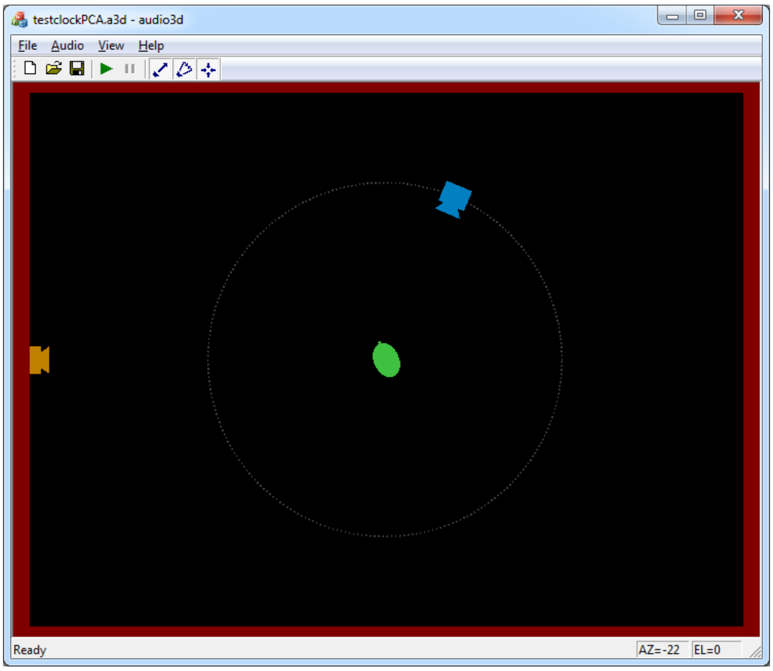
Audio3D – virtual audio space generator

Mark Huckvale
September 2015

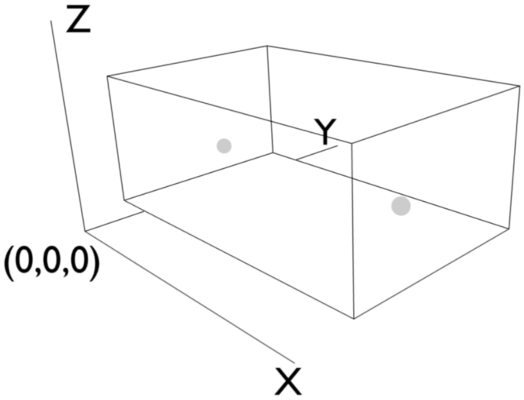


Introduction

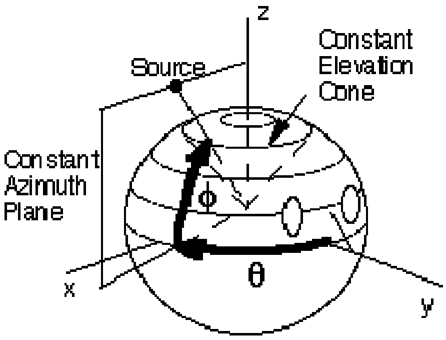
- Audio3D is a Windows program to generate virtual auditory spaces in 3 dimensions.
- Audio3D creates a virtual room containing a listener and one or more sources of sound which may be fixed or moveable.
- Audio3D generates a real-time binaural auditory image of the room and sound sources at the listener location which can be heard over headphones.
- Audio3D can be configured with HRTF data from different sources, supplied as HRIR.
- Audio3D separately computes the effect of direct path, early wall reflections and room reverberation on each of the sound sources.
- Audio3D supports the Zeiss HT-1 head tracker to change room orientation as the listener's head moves.
- Audio3D is optimised to run in real time on normal PC hardware.

Co-ordinate System

- Rooms are rectangular, specified by width, depth and height.
- The co-ordinate system is based in the lower left bottom corner of the room. X-axis goes across room width, Y-axis goes along room depth, Z-axis goes up room height.



- Angles are specified in vertical polar co-ordinates, with azimuth being a direction in the room horizontal plane between -180 and +180 degrees, with 0 degrees being straight ahead down the y-axis, while elevation being a tilt to the room horizontal plane between -90 (down) and +90 (up).



Vertical-Polar Coordinates

Head-Related Transfer Functions

HRTF data is taken from the CIPIC database [1]. This is a collection of head-related impulse responses measured from human subjects and the KEMAR manikin. Each HRIR consists of 200 samples at 44100 samples/sec for each of the left and right ears for each of 2500 spatial directions.

The CIPIC HR responses are processed to make them usable by Audio3D:

a. New directions are synthesized for azimuths of ±90° and elevations of 0° by averaging IRs at ±80° at all elevations.
b. The IRs are time-shifted to zero delay, using the delay values included in the database.
c. The set of IRs for each listener are scaled by a single factor so that the single largest IR in the set has sum-squared sample value of 1.
d. The set of IRs for each listener are decomposed using principal components analysis into a set of 16 base vectors (of 200 samples) and 16 weights per spatial direction.

Room reverberation modelling

Room reverberation is modelled using the image method [2]. Input is the size of the room, the reflection coefficients for walls and ceiling and the locations of a sound source and a listener. Output is an impulse response of a given maximum duration at a given sample rate.

The room impulse response generated for the room is then processed to remove the direct path and early reflections which are dealt with separately by Audio3D.

These are the paths removed from the room reverberation response. Assume the source is at (sx, sy, sz) and the room dimensions are W, D, H. Then the direct path and six early reflections originate at these co-ordinates:

0. Direct path is from (sx,sy,sz) weighted by 1
1. Wall 1 reflection from (-sx,sy,sz) weighted by wall reflection coefficient
2. Wall 2 reflection from (sx,-sy,sz) weighted by wall reflection coefficient
3. Wall 3 reflection from (2W-sx,sy,sz) weighted by wall reflection coefficient
4. Wall 4 reflection from (sx,2D-sy,sz) weighted by wall reflection coefficient
5. Floor reflection from (sx,sy,-sz) weighted by floor reflection coefficient
6. Ceiling reflection from (sx,sy,2H-z) weighted by ceiling reflection coefficient

HRTF implementation

Audio3D has two methods of operation: conventional and PCA. In conventional mode, all sound sources are processed separately, with each direct path of the source to the listener and each early reflection treated as independent signals. This approach can lead to significant amounts of computation and is only suitable for one or two sound sources. In PCA mode, the sound sources are first combined in terms of their effect on a set of principal IR vectors, and then convolution is only applied once per PCA vector. This approach scales well with increasing numbers of sound sources and is required when using more than 2 sources.

We give a brief mathematical justification for the PCA approach below, more details can be found in [3].

Assume we have N noise sources S_1..S_N. Each generates a direct path and six early reflections: s_{10}..s_{16}, s_{20}..s_{26}, etc. Each signal path is weighted by the reflection coefficients: w_{10}..w_{16}, w_{20}..w_{26}, etc. Each of these is associated with a direction to the listener, and hence to a set of HR impulse responses: H_{10}..H_{16}, H_{20}..H_{26}, etc (actually one of these per ear).

The required signal (y) to be input to one ear of the listener is then

$$y = \sum_{i=1}^{N} \sum_{j=0}^{6} w_{ij} s_{ij} * H_{ij}$$

This requires 7N convolutions.

Assume we decompose the set of HR impulse responses using PCA, so that any H_i can now be written:

$$H_i = \sum_{k=1}^{P} p_{ik} h_k$$

Where P is the number of components kept, {h} are the PCA basis vectors, and $\{p_i\}$ are the PCA weights for one spatial direction.

The full convolution sum for the required signal y is now:

$$y = \sum_{i=1}^{N} \sum_{j=0}^{6} w_{ij} s_{ij} * \sum_{k=1}^{P} p_{ijk} h_k$$

Which may be re-arranged into:

$$y = \sum_{k=1}^{P} \left(\sum_{i=1}^{N} \sum_{j=0}^{6} w_{ij} s_{ij} p_{ijk} \right) * h_k$$

Which requires only P convolutions regardless of N, and is preferred when P < 7N.

One thing missing from this presentation is the different arrival times at the listener of the direct path and early reflections of each source. This can be accommodated by introducing different delays at the signal source for each of the seven paths.

Head Tracker

The program supports the Zeiss HT-1 Head Tracker

The HT-1 head tracker is a USB device that sends real-time orientation data of the head tracker pod. The head tracker pod can be attached to the listener's headphones to input the pitch (elevation) and yaw (azimuth) movements of the listener's head into the generator.

The head tracker sends orientation data in quaternion format (qw,qx,qy,qz) . These are converted to yaw and pitch using

```
                    pitch = asin(2*qw*qy - 2*qz*qx)
           yaw = atan2(2*qw*qz+2*qx*qy , 1 - 2*qy*qy - 2*qz*qz)
```

Configuration File

- The configuration file is a text file with file extension "a3d".
- Comment lines must be prefixed with '#'.
- Sections in the file are marked with '[*name*]'
- Attribute value pairs are indicated as 'attribute=value', with one attribute per line.
- Currently these sections are supported:
  ```
  [config]
  [listener]
  [source]
  [noise]
  ```
- The [config] section describes the basic parameters of the auditory space, and supports the following attributes:
  - **room**: sets the size of the room (width, depth, height) in metres. Default:
    `room= 4 3 2.5`
  - **reflect:** sets the wall, floor and ceiling reflection coefficients. Default:
    `reflect= 0.9 0.7 0.7`
  - **srate**: sets the operational sampling rate (samples/sec). Note all impulse responses must be supplied at this rate. Default:

    `srate= 44100`
  - **level**: sets the reference level in dB for the audio mixer for RMS of full wave sine. This is also the maximum level before distortion. Default:
    `level= 80`
  - **reverb**: sets duration of the reverberation impulse response in seconds. Maximum 1 second. Default 0.05 seconds.
    `reverb=0.05`
- The [listener] section describes the characteristic of the listener and supports the following attributes
  - **position**: sets the position of the listener (x, y, z) in metres. Default is centre of room. The position of the listener is currently fixed.
    `position= 2 1.5 1.25`
  - **facing**: sets the direction that the listener is facing. The orientation of the listener can be manipulated using keyboard commands or the head tracker. Two values specify azimuth and elevation in degrees. Three values specify a location in space in metres. Default:
    `facing= 0 0`
  - **hrir**: gives filename of head-related impulse responses in either full format (.hrir) or PCA component format (.hrpca). This value can also be supplied/overridden by a menu option. Default none.
    `hrir= subject_003.hrir`
  - **npca**: sets the maximum number of components to use for PCA coded HRIR. Zero means use all available components. Default:
    `npca= 0`
  - **equalisation**: gives filename of an impulse response of an equalisation filter for the operators headphones. Text file contains float sequence. Default none. [NOT YET IMPLEMENTED]
    `equalisation= headphone.ir`
- The [source] section describes the characteristics of the source audio and supports the following attributes
  - **audio**: gives name of audio file containing source signal. Will be changed to single channel at mixer sampling rate if required. Relative filenames are resolved with respect to the folder containing the configuration file. This value can also be supplied/overridden by a menu option.
    `audio= filename.wav`
  - **level**: specifies the presentation level of the source as if measured at 1m (dB).
    `level= 65`
  - **position**: sets the (initial) position of the source (x, y, z) in metres.
    `position= 2 2.5 1.25`
  - **loop**: plays the audio in a continuous loop. Values: on/off. Default: on.
    `loop= on`
  - **circuit**: allows the source to move in a circuit around the listener, controlled by the mouse wheel. The circuit is either of constant elevation (horizontal) or constant azimuth (vertical). Values: horizontal/vertical/off. Default: off.
    `circuit=horizontal`
- Each [noise] section describes the characteristics of a single noise audio source and supports the following attributes
  - **audio**: gives name of audio file containing noise signal. Will be changed to single channel at mixer sampling rate if required. Relative filenames are resolved with respect to the folder containing the configuration file.

```
audio= filename.wav
```

- **level**: specifies the presentation level of the noise as if measured at 1m (dB).
  ```
  level= 65
  ```
- **position**: sets the position of the noise source (x, y, z) in metres.
  ```
  position= 2 2.5 1.25
  ```

Example Configuration file:

```
# audio3d config file
#
# movable speech source plus fixed mechanical clock
#
[config]
room=4 3 2.5
reflect=0.5 0.5 0.6
level=80
#
[listener]
position= 2 1.5 1.25
facing= 2 2.5 1.25
hrir=subject_003.hrpca
npca=8
#
[source]
position= 2,2.5,1.25
circuit=1
audio=six44100.wav
loop=1
level=65
#
[noise]
position= 0,1.5,1.25
audio=130388__olver__clock-ticking.wav
loop=1
level=45
```

Program Operation

Open Configuration

Opens a new configuration, replacing current

Open HRIR

Opens a new HRIR source file, replacing any specified in config file

Open Audio

Opens a new audio file to replace any audio file specified in the [source] section of the config file.

Save Audio

Saves the generated audio as stereo file [NOT YET IMPLEMENTED].

Audio Play

Starts replay of all sound sources

Audio Pause

Pauses replay of all sound sources

Audio Direct Path

Toggles inclusion of direct path in synthesis

Audio Reflections

Toggles inclusion of early reflections in synthesis

Audio Reverberation

Toggles inclusion of room reverberation in synthesis

View Head Tracker

Toggles on and off operation of the head tracker (if connected).

References

[1] http://interface.cipic.ucdavis.edu/sound/hrtf.html

[2] J. Allen and D. Berkley, Image method for efficiently simulating small-room acoustics, Journal of the Acoustical Society of America, vol. 65(4), pp. 943-950, April 1979.

[3] C.Y. Zhang, B. Xie, Platform for dynamic auditory environment real-time rendering system, Chinese Science Bulletin, 58 (2013) 316-327.