LOCATA

IEEE-AASP Challenge on Acoustic Source Localization and Tracking

- Documentation for Participants -

Version 3.0 (April 17, 2018)

`www.locata-challenge.org`

Heinrich W. Löllmann[1], Christine Evers[2], Alexander Schmidt[1], Heinrich Mellmann[3], Hendrik Barfuss[1], Patrick A. Naylor[2], and Walter Kellermann[1]

[1] Chair of Multimedia Communications and Signal Processing
Friedrich-Alexander University Erlangen-Nürnberg
[2] Dept. of Electrical and Electronic Engineering, Imperial College London
[3] Institut für Informatik, Humboldt Universität zu Berlin

Abstract

This document describes the LOCATA Challenge, its tasks, corpus data, and the provided MATLAB software.

1 LOCATA Challenge Description

This section provides an overview about the goals and tasks of the LOCATA Challenge. Further information, updates and details about the schedule can be found on the LOCATA website `www.locata-challenge.org`.

1.1 Aims & Motivation

The challenge of sound source localization and tracking in realistic environments has attracted widespread attention in the Audio and Acoustic Signal Processing (AASP) community in recent years. Source localization approaches in the literature address the estimation of positional information about acoustic sources using a pair of microphones, microphone arrays, or networks with distributed acoustic sensors. However, despite the substantial interest in sound source localization and tracking approaches, a comprehensive, objective benchmarking campaign of state-of-the-art algorithms has not been conducted up to now.

The IEEE-AASP challenge on acoustic source *LOCalization And TrAcking* (LOCATA) provides researchers in acoustic source localization and tracking with a common, publicly released data corpus and a corresponding framework to objectively benchmark their results against competing algorithms. The LOCATA corpus includes real-life recordings for a range of scenarios with ground-truth data of the positional information of the sources and sensors.

1.2 Tasks

The scope of the IEEE-AASP LOCATA Challenge is the localization and/or tracking of sound sources in realistic acoustic environments. The challenge offers the following six tasks:

Task 1: Localization of a single, static loudspeaker using static microphone arrays

Task 2: Localization of multiple static loudspeakers using static microphone arrays

Task 3: Tracking of a single, moving talker using static microphone arrays

Task 4: Tracking of multiple, moving talkers using static microphone arrays

Task 5: Tracking of a single, moving talker using moving microphone arrays

Task 6: Tracking of multiple moving talkers using moving microphone arrays.

For the purpose of the LOCATA Challenge, participants are provided with the ground-truth values of the microphone positions and array orientations for all scenarios.

Participants of the challenge can submit results either for a single microphone configuration or several configurations as well as a single or multiple tasks. Moreover, results for multiple algorithms can be submitted per task and microphone configuration (see also Sec. 6.3).

2 Datasets

The following data sets are provided

Development dataset: Multichannel audio recordings and ground-truth data for microphone positions, array orientations and source positions of all provided recordings. It comprises 3 recordings for each of the 6 tasks and each of the 4 microphone configurations, i.e., 72 recordings in total (Release: Feb. 2018).

Evaluation data set: Multichannel audio recordings and ground-truth data for the microphone positions and array orientations of all provided recordings. For Task 1 and 2, it comprises 13 recordings for each microphone configuration and task (static scenarios), and 5 recordings per task and array otherwise, i.e., 184 recordings in total (Release: Apr. 2018).

The databases provided as part of the LOCATA Challenge can be downloaded by registered users from the LOCATA website.

3 Microphone Arrays and Sound Sources

This section provides a brief overview about the microphone arrays and sound sources used to record the development and evaluation dataset for the LOCATA Challenge.

3.1 Acoustic Sources & Speech Material

All recordings were conducted in a computing laboratory of the Department of Computer Science at Humboldt University Berlin. The room is equipped with an optical tracking system, which is typically used to track the positions of NAO robots in preparation of the soccer competition RoboCup. This tracking system has provided the positions and orientation of the speakers (talkers and loudspeakers) and microphone arrays as described in Sec. 4 in more detail.

The position of each object (speakers and microphone arrays) is determined by a reference point and its orientation by a reference vector. A local coordinate system (local reference frame) is defined for each object where the reference point of the object coincides with its origin and the reference vector lies on the positive axis as shown in Fig. 1. An elevation of $\theta = 0$ corresponds to the positive $z-$ axis and an azimuth of $\phi = 0$ to the $y-$axis. This local coordinate system is considered in the following to describe the microphone positions for each

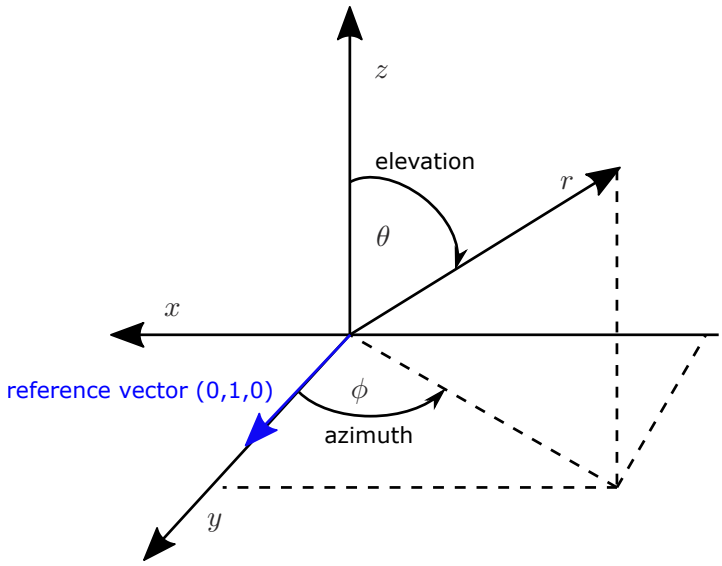

Figure 1: *Local coordinate system.*

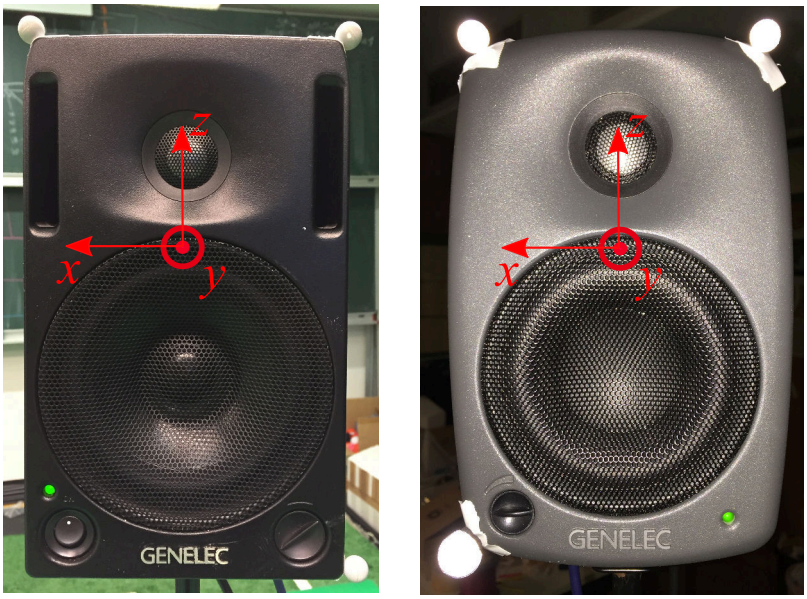

Figure 2: *Loudspeakers with markers: Genelec 1029A (left) and Genelec 8020C (left). The local coordinate system is marked by red color.*

array. Positional information about the speakers and microphone arrays will be described by a global coordinate system as explained later in Sec. 6.

For the recordings of Task 1 and Task 2, loudspeakers of type Genelec 8020C and Genelec 1029A were used as acoustic sources as shown in Fig. 2. The location of the local coordinate system is marked in Fig. 2 by red color. The reference vector coincides with the acoustic axis of the loudspeakers as specified by the data sheets of the manufacturer [1].

The reference point of the human talkers corresponds to the center of the mouth and the reference vector points towards the look direction of the head as indicated in Fig. 3.

For the recordings for Task 1 and Task 2, sentences selected from the CSTR VCTK database [2] were played back by static loudspeakers. The VCTK database provides over 400 newspaper sentences spoken by 109 native English speakers, recorded in a semi-anechoic environment with a sampling frequency of 96 kHz and downsampled to a sampling frequency of 48 kHz. The database is distributed under the Open Data Commons Attribution License, therefore permitting open access for participants. For the recordings for Tasks 3 to 6, speech utterances spoken by different persons were recorded. They spoke sentences which were randomly selected from the VCTK database. The recordings are representative of the practical challenges of data processing of conversational speech, such as natural speech inactivity during sentences, sporadic utterances as well as dialogues between multiple talkers.

Each talker was equipped with a microphone near the mouth[1] as shown in Fig. 3 to

[1]Microphones used: AKG CK 92 with AKG SE 300 B; DPA microphone d:screet 4060

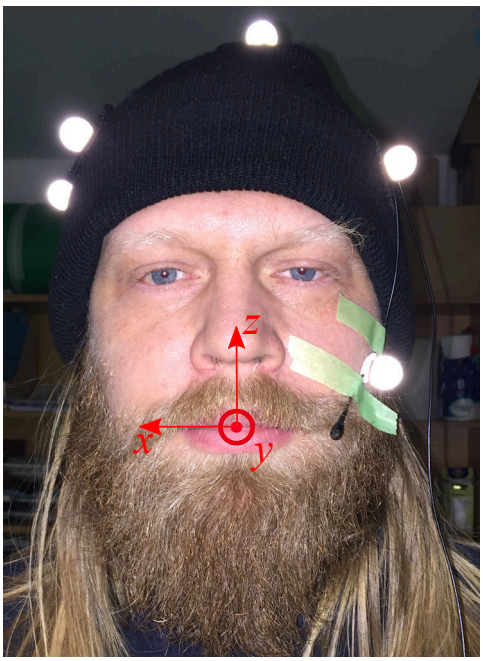

Figure 3: *Human speaker with markers. The local coordinate system is marked by red color.*

record close-talking speech signals. Participants are provided with the close-talking speech signals for the development phase, but will not have access to the close-talking signals for the evaluation phase of the challenge. The complete data set, including also the close-talking signals for the evaluation dataset, will be released as part of the LOCATA corpus once the challenge is completed.

The recordings were conducted in a real environment and are hence affected by measurement noise, traffic noise outside the recording environment, noise of the moving trolley in case of moving microphone arrays, etc.

3.2 Microphone Arrays

Four different microphone arrays were used for the measurements:

- Planar array with 15 microphones (DICIT array) which comprises different uniform linear sub-arrays
- Hearing aid dummies (Siemens Signia) with 2 microphones per hearing aid
- Pseudo-spherical array of 12 microphones integrated in the head of a humanoid robot
- Spherical array with 32 microphones (Eigenmike).

A picture of the used microphone arrays is shown in Fig. 4. All audio recordings have been

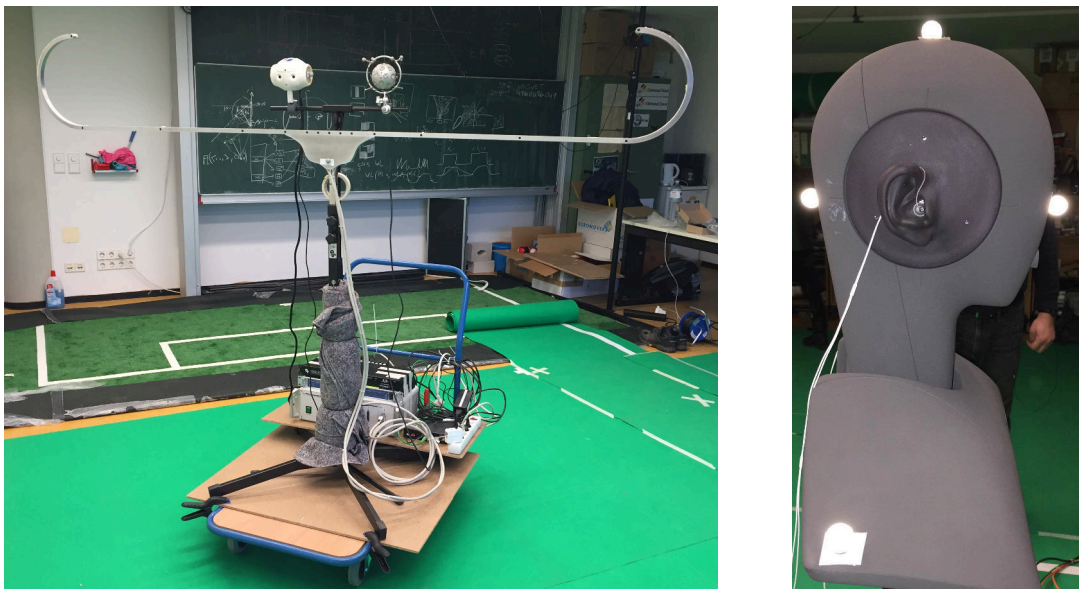

Figure 4: *Microphone arrays (with markers) used for the recordings: DICIT array, robot head, Eigenmike (left), and hearing aid dummies on an artificial head (right).*

performed with a sampling frequency of 48kHz. For the measurements of Task 5 and 6, the arrays have been moved by the depicted trolley.

The deployed microphone arrays account for typical application scenarios of acoustic source localization and tracking algorithms like, e.g., smart TVs and homes, hearing aids, robot audition, or tele-conference systems.

3.2.1 Planar array

The planar array with 15 microphones has been developed as part of the EU-funded project Distant-talking Interfaces for Control of Interactive TV (DICIT), cf., [3], hence denoted as DICIT array in the following. It was selected to account for the opportunities and challenges of arrays with large microphone spacings. It contains four linear uniform sub-arrays with microphone spacings of 4, 8, 16 and 32 cm. A technical drawing of the microphone geometry and the axis of the local coordinate system (local reference frame) is provided in Fig. 5. The locations of the microphones w.r.t. to the local reference frame are listed in Table 1. The listed spherical coordinates are related to the Cartesian coordinates according to Fig. 1. As for all objects, the normalized reference vector lies on the positive y-axis and the reference point coincides with the origin of the local coordinate system (local reference frame). The signals recording by the DICIT arrays (after A/D conversion) have been processed by an equalizer to account for the individual transmission characteristic of its microphones.

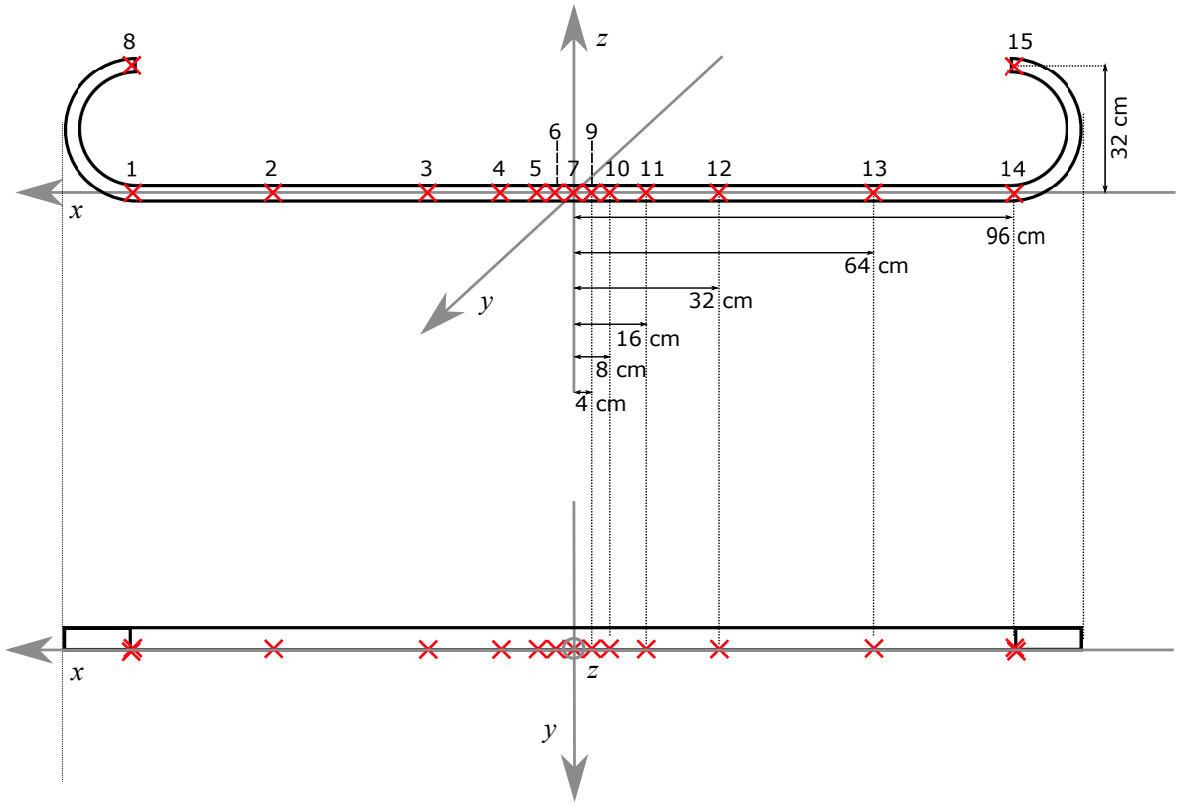

Figure 5: *Microphone positions for the DICIT array (marked by red crosses).*

3.2.2 Hearing aids

Hearing aid dummies of type Pure 7mi (Siemens Signia) of the hearing aid manufacturer Sivantos were mounted on an artificial head (HMS II.3 of HEAD acoustics) for the measurements. Each hearing aid contains two microphones (Sonion, type 50GC30-MP2) with a distance of 9 mm. A technical drawing of the microphone positions is shown in Fig. 6. The coordinates of the microphone positions are listed in Table 2.

3.2.3 Robot head

The deployed prototype head of the humanoid robot NAO, manufactured by Softbank Robotics (former Aldebaran Robotics), was developed as part of the EU-funded project Embodied Audition for Robots (EARS).[2] This prototype head is equipped with 12 microphones positioned in a pseudo-spherical arrangement.[3] The microphone positions, which have been determined by numerical optimization, cf., [4, 5], are listed in Table 3 and a drawing of the microphone

[2] `https://robot-ears.eu/`

[3] The commercially available head for this robot contains 4 microphones.

	Cartesian Coordinates			Spherical Coordinates		
Mic. no.	x [m]	y [m]	z [m]	ϕ [deg]	θ [deg]	r [m]
1	0.960	0.000	0.000	-90	90	0.960
2	0.640	0.000	0.000	-90	90	0.640
3	0.320	0.000	0.000	-90	90	0.320
4	0.160	0.000	0.000	-90	90	0.160
5	0.080	0.000	0.000	-90	90	0.080
6	0.040	0.000	0.000	-90	90	0.040
7	0.000	0.000	0.000	-90	90	0.000
8	0.960	0.000	0.320	-90	72	1.012
9	-0.040	0.000	0.000	90	90	0.040
10	-0.080	0.000	0.000	90	90	0.080
11	-0.160	0.000	0.000	90	90	0.160
12	-0.320	0.000	0.000	90	90	0.320
13	-0.640	0.000	0.000	90	90	0.640
14	-0.960	0.000	0.000	90	90	0.960
15	-0.960	0.000	0.320	90	72	1.012

Table 1: *Microphone positions for the DICIT array.*

	Cartesian Coordinates			Spherical Coordinates		
Mic. no.	x [m]	y [m]	z [m]	ϕ [deg]	θ [deg]	r [m]
1	-0.079	0.000	0.000	90	90	0.079
2	-0.079	-0.009	0.000	97	90	0.079
3	0.079	0.000	0.000	-90	90	0.079
4	0.079	-0.009	0.000	-97	90	0.079

Table 2: *Microphone positions for the hearing aid dummies.*

geometry is provided by Fig. 7.[4]

[4]Modification of a technical drawing kindly provided by Softbank Robotics

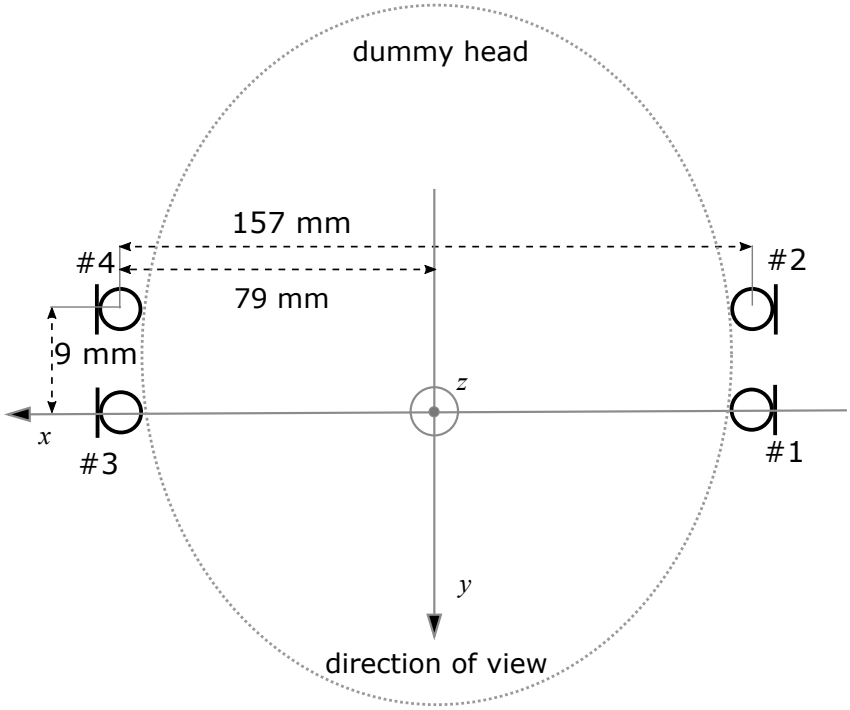

Figure 6: *Microphone positions of the hearing aids mounted on an artificial head.*

	Cartesian Coordinates			Spherical Coordinates		
Mic. no.	x [m]	y [m]	z [m]	ϕ [deg]	θ [deg]	r [m]
1	-0.028	0.030	-0.040	43	134	0.057
2	0.006	0.057	0.000	-6	90	0.057
3	0.022	0.022	-0.046	-46	146	0.056
4	-0.055	-0.024	-0.025	114	112	0.065
5	-0.031	0.023	0.042	54	43	0.057
6	-0.032	0.011	0.046	71	36	0.057
7	-0.025	-0.003	0.051	98	26	0.057
8	-0.036	-0.027	0.038	127	50	0.059
9	-0.035	-0.043	0.025	141	66	0.060
10	0.029	-0.048	-0.012	-149	102	0.057
11	0.034	-0.030	0.037	-131	51	0.059
12	0.035	0.025	0.039	-55	48	0.058

Table 3: *Microphone positions for the robot head.*

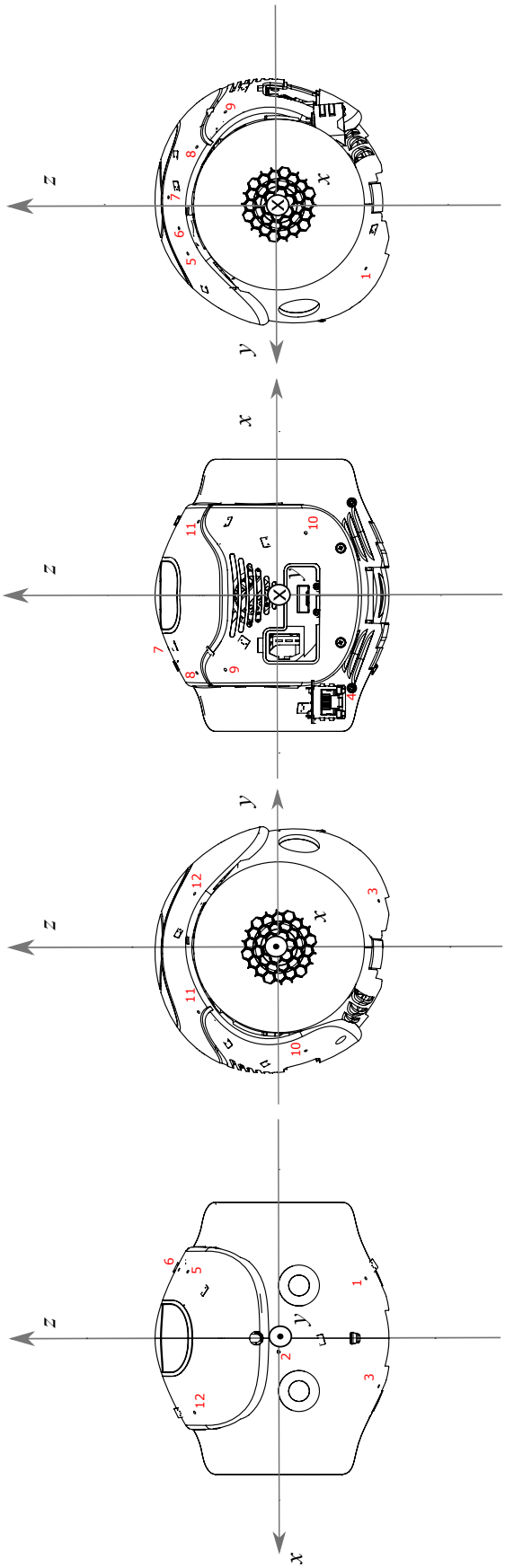

Figure 7: *Microphone positions for the robot head.*

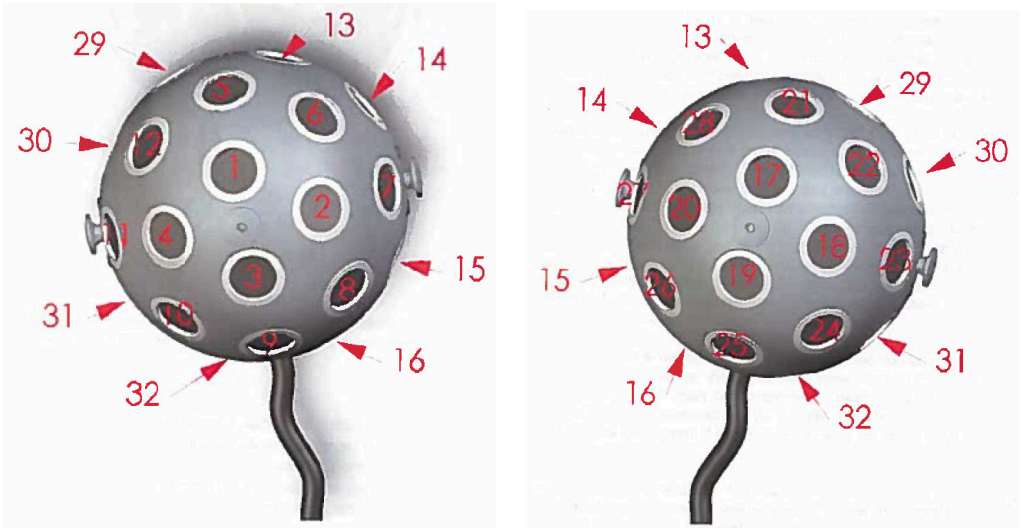

Figure 8: *Microphone positions for mh-acoustics Eigenmike [6].*

3.2.4 Spherical array

A spherical array of type Eigenmike[5] manufactured by mh acoustics was used for the recordings. The Eigenmike contains 32 microphones mounted on a sphere with a diameter of 84mm [6]. The positions of the microphones are listed in Table 4.

The reference point is located at the center of the sphere. Each microphone therefore corresponds to a radius of 0.042m relative to the local reference frame. The microphone geometry is illustrated in Fig. 8. Microphones $1-4$ are positioned towards a blue anodized shock mount [6]. It is important to notice that the blue anodized shock mount was pointed along the y-axis during the recordings (see also Fig. 9).

[5]Purchased in 2009, corresponding to release notes v8.0.

	Cartesian Coordinates			Spherical Coordinates		
Mic. no.	x [m]	y [m]	z [m]	ϕ [deg]	θ [deg]	r [m]
1	0.000	0.039	0.015	0	69	0.042
2	-0.022	0.036	0.000	32	90	0.042
3	0.000	0.039	-0.015	0	111	0.042
4	0.022	0.036	0.000	-32	90	0.042
5	0.000	0.022	0.036	0	32	0.042
6	-0.024	0.024	0.024	45	55	0.042
7	-0.039	0.015	0.000	69	90	0.042
8	-0.024	0.024	-0.024	45	125	0.042
9	0.000	0.022	-0.036	0	148	0.042
10	0.024	0.024	-0.024	-45	125	0.042
11	0.039	0.015	0.000	-69	90	0.042
12	0.024	0.024	0.024	-45	55	0.042
13	-0.015	-0.000	0.039	91	21	0.042
14	-0.036	0.000	0.022	90	58	0.042
15	-0.036	0.000	-0.022	90	121	0.042
16	-0.015	0.000	-0.039	89	159	0.042
17	0.000	-0.039	0.015	180	69	0.042
18	0.022	-0.036	0.000	-148	90	0.042
19	0.000	-0.039	-0.015	180	111	0.042
20	-0.022	-0.036	0.000	148	90	0.042
21	0.000	-0.022	0.036	180	32	0.042
22	0.024	-0.024	0.024	-135	55	0.042
23	0.039	-0.015	0.000	-111	90	0.042
24	0.024	-0.024	-0.024	-135	125	0.042
25	0.000	-0.022	-0.036	180	148	0.042
26	-0.024	-0.024	-0.024	135	125	0.042
27	-0.039	-0.015	0.000	111	90	0.042
28	-0.024	-0.024	0.024	135	55	0.042
29	0.015	-0.000	0.039	-91	21	0.042
30	0.036	0.000	0.022	-90	58	0.042
31	0.036	0.000	-0.022	-90	122	0.042
32	0.015	0.000	-0.039	-89	159	0.042

Table 4: *Microphone positions for the Eigenmike.*

4 Ground-Truth Position Data

The positions of all microphone arrays and sound sources were recorded during the measurements with the help of an optical tracking system. This section describes briefly the process by which the positioning data was obtained and discusses its accuracy.

The ground truth for the positions and orientations of all microphone arrays and sound sources was determined by means of the optical tracking system *OptiTrac* [7]. For this purpose, reflective markers with diameters of 11.1mm and 15.9mm were attached to each object (i.e., loudspeakers, microphone arrays and human talkers) as shown, e.g., in Fig. 2, Fig. 3 and Fig. 4. These markers were detected by 10 spatially distributed infrared cameras of type *Flex 13*.[6] The synchronized camera signals allowed to determine the marker positions by triangulation using the software *Tracking Tools* (version 2.5.3). At least three markers were attached to each object to track their positions and orientations. Unique marker geometries were used to distinguish between the objects. Human talkers wore hats with attached markers. In addition, one marker was attached to the cable of each DPA mouth microphone close to the mouth as shown in Fig. 3.

A group of markers marking one object was identified as a singular rigid pattern called *trackable* by the tracking software. Thus, each object was described by a trackable whose position and orientation were tracked as a unit by the *OptiTrac* software in addition to the positions of the individual markers. The used tracking system is able to simultaneously track the positions and the orientations of multiple trackables. A trackable can be successfully tracked even if some of the markers are occluded as long as at least three markers are visible for the cameras. All trackables and markers were tracked with respect to a global coordinate system whose origin and orientation was defined by a 'calibration square' placed on the floor.[7] The positions of the markers and trackables were captured with a rate of 120 frames per second and the audio recordings were performed with a sampling rate of 48kHz. All data streams were stored on the same computer whose timestamps generated by its system clock were stored for each data sample. These timestamps were used in a post-processing step to synchronize the position data and audio data streams with an accuracy of about $\pm$1ms.

The recorded (noisy) measurements for the marker positions have been used in a post-processing step to determine the ground-truth positions and orientations of all objects (trackables), i.e., the reference points and reference vectors as defined in Sec. 6.1. While a detailed description of this post-processing step is beyond the scope of this manual, the main sources for erroneous position data should be briefly discussed to provide an assessment for the accuracy of the tracking procedure.

[6] `http://optitrack.com/products/flex-13/specs.html`

[7] `https://v20.wiki.optitrack.com/index.php?title=Calibration`

The main cause for measurement errors were reflections of the infrared light, which was emitted by the tracking system, at the surfaces of the microphone arrays. Such reflections led to the detection of non-existing markers ('ghost markers') as well as missing detections of markers. In addition, some markers were occasionally occluded during the measurements with multiple moving objects. These erroneous detections led in isolated instances to a tracking loss or misalignment by the *OptiTrac* system for some objects, resulting in outliers for the orientation and position of the trackables. In some cases, the missing marker detections also resulted in high frequency noise (jitter) for the measured trajectories in the range of a few millimeters. Outliers and missing data points have been replaced by estimated values in the course of the post-processing step.

In order to estimate how well the reconstructed trajectories correspond to the originally recorded marker positions, the mean-square error between the recorded marker positions and the marker positions of the reconstructed trackables were calculated for all positions. The results indicate deviations of less than 12mm for all objects, where the deviations for most objects are below 6mm.

The positions of the microphones in relation to the marker positions were derived by means of technical drawings of the microphone positions and caliper measurements with an estimated accuracy of a few millimeters. The movement of the arrays might have caused additional measurement errors: The DICIT array was slightly bending back and forth while being moved during the measurements for Task 5 and 6 due to its large dimensions(cf., Fig. 4), where a rigid array was assumed for the determination of the microphone positions.

The Eigenmike used in the measurements is a 2009 version, and is therefore mounted in a Samson SP01 spider shockmount holder using rubber bands. To minimize the effects of shadowing and scattering, the markers were attached to the edges of the shockmount holder as shown in Fig. 9. The use of the rubber bands can lead to small rotations and offsets of the microphone array relative to the measured marker positions.

5 Data & Software

This section provides a description of the datasets provided for the development and evaluation phase of the challenge.

The LOCATA *development database* consists of 6 zip-files (one archive for each task, named `task1.zip`, ..., `task6.zip`) and a corresponding SHA256 checksum file, `dev_release.sha256`. The folders `task1`, ... `task6` contain multichannel audio recordings and ground-truth data of the development database. Each task directory comprises one sub-directory per recording, which contains sub-directories with the wav-files (audio recordings) and csv text-files

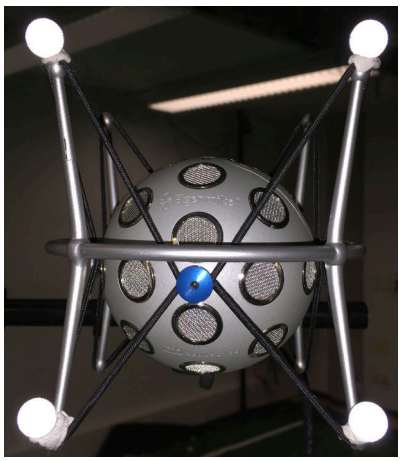

Figure 9: *Eigenmike mounted in a spider shockmount holder with attached markers.*

(positional information) for each array.

The SHA256 checksums contained in `dev_release.sha256` should be used to check that the downloaded files are not corrupted, e.g., by executing the following commands:

- Linux: `sha256sum -c dev_release.sha256`
- macOS: `shasum -c dev_release.sha256`
- Windows: Enter `certutil -hashfile task1.zip SHA256` etc. on the Command Prompt or use programs like *HashMyFiles* (`www.nirsoft.net`).

The LOCATA *evaluation database* consists of 8 zip-files each restricted to a maximum file size of 2 GB (`eval_task1.zip`, `eval_task2.zip`, `eval_task3.zip`, `eval_task4.zip`, `eval_task4.z01`, `eval_task5.zip`, `eval_task6.zip`, `eval_task6.z01`) and one SHA256 checksum file
`eval_release.sha256`. The extraction of the multiple parts of the zip-archives can be done for Windows, e.g., by using programs like *WinRar* and for Linux and macOS, e.g., by the commands `zip -FF eval_task4.zip --out task4_full.zip` and `unzip task4_full.zip`.

The zip-file `matlab_v2.zip` contains a folder `matlab` with all MATLAB functions to read the data, estimate the source locations and write the results to csv text-files for the development and evaluation database. The zip-file `Table4Results.zip` contains a table in an Excel and csv format (`Table4LOCATA_submission.xls` and `Table4LOCATA_submission.csv`) for the submission of the challenge results. Moreover, a pdf-file with this documentation is provided.

For each recording, the following files are provided for the development and evaluation database:

- One file named `required_time.txt`, containing the synchronized system timestamps. Estimates must be provided for **each** of the timestamps specified in `required_time.txt` for the evaluation database by the participants of the LOCATA Challenge.
- One file named `audio_array_$array.wav` for each array, containing the multichannel recordings of the respective microphone array (`dicit`, `benchmark2`, `eigenmike`, `dummy`).
- One file named `audio_array_timestamps_$array.txt` for each array, containing the synchronized system timestamps for each sample in the corresponding file `audio_array_$array.wav`.
- One file named `position_array_$array.txt` for each array, containing the ground-truth position and orientation of the array reference point and microphone positions as documented in Sec. 3.2. Note that the positions are synchronized with the audio data with a sampling frequency of 120Hz, cf., Sec. 4.

For the development database, the following files are provided in addition:

- One file named `audio_source_$source.wav` for each source, containing the single-channel source signals. For Tasks 1 and 2, the clean speech signals from the VCTK database are provided. For the remaining tasks, the close-talking signals recorded by the reference microphones are provided (see also Sec. 3.1).
- One file named `audio_source_timestamps_$source.txt` for each source, containing the synchronized system timestamps for each sample in the corresponding file `audio_source_$source.wav`.
- One file named `position_source_$source.txt` for each source, containing the ground-truth position and orientation of the source reference point as documented in Sec. 3.1 and Sec. 6.

MATLAB programs are provided to demonstrate the use of the LOCATA databases. For a smooth execution, participants must ensure that the folders containing the unarchived data form the following structure for the development database:

```
matlab
data
  task1
    recording1
      ($arrayname1)
        audio_array_benchmark2.wav
        audio_source_($sourcename1).wav
        audio_source_($sourcename2).wav
        ...
        audio_array_timestamps.txt
        audio_source_($sourcename1).wav
        audio_source_($sourcename2).wav
        ...
```

A folder `recording$` contains speech recordings for the same speaker(s).

The required structure for the evaluation database is the same with the difference that the files `audio_source_($sourcename$).wav` and `position_source_($sourcename$).txt` are not provided.

The provided MATLAB functions are located in the folder `./matlab/`. The main function is `./matlab/main.m`. The interface of this function reads:

`main( data_dir, results_dir, is_dev, arrays, tasks).`

The path string for either the development or evaluation database is given by `data_dir` and the path string pointing to the directory where the results `main.m` are to be written to is given by `results_dir`. A value of 1 for `is_dev` indicates that the development database is provided where a value of 0 indicates the use of the evaluation database. The cell array `arrays` and the vector `tasks` specify the microphone arrays and tasks for which the evaluation is performed. These two input arguments are optional; the default settings use all arrays ({`'benchmark2', 'eigenmike', 'dicit','dummy'`}) and tasks (`[1,2,3,4,5,6]`).

The function name for the participant's localization algorithm is specified by the string variable `my_alg_name` within the function (line 79) where the *Multiple SIgnal Classification* (MUSIC) algorithm (e.g., [8]) is provided as an example.

The results of `main.m` are written into csv text-files and stored in a similar folder structure to the LOCATA databases, further structured by array (`benchmark2`, `dicit`, `dummy`,

eigenmike) and algorithm (e.g., MUSIC as in the example below). In the case of running main.m over Task 1 for all arrays, the folder structure will be as follows:

```
results
└── task1
    └── recording1
        ├── benchmark2
        │   └── MUSIC
        ├── dicit
        │   └── MUSIC
        ├── dummy
        │   └── MUSIC
        └── eigenmike
            └── MUSIC
```

The sub-folder MUSIC contains N csv text-files with the localization results (source_$.txt where N denotes the number of estimated sound sources, and one file containing the elapsed computation time of the localization algorithm (telapsed.txt). For the development database, MATLAB figures with localization results and ground-truth data are saved in addition. For the evaluation database, only the text files but not the MATLAB figures are to be saved to the folder with the results.

The main function iterates over all recordings in data_dir for the specified tasks and arrays and calls the following functions:

- ./matlab/utils/load_data.m: Loads audio data from wav-files recorded by the microphone arrays into the structure audio_array. For the development database, the reference microphone signals are loaded into the structure audio_source. In addition, ground-truth *OptiTrac* positions for the microphone arrays and, in the case of the development database, sound sources are loaded into the structures position_array and position_source, respectively. Thus, the development database provides all four structures such that participants have access to the ground-truth positions of the sound sources as well as the close-talking speech signals where the evaluation database provides only the structures audio_array and position_array.

- ./matlab/utils/get_truth.m: Evaluates the source directions (azimuth/elevation) w.r.t. the local coordinate system of each microphone array as specified in Sec. 3.2. The function demonstrates the use of rotation matrices and translation vectors where positional information about the sound sources is returned only for the development database.

- ./matlab/utils/MUSIC.m: Demonstrates the use of the LOCATA data for DOA estimation using MUSIC.[8] For the LOCATA Challenge, DOA estimates are required at

[8]The frequency-domain processing of multichannel recordings by MUSIC might require to increase the system swap space (virtual memory) such that MATLAB can allocate enough memory for the execution of this function.

the timestamps defined by the *OptiTrac* system. `MUSIC.m` therefore also shows as an example how the DOA estimates obtained from the multichannel speech data can be interpolated to the *OptiTrac* update rate. Participants of the challenge are free to use their own interpolation techniques (if required).

- `./matlab/utils/plot_results.m`: Plots the estimated results. For the development but not the evaluation database, MATLAB figures with the localization results and ground-truth data are saved in the folders with the results.
- `./matlab/utils/results2csv.m`: Checks if only valid fields are provided and saves the MATLAB struct with the results to csv files in the directory `results_dir`.

The provided MATLAB software has been created and tested with MATLAB version 9.3 (R2017b). The use of MATLAB version 8.6 (R2015b) or an older version may cause problems.

6 Coordinate Systems and Challenge Requirements

This section provides an overview of the used coordinate systems and the kind of estimates that should be provided by the participants of the challenge.

6.1 Reference Frames

Participants are required to provide estimates of the source positional information relative to the microphone array used for the audio recordings. The coordinate system used for LOCATA, as illustrated in Fig. 10, is defined as follows: The origin of the global reference frame is defined as the origin of the *OptiTrac* system, which was defined by a calibration square lying on the floor, cf., Sec. 4. It should be noted that the position of the calibration square, and hence the location of the global reference frame origin within the enclosure, may have changed between recordings. This does no effect the results for the LOCATA Challenge since the information required for reference frame transformations is provided specific to each recording.

The x-, y-, and z-axes are defined as the East, North and Up direction relative to the global origin. The 3D position of the array reference point (see Sec. 3.2) in x-, y-, and z-coordinates is defined as the East, North and Up position, $(x_{t,r}, y_{t,r}, z_{t,r})$, of the reference point relative to the global reference frame. For each recording, the file `position_array_$array.txt` for the corresponding array provides the 3D position of the reference point by the field `position` as a $3 \times T$ matrix, where T is the number of *OptiTrac* samples for the corresponding recording. The 3D microphone positions, defined similar to the reference point, are provided in the field

`mic` as a $3 \times T \times M$ matrix, where M is the number of microphones in the corresponding array. The rotation of the array is provided by a 3D rotation matrix, which is contained in the field `rotation` as a $3 \times T \times 3$ matrix. The normalized reference vector w.r.t. the global reference frame is provided by the $3 \times T$ matrix in `ref_vec`.

Fig. 10 shows an illustration of the LOCATA reference frames defined above. The global reference frame is shown in black. The local reference frame relative to any selected array is shown in green. Note that the origin of the local reference frame corresponds to a translation by the array reference point position and a rotation by the array orientation. The figures depict in red the spherical and Cartesian coordinates of a source relative to the microphone array.

6.2 Coordinate System

The positional source information is stored in the file `position_source_$source.txt` for each recording of the corresponding source. The file provides the absolute position in Cartesian coordinates $(x_{t,n}, y_{t,n}, z_{t,n})$ of the source reference point in the global reference frame in the field `position` as a $3 \times T$ matrix for each of the T *OptiTrac* samples. The source rotation in the global reference frame is provided in the field `rotation` by a $3 \times T \times 3$ matrix. The normalized reference vector w.r.t. the global reference frame is provided by the $3 \times T$ matrix in `ref_vec`.

The spherical coordinates of any source in the local reference frame relative to any selected array are defined by the source azimuth, elevation and range in compliance with Fig. 1. The source azimuth $\phi_{t,n}$ at any time step t of source n relative to an array is defined as the horizontal direction, where $\phi_{t,n} = 0$ rad is pointing along the positive y-axis of the local reference frame, i.e., in the 'look direction' of the array. The azimuth is rotated counter-clockwise and is defined between $-\pi \leq \phi_{t,n} < \pi$, i.e., the negative x-axis of the local reference frame corresponds to $\phi_{t,n} = \frac{\pi}{2}$ rad, the positive x-axis corresponds to $\phi_{t,n} = -\frac{\pi}{2}$ rad, and the negative y-axis corresponds to $\phi_{t,n} = \pi$ rad. The elevation, $0 \leq \theta_{t,n} \leq \pi$, is defined as the vertical direction, where $\theta_{t,n} = 0$ rad is pointed upwards along the positive z-axis of the local reference frame, $\theta_{t,n} = \pi$ rad is pointed downwards along the negative z-axis and $\theta_{t,n} = \frac{\pi}{2}$ is pointed along the horizontal plane, i.e., in the look direction of the array. The source-sensor range, $r_{t,n} \geq 0$, is defined as the Euclidean distance between the reference points of source and sensor.

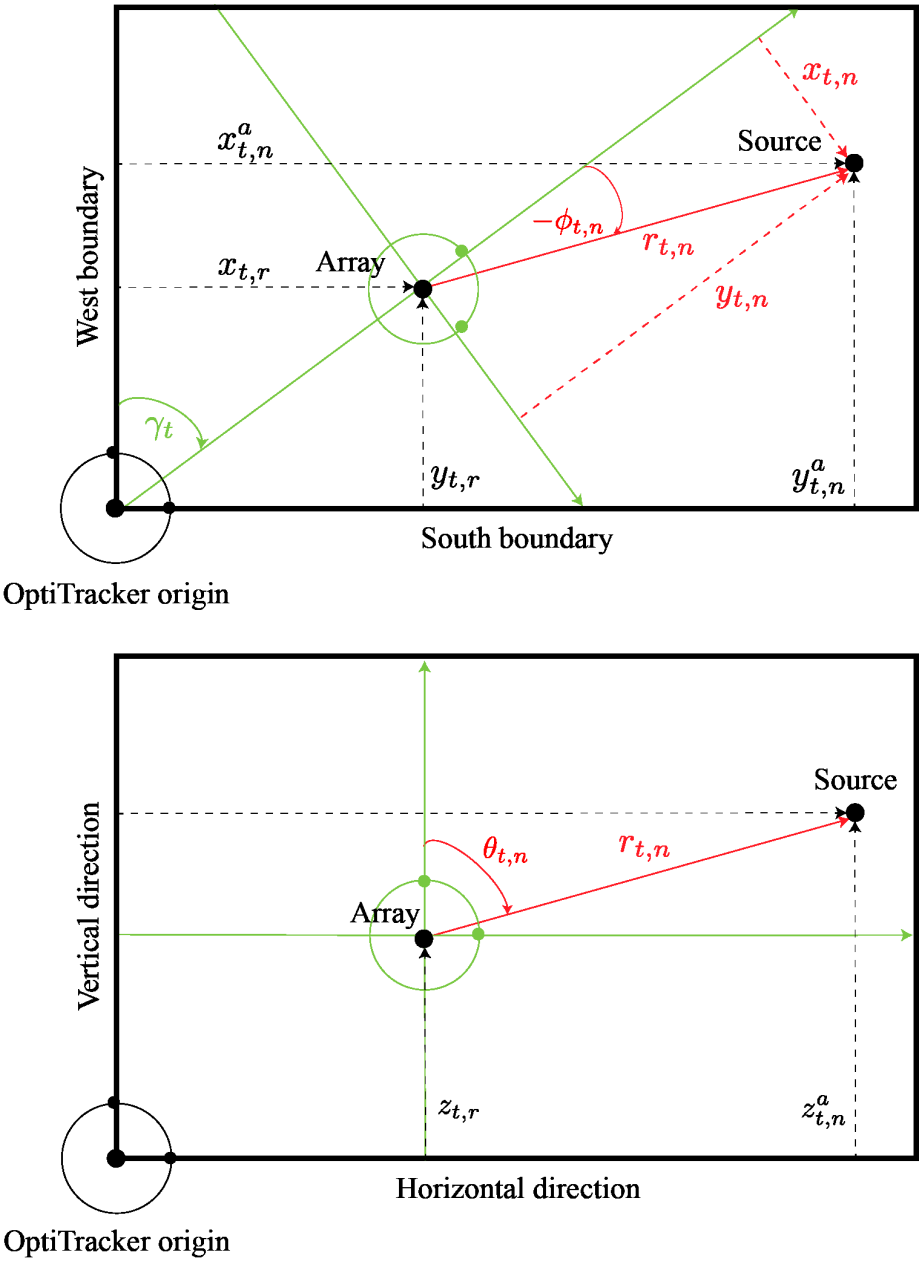

Figure 10: *Illustration of LOCATA reference frames.*

6.3 Challenge Requirements

The organizers strongly recommend that challenge participants use the provided MATLAB program `main.m` to evaluate and store their results for the evaluation database. The participant's code should be inserted as indicated by the comments where the required interface for the participants' function call is also explained.

Challenge participants are required to submit one zip-file containing a directory with the

results generated by `main.m`. Archives exceeding a limit of 2 GB per file should be split into multiple zip-files. For Linux and macOS, this can be done, e.g., via the command line statement `zip -r -s 2g results.zip results/` .

The zip-archive must contain the files `source_*.txt` and `telapsed.txt` as detailed in Sec. 5. The files `source_*.txt` must satisfy the following requirements:

- Azimuth estimates w.r.t. the local reference as defined in Sec. 6.2 must be provided. Each azimuth value must be associated with a unique source identifier (ID). The function `./matlab/main.m` demonstrates how to specify and save this data.
- In addition to the estimated azimuth values, participants are encouraged to provide the following estimates at each timestamp defined in `required_time.txt`:
  - Azimuth [rad] (**mandatory**)
  - Source ID (**mandatory**)
  - Elevation [rad] (optional)
  - Source-to-sensor range [m] (optional)
  - Cartesian x-position [m] (optional)
  - Cartesian y-position [m] (optional)
  - Cartesian z-position [m] (optional)
  - Speed [m/s] (optional).
- **All** estimates **must** be provided at the predefined timestamps which are linked to the update rate of 120Hz of the utilized *OptiTrac* system. The required timestamps are defined for each recording in the file `required_time.txt`. How DOA estimates at these exact timestamps are obtained is demonstrated in `./matlab/utils/MUSIC.m` as described before.

In addition to the folders with the results, the participants need to submit a table with further information about their submissions. A corresponding Excel and csv table template along with further instructions is provided as part of the evaluation dataset (see Sec. 5).

The evaluation of the challenge submissions is based on the accuracy of the estimated locations, IDs and number of sources during periods of source activity. The source activity periods are hand-labeled by means of the anechoic and/or close-talking speech signals to determine the ground-truth values for the evaluation. Thereby, a source activity period corresponds to a speech utterance by a source and, therefore, includes voiced as well as unvoiced speech. Measurements of head-related transfer functions (HRTFs) or steering vectors for the microphone arrays used for the LOCATA recordings are not be provided. Nevertheless, participants of the challenge are allowed to use any measured or simulated HRTFs that are available in the public domain or have been acquired using the participant's own equipment.

Acknowledgments

The organizers would like to thank Claas-Norman Ritter and Ilse Sofía Ramírez Buensuceso Conde for their contributions as well as the hearing aid manufacturer Sivantos for providing the hearing aid dummies.

References

[1] Genelec, "Genelec Speaker Manuals," Available online, Dec. 2017, https://www.manualslib.com/brand/genelec/speakers.html.

[2] C. Veaux, J. Yamagishi, and K. MacDonald, "English Multi-speaker Corpus for CSTR Voice Cloning Toolkit," [Online] http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html, 2018.

[3] A. Brutti, L. Cristoforetti, W. Kellermann, and L. Marquardt, "WOZ Acoustic Data Collection for Interactive TV," *Language Resources and Evaluation*, vol. 44, no. 3, pp. 205–219, Sept. 2010.

[4] V. Tourbabin and B. Rafaely, "Optimal design of microphone array for humanoid-robot audition," in *Proc. of Israeli Conf. on Robotics (ICR)*, Herzliya, Israel, Mar. 2016, (abstract).

[5] V. Tourbabin and B. Rafaely, "Theoretical Framework for the Optimization of Microphone Array Configuration for Humanoid Robot Audition," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 22, no. 12, Dec. 2014.

[6] mh acoustics, *EM32 Eigenmike microphone array release notes (v17.0)*, Oct. 2013, www.mhacoustics.com/sites/default/files/ReleaseNotes.pdf.

[7] OptiTrack, *Product Information about OptiTrack Flex13*, [Online], http://optitrack.com/products/flex-13/, Feb. 2018.

[8] H. L. van Trees, *Optimum Array Processing: Detection, Estimation, and Modulation Theory*, vol. Part IV, Wiley, 2004.