but it gives no insight into the strengths and weaknesses of the underlying technique. The limitations of each approach can become visible by triggering algorithm failure and to this end the algorithms are tested with manually selected scenarios. Each scenario has an extreme source configuration which is likely to be problematic for localization. Examples for such problematic configurations are very small source distances or a very high noise level

5.3 Real-world recordings

Recordings created in a normal reverberant room are the second part of the experiments. In this part sounds are presented through loudspeakers to the KEMAR in a real-world environment. Between the recording of different sound scenes the loudspeakers have to be moved manually and the new positions have to be measured precisely. This procedure is cumbersome and requires multiple minutes of work to record a few seconds of audio. For practical purposes the real-world experiments are split in a manual part and an automated part. Both parts use a slightly different hardware setup and will be discussed in the following sections.

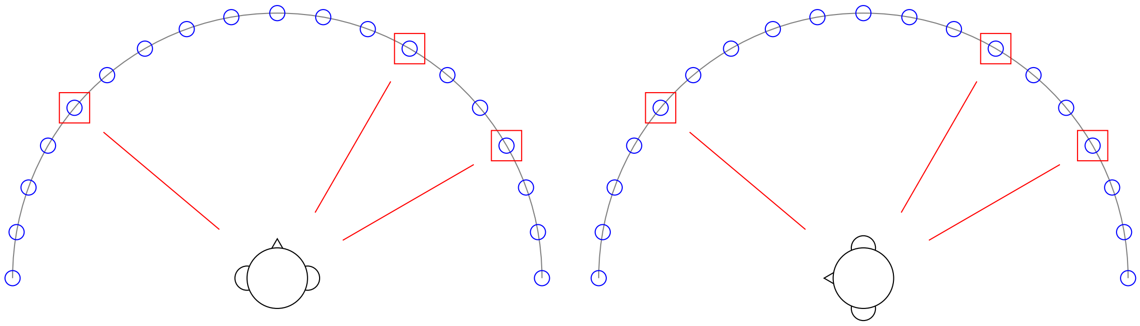


Figure 5.4: Top view of the manual experiment setup. The TFs are measured at 3 elevations and 19 azimuths on a circle with radius 1.3 m. Recorded scenarios consist of up to three active sources at different positions and all recordings are created with two KEMAR orientations.

5.3.1 The experimental environment

All real-world recordings are performed in a office room with dimensions 5.10×3.49×3.09 m (L×W×H). The reverberation time RT60 of the room is 0.64 s and was measured with the switch off method at different positions in the room. Due to a high noise floor, the RT60 measurement could not be obtained directly, but was calculated from T20 and T30 measurements according to ISO 3382-2:2008 [48].

5.3.2 Manual experiments

Real world sound scenes are set up and recorded in manual experiments. They closely represent the localization problem that has to be solved by a cognitive robot. The results obtained by the analysis of these recordings are a very good prognosis for the algorithm performance in real-world scenarios.

Special hardware, namely the arc, for automatically moving a loudspeaker is available in the anechoic chamber. A comparable construction is not available for office environment and the loudspeakers have to be positioned manually. One interesting evaluation result will be the direct comparison between a recorded sound scene and its simulation. To be able to use the manual recordings as reference data, all loudspeakers and the KEMAR have to be precisely positioned in the room.

The setup of the manual experiments can be seen in Figure 5.4. All positions are in the far field at a distance of 1.3 m to the KEMAR. The spatial grid resolution is 10° in elevation and azimuth direction, which corresponds to a maximum grid point distance of approximately 22 cm. The loudspeakers are cube-shaped with an edge length of 20 cm. The grid point distance is slightly above the minimal distance that can be achieved with the hardware.

Due to spatial constraints in the office, it is not possible to place the loudspeakers in a full circle

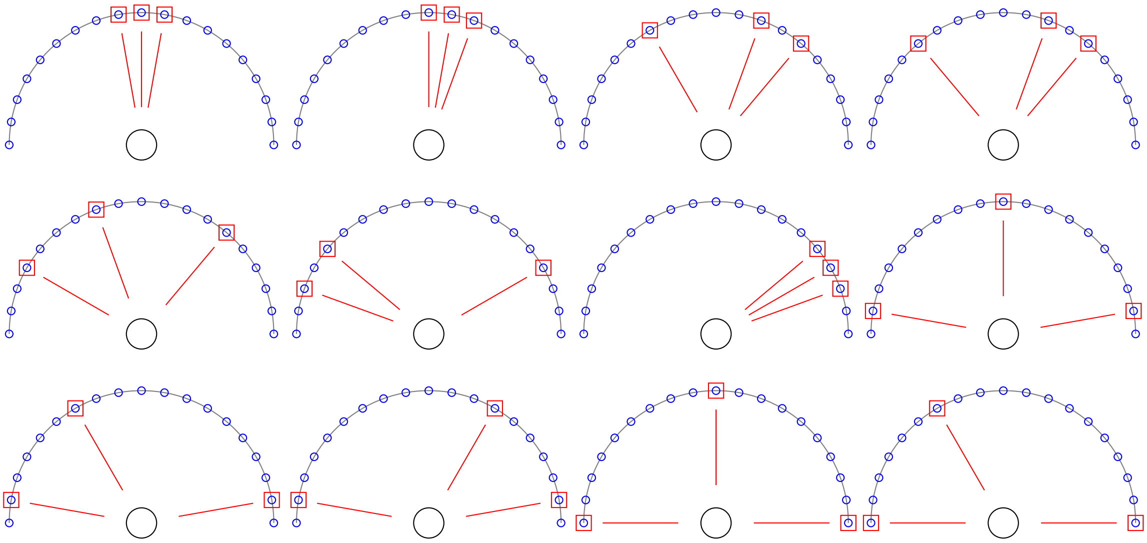


Figure 5.5: The loudspeaker positions for all sound scenes. Each scene was recorded with both KEMAR orientations.

around the KEMAR, so the possible source locations are restricted to a semicircle. To cover all sound scenes of interest, two different KEMAR orientations in the room are used. On the left side of Figure 5.4 the KEMAR is facing the semicircle and all possible source positions are in the front hemisphere. Scenarios with sources lying in opposite hemispheres can be recorded with this orientation. In the second setup the KEMAR is rotated 90° counter-clockwise such that its right ear is facing the semicircle. All possible source positions are now in the right hemisphere. Most notably, scenarios with source positions that are each others front-back confusion can be studied with this setup. For both KEMAR orientations the TF database is measured at 3 elevations and 19 azimuths. The elevation planes are at $-10°$, $0°$ and $10°$.

Up to three sound sources are presented to the KEMAR simultaneously. The number of different loudspeaker positions had to be limited since the exact positioning of all loudspeakers in respect to the KEMAR is a complicated and time consuming procedure. The loudspeaker configurations can be seen in Figure 5.5. The combination of different speaker configurations and both KEMAR orientations yields 24 different physical setups and with each setup multiple different sound scenes are recorded.

A total of 858 sound scenes with a combined length of approximately 1.3 hours were recorded in the manual experiments. Each recording has a length of 5.5 s and was checked by a human listener for unwanted additional sources from neighboring rooms. Affected recordings were repeated until they fulfilled the required quality.

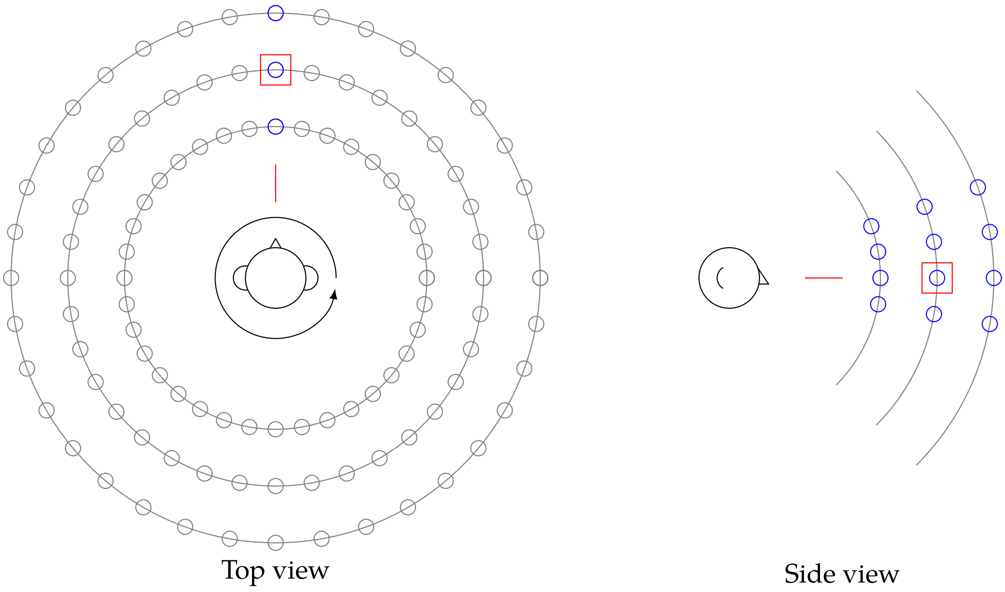


Figure 5.6: The automated experiment setup uses loudspeaker positions (blue circles) at different elevations and distances. Source position changes in azimuth direction are simulated by rotating the KEMAR around its vertical axis and virtual source positions (gray circles) are yielded.

5.3.3 Automated experiments

The limited number of sound scenes from the manual experiments is insufficient for an in-depth analysis of COMPaSS's properties. The required amount of data is so high that it has to be created in automated experiments. To eliminate the need for a frequent repositioning of multiple loudspeakers, the automated experiments are a combination of real measurements and artificial mixing.

In contrast to the previous experiments, where a complete sound scene with multiple sources was recorded at once, the hybrid approach taken here records only one source at a time. The observations of multiple sources from different directions are later mixed to create any sound scene. Observation mixing has the advantage that each sound source has to be presented only once from each position. This approach reduces the number of required loudspeakers for the experiment to one. The complexity to carry out the automated experiments lies between simulations and manual experiments.

Recording

To speed up the recording process, the loudspeaker is positioned at a fixed azimuth angle in respect to the KEMAR and only its elevation and range in respect to the KEMAR are changed manually throughout the experiment. The different combinations of elevation ($-10°, 0°, 10°, 20°$) and distance

(0.7 m, 1.0 m, 1.3 m) can be seen on the right side of Figure 5.6. Similarly to the previous recordings in the anechoic chamber, the azimuth angle of incidence can be controlled automatically with the KEMAR rotation around its vertical axis using a turntable. The loudspeaker has to be set up only once at the beginning of a recording session. The turntable is rotated in 2.5° steps and the observations of all sound signals are automatically recorded. The resulting virtual sound source positions are depicted on the left of Figure 5.6.

Using this setup and procedure 18144 individual recordings with a total length of approximately 27.72 hours were created. The automated experiments use the same test signals as the simulations. All signals are normalized to the same gain prior to their presentation. As in the manual experiments each recording was checked by a human listener for identical recording conditions and repeated if necessary. The automatic procedure allowed for a finer azimuthal grid than in the manual experiments. The angular resolution of 2.5° results in a distance of 5.67 cm between neighboring grid points on the horizontal plane at a recording distance of 1.3 m. This is far below the speaker side length of 20 cm.

Observation mixing

Sound scenes are created by mixing individually recorded observations. Compared to a full sound scene simulation, observation mixing is much simpler, as sound sources do not have to be spatialized. Instead, binaural source observations can be loaded and combined to retrieve a sound scene. Most notably, observation mixing does not require a TF database.

The binaural sound scene simulator normalizes the source signals and applies the gain right before the spatialization. For the automated experiments, the gain step has to be postponed and applied during the final mixing. Since source signals are normalized prior to the automated recording, the observations can be directly multiplied with the gain factor.

A sound scene is the superposition of multiple sources and is calculated by adding observations in the time domain. For the regarded signals and source positions, this mixing approach allows for the same degree of freedom as the simulations in terms of the number of possible sound scenes.

Validity of the taken approach

The automated experiments introduce two modifications to the manual recording process to create a feasible automated test data acquisition approach. The question aries how well sound scenes from observation mixing represent real-world scenarios.

The first modification is the instantaneous mixing of recorded source observations. This approach assumes that simultaneously active sources do not interfere with each other acoustically and that the sound field at an ear microphone is the superposition of multiple individual sound fields. The same assumption is made for the evaluation of sound signal separation algorithms [5, 33], where individual source observations are used as the ground truth for separation results. The instantaneous mixing step can therefore be expected to have little or no impact on evaluation results.

The second modification simulates different source positions by KEMAR rotation, similar to the TF database measurement in the anechoic chamber. While this method is unproblematic in anechoic environments, it can cause inaccuracies in the presence of reverberation. Sound reflections in a room also introduce spectral changes to a perceived sound signal. These changes depend on the positions of both source and observer. In regard to reverberant environments, source movement and KEMAR rotation are not equivalent as the observations are subject to different room related effects. The impact of the modification is hard to predict, but the localization problem should not be severely affected as observations are still subject to realistic reverberation.

Observation mixing introduces unapparent inaccuracies with the way noise is handled. Measurement noise and interfering noise from recording equipment are present in each recording. Normally, the observation of a louder source has a better SNR, as the source's signal level is higher. Observation mixing cannot change the source gain at playback. Instead, it applies the gain to the source recording. At this point noise is already present in the recordings and is unavoidably also amplified or attenuated. The SNR of each recording stays exactly the same. Additionally, the instantaneous mixing step amplifies interfering noise sources. When a sound scene is recorded manually, each interfering sound source is recorded once. Opposed to this, the noise sources are present in each single observation that is mixed into a virtual sound scene. This mixing amplifies the signals of the noise sources by a factor proportional to the number of active sources in the sound scene.

In summary, the simplifications of the experimental process change several properties of the generated sound scenes. The virtual room has a higher noise level and its reverberation is different. Neither of the two modifications makes the localization problem easier to solve. Evaluation results obtained with observation mixing should closely resemble real-world tests. This theoretical analysis will also be verified with measurements in the evaluation.

6 Evaluation

This chapter analyzes the different properties of COMPaSS and compares the algorithm's general performance to other state-of-the-art techniques.

6.1 Quality metric

Many of COMPaSS's properties are given by the algorithm design and can be inferred with a theoretical analysis. However, some properties like COMPaSS's reverberation tolerance or noise tolerance cannot be predicted and have to be measured, as they partially depend on the algorithm's input data. An experimental evaluation of these properties requires a measure for the performance of the localization algorithm.

6.1.1 Localization accuracy

Localization accuracy describes how well an algorithm determines the positions of the sound sources that are active in a presented sound scene. There is a number of possibilities for defining the localization accuracy and in this evaluation it is measured by three individual numbers:

Exact accuracy This number is the percentage of correctly localized positions. A result is deemed correct if the closest spatial grid point to the actual source position is chosen by the algorithm. As TF-based algorithms select the best matching filter pairs from the database, their results will lie on the same spatial sampling grid as the TF database. Therefore, the exact accuracy has an implicit tolerance of half the grid point distance. When comparing TF-based algorithms with algorithms whose localization results are continuous, this implicit tolerance must be taken into account.

Tolerance region Some applications will not require exact accuracy and allow a certain deviation of the localization results. In those cases the localization result will suffice if it lies in a tolerance region around the correct source position. The tolerance region measure indicates the fraction of localization results within the allowed deviation over the number of total localizations. On a spatial sampling grid this means that not only the closest sampling point is deemed correct, but also a number of nearest neighbors of the correct point. Again, an additional implicit tolerance of half a grid point distance will have to be added to calculate the equivalent tolerance for continuous localization results. In this evaluation the tolerance region includes the neighboring grid points of the correct source location. Thus, a localization result may deviate one and a half times the grid point distance from an actual position.

Mean angular error The first two performance values indicate the percentage of correct localizations, but do not give a qualitative measure for the magnitude of the errors made by the algorithm. The third value is therefore the mean angular error (MAE). It can be seen as an uncertainty measure for the localization results of an algorithm.

6.1.2 Normalization due to sparseness

The sources that are present in a sound scene are not necessarily observable at all times. The mere presence of a sound source cannot be detected by a localization algorithm if the source is not emitting sound. For calculating of the localization accuracy it is important to distinguish between the existence of a sound source and its observability.

By its design COMPaSS processes sound streams and delivers localization results per sound frame. In the following evaluations the sources of a sound scene are permanently active. Nevertheless, due to the sparseness of a sound signal, even an active source may not be observable in some sound frames. To account for this circumstance, the localization accuracy is calculated by

$$\text{accuracy} = \frac{\text{number of results in tolerance region}}{\text{number of localizable frames}},$$

which can be seen as a normalization of the localization accuracy. The reasoning behind this approach is that the number of correctly localizable frames is limited by the number of frames in which the source is observable.

This evaluation treats a source as observable if its signal power level is higher than the noise floor. A sound signal is amplified or attenuated by reflections in the environment and these level changes can affect the observability of a source. In consequence, the observability classification should be performed with the signals that arrive at the microphones, rather than with the emitted source signal. For real-world recordings the classification can use the individual recordings of a source from its respective direction. For simulations it can use the spatialized signals of the individual sources.

6.2 Evaluation procedure

The evaluation of COMPaSS's individual properties and its overall performance follow an identical procedure:

- Identifying the sound scene parameters that influence one algorithm property
- Creating multiple sound scenes
- Calculating and averaging of the localization accuracies
- Variating the identified parameters

Each of COMPaSS's properties is evaluated individually to isolate its effect on the localization accuracy. The evaluation starts with the identification of the sound scene parameters that have an influence on the tested property. These parameters are set to expedient values in the subsequent sound scene simulation or recording. All other parameters are set to default values or random values in order to create a statistically significant number of representative sound scenes. The evaluation calculates the average localization accuracy over all frames of all sound scenes while accounting for the observability of the sources. The sound scene creation and accuracy measurement steps can be repeated for different values of the identified sound scene parameters.

6.3 Analysis of COMPaSS's properties

This section analyzes the properties of COMPaSS in detail. The evaluation requires a huge number of different sound scenes and is therefore performed mainly with simulated sound signals. Whenever possible identical sound scenes are also created from real-world recordings with observation mixing. Using real-world data is not possible when source positions outside the recorded elevation range of $[-10°, \dots, 20°]$ are required. The results from observation mixing serve as a prediction for the loss of accuracy that the localization algorithm will suffer under real conditions.

As COMPaSS can work in either Fourier or DCT domain, all evaluations are performed with both variants of the algorithm. A comparison will determine the strengths and weaknesses of each variant.

For all tests the TF databases are sampled at a resolution of 5° in azimuth direction. All signals are down-sampled to 16 kHz and a frame size of 1024 samples (64 ms) is chosen. The length of the filter impulse responses for the simulations is 128 samples which is long enough for the impulse response to attenuate. The impulse responses created from real-world data are substantially longer, but are truncated to 256 samples. This length is a compromise between keeping the most important features and achieving a higher degree of independence from room related effects.

Sound scene	Coherence			DCT		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
simulation						
speech	1.00	1.00	0.00	1.00	1.00	0.00
music	1.00	1.00	0.00	1.00	1.00	0.00
noise	1.00	1.00	0.00	1.00	1.00	0.00
observation mixing						
speech	0.97	1.00	0.92	0.97	1.00	0.70
music	0.92	1.00	1.12	0.97	1.00	0.62
noise	0.94	1.00	0.61	0.97	1.00	0.32

Table 6.1: Localization accuracies for different types of sources averaged over 25 randomly chosen positions. At each position three different speech and music signals and one noise signal are tested.

6.3.1 Different types of single sound sources

This test analyzes how well COMPaSS localizes single sound sources of different type, namely speech, music, and noise. The results are particularly interesting as each signal type has different sparseness and observability properties. Test sound scenes are created with the following parameters:

Sound sources In each sound scene one source is active. The tests use three different speaker signals, three music signals, and one broadband noise signal.

Positions 25 source positions are randomly chosen.

Table 6.1 shows the resulting localization accuracies. In the simulations both COMPaSS variants can localize all three sound source types perfectly. As expected, the accuracy deteriorates when real-world recordings are used. The exact accuracy of the DCT variant drops only slightly by 3%.

6.3.2 Sound source pairs of the same type

This test analyzes how COMPaSS's localization accuracy changes when a second source of the same type as the first one is added to the sound scene:

Sound sources In each sound scene two sources are active. For speech and music, three different combinations of signal pairs are tested. The broadband noise signal is paired with a second broadband noise signal.

Positions 25 source positions pairs for two simultaneous sources are randomly chosen.

Sound scene	Coherence			DCT		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
simulation						
speech-speech	0.93	0.93	6.50	0.93	0.93	6.40
music-music	0.64	0.69	27.68	0.74	0.77	21.53
noise-noise	0.56	0.58	35.50	0.54	0.54	40.17
observation mixing						
speech-speech	0.66	0.80	19.53	0.72	0.78	18.42
music-music	0.41	0.61	32.49	0.53	0.64	28.23
noise-noise	0.46	0.53	46.28	0.53	0.58	41.11

Table 6.2: Localization accuracies for signal pairs of the same type averaged over 25 randomly chosen position pairs. At each position three different speech and music pairs and one noise signal pair are tested.

Table 6.2 shows the resulting localization accuracies. COMPaSS achieves excellent results localizing combinations of two speech signals. The exact accuracy for the detection of the music signals drops to approximately 70%, which can be attributed to a higher overlap of the signals in time-frequency domain due to their lower sparseness. Since the noise signals are not sparse at all, COMPaSS is able to pickup only one of the two sources correctly most of the time. Therefore, the localization accuracy approaches 50% for this type of source pair.

6.3.3 Sound source pairs of different type

The previous test has shown how the pairing of sound sources with similar signal characteristics influences the localization accuracy. This test uses pairs of one speech signal and one music signal to analyze how source pairs of different type are handled by the algorithm. Two different outcomes are likely for this test. On the one hand, the speech-music localization accuracy could be between the previous two accuracies as the overall sparseness of the sound signals also lies between the previous two. On the other hand, the music source could completely obscure the speech source, as its signal is more aggressive in terms of covering the time-frequency spectrum. Subsequently, the localization accuracy could be even lower as in the music-music case. The test is conducted with the following parameters:

Sound sources In each sound scene two sources are active, one speech signal and one music signal. At each position three different signal pairs are tested.

Positions 25 source position pairs for two simultaneous sources are randomly chosen.

Table 6.3 shows the resulting localization accuracies. The accuracy of the speech-music sound scenes lies between the speech-speech and music-music accuracies from the previous test. This

Sound scene	Coherence			DCT		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
simulation						
speech-music	0.85	0.86	15.65	0.82	0.82	18.84
observation mixing						
speech-music	0.60	0.72	24.87	0.66	0.73	23.36

Table 6.3: Localization accuracies for signal pairs of one speech and one music signal averaged over 25 randomly chosen position pairs. At each position three different pairs of speech and music signals are tested.

Sound scene	Coherence			DCT		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
simulation						
speech-speech-speech	0.85	0.87	12.33	0.91	0.91	8.24
speech-speech-music	0.68	0.70	29.25	0.69	0.70	31.84
observation mixing						
speech-speech-speech	0.50	0.65	29.62	0.57	0.65	29.37
speech-speech-music	0.43	0.58	34.02	0.50	0.59	34.16

Table 6.4: Localization accuracies for signal triples averaged over 25 randomly chosen position triples. At each position three different source signal combinations are tested.

observation supports the assumption that the localization accuracy scales with the average sparseness of the sound sources in a sound scene.

6.3.4 Three sound sources

This test analyzes how COMPaSS's localization accuracy changes when a third source is added to the sound scene:

Sound sources In each sound scene three sources are active. There are two types of source triples, one consisting of three speech signals and the second consisting of two speech and one music signal. For each type three different triples are tested at each position.

Positions 25 source positions triples for three simultaneous sources are randomly chosen.

Table 6.4 shows the resulting localization accuracies. In comparison to the two-source tests, the accuracies are lower. This can be explained by the fact that signal disjointness becomes more improbable as sources are added. The lower accuracy for mixtures containing music can be attributed to the same effect. However, the results for three speech sources are consistent with the previous tests, as the accuracy scales almost linearly with the number of sources. Interestingly, in the sim-

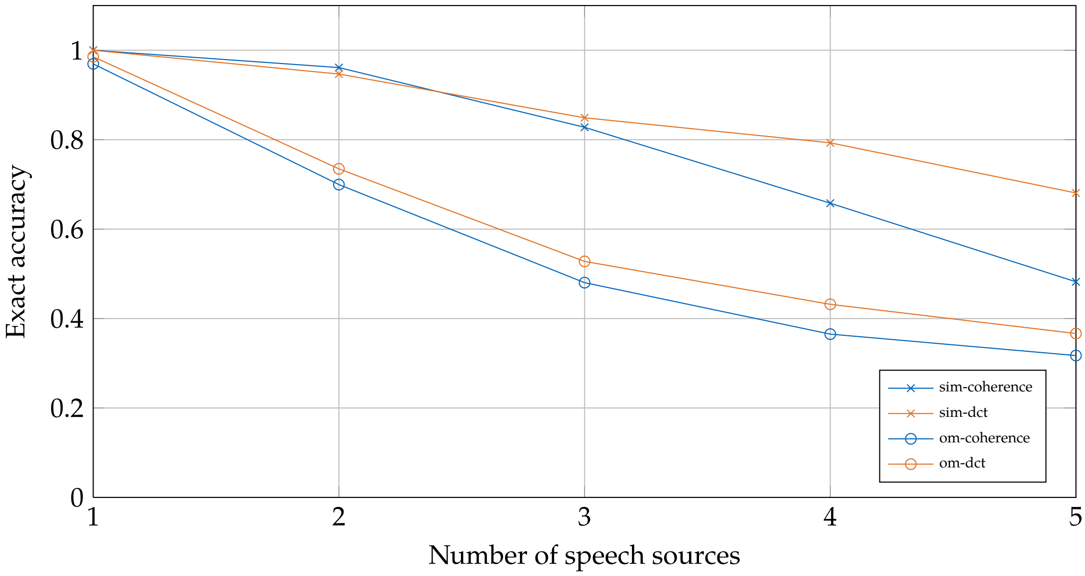


Figure 6.1: The exact localization accuracy plotted over the number of active speech sources. COMPaSS's accuracy decreases when sources are added, because the assumed W-disjointness of the source spectra decreases.

ulation the mixture of three speech sources shows a better result than the speech-music mixture from the previous test. However, with real-world recordings this effect cannot be reproduced.

6.3.5 Number of simultaneous speech sources

This test investigates how the accuracy scales with the number of active speech sources:

Sound sources The number of sound sources varies between one and five. All source signals are speech recordings.

Positions 25 source configurations for one to five sound sources are randomly chosen.

The results are plotted in Figure 6.1. COMPaSS's accuracy is almost perfect in the one-source case and drops with increasing number of sources. With each additional source the spectrum of the observations gets more populated and the probability of W-disjointness and therefore also the localization accuracy decreases. Interestingly, accuracy decreases less than linearly when more than three sources are present in a sound scene. In the case of five present sources, COMPaSS's accuracy is still better than 35% in the observation mixing case. This is a strong indicator that COMPaSS's ability to localize sound sources does not completely break down with an increasing number of sources and that COMPaSS is at least able to localize a subset of all active sources. It is to be noted

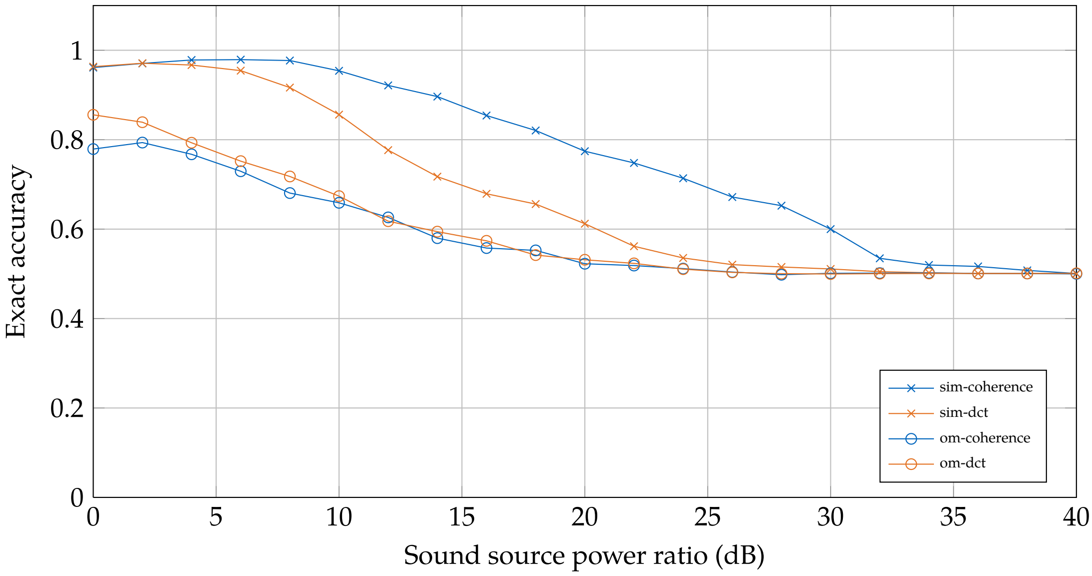


Figure 6.2: The exact localization accuracy plotted over the sound source power ratio. At a low ratio COMPaSS detects both sources with a good accuracy, but at higher ratios only the louder source is visible to the algorithm and the localization accuracy approaches 50%.

that the localization problem in this test is particularly hard to solve, as each sound scene consists of multiple equally dominant source signals.

6.3.6 Sound source power ratio

In real environments usually some sound sources are louder than others. This test evaluates the influence of the signal power ratio of two sound sources on the localization accuracy:

Sound sources The same two speech sources are present in every sound scene.

Positions Eight position pairs are randomly chosen for the two sources.

Others The signal power of the first source is fixed in all trials. The power ratio of both sources is varied between 0 and 30 dB.

A plot of the localization accuracies is shown in Figure 6.2. At a low ratio COMPaSS detects both sources with a good accuracy, but at higher ratios only the louder source is visible to the algorithm as it dominates the spectrum of the observations. At a ratio of approximately 10 dB the localization accuracy starts deteriorating in the simulations. In the case of observation mixing, the deterioration starts almost instantly and at 10 dB the exact accuracy is approximately 0.75. A likely interpretation

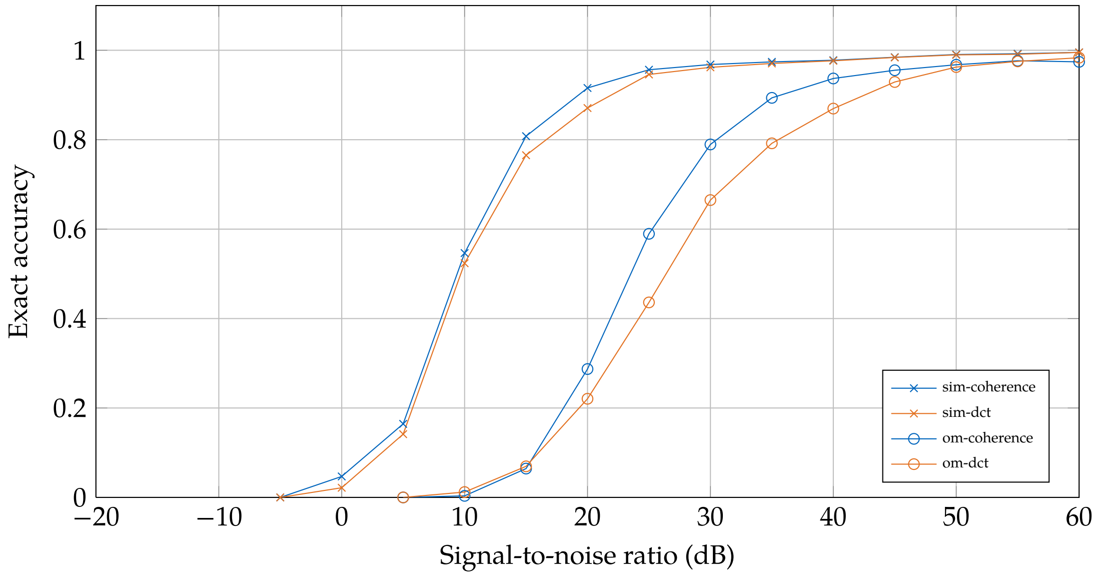


Figure 6.3: Localization accuracy over the signal-to-noise ratio of an interfering white noise signal. The noise is added to the observations after mixing.

for this number is that one source is correctly localized 50% of the time when the other source has a ten times larger signal power.

6.3.7 Signal to noise ratio

Under realistic conditions the observed sound signals will be subject to noise. This test evaluates the noise tolerance of COMPaSS:

Sound sources Every sound scene has one active sound source. At each position three speech and three music signals are tested.

Positions Eight positions are randomly chosen.

Others A white noise or a low-frequency noise is added to the observations after the mixing. The signal-to-noise ratio (SNR) is varied between -20 dB and 60 dB.

The noise signals are purely additive and not processed by a direction dependent filtering. Thus, COMPaSS cannot determine the location of the noise source in the room. In practice, such distortions are caused, for example, by measurement noise of the recording equipment. Figure 6.3 shows the localization accuracy over the SNR for the white noise signal. In the simulations COMPaSS achieves good localization results with an SNR of 15 dB and higher. Observation mixing shows similar results at 30 dB.

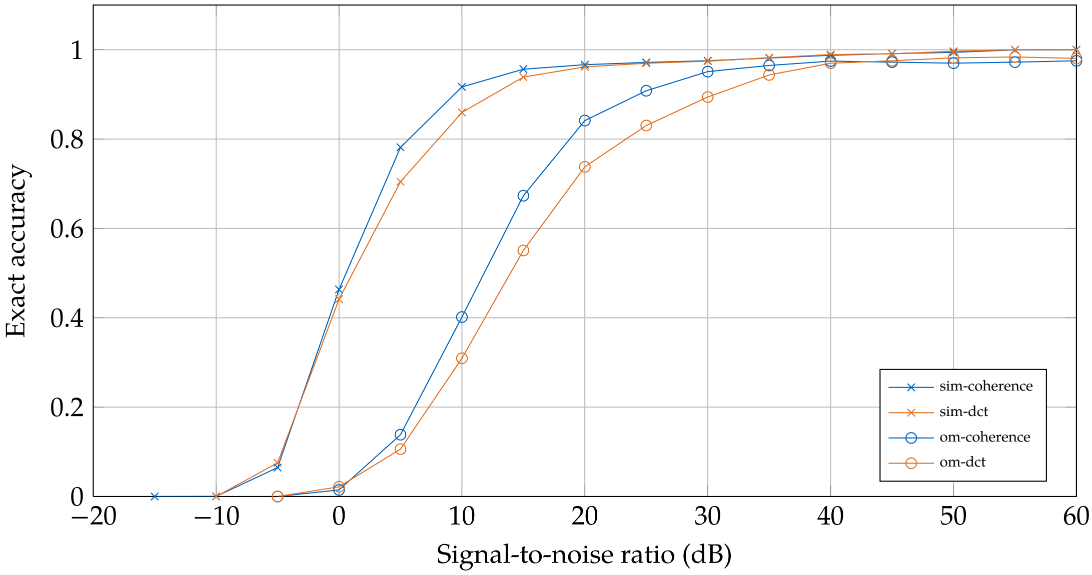


Figure 6.4: Localization accuracy over the signal-to-noise ratio of an interfering low-frequency noise signal. The noise is added to the observations after mixing.

The low-frequency noise was generated by applying a lowpass filter with a cutoff-frequency of 2000 Hz to a white noise signal. Time delay of arrival based localization algorithms are specifically susceptible to low-frequency noise as they cannot use higher frequencies due to aliasing problems. This test investigates how COMPaSS is affected when only the low-frequency region is noisy. The resulting accuracy is shown in Figure 6.4. Compared to the white noise signal, the localization accuracy is approximately 10 dB better, which indicates that COMPaSS is implicitly using the whole spectrum for localization.

6.3.8 Noise source in the virtual room

In the previous section a noise signal was added to the final sound scenes. This section evaluates the influence of a noise source in the virtual room:

Sound sources Every sound scene has one active sound source. At each position three speech and three music signals are tested.

Positions Eight positions are randomly chosen for the sound source.

Others A noise source is present at a fixed position in the room. Its SNR is varied between -20 dB and 60 dB.

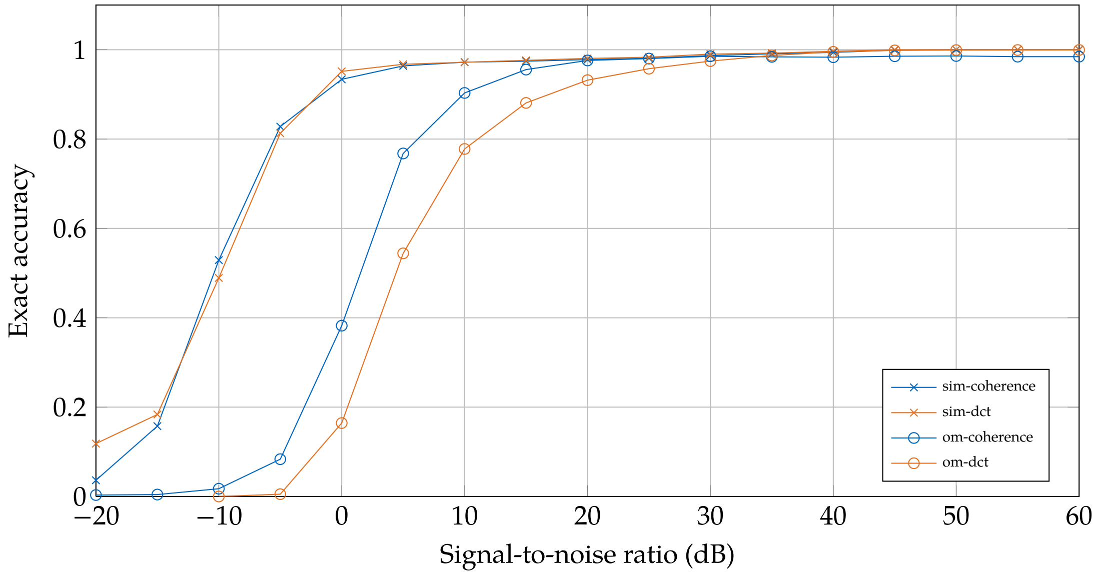


Figure 6.5: Localization accuracy over the signal-to-noise ratio of an interfering noise signal. A computer fan noise signal is spatialized to a position in the virtual room.

The noise signal is the recording of a computer fan and corresponds to one typical type of noise source on a robotic platform. Figure 6.5 shows the localization accuracy over the SNR. Compared to the global noise signals of the previous test, lower SNRs are sufficient to reach the same localization accuracy.

6.3.9 Noisy transfer functions

Not only the observations during the localization process can be subject to noise, but also the TF database entries can be noisy if their recording conditions were not perfect. This test analyzes how noisy TFs affect COMPaSS and which SNR is required to guarantee a good localization quality:

Sound sources Every sound scene has one active sound source. At each position three speech and three music signals are tested.

Positions Eight positions are randomly chosen.

Others White noise is added to each impulse response from the TF database. The signal-to-noise ratio (SNR) is varied between 0 dB and 30 dB.

The resulting accuracies are plotted in Figure 6.6. Regardless of the environment, COMPaSS performs poorly at low SNRs and requires around 20 dB SNR to reach its maximum performance. In

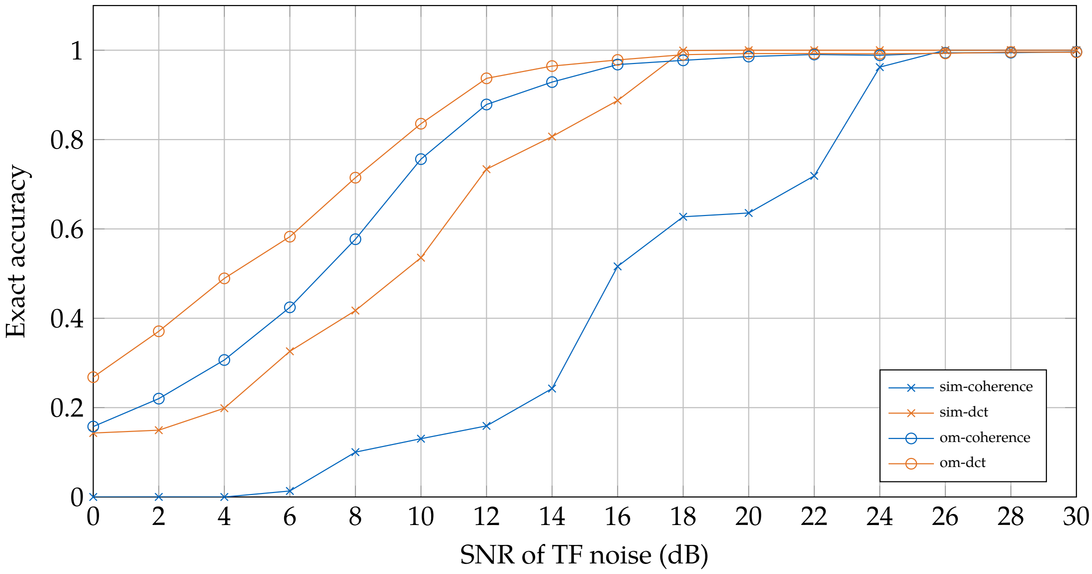


Figure 6.6: Localization accuracy over the signal-to-noise ratio of the impulse responses from the TF database. With observation mixing COMPaSS reaches its maximum performance for SNRs over 20 dB. The better performance of observation mixing compared to the simulations can be attributed to their longer impulse responses.

the simulations, the coherence similarity measure performs approximately 10 dB worse than the DCT, but this difference is not that pronounced with real-world data. Interestingly, observation mixing performs better than the simulations. This might seem curious, but can be explained by the longer impulse responses that are used in the real environment.

6.3.10 Single source in the horizontal plane

All previous tests average the localization accuracy over multiple random source positions, since the source positions themselves influence the algorithm performance. This test evaluates COMPaSS's performance for a single source for different lateral angles:

Sound sources Each sound scene has one sound source. At each position three speech and three music signals are tested.

Positions The single source is always on the zero elevation plane. Its position is varied in 5° steps around the listener.

Others COMPaSS localizes sound scenes with only one source with a very high accuracy. To ensure that localization errors occur frequently, noise is added to the simulations (20 dB) and to observation mixing (30 dB).

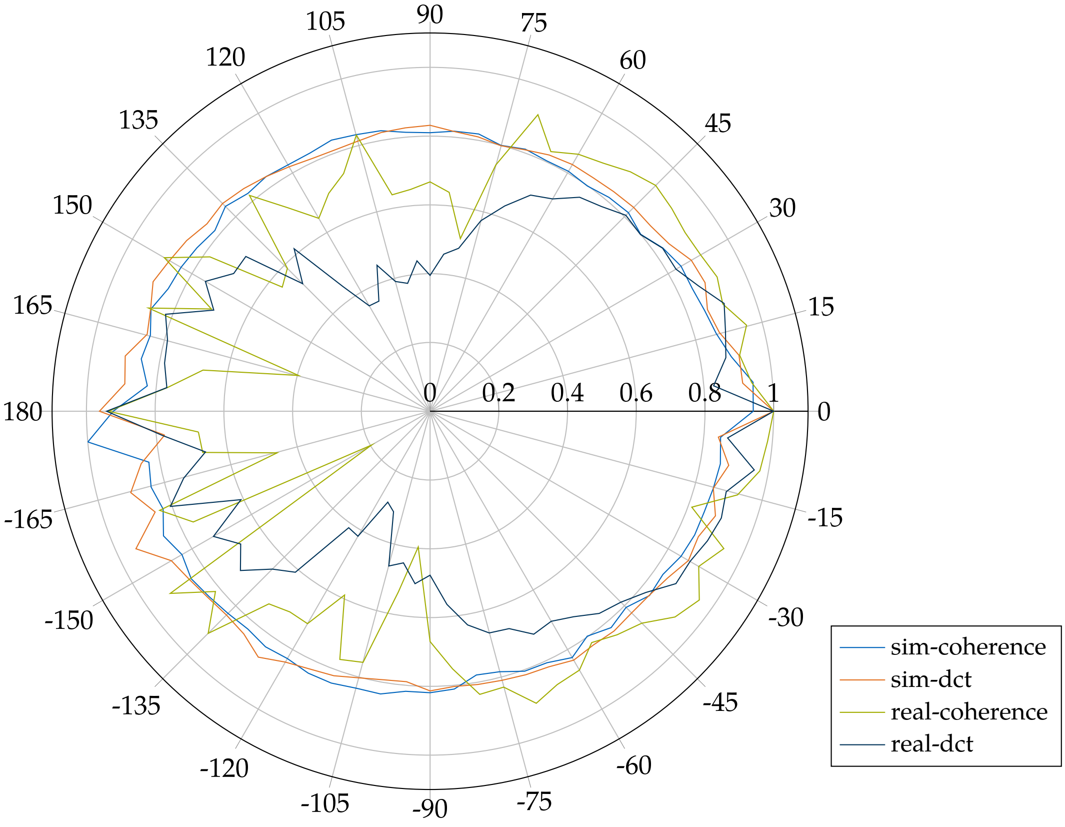


Figure 6.7: Radial plot of the localization accuracy over the azimuth angle. The KEMAR is viewed from above and is facing 0°. The sound source is on the horizontal plane and is moving around the listener. The achieved accuracy at each angle is indicated by the radius of the curve in the plot.

Figure 6.7 shows a radial plot of the localization accuracy over the azimuth angle. The sound scene is seen from above with the listener looking towards 0°. The radius indicates the localization accuracy in the radial plot. As the simulations and observation mixing are created at different SNRs the results are only comparable qualitatively. In general, COMPaSS behaves similarly in both environments. It performs best in the vicinity of 0° and ±180° and has its worst accuracy near ±90°. The accuracy drop on left and right side comes from the low signal power and therefore bad SNR at the contralateral ear, where the source is fully obscured by the head.

6.3.11 Single source on the cone of confusion

The previous section analyzes a single source moving in azimuth direction on the horizontal plane. An identical test can be performed with a source moving in elevation direction on a cone of confusion. On such a source trajectory, all different source positions have identical ITD and ILD values. Source trajectories on two different cones of confusion are tested with the following parameters:

Sound sources Each sound scene has one sound source. At each position three speech and three music signals are tested.

Positions The first trajectory lies on the cone of confusion in the median plane ($\varphi = 0°$) and the source moves on a circle in 5° steps from $-45°$ elevation to 225°. Those are all elevations that could be recorded in the anechoic chamber. The second cone of confusion is at azimuth angle $\varphi = 60°$. The radius of the corresponding trajectory is smaller by a factor of $\cos(60°)$, as the grid points of the TF database lie on a sphere.

Others Observation mixing is omitted in this test, as not all necessary elevations could be recorded in the office environment. As in the previous section a 15 dB noise signal is added to the simulations to ensure that localization errors occur frequently.

Figure 6.8 shows radial plots of the resulting localization accuracies. The KEMAR is located in the center of the plot and is looking towards $\theta = 0$, the elevation $\theta = 90$ corresponds to a point above the KEMAR. The accuracy on the zero azimuth plane (left plot) is almost perfect, with some minor errors occurring in a small region directly above the listener. At an azimuth of 60° (right plot) the accuracy is worse. This can be attributed to the same occlusion effects as observed in the previous test on the horizontal plane. Additionally, in the second case the distance between neighboring grid points is smaller, which possibly makes it harder to distinguish from one grid point to another.

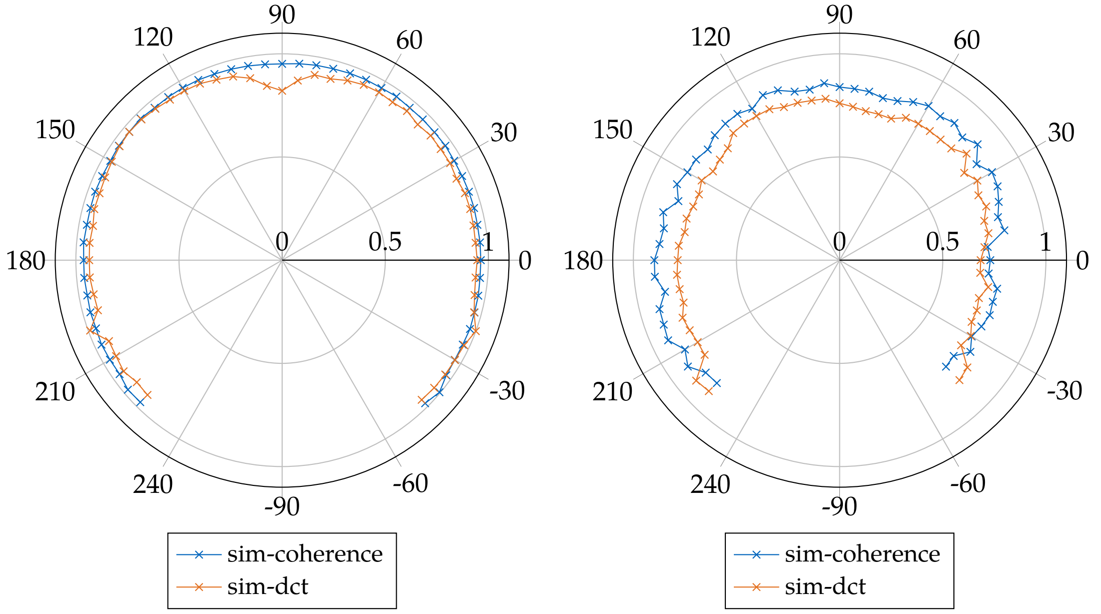


Figure 6.8: Localization accuracies over the elevation angle on two cones of confusion. The KEMAR is located in the center of the plot and is looking towards $\theta = 0$. The left and right plot show the cones of confusion at azimuth $\varphi = 0^\circ$ and $\varphi = 60^\circ$ respectively.

6.3.12 Minimal distance between two sources on the horizontal plane

If the distance between two active sources becomes too small, a localization algorithm might become unable to recognize them as two individual signals. This test analyzes the localization accuracy as a function of the distance between two sources on the horizontal plane:

Sound sources Each sound scene has two active sound sources. Both sources emit a speech sound signal.

Positions The first sound source is located at eight different positions on the horizontal plane. For each position the second sound source starts at a distance of 50° and moves in 5° steps towards the first source.

Others As done in previous tests, noise is added to the sound scenes to ensure that localization errors occur frequently.

As can be seen in Figure 6.9, the performance of the algorithm is constant for large distances and starts to deteriorate for distances below 15°. At 5° the coherence variant is almost completely unable to detect the two sound sources individually. However, the accuracy of the DCT variant remains

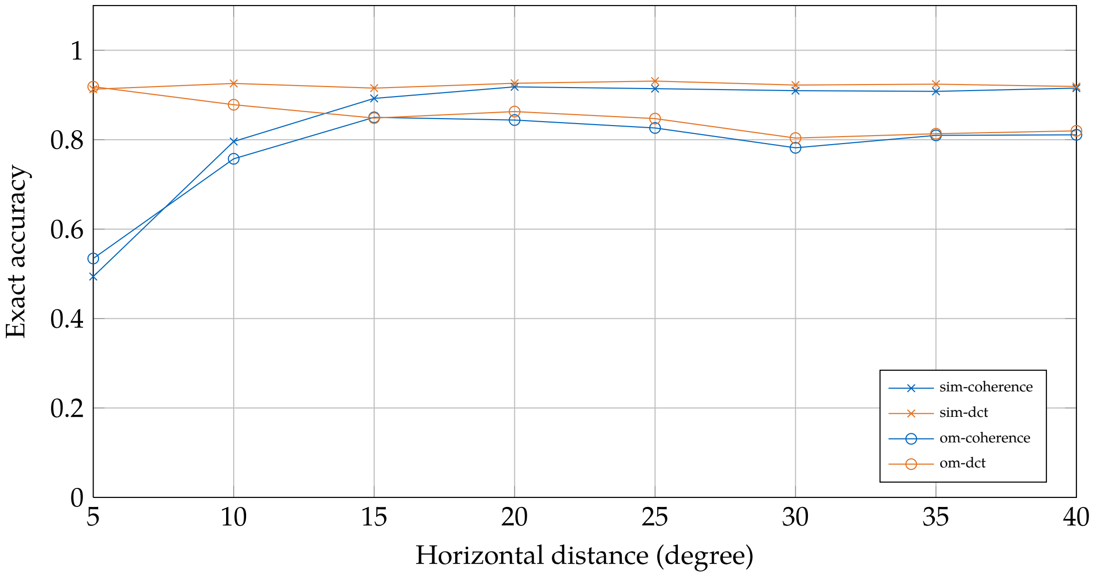


Figure 6.9: The localization accuracy for two sound sources on the horizontal plane as a function of the distance of the sources.

stable even at the short distance of 5°. This strength is caused by the DCT variant's narrower areas of high similarity.

6.3.13 Minimal distance between two sources on the cone of confusion

This test evaluates the minimal distance between two sources on the cone of confusion, as conducted previously for the horizontal plane:

Sound sources Each sound scene has two active sound sources. Both sources emit a speech sound signal.

Positions The first sound source is located at eight different positions on the median plane ($\varphi = 0$). For each position the second sound source starts at a distance of 50° and moves in 5° steps towards the first source.

Others Observation mixing does not have recordings for all necessary elevations and is omitted in this test. As done in previous tests, noise is added to the sound scenes to ensure that localization errors occur frequently.

Figure 6.10 shows the localization accuracy over the distance of the sources. The behavior is very similar to the previous test. The DCT variant has a good accuracy regardless of the distance of the

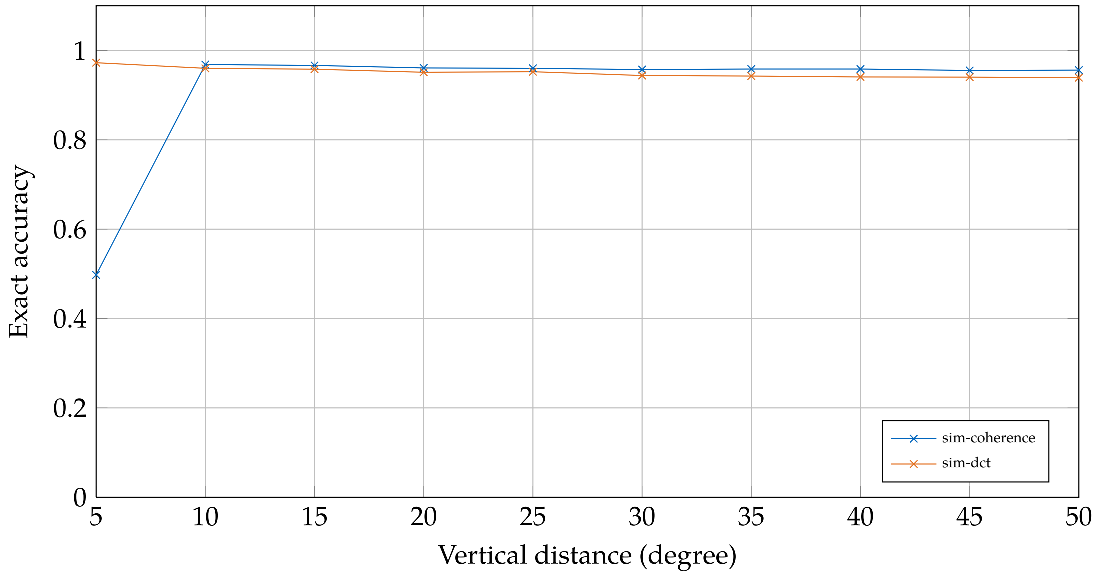


Figure 6.10: The localization accuracy for two sound sources on the median plane as a function of the distance of the sources.

two sources. The coherence variant has a comparable accuracy, but is unable to detect both sources if their distance is 5° in elevation direction.

6.3.14 Length of the impulse responses in the TF database

In reverberant environments, the transfer functions between a sound source and the microphones are influenced by room effects. Room effects are unique to each pair of source and microphone positions and can therefore improve the localization accuracy. The amount of room effects that is used for the localization can be regulated implicitly with the length of the TFs (in samples) [31]. This test investigates the dependence of the localization accuracy on the TF length:

Sound sources In each sound scene three speech sources are active.

Positions 25 source positions triples for three simultaneous sources are randomly chosen.

Others The length of the impulse responses is varied between 128 and 1024 samples.

Table 6.5 shows the resulting localization accuracies. In the simulations the TF length does not have a large influence, as the impulse responses are recorded in the anechoic chamber and decay quickly within the first 128 samples. In contrast to this, the results obtained with the observation mixing data are strongly dependent on the TF length. At an impulse response length of 1024 samples,

TF length	Coherence			DCT		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
simulation						
128	0.83	0.85	14.63	0.84	0.84	16.24
256	0.90	0.90	9.52	0.87	0.88	13.45
512	0.90	0.91	9.34	0.87	0.88	12.94
1024	0.89	0.90	10.39	0.87	0.87	13.04
observation mixing						
128	0.07	0.19	57.90	0.26	0.37	45.14
256	0.50	0.64	30.45	0.56	0.63	31.92
512	0.94	0.95	5.14	0.93	0.94	6.18
1024	0.96	0.97	3.18	0.96	0.97	3.12

Table 6.5: Localization accuracy for sound scenes with three speech signals, while varying the length of the impulse responses in the TF database.

COMPaSS localizes the three speech signals with an accuracy of 0.96, which is higher than in the anechoic simulations. A downside of using more room related effects for localization is that the TF database becomes dependent on the position of the KEMAR in the room [31].

6.3.15 Localization with coarse databases

As TF databases are sampled on a spatial grid, actual source positions will often lie between grid points. Coarse databases are sometimes preferable, because COMPaSS's computational complexity scales with the number of TFs in the database. On the one hand, smaller databases are sampled on a coarser spatial grid and subsequently COMPaSS has to take fewer possible source locations into account. On the other hand, the resolution of the localization results decreases with the grid point density. A smaller density increases the maximum possible distance between an actual sound source and its nearest grid point. The following test investigates COMPaSS's behavior when a sound source lies between two grid points:

Sound sources Every sound scene has one active sound source. At each position three speech, three music, and one noise signal are tested.

Positions The 25 source positions are on the horizontal plane. Their azimuth angles are randomly chosen from the set of intermediate azimuths $\{k \cdot 5^\circ + 2.5^\circ\}, k \in [-36, \dots, 35]$.

Table 6.6 shows the resulting localization accuracies. As the localization algorithm cannot determine the true position of the source, the localization of any of the closest neighboring grid points is deemed correct. The coherence variant determines location of the sound source without errors.

Source type	Coherence			DCT		
	Exact	**Tolerance**	**Front-back**	**Exact**	**Tolerance**	**Front-back**
speech	1.00	1.00	0.00	0.75	0.75	0.25
music	1.00	1.00	0.00	0.71	0.71	0.29
noise	1.00	1.00	0.00	0.78	0.78	0.22

Table 6.6: Localization accuracy of sound sources with a coarse database. The simulated position of the sound source lies on the horizontal plane between two neighboring grid points of the TF database. The columns titled Front-back list the percentage localization results that were subject to front-back confusions.

This result was expected as the coherence similarity measure yields also a high similarity for positions that are near the true source location. The DCT variant seems error-prone and achieves an exact accuracy of approximately 0.75 for all three signal types. The similarity values yielded with the DCT decrease quickly with azimuth. Thus, the algorithm sometimes does not identify the neighboring grid points as the source position.

Instead of the MAE, the percentage of front-back confusions is given in Table 6.6. For the DCT variant these numbers indicate that all its localization errors are caused by front-back confusions. In all previous tests the confusion effects were also evaluated and neither of the two COMPaSS variants showed to be prone to this type of error. The present test indicates that the DCT variant of COMPaSS sometimes favors the front-back confusion instead of a neighboring position, when a sound source is not located on a grid point of the TF database.

6.4 COMPaSS in a real environment

An evaluation of COMPaSS with real-world recordings is necessary to verify that its performance holds up under realistic conditions.

6.4.1 Verification of the observation mixing method

The evaluation of COMPaSS's properties uses sound scenes that are created by simulations or observation mixing. In Chapter 5, I have proposed the use of observation mixing, when the recording of a large number of sound scenes is practically unfeasible.

The following evaluation investigates the validity of the observation mixing method experimentally. Its goal is to measure the difference between real-world sound scenes and sound scenes created with observation mixing. To this end the evaluation recreates the recorded sound scenes from the manual experiments with observation mixing and subsequently processes both types of sound scenes with different localization algorithms. The localization accuracies for both types of sound scenes are given in Table 6.7. The difference of the accuracy values between real recordings and

Algorithm	Real recordings			Observation mixing		
	Exact	Tolerance	MAE (°)	Exact	Tolerance	MAE (°)
compass-coh	0.82	0.85	12.51	0.80	0.83	13.84
compass-dct	0.82	0.84	13.03	0.80	0.83	13.67
csscl	0.35	0.42	37.48	0.35	0.43	36.30
fdbm	0.39	0.54	30.52	0.38	0.53	30.46
srp-phat	0.29	0.49	43.87	0.29	0.48	44.72
srp-phat-pf	0.44	0.79	17.78	0.45	0.76	18.60

Table 6.7: Evaluation results for real recordings and observation mixing. The numbers are virtually identical and indicate that evaluations performed with observation mixing are valid.

observation mixing is maximally three percentage points and the MAE deviates by circa 1°. The small deviation of the measured values indicate that observation mixing creates sound scenes that closely resemble real recordings.

6.4.2 Comparison with state-of-the-art techniques

The following comparison of COMPaSS to the viable state-of-the-art methods, which were identified in Section 3.2, is split into two parts. The first one looks at the per frame localization results of all algorithms and interprets these results. The second one is a comparison of the overall localization performance of the different algorithms.

Per frame localization results

A sound scene with three speaker signals at azimuth angles −80°, −30°, and 80° is processed with all localization algorithms. The per frame localization results of each algorithm are shown in Figure 6.11. The positions of the actual sources are indicated by thin horizontal lines in the plots, the localization results are indicated by the colored markers. The algorithms behave quite differently:

- With both similarity measures COMPaSS estimates source positions almost 100% correctly. Only the source at −80° (blue) is not detected right from the beginning of the recording.

- The real-world evaluation of CSSCL revealed that the algorithm has to process sound data in large blocks to create reliable localization results. Due to this high algorithmic latency, CSSCL is not able to create per frame localization results and calculates one location estimate for the whole recording instead. CSSCL incorrectly detects a source at 0° instead of finding the real source at −80° (blue).

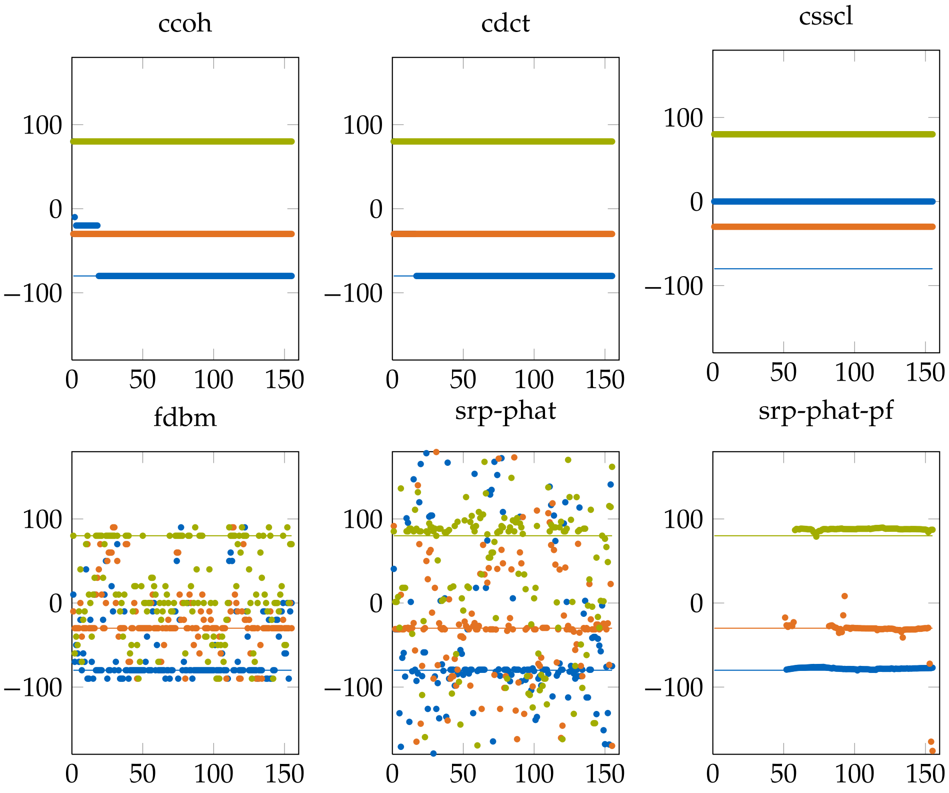


Figure 6.11: Each plot shows the detected azimuth angles of the three active sources at every time instance. The thin lines indicate the actual source position (−80°, −30° and 80°) and the markers depict the respective algorithm's estimation.

- The results of FDBM appear noisy, but the estimated positions form visible clusters at the real source locations. The source at 80° (green) is rarely detected correctly and erroneous detections cluster around 0°.
- SRP-PHAT behaves similar to FDBM. The major difference is that the clustering at 80° (green) is more pronounced and that the erroneous detections do not form clusters.
- SRP-PHAT-PF does not identify the positions of any sound source until the 50th frame, which corresponds to an internal latency of circa 1.5 s caused by the particle filter. In contrast to the previous algorithms, its results are continuous on the azimuth scale. The source at −30° (orange) disappears shortly after it was detected and reappears approximately 30 frames later. Apart from that, the results closely follow the correct source locations.

COMPaSS shows the best performance in this per-frame comparison. CSSCL disappoints with its huge latency requirement, as this property contradicts the constraints of the robot auditory system. FDBM and SRP-PHAT create results with many erroneous detections. Given the noisy results of SRP-PHAT, the performance of SRP-PHAT-PF after the initial delay is amazing as the noisy results are almost perfectly filtered.

Overall localization performance

The localization success rates of all algorithms for sound scenes with one, two, and three sound sources are shown in Table 6.8. As COMPaSS (coh) and COMPaSS (dct) have almost identical numbers, they are not discussed separately.

In the simulations, COMPaSS achieves an exact localization accuracy of 100% for the one-source scenarios and drops only little to 96% in the multi-source scenarios. CSSCL can also flawlessly localize one source, but its accuracy drops to 71% in the three-source case. FDBM has a single-source accuracy of 88% and suffers a big accuracy loss of 36 percentage points when two sources are added. The SRP-PHAT beamformer has a much lower exact accuracy of 42% to 28%. The combination of the beamformer with a particle filter SRP-PATH-PF has a single-source exact accuracy of 51% and drops only little to 48% for three sources. The behavior of the TF-based algorithms does not change notably when the exact accuracy requirement is eased to the accuracy in a small tolerance region. However, the performance of SRP-PHAT-PF increases drastically and surpasses FDBM in all cases and CSSCL in the presence of three sources. In the three-source case COMPaSS and SRP-PHAT-PF make only small mean angular errors (MAEs) of approximately 3° and 11° respectively. CSSCL and FDBM have MAEs of 19.77° and 27.76° and estimated source positions will therefore, on average, deviate significantly from the actual source locations.

Algorithm	Simulation			Real recordings		
	Exact	**Tolerance**	**MAE (°)**	**Exact**	**Tolerance**	**MAE (°)**
one source						
compass-coh	1.00	1.00	0.00	1.00	1.00	0.00
compass-dct	1.00	1.00	0.00	1.00	1.00	0.00
csscl	1.00	1.00	0.00	0.72	0.72	18.25
fdbm	0.88	0.90	6.79	0.50	0.61	28.41
srp-phat	0.42	0.78	23.80	0.42	0.76	30.55
srp-phat-pf	0.51	0.98	5.51	0.47	0.92	11.01
two sources						
compass-coh	0.97	0.97	2.33	0.92	0.93	5.83
compass-dct	0.97	0.98	1.98	0.92	0.93	5.55
csscl	0.90	0.90	8.28	0.42	0.47	39.57
fdbm	0.59	0.65	24.94	0.39	0.52	33.13
srp-phat	0.33	0.55	38.68	0.33	0.56	41.94
srp-phat-pf	0.49	0.87	11.07	0.44	0.82	16.87
three sources						
compass-coh	0.96	0.97	3.14	0.82	0.85	12.51
compass-dct	0.96	0.97	2.69	0.82	0.84	13.03
csscl	0.71	0.73	19.77	0.35	0.42	37.48
fdbm	0.52	0.59	27.76	0.39	0.54	30.52
srp-phat	0.28	0.46	43.19	0.29	0.49	43.87
srp-phat-pf	0.48	0.84	11.46	0.44	0.79	17.78

Table 6.8: Localization results for identical sound scenes obtained by simulations and recordings in real environments. The percentage of correctly localized sources and the percentage of localizations lying in a 15° tolerance region are given. Additionally, the mean angular error (MAE) was measured.

The differences between the simulations and the real recordings are the presence of noise and possible deviations of the measured TFs. These deviations may arise due to changes in air pressure, room temperature, and the finite length of the measured TFs.

COMPaSS and SRP-PHAT-PF prove to be robust towards real conditions as their performance is not affected drastically. COMPaSS shows an excellent performance in the single-source case, and a still acceptable accuracy of 82% in the three-source case. The measurements for SRP-PHAT-PF change only slightly and its exact accuracy for three sources is now in second place after COMPaSS. CSSCL and FDBM disappoint in the real environment compared to the promising numbers of the simulations.

6.5 Discussion

This chapter evaluated COMPaSS's properties and compared its performance to state-of-the-art techniques. The comparison revealed that COMPaSS and SRP-PHAT-PF are suitable candidates for practical use in real environments. Both have a high localization accuracy in the tolerance region paired with a small MAE. If exact localization results are required, COMPaSS is the better choice as its accuracy is significantly higher in all cases.

COMPaSS, FDBM, and SRP-PHAT estimate source positions for every sound frame. The three algorithms do not introduce any additional latency. The evaluation of CSSCL uncovered that a practical implementation of the algorithm has to process sound data in large sound blocks resulting in a high latency. The particle filter of SRP-PHAT-PF has an internal latency and the algorithm requires a few sound frames until it starts localizing a sound source. Although CSSCL and FDBM show promising results in the simulations, their localization results are flawed in reality. Apparently, despite all efforts, the simulations do not properly account for the effects that are present under real conditions. While simulations are easy to perform, all results obtained from simulations have to be handled with reservations and should always be verified with real-world tests.

In the real-world evaluation the coherence and DCT similarity measures showed almost the same performance. The testing of COMPaSS's properties revealed that both approaches have their strengths. For example, the coherence measure performs more accurately with coarse databases, while the DCT measure can better distinguish sources that are close together. The choice which method to use will depend on the needs of the application.

It is also imaginable to use both measures simultaneously in a coarse to fine grid search. A rough source localization could be performed using the coherence measure with its broad peaks and a coarse TF database. Afterwards, the search could be refined around these positions using the DCT measure and a TF database with a finer grid.

7 Tracking of dynamic sound scenes

The previous chapters introduced the COMPaSS algorithm and presented an evaluation of its performance for sound scenes with stationary parameters, most importantly the number of active sound sources and their respective positions. This chapter examines dynamic sound scenes and proposes an approach for tracking of scene parameters.

7.1 Dynamic sound scenes

In static sound scenes every parameter that defines the sound scene is fixed. However, in the real world some of these parameters are likely to change and a sound processing system has to take dynamic sound scenes into account. The most likely parameter changes include:

Source movement The position of a source can change over time due to movement. Sound sources can exhibit any movement pattern and also change their state from stationary to moving and vice versa. If no high level information about a sound source is available, no assumptions about its movement behavior can be made. In this case the localization system has to be able to follow arbitrary movements. A robot's ego motion also changes the relative positions of the sources to the robot. However, this motion is known to the robot, and the auditory system can take it into account.

Appearing of new sources In static sound scenes, all sources are emitting a signal all the time. In reality, the activity of a sound source can change dynamically. New sound sources can become active at any time and the auditory system has to be able to determine both the number of currently active sources and their positions.

Vanishing of existing sources An active source can stop emitting sound, which has to be taken into account by the auditory system. If the auditory system does not observe a sound source for a certain amount of time, it can assume that the source has vanished.

The definition of vanishing given above is vague, as different types of sources require different inactivity times in order to confidently establish that they have vanished. On the one hand, the processing should use the shortest possible inactivity time to be able to detect the disappearing of

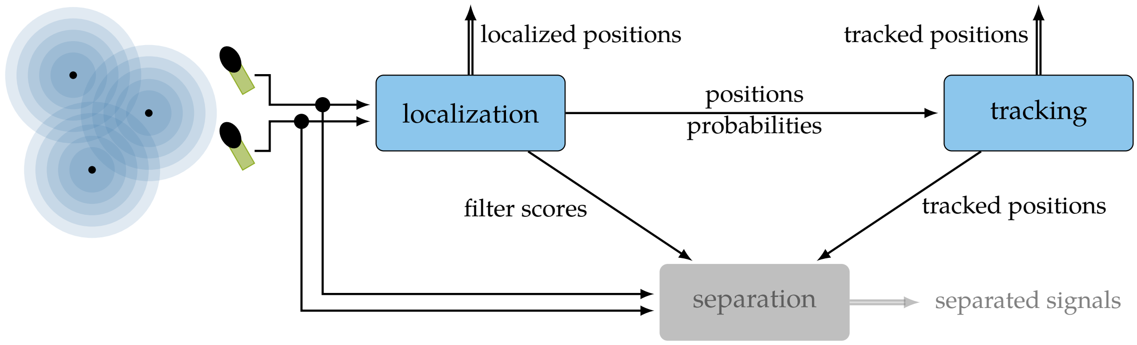


Figure 7.1: Signal flow diagram for the complete auditory system. The tracking processes the localization results and provides the tracked source positions to other auditory system modules.

sources as soon as it happens. On the other hand, some sound signals have natural pauses and a localization system that uses too short inactivity times will e.g. detect individual sound sources for each spoken word of a sentence. If no high level knowledge about the type of sound source is available, the inactivity time has to be set to a tradeoff between fast response and robustness towards short pauses.

In the context of dynamic sound scenes, tracking of given parameters becomes important. Due to its low algorithmic latency, COMPaSS is already able to follow source movements and a subsequent tracking stage should render the positions more precisely and add and remove sound sources from the tracking on-line. A signal flow diagram for COMPaSS with subsequent source tracking is given in Figure 7.1.

7.2 Complications caused by the localization

Some unavoidable properties of COMPaSS's localization results will cause some complications for the tracking:

Source temporarily not detectable Sometimes COMPaSS cannot detect an active sound source in one sound frame. This happens mostly, when the other active sources are more dominant in a time interval. The tracking approach cannot distinguish between inactivity of a sound source and the temporal non-detectability by the localization algorithm. Sporadic non-detectability of sound sources can partly be remedied by the tracking system with requiring longer inactivity times for the removal of sources.

Erroneous source detection COMPaSS determines the source locations where actual sound sources are most likely to be present. This estimation can be erroneous mostly due to noise and reverberation. The tracking system needs to filter out these outliers.

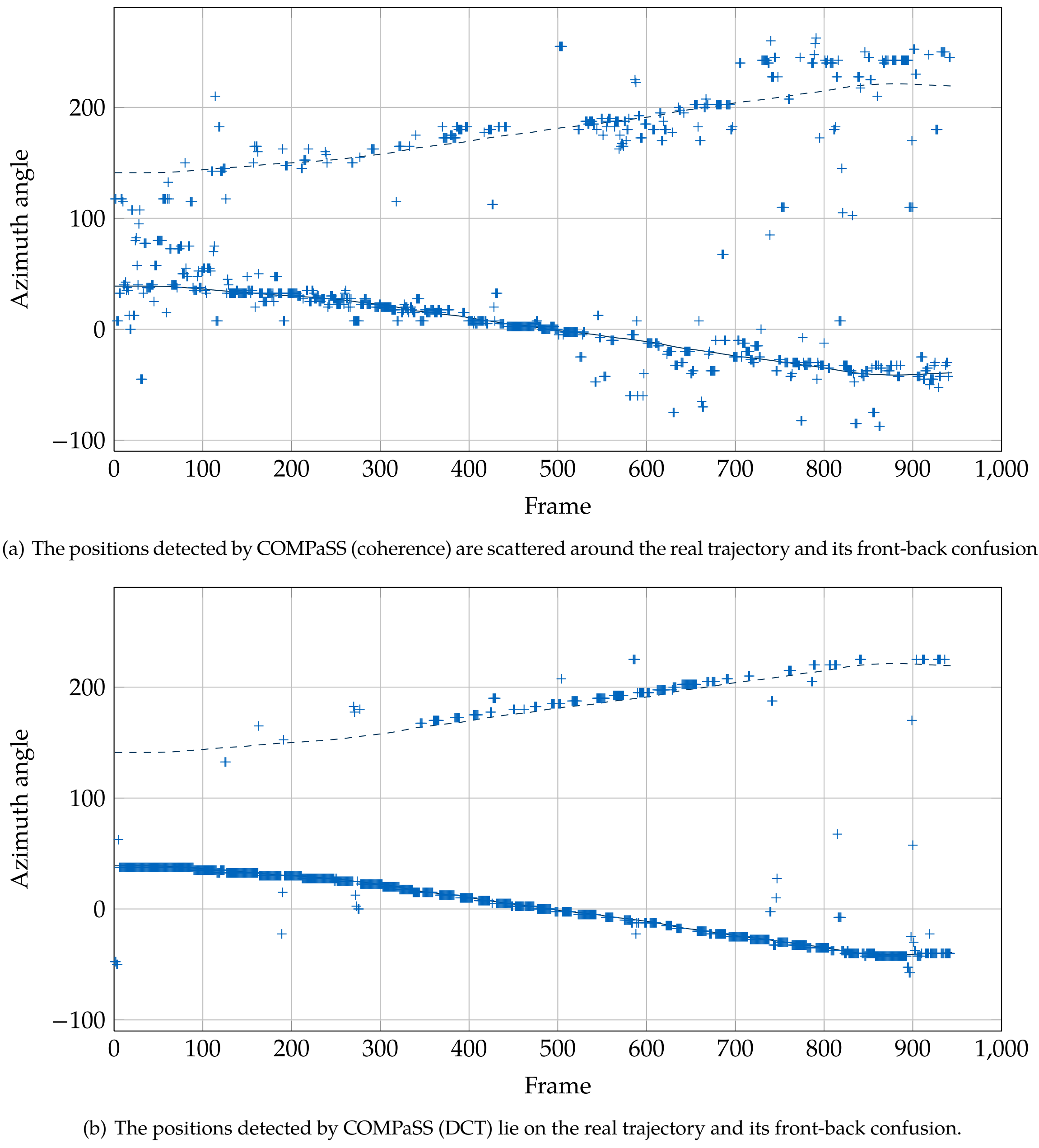
Observation assignment The sound localization is performed in each sound frame individually and estimates the likely source positions. Even if the positions of multiple sound sources are stationary, the probabilities of COMPaSS's results will change over time as the signals change. This leads to a permutation ambiguity, as the most likely position may not always correspond to the same source in different frames.

The observation assignment is the most complicated problem to solve. In theory, it should be easy to permute the localization results of two consecutive sound frames such that the pairwise Euclidean distance between the estimated positions is minimized. This would ensure that even observations of sources that are moving are assigned to the same tracked source. However, due to temporary non-detectability and erroneous source detection, observation assignment becomes complicated, as the minimization of pairwise distances cannot account for those effects. In [108] a solution of this problem was implemented and the tracking system for COMPaSS will incorporate some of the algorithm's ideas.

7.3 Effects of source movement on the localization

In Figure 7.2 the result of COMPaSS localizing a moving speech source is presented. The plots show the estimated azimuth angle in each sound frame. In 7.2(a) the coherence measure and in 7.2(b) the DCT measure is used for similarity calculations. The correct trajectory of the sound source is indicated by the solid line and its front-back confusion is given by the dashed line. As the front-back confusion passes through azimuth $\pm 180°$ and the plot is periodic in y-direction, the azimuth angle is shown in the range between $[-90°, 270°]$ to remove the discontinuity at $\pm 180°$.

A high percentage of the positions estimated by COMPaSS shown in Figure 7.2(a) lies on the actual trajectory and is scattered in the vicinity of $\pm 20°$ around the real positions. Another significant part of the localization results is scattered around the trajectory of the sound source's front-back confusion. Only a few results are complete outliers and cannot be attributed to any of the two previous classes. The DCT measure shown in Figure 7.2(b) differs from the coherence measure in that the results are far less noisy and follow the actual source trajectory more precisely. Front-back confusions are also present, but they are not as pronounced as with the coherence measure. Furthermore, the number of complete outliers is also significantly smaller. The DCT measure seems to produce much better results for the localization of moving sound sources than the coherence measure.



(a) The positions detected by COMPaSS (coherence) are scattered around the real trajectory and its front-back confusion.

(b) The positions detected by COMPaSS (DCT) lie on the real trajectory and its front-back confusion.

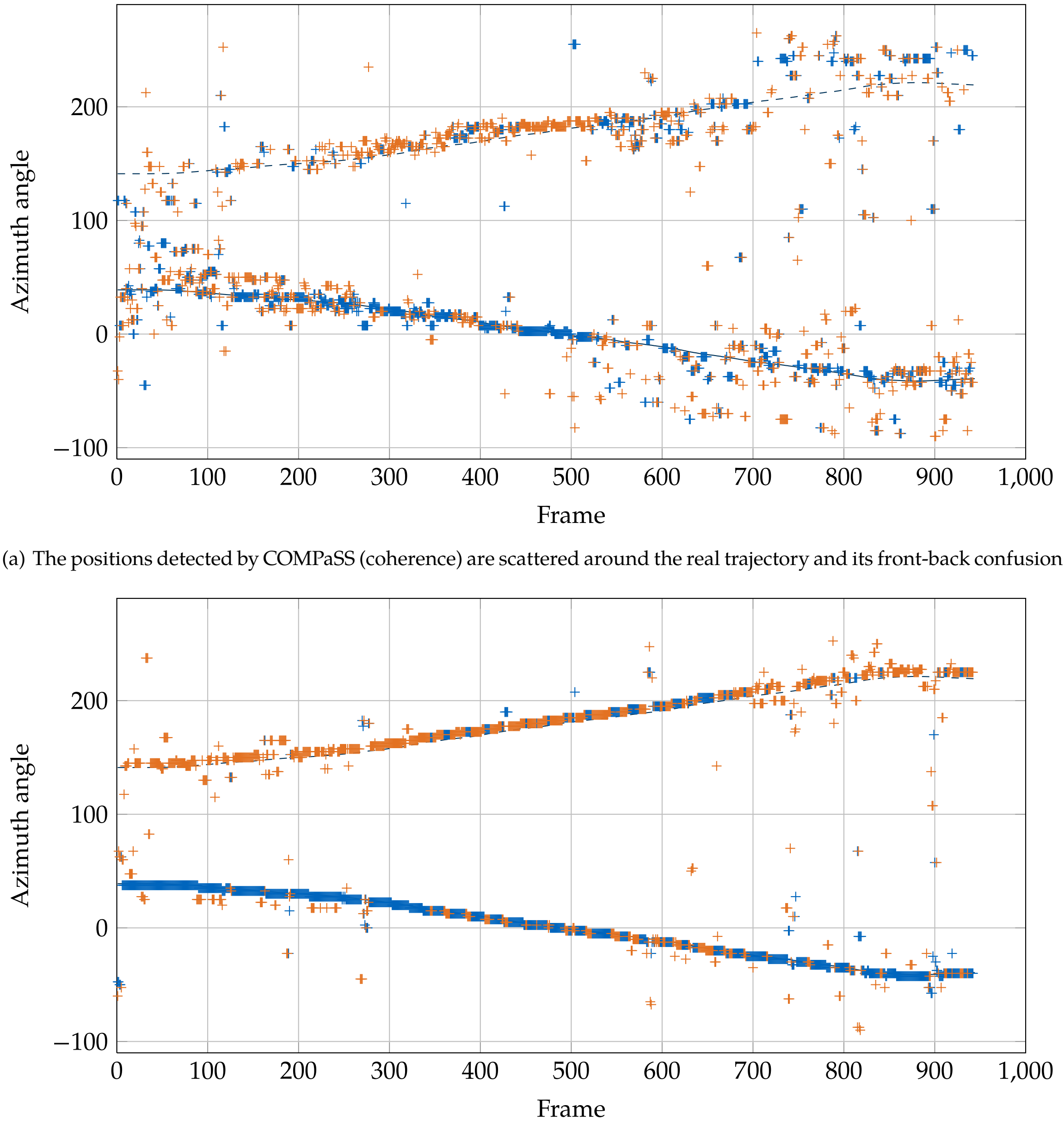
Figure 7.2: Raw COMPaSS localization for a moving sound source. The source is moving along the solid line and the position of its front-back confusion is indicated by the dashed line. The plot is periodic in y-direction and the azimuth angle range between [−90°, 270°] is shown to remove the discontinuity at ±180°. Comparing the two similarity scoring approaches, the DCT measure shows significantly less noisy results and fewer front-back confusions.

The most interesting result from this simple experiment is the occurrence of front-back confusions, as those were less pronounced during the test of COMPaSS with static sound sources. One thing to note is the slight asymmetry between the correct and front-back confused localization results, which can be seen best in Figure 7.2(b). While the correct localizations lie exactly on the real source trajectory, the detected front-back confusions are slightly off the expected trajectory. The deviation is small and corresponds to the grid point distance of the sampled HRTF database. The sound of a source reaches the contralateral ear by refraction around the head and the asymmetry of a head's front and back can slightly shift the positions of equal ITD and ILD. Therefore, the front-back confusion of a source at $[x, y, z]^T$ does not necessarily lie at $[x, -y, z]^T$. When using spatial sampling this means that the front-back confusion of one point is possibly not on the spatial grid but lies between two sampled positions.

The fewer occurrences of front-back confusions in static sound scenes can be explained by COMPaSS's winner-takes-all step in the determination of the most likely positions. It completely dismisses the similarity scores of the front-back confusion when the correct source position is dominant. However, if the sound source does not lie on a grid point, the similarity scores are distributed among the closest neighbors. Additionally, the position between two grid points can correspond to the front-back confusion of a point on the opposite side of the head. The co-occurrence of these two effects causes the detection of the front-back confusion instead of one of the geometrically closer points.

To better illustrate COMPaSS's behavior, the same single-source sound scene is localized again and the two most likely sound source positions are extracted and plotted in Figure 7.3. The first thing to note is that the results of the first sound source have not changed at all. This is due to COMPaSS's iterative extraction process of the most likely sound sources from the similarity values. The additional localization results for the nonexistent second source are mostly clustered around the front-back confusion. Interestingly, when a front-back confusion occurs in the localization of the first source, the results of the second source are also front-back confused yielding the actual position of the source. This means that the similarity values of the actual source are still second highest in case of a front-back confusion.

In summary, COMPaSS produces comprehensible localization results for moving sound sources. Especially the DCT measure yields stable position estimations that should be a suitable input for a subsequent tracking approach. One problem that arises is the occurrence of front-back confusions, but even in this case COMPaSS behaves predictably.



(a) The positions detected by COMPaSS (coherence) are scattered around the real trajectory and its front-back confusion.

(b) The positions detected by COMPaSS (DCT) lie on the real trajectory and its front-back confusion.

Figure 7.3: Raw COMPaSS localization of two sources for a sound scene with only one moving source. The source is moving along the solid line and the position of its front-back confusion is indicated by the dashed line. The different marker colors correspond to the different detected sources. The front-back confusion is identified as the nonexistent second source. Comparing the two similarity scoring approaches the DCT measure shows significantly less noisy results.

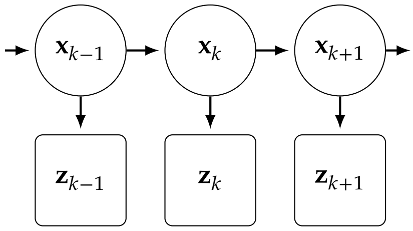


Figure 7.4: A particle filter assumes the state to be a first order Markov process with the unobserved state variables $\mathbf{x}_k$. At each time instance the measurements $\mathbf{z}_k$ are taken.

7.4 Design of a tracking approach

The previous section showed COMPaSS's behavior when the sound sources are moving. Starting with this knowledge this section will derive a suitable tracking approach.

7.4.1 Sequential Monte Carlo simulation

Different types of Bayesian filters have been used for sound source tracking in the past. Kalman filters assume that the state transition and observation models are linear and that the process and measurement noises are Gaussian. The front-back confusions in COMPaSS's localization results breach these conditions, hence the Kalman filter is not applicable. The extended Kalman filter is also not usable here, since the measurement density is a non-Gaussian, multimodal distribution due to front-back confusions.

Sequential Monte Carlo simulations, also known as particle filters, are suited for nonlinear or non-Gaussian Bayesian tracking [7]. A particle filter estimates the hidden state variable $\mathbf{x}_k$ at each discrete time step k based on measurements $\mathbf{z}_k$. Therefore, it estimates the posterior distribution $p(\mathbf{x}_k|\mathbf{z}_{1:k})$, where $\mathbf{z}_{1:k}$ denotes all measurements $(\mathbf{z}_1, \ldots, \mathbf{z}_k)$. The evolution of the unobserved state variable is assumed to be a first order Markov process and is schematically shown in Figure 7.4. In this Markov chain, the next state of the variable only depends on the previous state and the posterior can be computed with a two-step recursion. The first step of this recursion is a prediction

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})\, d\mathbf{x}_{k-1}, \tag{7.1}$$

where $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ denotes the state transition probability, which is given by a model of the system dynamics, and $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ is the previous posterior distribution. In the filtering or update step, the new posterior is given by

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} = \alpha\, p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1}), \tag{7.2}$$

where $p(\mathbf{z}_k|\mathbf{x}_k)$ is the measurement density, and $p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})d\mathbf{x}_k$ is a normalizing factor.

In practice the probability distributions cannot be given analytically and have to be approximated. Particle filters represent the posterior by a set of N_p weighted samples $\{\mathbf{x}_k^{(i)}, w_k^{(i)}\}_{i=1}^{N_p}$ that are drawn from the probability distribution. Using these particles the prediction step corresponds to calculating the next state of a particle $\mathbf{x}_k^{(i)}$ from its previous state $\mathbf{x}_{k-1}^{(i)}$

$$\mathbf{x}_k^{(i)} = \mathbf{g}(\mathbf{x}_{k-1}^{(i)}), \tag{7.3}$$

where $g(\cdot)$ models the system's dynamics including process noise. In the next step, the samples are reweighted via

$$w_k^{(i)} = \alpha w_{k-1}^{(i)} p(\mathbf{z}_k|\mathbf{x}_k^{(i)}), \tag{7.4}$$

where α is a normalizing constant that forces the sum of the new weights to be equal to one. The term $p(\mathbf{z}_k|\mathbf{x}_k^{(i)})$ is given by

$$p(\mathbf{z}_k|\mathbf{x}_k^{(i)}) = \mathbf{h}(\mathbf{z}_k, \mathbf{x}_k^{(i)}), \tag{7.5}$$

with $\mathbf{h}(\cdot)$ being a model for the observation probability. To implement sound source tracking with particle filters, the system dynamics model and the observation model have to be properly defined.

7.4.2 System dynamics model

The state of a possibly moving sound source can be modeled by the position and velocity of the source. The state vector $\mathbf{x}_k^{(i)}$ is defined by

$$\mathbf{x}_k^{(i)} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T, \tag{7.6}$$

where (x, y, z) are the position and $(\dot{x}, \dot{y}, \dot{z})$ the velocity. Previous sound source tracking research [112, 113] proposes to model source movement as Langevin motion. Its biggest advantage is that it does not make any assumptions about movement patterns and is unbiased towards movement directions. Instead, acceleration in each direction is equiprobable, and is updated with a random term in each iteration. The resampling step of the particle filter implicitly favors particles whose state better represents the observed movement behavior. Langevin motion models source movement as an independent first order process in each direction and the state update is computed by

$$\mathbf{x}_k^{(i)} = \mathbf{g}(\mathbf{x}_{k-1}^{(i)}) = \begin{bmatrix} 1 & 0 & 0 & a\Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 & a\Delta T & 0 \\ 0 & 0 & 1 & 0 & 0 & a\Delta T \\ 0 & 0 & 0 & a & 0 & 0 \\ 0 & 0 & 0 & 0 & a & 0 \\ 0 & 0 & 0 & 0 & 0 & a \end{bmatrix} \cdot \mathbf{x}_{k-1}^{(i)} + \mathbf{n}_k, \tag{7.7}$$

where the noise term $\mathbf{n}_k$ is drawn from a multidimensional Gaussian distribution $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} b^2\Delta T^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & b^2\Delta T^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & b^2\Delta T^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & b^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & b^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & b^2 \end{bmatrix}, \tag{7.8}$$

with μ being the mean vector and Σ the covariance matrix. The model parameters are calculated by

$$\begin{aligned} a &= e^{(-\beta\Delta T)} \\ b &= v\sqrt{1-a^2}, \end{aligned} \tag{7.9}$$

where the steady state velocity v and rate constant β are configuration parameters.

7.4.3 Observation probability model

At each time step COMPaSS estimates the position of each source $\mathbf{z}_k = [x, y, z]^T$. Using the observation probability model, the particle filter calculates the probability of the current observation for each proposed sample i. The probability depends on the Euclidean distance between the observation and the sample. For object tracking the probability is often drawn from a normal distribution using

$$\begin{aligned}\hat{\mathbf{h}}(\mathbf{z}_k, \mathbf{x}_k^{(i)}) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{d(\mathbf{z}_k, \mathbf{x}_k^{(i)})^2}{2\sigma^2}} \\ d(\mathbf{z}_k, \mathbf{x}_k^{(i)}) &= \left\| \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_k^{(i)} - \mathbf{z}_k \right\| .\end{aligned} \tag{7.10}$$

The variance σ of the distribution is a configuration parameter that sets how fast the observation probability decays with increasing distance.

With moving sources COMPaSS is prone to front-back confusions, which has to be taken into account by the observation model. For the tracking of TDOA-based localization results, where a localization result and its front-back confusion are equally probable, the use of bimodal probability densities was proposed in [65, 64]. COMPaSS estimates the correct position with a higher probability than a front-back confusion and the peaks in the bimodal density should reflect this fact. Therefore, the observation probability model is given by

$$\mathbf{h}(\mathbf{z}_k, \mathbf{x}_k^{(i)}) = (1 - p_{fb}) \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{d(\mathbf{z}_k, \mathbf{x}_k^{(i)})^2}{2\sigma^2}} + p_{fb} \frac{1}{\sqrt{2\pi\sigma_{fb}^2}} e^{-\frac{d(\mathbf{z}_{fb,k}, \mathbf{x}_k^{(i)})^2}{2\sigma_{fb}^2}} , \tag{7.11}$$

where p_{fb} is the probability for the occurrence of a front-back confusion, σ_{fb} is the variance of the normal distribution of the second peak and $\mathbf{z}_{fb,k}$ is the front-back confused observation.

7.4.4 Solving the assignment problem

When COMPaSS observes a sound scene it can make an estimation about the number of currently observable sound sources using the similarity scores. The iteratively extracted sources will have a decreasing probability, and a sharp bend in this curve will indicate the number of sources. In a realistic scenario, active sound sources overlap in the TF-domain and an active source will sometimes not be observable due to the presence of more dominant sources. Therefore, COMPaSS is set to extract the first N_s sources regardless of their estimated probabilities. The value of N_s is chosen to be slightly higher than the number of sources that COMPaSS is able to observe at the same time, but its value does not matter that much as unwanted potential sources will be filtered out by the tracking algorithm.

The tracking of multiple sources can be performed either by one particle filter tracking all sources or by multiple filters each tracking one individual source. In the former approach, particles form clusters around the tracked source positions. When sources are not observable for a short time, the tracking has to prevent that its respective particles are attributed to another source and that the clusters converge to the same positions. The latter approach is more suitable for sources that are frequently not observable. Each active sound source is tracked by one individual particle filter and in each time step COMPaSS's observations are assigned to the correct filter. The following solution for the observation assignment problem is based on ideas presented in [108]. Each potential sound source that is detected by COMPaSS can fall in one of three categories:

Tracked source The observation corresponds to a sound source that is currently being tracked by a particle filter. Maximally one observation can be assigned to a filter.

New source The observation belongs to a new sound source that has not been observed previously. A new particle filter will be spawned for this source.

Faulty detection The observation does not correspond to an active sound source and at the same time the similarity score calculated by COMPaSS is too low to confidently identify the observation as a new sound source.

New sources are only added to the tracking if the observation's similarity score is above a threshold. This ensures that the tracking does not create particle filters for erroneous localization results. On the other hand a new source is added to the tracking only when it is dominant at a time instance. However, once a particle filter exists for a sound source, an observation's similarity scores can be substantially lower without affecting the correct assignment of the observation to a tracked source.

Each time new localization results become available the tracking system performs the prediction step for each of the tracked sources. Next, it calculates the probabilities for all possible assignments of each observation. Let COMPaSS's probability of the q-th observation be P_q. The assignment probabilities for a new source or false detection are then calculated by

$$P_{q,n} = P_q \cdot P_{new} \tag{7.12}$$

$$P_{q,f} = (1 - P_q) \cdot P_{false}, \tag{7.13}$$

where P_{new} and P_{false} are configuration parameters. The probability that observation q corresponds to particle filter j is given by

$$P_{q,j} = P_q \cdot \sum_{i=0}^{N_p} \mathbf{w}_{k-1}^{(i)} \mathbf{h}(\mathbf{z}_k, \mathbf{x}_k^{(i)}), \tag{7.14}$$

where $\mathbf{z}_k$ is the q-th localized position, $\mathbf{x}_k^{(i)}$ the predicted samples of filter j and $\mathbf{w}_{k-1}^{(i)}$ the sample weights of filter j from the previous iteration.

The above formulas yield the individual assignment probabilities for each observation. The tracking algorithm has to assign all observations at once and at the same time exclude cases where two or more observations are assigned to the same tracked source. Under the assumption that the probabilities of the observations are statistically independent, the tracking system can calculate the probabilities for all possible assignment combinations. The presented tracking approach differs from the one in [108] significantly in that the most probable combination is picked and the others are excluded from further computations. When each observation is categorized the existing particle filters will belong to one of two groups:

Observed An observation has been assigned to the particle filter. The particle filter can perform its update step normally.

Unobserved The tracked sound source was not observed at the current time instance. This could mean that other sources were more dominant, or that the source has disappeared.

The sample weights of the observed and the unobserved particle filters have to be updated in the next step. If an observation is available for filter j, the observation probability model $\mathbf{h}(\mathbf{z}_k, \mathbf{x}_k^{(i)})$ can be used to adjust the weights according to the new measurement. Regardless of the observation, the prediction step of the filters moves the particles along the last know trajectory of the source. In the absence of a current observation, the certainty of the source position decreases, which corresponds to the particle weights becoming equiprobable. Therefore, the update step of an unobserved source is performed by

$$w_k^{(i)} = \frac{1}{2}(w_{k-1}^{(i)} + \frac{1}{N_p}), \tag{7.15}$$

which pulls the sample weights slowly towards the uniform distribution. If a tracked sound source was not observed for a number of consecutive frames the tracking system can assume that it has become inactive and remove its particle filter.

If necessary, the final step of the tracking algorithm resamples the observed particle filters using the sampling importance resampling method [35] to avoid sample impoverishment.

7.5 Testing the tracking with real data

In this section test results with for the tracking system with a few sound scenes are presented. In Figure 7.3(b) the localization results for a sound scene with one moving source are shown. COMPaSS estimates the two most likely potential sound source positions and the results cluster around the real trajectory of the sound source and its front-back confusion. Figure 7.5 shows a plot of the tracking results for this localization data. The tracking follows the trajectory of the real source accurately and the front-back confusion is correctly not recognized as an individual source.

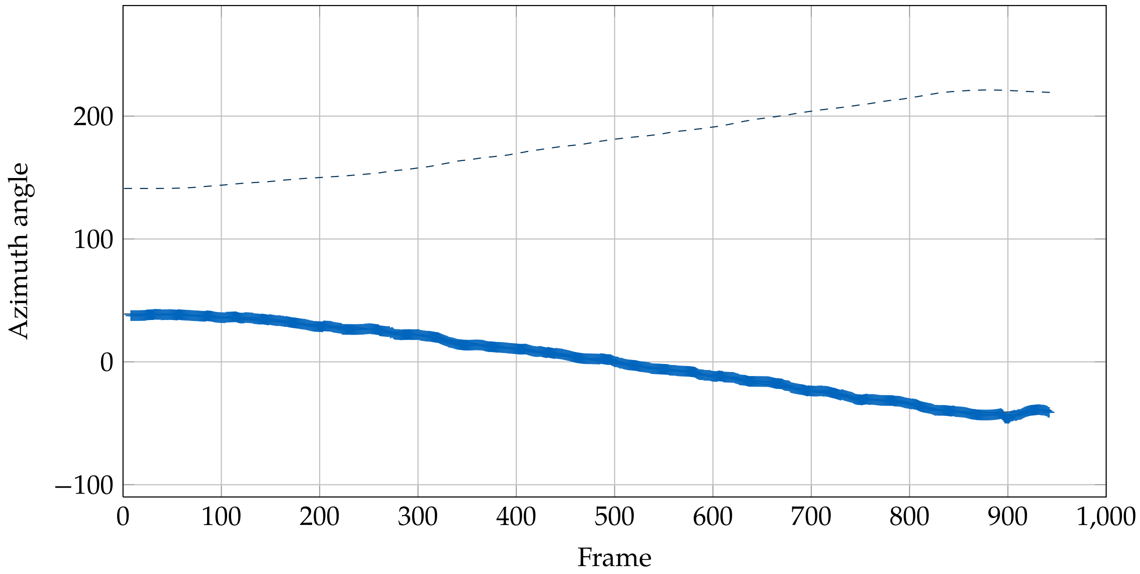
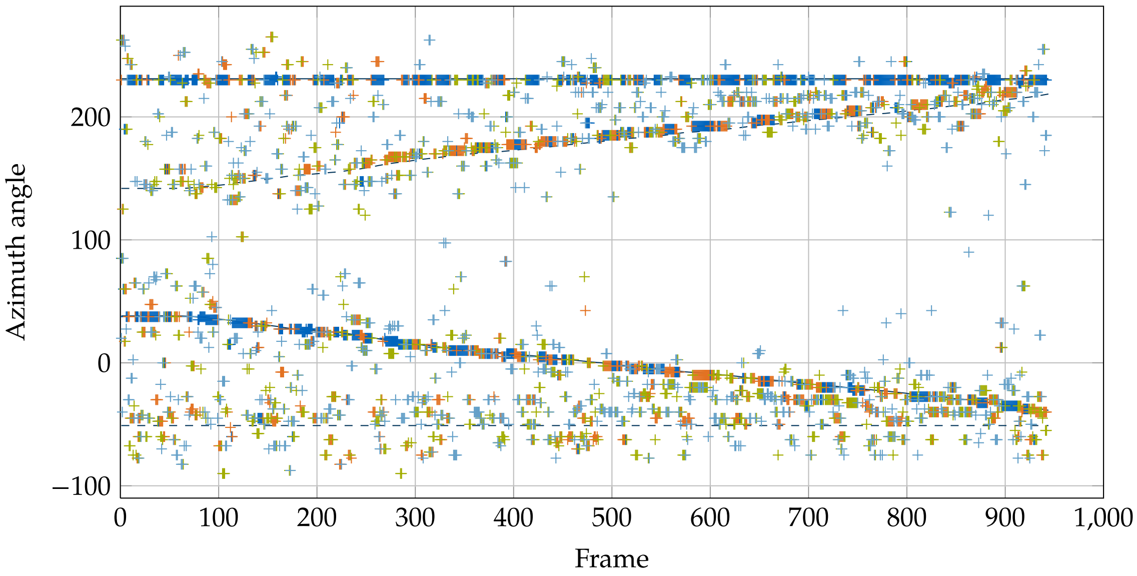


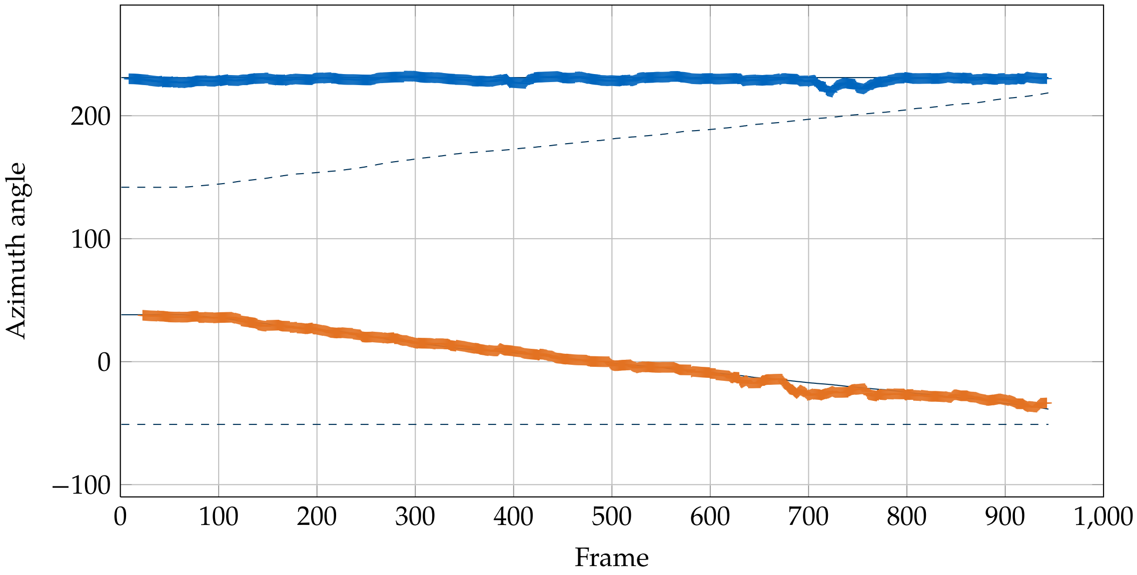
Figure 7.5: Tracking results for a sound scene with one single moving sound source. The tracking follows the trajectory of the real source accurately. The front-back confusions that are present in the localization results are completely removed by the filtering algorithm.

The second sound scene has two active sound sources, one moving and one stationary source. The localization results for the first four potential sources are shown in Figure 7.6(a). The observations cluster mostly on the two actual source trajectories and around the front-back confusion of the moving source. Sometimes the correct positions are not detected until the third or fourth observation, and sometimes they are missed completely. There is no clear clustering around the front-back confusion of the source with the stationary position. The tracking results for this localization data are shown in Figure 7.6(b). The tracking system recognizes both sources correctly and follows their trajectories precisely. The front-back confusion and the noisy additional observations are filtered out. Around frame 700 the tracking of both sources shortly deviates, which is most likely caused by the clustering of potential sound sources between the real trajectories and the front-back confusion of the opposite source.

The third sound scene has two moving sound sources. One source moves approximately twice as fast as the other, and the real trajectories intersect twice with the opposite front-back confusions. The localization results for the first four potential sources are shown in Figure 7.7(a). They are very similar to the previous sound scene, except that now both front-back confusions are observed. The results of the tracking system are shown in Figure 7.7(b). As before, the source trajectories have been correctly identified and only the actual sources were added to the tracking. Between the two cross points the straight trajectory and its front-back confusion are localized more often

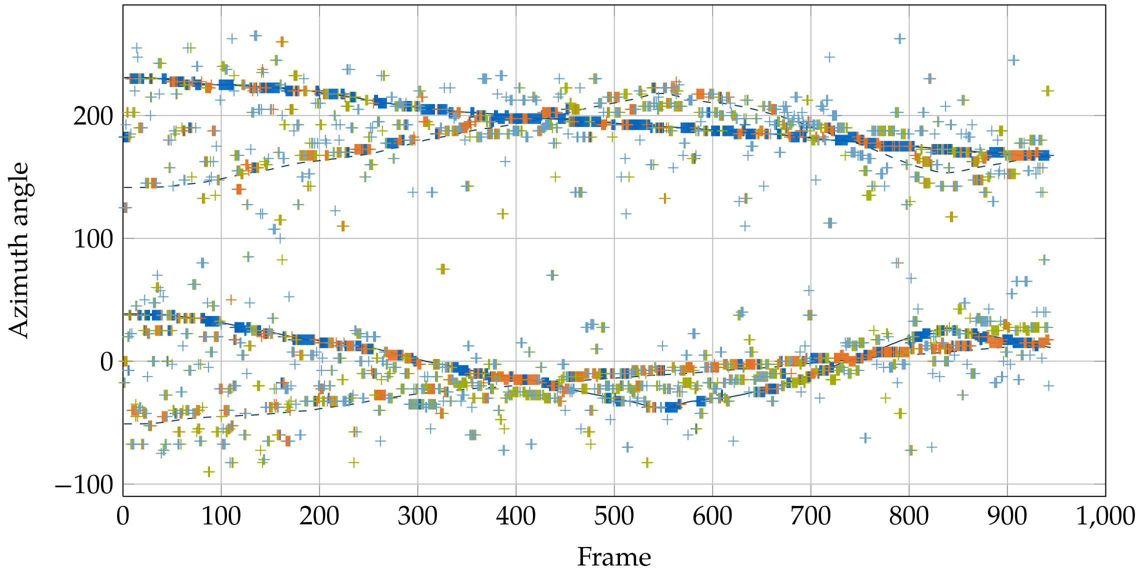


(a) Localization results of COMPaSS (DCT) for the first four potential sources. There is a clear clustering around the real trajectories, but also a lot of noise.

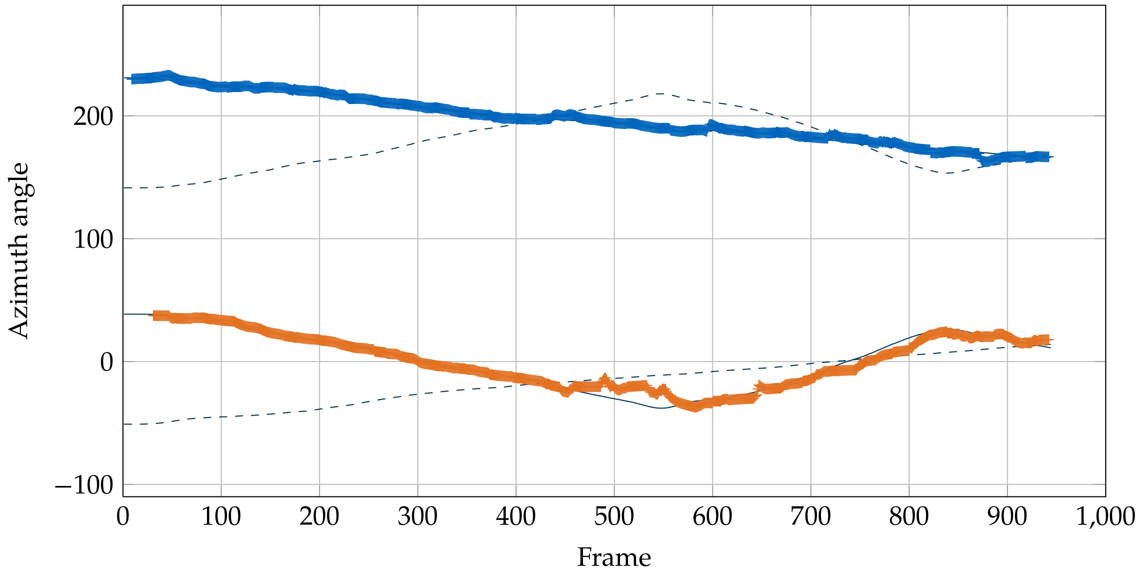


(b) The tracking algorithm recovers both sources with a good accuracy. Only around frame 700 the tracking shortly deviates.

Figure 7.6: Raw COMPaSS localization and tracking results for a sound scene with two sources. One sound source is moving, the other is stationary. The correct source positions are indicated by the solid lines and their front-back confusions by the dashed lines.



(a) Localization results of COMPaSS (DCT) for the first four potential sources. There is a clear clustering around the real trajectories and their front-back confusions.



(b) The tracking algorithm recovers both sources with good accuracy. Even the points where the real trajectories cross with the front-back confusion of the opposite source are recovered correctly.

Figure 7.7: Raw COMPaSS localization and tracking results for a sound scene with two moving sources. The correct source positions are indicated by the solid lines and their front-back confusions by the dashed lines.

than the curved trajectory. Nevertheless, the subsequent tracking correctly follows the true source trajectories, which can be attributed to the bimodal probability densities in the observation model.

7.6 Discussion

This chapter presented a tracking system for COMPaSS. The system is optimized for COMPaSS and its properties, primarily front-back confusions of moving sources. The solution of the assignment problem is mainly inspired by [108], but differs in the selection of the most likely assignment permutation. In contrast to [108], the presented observation model has to account for the possibility that sources are not observed in short time intervals. The correct handling of front-back confusions and non-observable sources is verified with experiments. The results reveal that the tracking is robust towards front-back confusions and does not break down if a source is not observed in a short time interval.

The tracking system has a number of configuration parameters that can be adjusted. Depending on the actual application, different parameter sets can be used to optimize certain properties of the tracking. Additionally, in an actual application high-level knowledge about the sources and their movement behavior is often known a priori and can be incorporated into the system to further enhance the performance.

In summary, the presented tracking system is a generic solution for post-processing COMPaSS's localization results. The tracking system can be further configured and extended for specialized sound source tracking problems.

8 Separation of multiple sound sources

This chapter presents a sound separation algorithm that can be used together with COMPaSS in an auditory system. The algorithm builds completely on existing ideas and only integrates them to work together with the localization and tracking systems.

8.1 Sound source separation algorithm

From the general separation approaches presented in Section 2.5 only two are practically applicable for a binaural robotic auditory system, namely inverse filtering and binaural masking. Inverse filtering performs well in the absence of reverberation. But as the level of reverberation increases the inverse filters are more likely to become unstable. Binary masking approaches on the other hand do not have this problem and usually have a lower computational complexity.

The separation algorithm presented here is based on binary masking. The separation is integrated with COMPaSS and reuses some of its intermediate results. The signal flow diagram of the localization system is shown in Figure 8.1. The separation algorithm gets the filtered source positions from the tracking, and the similarity scores of all transfer functions from COMPaSS. The separation algorithm works in the frequency domain and processes each sound frame individually. The goal of the algorithm is to segregate the information of each active source into an individual

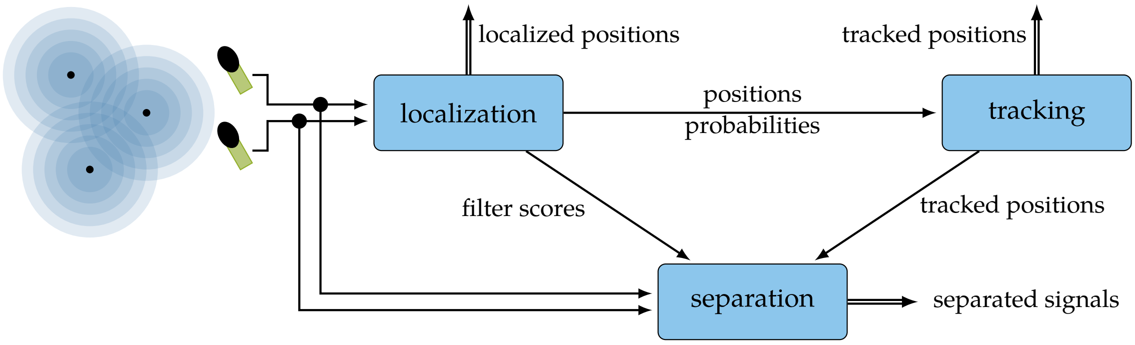


Figure 8.1: Signal flow diagram for the complete auditory system. The separation uses the tracked positions of the sound sources and the filter scores that were calculated by COMPaSS.

sound stream. Therefore, it has to determine which parts of the spectrum are dominated by which sound source. Once this information is available the algorithm can create separate sound streams from each of the observations. The binaural robot will therefore have two separated versions of each sound source. The ipsilateral ear will usually record a better observation of a sound source than the opposite ear, and the robot should use this version for further processing.

Let N denote the number of active sound sources found by the tracking, and let p_n be the index of the TF in the database that is closest to the filtered position of the source n. From the similarity scores $\boldsymbol{C_t}(f, \nu, \eta)$ of the current sound frame ν the algorithm creates the matrix $\boldsymbol{V_\nu} \in \mathbb{R}^{F \times N}$ by

$$\boldsymbol{V_\nu}(f, n) = \boldsymbol{C_t}(f, \nu_{sep}, p_n), \tag{8.1}$$

where F is the number of frequency bins created by the frequency transform. The columns of $\boldsymbol{V_\nu}$ correspond to each active sound source and its entries indicate the similarity at each frequency bin. The higher the value of an entry of $\boldsymbol{V_\nu}$ the higher the probability that the corresponding source is dominant at the corresponding frequency. Exploiting this fact, the algorithm calculates the entries of the binary masking matrix $\boldsymbol{M_\nu} \in \mathbb{R}^{F \times N}$ by

$$\boldsymbol{M_\nu}(f, n) = \begin{cases} 1 & \text{if } n == \arg\max_{n_i} \boldsymbol{V_\nu}(f, n_i) \\ 0 & \text{otherwise.} \end{cases} \tag{8.2}$$

Each row of $\boldsymbol{M_\nu}$ has exactly one non-zero entry that indicates which source is dominant at the frequency. In the last step the observations $\boldsymbol{X_{j,\nu}} \in \mathbb{R}^F$ are segregated into the source spectra $\boldsymbol{S_{j,\nu}} \in \mathbb{R}^{F \times N}$ by calculating

$$\hat{\boldsymbol{S}}_{\boldsymbol{j,\nu}} = \left(\boldsymbol{X_{j,\nu}} \cdot \boldsymbol{u_N^T}\right) \circ \boldsymbol{M_\nu}, \tag{8.3}$$

where $\boldsymbol{u_N^T}$ is an all-ones row vector with N entries and $(\circ)$ denotes the Hadamard product. The columns of $\hat{\boldsymbol{S}}_{\boldsymbol{j,\nu}}$ are the spectra of the separated sound sources.

8.2 Transform domain considerations

Sound signal separation by masking can be performed in any invertible transform domain where the signal admits a sparse representation [94]. COMPaSS calculates the similarity data $\boldsymbol{C_t}$ either in the Fourier or DCT domain and the separation can use this data directly to perform the separation in the same domain. Not only the transform domain but also the exact transform parameters, like for example the window length of the transform, have to be taken from the localization.

Signal separation can be performed in another transform domain or with different transformation parameters than the localization. In this case $\boldsymbol{C}_t$ cannot be shared between localization and separation anymore. The performance penalty is moderate, as the separation does not require the complete similarity data, but only the entries that belong to active source positions. The separation can calculate them using its own transform parameters. If the separation uses different frame lengths than the localization, it also has to adjust the estimated source positions, as the frames from both algorithms are not time-aligned. The tracking algorithm models the positions and velocities of all sound sources and with this data the separation algorithm can generate arbitrary intermediate positions and even predict positions in the near future if necessary.

8.3 Evaluation

This section compares the separation performance of the presented approach to state-of-the-art techniques.

8.3.1 Evaluation criteria

In literature, different methods have been used to objectively measure the quality of a signal separation. The most meaningful measure for blind source separation of mixed audio signals comes from the BSS_EVAL toolbox [114], which is also used by the Signal Separation Evaluation Campaign (SiSEC) [5].

BSS_EVAL requires the original source signals $s_i(t)$ and calculates three different signal ratios for each demixed signal. In its first step, it decomposes an estimated source signal $\hat{s}_k(t)$ into three parts. To this end, it projects an estimated signal onto different orthogonal subspaces spanned by the original source signals. The projection onto the space spanned by the desired source signal $s_{i=k}(t)$ yields $s_{target}(t)$, which is the part of $\hat{s}_k(t)$ that can be explained by a possibly convolved version of the original source. Analogously, the projection onto the space of the remaining signals $s_{i\neq k}(t)$ yields $e_{interf}(t)$, which represents the error in $\hat{s}_k(t)$ that is caused by possibly convolved versions of the interfering source signals. The signal parts that cannot be explained by the sources form the third signal $e_{artif}(t)$ and are called the artifact error. From the decomposed signals BSS_EVAL calculates the three signal ratios of interest using

$$SDR = 10\log_{10} \frac{||s_{target}||^2}{||e_{interf}||^2 + ||e_{artif}||^2}, \tag{8.4}$$

$$SIR = 10\log_{10} \frac{||s_{target}||^2}{||e_{interf}||^2}, \text{ and} \tag{8.5}$$

$$SAR = 10\log_{10} \frac{||s_{target}||^2 + ||e_{interf}||^2}{||e_{artif}||^2}. \tag{8.6}$$

These ratios are called signal-to-distortion ratio (SDR), signal-to-interference ratio (SIR), and signal-to-artifact ratio (SAR). The most important of the three ratios is the SDR as it measures the energy ratio of the desired signal to all possible errors. In [62] the authors found that SDR values are correlated to the performance of automatic speech recognition (ASR) on the separated signals. This makes SDR a good performance measure for sound separation in robotic auditory systems.

8.3.2 The 0-dB mask

The SDR, SIR, and SAR values are good measures to compare the separation results from different algorithms. The SDR values of an algorithm highly depend on the testing conditions, therefore, comparability between different experiments is not given. No clear conclusions about the absolute quality of a signal separation can be drawn from the signal ratios alone, and some point of reference to compare the results against is necessary.

Since the assumption of W-disjont orthogonal source signals does only approximately hold true in reality, there is an upper bound for the separation performance that can be achieved with binary masking. This upper bound can serve as a reference point and it can be calculated from a separation result that was created by an algorithm that is optimal in some sense. The ideal binary mask in terms of signal-to-interference ratio is the 0-dB mask [120]. The entries of this mask are 1 at points where the target signal is at least as loud as the sum of all interfering sources and 0 otherwise. The 0-dB mask is calculated from ground truth data that is usually not available to separation algorithms.

The results of the 0-dB mask are only an upper bound for binary masking approaches and source separation with a blind approach, beamforming, or filter inversion can theoretically achieve even better results. The usefulness of the 0-dB mask as a reference algorithm is not affected by this fact.

Algorithm	Simulation			Office environment		
	SDR	**SIR**	**SAR**	**SDR**	**SIR**	**SAR**
ccoh	10.14	19.05	12.05	3.90	11.03	6.51
cdct	11.35	19.51	13.28	4.53	9.17	8.07
csscl	13.53	25.67	13.86	-12.03	10.87	-11.38
fdbm	9.33	22.20	11.88	2.13	10.13	7.19
srp-phat	4.82	5.83	19.09	-14.14	10.13	-12.62
srp-phat-pf	8.71	10.37	19.11	-11.76	13.30	-11.53
0-dB mask	15.38	25.07	16.49	12.02	21.33	15.95

Table 8.1: Performance comparison for the separation quality of two active sound sources. All values are given in decibel.

8.3.3 Separation performance

The separation performance is evaluated under the same real-world conditions as the localization in Section 6.4. The test data includes real-world recordings of sound scenes with two or three speech signals that were recorded in a reverberant office environment and simulations of the same sound scenes. The performance of COMPaSS is measured for masking in the Fourier domain (ccoh) and masking in DCT domain (cdct). FDBM also performs binary masking, CSSCL uses filter inversion and both SRP-PHAT approaches use geometric source separation, which is based on a steered beamformer.

Table 8.1 shows the separation performance for sound scenes with two sources. With real-world recordings, the two COMPaSS based approaches achieve the highest SDR values. Their distance to the SDR value obtained by the 0-dB mask is approximately 4 dB in the simulations and approximately 7.5 dB in the real environment. The theoretical advantage of the filter inversion approach becomes evident with the SDR value of CSSCL in the simulation, which is about 2.2 dB higher than the SDR of COMPaSS. In the real environment however, the SDR of CSSCL drops significantly and the low SAR value indicates that the separation introduces substantial artifacts. The filter inversion is more likely to become unstable in the presence of reverberation, which causes audible artifacts in the separated signals. The separation performance of the beamformer is comparatively poor and suffers from distortions under real conditions.

The separation performance for three sound sources is given in Table 8.2. As expected, the measured values are lower but overall consistent with the two-source case. Most notably, COMPaSS's distance to the SDR of the 0-dB mask has not significantly changed. Under real conditions, COMPaSS based separation achieves better SDR values than any of the other algorithms.

In [63] a comparison of several state-of-the-art blind source separation algorithms was conducted. The authors reported the SDR values of the evaluated methods along with the SDRs of the 0-dB

Algorithm	Simulation			Office environment		
	SDR	SIR	SAR	SDR	SIR	SAR
ccoh	6.52	13.72	8.26	2.09	6.36	4.27
cdct	7.87	15.72	10.01	2.54	5.24	5.74
csscl	8.75	15.04	12.40	-10.78	9.68	-9.90
fdbm	7.15	17.33	9.81	0.97	5.88	6.95
srp-phat	2.23	4.86	15.24	-13.90	5.50	-11.45
srp-phat-pf	3.42	4.73	17.34	-12.85	9.02	-11.25
0-dB mask	12.65	21.03	13.93	9.72	18.29	11.23

Table 8.2: Performance comparison for the separation quality of three active sound sources. All values are given in decibel.

mask. A comparison of their results with the measurements from this evaluation indicates that the separation performance of COMPaSS is competitive with state-of-the-art blind source separation algorithms.

8.4 Discussion

The presented separation algorithm integrates ideas from existing approaches into the COMPaSS framework. The actual signal separation is performed by binary masking based on the similarity scores of the estimated source positions. The spectrum of each observation is individually segregated into the source spectra, thus, creating two estimations for each separated sound source. The quality of the estimated signal from the ipsilateral sensor is usually better and the auditory system can discard the estimation from the contralateral sensor.

The separation algorithm is designed to introduce only little additional computational complexity to the auditory system and for most of its calculations it can reuse intermediate results from the localization. The performance of the algorithm was evaluated in simulations and in real-world experiments. The quality of the proposed signal separation compares well to other robotic auditory systems. Its performance is even comparable to state-of-the-art blind source separation approaches that do not have low computational complexity or low latency requirements.

9 Conclusion

In this work, I presented the design of three low-level modules for an auditory system on cognitive robots. On robots, auditory processing algorithms have to meet a number of requirements and constraints in order to be applicable in the real world. Existing techniques are often specialized to solve one particular problem and they fulfill the other requirements and constraints only partly.

Out of the need for more generally applicable solutions, I designed and implemented the following modules in the course of this work:

- A module that can determine the position of a sound source in the environment. My localization algorithm COMPaSS follows a binaural approach and uses the observers transfer functions to estimate the position of a source. In the presence of multiple active sources, COMPaSS exploits the sparseness of sound signals in different transform domains for localization.
- My sound source tracking module is specifically designed for post processing the output of COMPaSS. My tracking system performs a Sequential Monte Carlo simulation to filter and stabilize the localization results. The tracking models the source motion with a Langevin process and accounts for the typical error behavior of COMPaSS in the probability calculation. The tracking system is designed to correctly map current observations to already tracked sources.
- The sound source separation module is build on top of COMPaSS and reuses its internal results to keep the computational complexity low. With knowledge about the estimated positions of the sound sources the algorithm creates binary masks to segregate the spectra of the sound sources.

For the evaluation of COMPaSS and its comparison to existing techniques, I recorded an extensive test set with over 24 hours of sound data. The evaluation revealed that COMPaSS can localize a sound source with good accuracy even in the presence of interfering sources, noise, and reverberation. On average, COMPaSS achieves more accurate localization results compared to existing localization systems, while at the same time operating inside the requirements and constraints of robotic systems.

The tracking module is a framework that can be optimized by parameter-tuning for a specific task. I verified the functional capability of the tracking with adequate tests, but did not conduct an in-depth evaluation of the tracking performance with a specialized parameter set, as these results would give only little objective insight into the general properties of the tracking system.

The evaluation of the separation module uses a large data set of real-world recordings and simulations. The experiments clearly show that the separation quality of the proposed approach is better than the compared state-of-the-art separation systems for robots. The measured separation performance is even comparable to state-of-the-art blind source separation systems that are not subject to same restrictions.

In regards to sound source localization, tracking, and separation systems for robots, my results show that more generally applicable designs of these systems are feasible and that an improved accuracy is possible at the same time.

Bibliography

1. P. Aarabi and S. Zaky. Robust sound localization using multi-source audiovisual information fusion. In *Information Fusion*, 2(3), pp. 209--223, 2001. doi:10.1016/S1566-2535(01)00035-5.

2. M. Aharon, M. Elad, and A. Bruckstein. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. In *IEEE Transactions on Signal Processing*, 54(11), pp. 4311--4322, 2006. doi:10.1109/TSP.2006.881199.

3. S.B. Andersson, A.A. Handzel, V. Shah, and P.S. Krishnaprasad. Robot phonotaxis with dynamic sound-source localization. In *IEEE International Conference on Robotics and Automation*, pp. 4833--4838. 2004. doi:10.1109/ROBOT.2004.1302483.

4. S. Araki, S. Makino, H. Sawada, and R. Mukai. Reducing musical noise by a fine-shift overlap-add method applied to source separation using a time-frequency mask. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 81--84. 2005. doi: 10.1109/ICASSP.2005.1415651.

5. S. Araki, A. Ozerov, V. Gowreesunker, H. Sawada, F. Theis, G. Nolte, D. Lutter, and N.Q.K. Duong. The 2010 signal separation evaluation campaign (SiSEC2010): audio source separation. In *Latent Variable Analysis and Signal Separation*, 6353, pp. 114--122, 2010. doi: 10.1007/978-3-642-15995-4_15.

6. S. Arberet, R. Gribonval, and F. Bimbot. A robust method to count and locate audio sources in a stereophonic linear instantaneous mixture. In *Independent Component Analysis and Blind Signal Separation*, pp. 536--543. Springer, 2006. doi:10.1007/11679363_67.

7. M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. In *IEEE Transactions on Signal Processing*, 50(2), pp. 174--188, 2002. doi:10.1109/78.978374.

8. F. Asano, H. Asoh, and T. Matsui. Sound source localization and signal separation for office robot "JiJo-2". In *IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 243--248. 1999. doi:10.1109/MFI.1999.815997.

9. F. Asano, M. Goto, K. Itou, and H. Asoh. Real-time sound source localization and separation system and its application to automatic speech recognition. In *Eurospeech*, pp. 1013--1016. 2001.

10. F. Asano, Y. Motomura, and S. Nakamura. Fusion of Audio and Video Information for Detecting Speech Events. In *International Conference on Information Fusion*, pp. 386--393. 2003. doi:10.1109/ICIF.2003.177472.

11. F. Asano, K. Yamamoto, I. Hara, J. Ogata, T. Yoshimura, Y. Motomura, N. Ichimura, and H. Asoh. Detection and separation of speech event using audio and video information fusion and its application to robust speech Interface. In *EURASIP Journal on Advances in Signal Processing*, 2004(11), pp. 1727--1738, 2004. doi:10.1155/S1110865704402303.

12. H. Asoh, F. Asano, T. Yoshimura, K. Yamamoto, Y. Motomura, N. Ichimura, I. Hara, and J. Ogata. An application of a particle filter to bayesian multiple sound source tracking with audio and video information fusion. In *International Conference on Information Fusion*, pp. 805--812. 2004.

13. H. Asoh, I. Hara, F. Asano, and K. Yamamoto. Tracking human speech events using a particle filter. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1153--1156. 2005. doi:10.1109/ICASSP.2005.1415614.

14. E. Berglund and J. Sitte. The parameterless self-organizing map algorithm. In *IEEE Transactions on Neural Networks*, 17(2), pp. 305--16, 2006. doi:10.1109/TNN.2006.871720.

15. E. Berglund, J. Sitte, and G. Wyeth. Active audition using the parameter-less self-organising map. In *Autonomous Robots*, 24(4), pp. 401--417, 2008. doi:10.1007/s10514-008-9084-9.

16. J. Blauert. *Spatial hearing: the psychophysics of human sound localization*. MIT Press, 1997. ISBN 0262024136.

17. M.S. Brandstein and H.F. Silverman. A practical methodology for speech source localization with microphone arrays. In *Computer Speech & Language*, 11(2), pp. 91--126, 1997. doi:10.1006/csla.1996.0024.

18. M.S. Brandstein and H.F. Silverman. A robust method for speech signal time-delay estimation in reverberant rooms. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 375--378. 1997. doi:10.1109/ICASSP.1997.599651.

19. R.A. Brooks, C. Breazeal, M. Marjanovi, B. Scassellati, and M.M. Williamson. The Cog project: building a humanoid robot. In *Computation for Metaphors, Analogy and Agents*, 1562, pp. 52--87, 1999. doi:10.1007/3-540-48834-0_5.

20. R.A. Brooks. *Cambrian Intelligence: The Early History of the New AI*. The MIT Press, 1999. ISBN 9780262522632.

21. M.D. Burkhard and R.M. Sachs. Anthropometric manikin for acoustic research. In *Manikin Measurements*, pp. 8--16. 1978.

22. M. Buss, A. Peer, T. Schauss, N. Stefanov, U. Unterhinninghofen, S. Behrendt, J. Leupold, M. Durkovic, and M. Sarkis. Development of a multi-modal multi-user telepresence and teleaction system. In *The International Journal of Robotics Research*, 29(10), pp. 1298--1316, 2009. doi:10.1177/0278364909351756.

23. N. Checka, K.W. Wilson, M.R. Siracusa, and T. Darrell. Multiple person and speaker activity tracking with a particle filter. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 881--884. 2004. doi:10.1109/ICASSP.2004.1327252.

24. Y. Chisaki, S. Kawano, K. Nagata, K. Matsuo, H. Nakashima, and T. Usagawa. Azimuthal and elevation localization of two sound sources using interaural phase and level differences. In *Acoustical Science and Technology*, 29(2), pp. 139--148, 2008. doi:10.1250/ast.29.139.

25. Y. Chisaki, K. Matsuo, K. Hagiwara, H. Nakashima, and T. Usagawa. Real-time processing using the frequency domain binaural model. In *Applied Acoustics*, 68(8), pp. 923--938, 2007. doi:10.1016/j.apacoust.2006.12.004.

26. Y. Chisaki, T. Nakanishi, H. Nakashima, and T. Usagawa. Concurrent speech signal separation based on frequency domain binaural model. In *International Workshop on Acoustic Echo and Noise Control*, pp. 255--258. 2003.

27. C. Choi, D. Kong, J. Kim, and S. Bang. Speech enhancement and recognition using circular microphone array for service robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3516--3521. 2003. doi:10.1109/IROS.2003.1249700.

28. I. Cohen and B. Berdugo. Speech enhancement for non-stationary noise environments. In *Signal Processing*, 81(11), pp. 2403--2418, 2001. doi:10.1016/S0165-1684(01)00128-1.

29. M. Cooke, G. Brown, and M. Crawford. Computational auditory scene analysis: Listening to several things at once. In *Endeavour*, 17(4), pp. 186--190, 1993.

30. M. Durkovic, T. Habigt, M. Rothbucher, and K. Diepold. Low latency localization of multiple sound sources in reverberant environments. In *The Journal of the Acoustical Society of America*, 130(6), pp. EL392--EL398, 2011. doi:10.1121/1.3659146.

31. M. Durkovic, F. Sagstetter, and K. Diepold. HRTF measurements with recorded reference signal. In *Audio Engineering Society Convention 129*, pp. 1--8. 2010.

32. Y. Ephraim and D. Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(6), pp. 1109--1121, 1984. doi:10.1109/TASSP.1984.1164453.

33. C. Févotte, R. Gribonval, and E. Vincent. *BSS_EVAL toolbox user guide*. Technical Report, IRISA, 2005.

34. W. Gardner and K. Martin. *HRTF Measurements of a KEMAR Dummy-Head Microphone*. Technical Report, MIT Media Lab Perceptual Computing, 1994.

35. N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE Proceedings F Radar and Signal Processing*, 140(2), p. 107, 1993. doi:10.1049/ip-f-2.1993.0015.

36. E. Grassi and S. Shamma. A biological inspired, learning, sound localization algorithm. In *Conference on Information Sciences and Systems*, pp. 344--348. 2001.

37. T. Habigt, M. Durkovic, M. Rothbucher, and K. Diepold. Enhancing 3D audio using blind bandwidth extension. In *Audio Engineering Society Convention 129*, pp. 1--5. 2010.

38. A.A. Handzel and P.S. Krishnaprasad. Biomimetic sound-source localization. In *IEEE Sensors Journal*, 2(6), pp. 607--616, 2002. doi:10.1109/JSEN.2002.807772.

39. I. Hara, F. Asano, H. Asoh, J. Ogata, N. Ichimura, Y. Kawai, F. Kanehiro, H. Hirukawa, and K. Yamamoto. Robust speech interface based on audio and video information fusion for humanoid HRP-2. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2404--2410. 2004. doi:10.1109/IROS.2004.1389768.

40. S. Hashimoto, S. Narita, H. Kasahara, A. Takanishi, S. Sugano, K. Shirai, T. Kobayashi, H. Takanobu, T. Kurata, K. Fujiwara, T. Matsuno, T. Kawasaki, and K. Hoashi. Humanoid robot-development of an information assistant robot Hadaly. In *IEEE International Workshop on Robot and Human Communication*, pp. 106--111. 1997. doi:10.1109/ROMAN.1997.646961.

41. A.D. Horchler, R.E. Reeve, B. Webb, and R.D. Quinn. Robot phonotaxis in the wild: a biologically inspired approach to outdoor sound localization. In *Advanced Robotics*, 18(8), pp. 801--816, 2004. doi:10.1163/1568553041738095.

42. T. Hromádka, M.R. Deweese, and A.M. Zador. Sparse representation of sounds in the unanesthetized auditory cortex. In *PLoS Biology*, 6(1), pp. 124--137, 2008. doi:10.1371/journal.pbio.0060016.

43. J. Huang, N. Ohnishi, X. Guo, and N. Sugie. Echo avoidance in a computational model of the precedence effect. In *Speech Communication*, 27(3-4), pp. 223--233, 1999. doi:10.1016/S0167-6393(98)00075-2.

44. J. Huang, N. Ohnishi, and N. Sugie. Sound localization in reverberant environment based on the model of the precedence effect. In *IEEE Transactions on Instrumentation and Measurement*, 46(4), pp. 842--846, 1997. doi:10.1109/19.650785.

45. J. Huang, T. Supaongprapa, L. Terakura, F. Wang, N. Ohnishi, and N. Sugie. A model-based sound localization system and its application to robot navigation. In *Robotics and Autonomous Systems*, 27(4), pp. 199--209, 1999. doi:10.1016/S0921-8890(99)00002-0.

46. R.E. Irie. Robust sound localization: an application of an auditory perception system for a humanoid robot. Master Thesis, 1995.

47. M. Isard and A. Blake. CONDENSATION - conditional density propagation for visual tracking. In *International Journal of Computer Vision*, 29(1), pp. 5--28, 1998.

48. ISO 3382-2:2008. *Acoustics -- Measurement of room acoustic parameters -- Part 2: Reverberation time in ordinary rooms*. ISO, Geneva, Switzerland, 2008.

49. A. Jourjine, S. Rickard, and O. Yilmaz. Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2985--2988. 2000. doi:10.1109/ICASSP.2000.861162.

50. R.E. Kalman. A new approach to linear filtering and prediction problems. In *Transactions of the ASME--Journal of Basic Engineering*, 82(Series D), pp. 35--45, 1960.

51. K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi. Humanoid robot HRP-2. In *IEEE International Conference on Robotics and Automation*, pp. 1083--1090. 2004. doi:10.1109/ROBOT.2004.1307969.

52. F. Keyrouz, W. Maier, and K. Diepold. A novel humanoid binaural 3D sound localization and separation algorithm. In *IEEE International Conference on Humanoid Robots*, pp. 296--301. 2006. doi:10.1109/ICHR.2006.321400.

53. F. Keyrouz and K. Diepold. Binaural source localization and spatial audio reproduction for telepresence applications. In *Presence: Teleoperators and Virtual Environments*, 16(5), pp. 509--522, 2007. doi:10.1162/pres.16.5.509.

54. F. Keyrouz, W. Maier, and K. Diepold. Robotic binaural localization and separation of more than two concurrent sound sources. In *International Symposium on Signal Processing and Its Applications*, pp. 1--4. 2007. doi:10.1109/ISSPA.2007.4555468.

55. F. Keyrouz, W. Maier, and K. Diepold. Robotic localization and separation of concurrent sound sources using self-splitting competitive learning. In *IEEE Symposium on Computational Intelligence in Image and Signal Processing*, pp. 340--345. 2007. doi:10.1109/CIISP.2007.369192.

56. F. Keyrouz, Y. Naous, and K. Diepold. A new method for binaural 3-D localization based on HRTFs. In *IEEE International Conference on Acoustics Speed and Signal Processing Proceedings*, pp. 341--344. 2006. doi:10.1109/ICASSP.2006.1661282.

57. H. Kitano, H.G. Okuno, K. Nakadai, T. Sabisch, and T. Matsui. Design and architecture of SIG the humanoid: an experimental platform for integrated perception in RoboCup humanoid challenge. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 181--190. 2000. doi:10.1109/IROS.2000.894602.

58. U. Klee, T. Gehrig, and J. McDonough. Kalman filters for time delay of arrival-based source localization. In *EURASIP Journal on Advances in Signal Processing*, 2006(1), pp. 1--16, 2006. doi:10.1155/ASP/2006/12378.

59. C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(4), pp. 320--327, 1976. doi:10.1109/TASSP.1976.1162830.

60. E.A. Lehmann, D.B. Ward, and R.C. Williamson. Experimental comparison of particle filtering algorithms for acoustic source localization in a reverberant room. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 147--150. 2003. doi:10.1109/ASPAA.2003.1285841.

61. D. Li and S.E. Levinson. A linear phase unwrapping method for binaural sound source localization on a robot. In *IEEE International Conference on Robotics and Automation*, pp. 19--23. 2002. doi:10.1109/ROBOT.2002.1013333.

62. M.I. Mandel, S. Bressler, B. Shinn-Cunningham, and D.P.W. Ellis. Evaluating source separation algorithms with reverberant speech. In *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7), pp. 1872--1883, 2010. doi:10.1109/TASL.2010.2052252.

63. M.I. Mandel, R.J. Weiss, and D.P.W. Ellis. Model-based expectation-maximization source separation and localization. In *IEEE Transactions on Audio, Speech, and Language Processing*, 18(2), pp. 382--394, 2010. doi:10.1109/TASL.2009.2029711.

64. I. Marković and I. Petrović. Applying von mises distribution to microphone array probabilistic sensor modelling. In *International Symposium on Robotics*, pp. 21--27. 2010.

65. I. Marković and I. Petrović. Speaker localization and tracking with a microphone array on a mobile robot using von Mises distribution and particle filtering. In *Robotics and Autonomous Systems*, 58(11), pp. 1185--1196, 2010. doi:10.1016/j.robot.2010.08.001.

66. E. Martinson and A. Schultz. Auditory evidence grids. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1139--1144. 2006. doi:10.1109/IROS.2006.281843.

67. T. Matsui, H. Asoh, J. Fry, Y. Motomura, F. Asano, T. Kurita, I. Hara, and N. Otsu. Integrated natural spoken dialogue system of Jijo-2 mobile robot for office services. In *National Conference on Artificial Intelligence and Eleventh Conference on Innovative Applications of Artificial Intelligence*, pp. 621--627. 1999.

68. Y. Matsusaka, S. Fujie, and T. Kobayashi. Modeling of conversational strategy for the robot participating in the group conversation. In *Eurospeech*, pp. 2173--2176. 2001.

69. Y. Matsusaka, T. Tojo, and S. Kubota. Multi-person conversation via multi-modal interface - a robot who communicate with multi-user. In *Eurospeech*, pp. 1723--1726. 1999.

70. J.C. Middlebrooks and D.M. Green. Sound localization by human listeners. In *Annual review of psychology*, 42, pp. 135--59, 1991. doi:10.1146/annurev.ps.42.020191.001031.

71. J. Mouba and S. Marchand. A source localization/separation/respatialization system based on unsupervised classification of interaural cues. In *International Conference on Digital Audio Effects*, pp. 233--238. 2006.

72. M. Murase, S. Yamamoto, J.M. Valin, K. Nakadai, K. Yamada, K. Komatani, T. Ogata, and H.G. Okuno. Multiple moving speaker tracking by microphone array on mobile robot. In *European Conference on Speech Communication and Technology*, pp. 249--252. 2005. doi:10.1.1.78.9438.

73. J.C. Murray and H.R. Erwin. A neural network classifier for notch filter classification of sound-source elevation in a mobile robot. In *International Joint Conference on Neural Networks*, pp. 763--769. 2011. doi:10.1109/IJCNN.2011.6033298.

74. J.C. Murray, H.R. Erwin, and S. Wermter. Robotic sound-source localisation architecture using cross-correlation and recurrent neural networks. In *Neural Networks*, 22(2), pp. 173--89, 2009. doi:10.1016/j.neunet.2009.01.013.

75. K. Nakadai, K.i. Hidai, H. Mizoguchi, H.G. Okuno, and H. Kitano. Real-time auditory and visual multiple-object tracking for humanoids. In *International Joint Conference on Artificial Intelligence*, pp. 1425--1436. 2001.

76. K. Nakadai, K.i. Hidai, H.G. Okuno, and H. Kitano. Real-time multiple speaker tracking by multi-modal integration for mobile robots. In *Eurospeech*, pp. 1193--1196. 2001.

77. K. Nakadai, K.i. Hidai, H.G. Okuno, and H. Kitano. Real-time speaker localization and speech separation by audio-visual integration. In *IEEE International Conference on Robotics and Automation*, pp. 1043--1049. 2002. doi:10.1109/ROBOT.2002.1013493.

78. K. Nakadai, T. Lourens, H.G. Okuno, and H. Kitano. Active audition for humanoid. In *National Conference on Artificial Intelligence*, pp. 832--839. 2000.

79. K. Nakadai, T. Matsui, H.G. Okuno, and H. Kitano. Active audition system and humanoid exterior design. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1453--1461. 2000. doi:10.1109/IROS.2000.893225.

80. K. Nakadai, D. Matsuura, H.G. Okuno, and H. Kitano. Applying scattering theory to robot audition system: robust sound source localization and extraction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1147--1152. 2003. doi:10.1109/IROS.2003.1248800.

81. K. Nakadai, H. Nakajima, M. Murase, S. Kaijiri, K. Yamada, T. Nakamura, Y. Hasegawa, H.G. Okuno, and H. Tsujino. Robust tracking of multiple sound sources by spatial integration of room and robot microphone arrays. In *International Conference on Acoustics Speed and Signal Processing Proceedings*, pp. 929--932. 2006. doi:10.1109/ICASSP.2006.1661122.

82. K. Nakadai, H. Nakajima, M. Murase, H.G. Okuno, Y. Hasegawa, and H. Tsujino. Real-time tracking of multiple sound sources by integration of in-room and robot-embedded microphone arrays. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 852--859. 2006. doi:10.1109/IROS.2006.281737.

83. K. Nakadai, H. Nakajima, K. Yamada, Y. Hasegawa, T. Nakamura, and H. Tsujino. Sound source tracking with directivity pattern estimation using a 64 ch microphone array. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1690--1696. 2005. doi:10.1109/IROS.2005.1544981.

84. K. Nakadai, H.G. Okuno, and H. Kitano. Epipolar geometry based sound localization and extraction for humanoid audition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1395--1401. 2001. doi:10.1109/IROS.2001.977176.

85. K. Nakadai, H.G. Okuno, and H. Kitano. Exploiting auditory fovea in humanoid-human interaction. In *National Conference on Artificial Intelligence*, pp. 431--438. 2002.

86. K. Nakadai, H.G. Okuno, and H. Kitano. Real-time sound source localization and separation for robot audition. In *IEEE International Conference on Spoken Language Processing*, pp. 193--196. 2002.

87. K. Nakadai, H.G. Okuno, and H. Kitano. Robot recognizes three simultaneous speech by active audition. In *IEEE International Conference on Robotics and Automation*, pp. 398--405. 2003. doi:10.1109/ROBOT.2003.1241628.

88. H. Nakashima, Y. Chisaki, T. Usagawa, and M. Ebata. Frequency domain binaural model based on interaural phase and level differences. In *Acoustical Science and Technology*, 24(4), pp. 172--178, 2003. doi:10.1250/ast.24.172.

89. T. Nishiura, M. Nakamura, A. Lee, H. Saruwatari, and K. Shikano. Talker tracking display on autonomous mobile robot with a moving microphone array. In *International Conference on Auditory Display*, pp. 1--4. 2002.

90. T. Nishiura, T. Yamada, S. Nakamura, and K. Shikano. Localization of multiple sound sources based on a CSP analysis with a microphone array. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1053--1056. 2000. doi:10.1109/ICASSP.2000.859144.

91. H.G. Okuno, K. Nakadai, K.i. Hidai, H. Mizoguchi, and H. Kitano. Human-robot interaction through real-time auditory and visual multiple-talker tracking. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1402--1409. 2001. doi:10.1109/IROS.2001.977177.

92. H.G. Okuno, K. Nakadai, and H. Kitano. Social interaction of humanoid robot based on audio-visual tracking. In *International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems: Developments in Applied Artificial Intelligence*, pp. 1--10. 2002.

93. K. Pearson. On lines and planes of closest fit to systems of points in space. In *Philosophical Magazine*, 2(11), pp. 559--572, 1901.

94. M.D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M.E. Davies. Sparse representations in audio and music: from coding to source separation. In *Proceedings of the IEEE*, 98(6), pp. 995--1005, 2010. doi:10.1109/JPROC.2009.2030345.

95. I. Potamitis, H. Chen, and G. Tremoulis. Tracking of multiple moving speakers with multiple microphone arrays. In *IEEE Transactions on Speech and Audio Processing*, 12(5), pp. 520--529, 2004. doi:10.1109/TSA.2004.833004.

96. E. Ravelli, G. Richard, and L. Daudet. Union of MDCT bases for audio coding. In *IEEE Transactions on Audio, Speech, and Language Processing*, 16(8), pp. 1361--1372, 2008. doi: 10.1109/TASL.2008.2004290.

97. L. Rayleigh. On our perception of sound direction. In *Philosophical Magazine*, 13(74), pp. 214--232, 1907. doi:10.1080/14786440709463595.

98. S. Rickard. The DUET blind source separation algorithm. In *Blind Speech Separation*, pp. 217--241. Springer Netherlands, 2007. ISBN 978-1-4020-6479-1. doi:10.1007/978-1-4020-6479-1_8.

99. S. Rickard and M. Fallon. The Gini index of speech. In *Annual Conference on Information Sciences and Systems*, p. 5. 2004.

100. M. Rothbucher, M. Durkovic, H. Shen, and K. Diepold. HRTF customization using multiway array analysis. In *European Signal Processing Conference*, pp. 229--233. 2010.

101. R. Roy and T. Kailath. ESPRIT-estimation of signal parameters via rotational invariance techniques. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7), pp. 984--995, 1989. doi:10.1109/29.32276.

102. R. Saab, O. Yilmaz, M.J. McKeown, and R. Abugharbieh. Underdetermined anechoic blind source separation via ℓ^q-basis-pursuit with $q < 1$. In *IEEE Transactions on Signal Processing*, 55(8), pp. 4004--4017, 2007. doi:10.1109/TSP.2007.895998.

103. R.O. Schmidt. Multiple emitter location and signal parameter estimation. In *IEEE Transactions on Antennas and Propagation*, 34(3), pp. 276--280, 1986. doi:10.1109/TAP.1986.1143830.

104. S.P. Thompson. On the function of the two ears in the perception of space. In *Philosophical Magazine Series 5*, 13(83), pp. 406--416, 1882. doi:10.1080/14786448208627205.

105. A. Torger and A. Farina. Real-time partitioned convolution for Ambiophonics surround sound. In *IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, pp. 195--198. 2001. doi:10.1109/ASPAA.2001.969576.

106. M. Usman, F. Keyrouz, and K. Diepold. Real time humanoid sound source localization and tracking in a highly reverberant environment. In *International Conference on Signal Processing*, pp. 2661--2664. 2008. doi:10.1109/ICOSP.2008.4697696.

107. J.M. Valin, F. Michaud, and J. Rouat. Robust 3D localization and tracking of sound sources using beamforming and particle filtering. In *IEEE International Conference on Acoustics Speed and Signal Processing*, pp. 841--844. 2006. doi:10.1109/ICASSP.2006.1661100.

108. J.M. Valin, F. Michaud, and J. Rouat. Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering. In *Robotics and Autonomous Systems*, 55(3), pp. 216--228, 2007. doi:10.1016/j.robot.2006.08.004.

109. J.M. Valin, F. Michaud, J. Rouat, and D. Letourneau. Robust sound source localization using a microphone array on a mobile robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1228--1233. 2003. doi:10.1109/IROS.2003.1248813.

110. J.M. Valin, F. Michaud, B. Hadjou, and J. Rouat. Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach. In *IEEE International Conference on Robotics and Automation*, pp. 1033--1038. 2004. doi:10.1109/ROBOT.2004.1307286.

111. J.M. Valin, S. Yamamoto, J. Rouat, F. Michaud, K. Nakadai, and H.G. Okuno. Robust recognition of simultaneous speech by a mobile robot. In *IEEE Transactions on Robotics*, 23(4), pp. 742--752, 2007. doi:10.1109/TRO.2007.900612.

112. J. Vermaak and A. Blake. Nonlinear filtering for speaker tracking in noisy and reverberant environments. In *IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, pp. 3021--3024. 2001. doi:10.1109/ICASSP.2001.940294.

113. J. Vermaak, M. Gangnet, A. Blake, and P. Perez. Sequential Monte Carlo fusion of sound and vision for speaker tracking. In *IEEE International Conference on Computer Vision*, pp. 741--746. 2001. doi:10.1109/ICCV.2001.937600.

114. E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. In *IEEE Transactions on Audio, Speech and Language Processing*, 14(4), pp. 1462--1469, 2006. doi:10.1109/TSA.2005.858005.

115. H. Viste and G. Evangelista. On the use of spatial cues to improve binaural source separation. In *International Conference on Digital Audio Effects*, pp. 209--213. 2003.

116. J.A. Wall, T.M. McGinnity, and L.P. Maguire. A comparison of sound localisation techniques using cross-correlation and spiking neural networks for mobile robotics. In *International Joint Conference on Neural Networks*, pp. 1981--1987. 2011. doi:10.1109/IJCNN.2011.6033468.

117. D. Wang and G.J. Brown. *Computational auditory scene analysis: Principles, algorithms, and applications*. IEEE Press, 2006. ISBN 0471741094. doi:10.1109/TNN.2007.913988.

118. D.B. Ward, E.A. Lehmann, and R.C. Williamson. Particle filtering algorithms for tracking an acoustic source in a reverberant environment. In *IEEE Transactions on Speech and Audio Processing*, 11(6), pp. 826--836, 2003. doi:10.1109/TSA.2003.818112.

119. D.B. Ward and R.C. Williamson. Particle filter beamforming for acoustic source localization in a reverberant environment. In *IEEE International Conference on Acoustics Speech and Signal Processing*, pp. 1777--1780. 2002. doi:10.1109/ICASSP.2002.5744967.

120. O. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. In *IEEE Transactions on Signal Processing*, 52(7), pp. 1830--1847, 2004. doi:10.1109/TSP.2004.828896.

121. S.H. Young and M.V. Scanlon. *Detection and localization with an acoustic array on a small robotic platform in urban environments*. Technical Report, DTIC Document, 2003.

122. Y.J. Zhang and Z.Q. Liu. Self-splitting competitive learning: a new on-line clustering paradigm. In *IEEE Transactions on Neural Networks*, 13(2), pp. 369--80, 2002. doi:10.1109/72.991422.