Binaural Source Localization and Filtering

System Block Diagram

By:
Daniel Hauer

Project Advisor:
Dr. Brian D Huggins

Date:
November 22, 2005

Project Description

The purpose of this project is to decode 3D spatial information audio information from stereo or binaural sources based on relevant psychoacoustic cues using DSP techniques. The goals would include graphing location–based magnitudes in a 2D plane environment against time. Also, the audio signal would be filtered based on its spatial properties (namely, the source azimuth angle), rather than frequency or other criteria. In this way, multiple signals could be decoded from the stereo signal to be sent to multiple strategically spaced speakers for a more realistic soundstage. In figure 1 and figure 2, this possible use of the system is illustrated. In both the diagrams, the speaker array is shown as the right-most output. The high level block diagram is shown as figure 1.

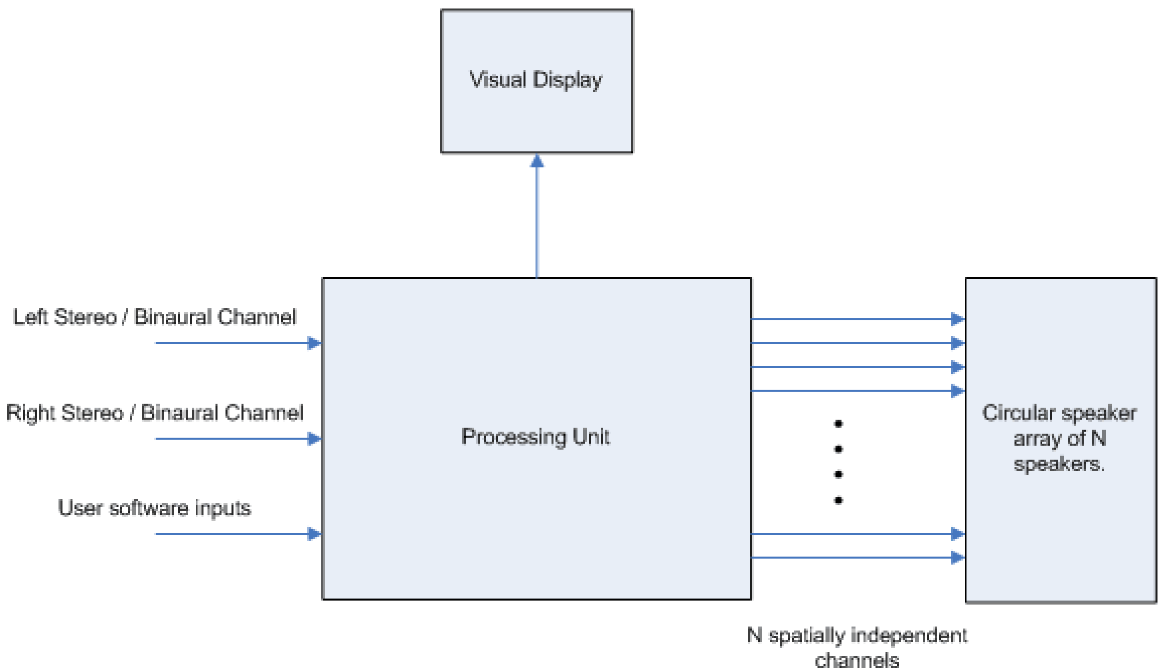

Figure 1 – System block diagram.

The inputs and outputs to the system will be in the form of waveform files as shown in figure 2. The waveform files will be either created artificially (as in test tones for testing), be custom recorded for the project, or be extracted from an audio CD. They will be in the format of Microsoft PCM wave (.wav) files.

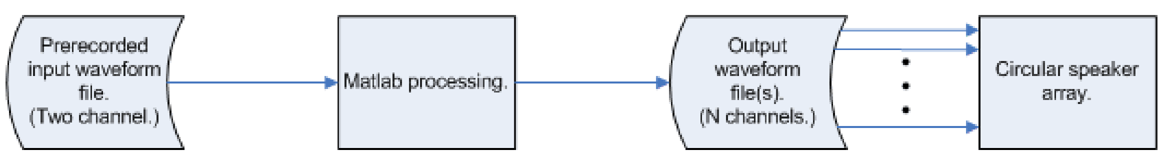

Figure 2 – System input and output.

Software Flowchart

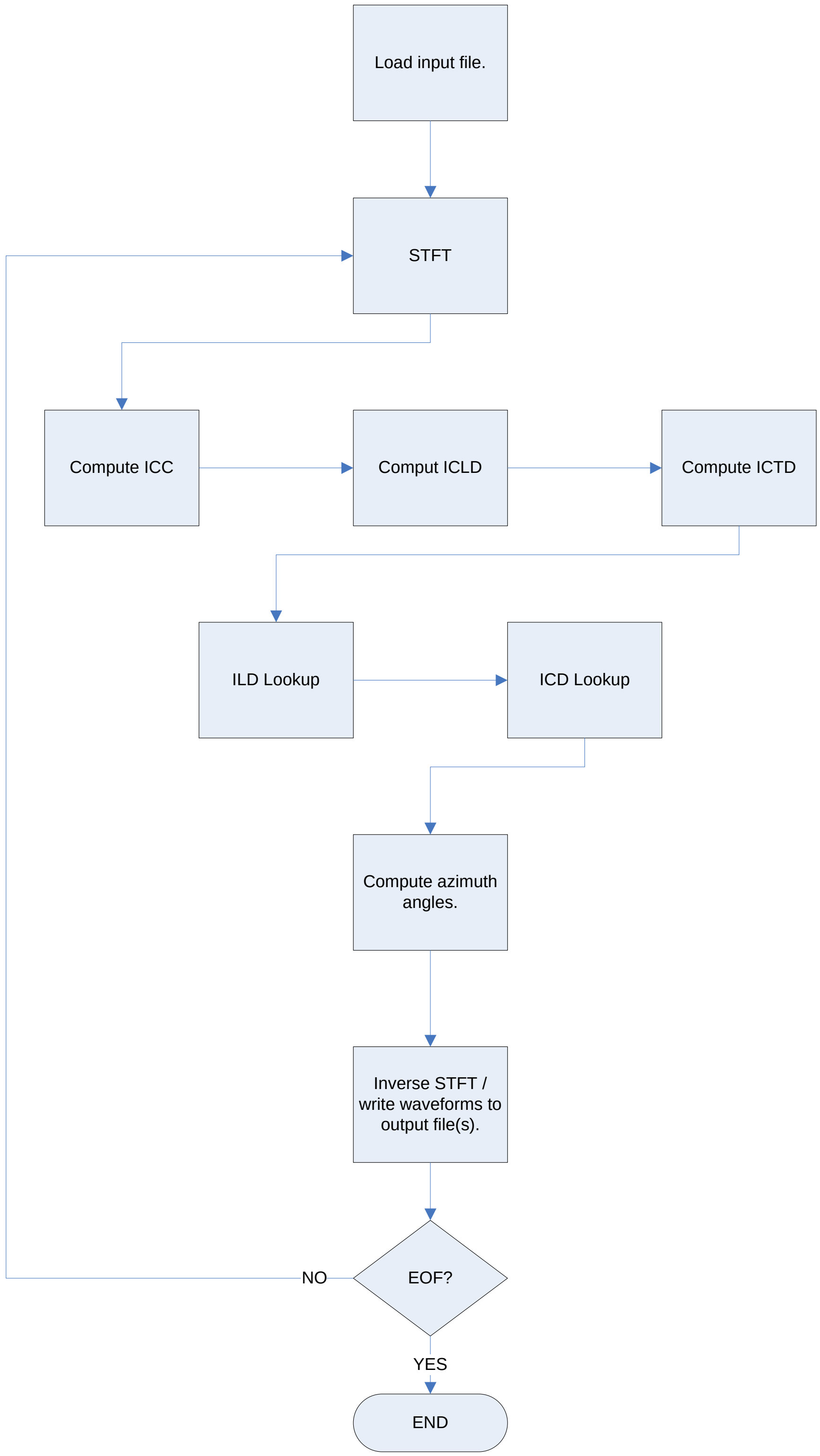

Figure 3 – System software flowchart.

Software Flowchart Subsystems

The overall software flowchart is shown as figure 3. Related subroutines are aligned together in horizontal levels. Descriptions of these levels follow:

1) Load input file

Input: Audio data in the form of .wav file.
Output: Matlab audio data variable to the workspace.
Description: In this step, the stereo signals are imported into the Matlab workspace to be processed.

2) Short-time Fourier transform (STFT)

Input: Audio data from Matlab workspace in time domain.
Output: Audio data to Matlab workspace in frequency domain.
Description: A short sample of audio data is taken (~1ms of audio) and converted to the frequency domain via an FFT. The 1ms STFT timeframe is chosen based of the speed of sound in air. In one millisecond, sound waves travel just over 13 inches; wider than the spacing between ears on a head. Thus, any sound arriving more that 1ms apart cannot have originated from the same source and is ignored.

3) Inter-channel information computed

Input: Frequency domain data of audio sample.
Output: ICLD, ICTD, ICC cues.
Description: From the frequency domain spectral coefficients, the three inter-channel cues are computed.

4) Head-related transfer function (HRTF) lookup

Input: Computed inter-channel cues.
Output: Separate azimuth angle estimates for ICTD and ICLD.
Description: Using HRTF lookup tables, the source azimuth angle is estimated for both the ICTD and ICLD.

5) Azimuth angle estimation calculation

Input: ICTD and ICLD azimuth angle estimates; ICC.
Output: Final azimuth angle estimate.
Description: Using the two intermediate azimuth angle estimates and the ICC, the final estimate of the azimuth angles is made.

6) I-STFT and file output

Input:	Final azimuth angle estimates.
Output:	N channels of audio data to the Matlab workspace, or an external file.
Description:	Using the azimuth angle estimation, the inverse STFT is performed on the spectral data, and the result is written to the appropriate file.

7) End of file detection and program termination.

Description:	The program checks for more audio data to be sampled. If it exists, the program returns to the STFT stage and begins again. If not, the program terminates.