

Cocktail-Party Recordings
by
Unnikrishnan, Harikrishnan (harikrishnan@uky.edu)
Center for Visualization and Virtual Environments
University of Kentucky
9/18/2008

Introduction

This document gives descriptions of the data for the cocktail party simulations recorded at Center for Visualization and Virtual Environments (CVVE), University of Kentucky during May – August 2008. The data were recorded so the cocktail party noise and the speaker of interest (SOI) can be separated and combined at various signal to noise (SNR) levels. This enables development and more efficient testing of beam forming algorithms, especially for microphone arrays with irregular spatial distributions in a “cocktail party” scenario.

Experimental setup

A 3.6x3.6x2.2 meter space surrounded by aluminum struts forms the audio cage. Microphones were mounted on the struts, and for all experiments 16 microphones were distributed in various configurations over the audio cage. The cage was situated in a typical office space with dimensions 6m by 6.7m by 2.25m with a carpet floor, acoustic ceiling tiles, plasterboard walls, and windows on one side. The natural noise sources included vents, florescent lights, computers, and traffic noise through the window. 3 acoustic foam pads (Auralex MAX-WALL 420) were set up to attenuate street noise and computer noise in the recording area, as well as to reduce room modes. For the cocktail party noise recordings, 6 or more people were present in the cage in sitting or standing postures and carrying on conversations in small groups. A total of 3 separate recordings were made of the party noise, where for each recording one person was removed to later be the SOI for that party recording. There were 3 SOI recording made of each individual person removed from the party. The microphone configurations remain the same and the SOI was talking either in a standing or sitting position. This resulted in 3 party noise recordings and 3 SOI recordings. Since the party and SOI recordings were made separately with the same room and microphone configurations, they can be scaled and linearly combined with party noise recordings to achieve various SNR levels. All the recordings were done at sampling rate of 22.05 kHz. The 3-D coordinates of the microphones, speed of sound, RT60 time, and SOI positions every 20 ms were estimated. Test data can be now be formed by linearly adding the individual recordings of the SOI to the party recording in which they were not present.

Data Organization:

Recorded data and support information for each experiment are stored in a folder named according to the following format: “cocktail_<date of recording>_<mic configuration description>”

Each folder contains following files:

1. Individual recordings are named as “*soi<n>.wav*” stored as 16 channel wave files. The integer n denotes the n^{th} SOI, with $n = 1, 2$ and 3 .
2. Party recordings without the n^{th} SOI are named as “*party<n>.wav*” stored as 16 channel wave files.
3. The position of SOI calculated at every 20ms saved as a text file in a file labeled

“soi<n>pos.txt.” Each row of this files has 4 values; the time value in seconds, followed by x, y and z coordinates of the estimated position.

4. Speed of sound c , RT60 time, and audio recording details are in file *“info.txt.”*
5. Microphone position coordinates are in a text file named *“mpos.txt”* and information is stored as a 3 X 16 matrix. Each row represents the x, y, and z position, while each column represents microphone 1, 2, ..., 16.

Measurement of environmental parameters

The parameters of interest in this setup are

1. Speed of sound
2. RT60 time
3. Position of microphones in the 3-D space

The following paragraphs describe how these were measured.

Measurement of speed of sound

Speed of sound measurements were made within 2 hours of the party and SOI recordings. The Matlab function *velest.m* from arraytoolbox[3] is used for the purpose. The measurement used a collinear placement of 3 microphones and a loudspeaker to form an endfire configuration. White Gaussian Noise (WGN) is played from the loudspeaker for 25 seconds. The velocity of sound for each microphone pair:

$$c_{ij} = \frac{T_{ij}}{d_{ij}} \quad (1)$$

where T_{ij} is the time of propagation and d_{ij} is the distance for each microphone pair. T_{ij} is estimated through cross correlation, and d_{ij} is measured using a measuring tape. c_{ijk} is computed for $k = 1, 2, \dots, 25$ time windows of 1 second duration. The c_{ijk} values corresponding to correlation magnitudes less than 0.4 were not used in the estimation. Of the remaining the most repeated value for c_{ijk} is selected as the velocity of sound. The velocity of sound for each experiment is provided in the *info.txt* file.

Measurement of RT60 time

A loud speaker excited the room with white Gaussian noise (WGN) for 30 seconds (until steady-state for the reverberations was reached). The WGN was cut off at 30 seconds and the room response $y(t)$ is recorded for 2 more seconds. The $y(t)$ drops linearly on a log scale after the loud speaker cuts off. The RT60 time is the amount of seconds it takes for $y(t)$ to fall 60 dB from its original magnitude right after cutoff. However the ambient room noise is often great than 60dB from the loudest sound, so the measurement was made based on the slope of the decaying sound right after cutoff and before it fall below the noise floor. The Matlab function *rt60est.m* from Arraytoolbox [3] is used for the purpose. RT60 for each simulation set is available in *info.txt*.

Estimation of microphone locations

The microphones M_n were arranged in a fixed geometry near the audio cage boundaries. Their location in 3D space was estimated by a triangulation method. Three reference points R_1 , R_2 and R_3 , whose coordinates are fixed (corners of the audio cage) were used. The distance of M_n from R_1 , R_2 and R_3 are measured using a measuring tape. Also the z coordinates (height from the floor) of M_n are measured using a laser beam (Leica DISTO A6). These data were used to compute the coordinates of M_n . The microphone coordinates of M_n is given as a 3 X N matrix in *mpos.txt*.

Source location algorithm

For estimating the position of the speaker, SRP-PHAT-Beta [1] algorithm is used. An approximate position of the speakers mouth was initially measured as in the case of the mic positions. A Steered Response Coherent Power (SRCP) was then applied in a 0.4 meter neighborhood around that point to estimate the sequence of positions of the SOI every 20 ms.

The SRCP was computed over a 3-D spatial grid of spatial points every 0.04 meters. A whitening parameter β determines how much contribution the magnitude of the source should have in the phase transformation (PHAT), which was set to 0.6 for the position estimates. For every 20 ms window in time, the maximum SRCP value in space was chosen as the SOI positions.

Let P_{ijk} be the detected peaks with i, j, k being the x, y and z co-ordinates. To reduce the effect of noise during the periods of silence a secondary threshold is applied. Any detection with coherent power less than 5 % of the maximum value is considered as absence of the sound source. The silence is represented by 'NaN' (Not A Number – Matlab notation). P_{ijk} is still spiky due to the presence of background noise or error in source location. Hence to get a smoother data P_{ijk} is passed through a sliding median filter over time with a window length of 21 samples to get the source location measurement.

(2)

References

- [1] K.D. Donohue, J. Hannemann, and H.G. Dietz, "[Performance for Phase Transform for Detecting Sound Sources in Reverberant and Noisy Environments](#)," *Signal Processing*, Vol. 87, no. 7, pp. 1677-1691, July 2007.
- [2] K.D. Donohue, K.S. McReynolds, A. Ramamurthy, "[Sound Source Detection Threshold Estimation using Negative Coherent Power](#)," *Proceeding of the IEEE, Southeastcon 2008*, pp. 575-580, April 2008.
- [3] K. D. Donohue, "Audio array toolbox", September 2008.
<http://www.engr.uky.edu/~donohue/audio/Arrays/MAToolbox.htm>