DESC9115 Digital Audio Systems - Lab Report 2

Matlab model for headphone based 3D sound from KEMAR based HRIR’s

Goal

To create a Matlab based operation for 3D sound in headphones. Drawing on HRIR data from a KEMAR manikin, the operation will have user defined input, and angles of elevation (vertical) and azimuth (horizontal).

Background

The Head Related Transfer Function (HRTF) is the processing that any signal goes through as it travels through space towards the body and into the ear canal. Countless studies have been done on the different processes involved to determine the cues involved in us characterizing the directivity of sound source. Many of these studies are based around the utilization of pre-measured HRTF’s or the derived impulse response of these functions, which is referred to as the Head Related Impulse Response (HRIR). This data can be achieved using a human subject and measuring the sound as it arrives into their ear, but more often than not a dummy is used to create these measurements. These impulse responses can then be used to filter audio samples to recreate the environment that those HRIR’s were recorded.

The chosen dataset for this function was recorded from Massachusetts Institute of Technology (MIT) and published by Bill Gardner and Keith Martin. The data was recorded by using a KEMAR dummy placed in MIT’s anechoic chamber and mounted on a motorized turntable. The sound source was then mounted on a boom and placed 1.4m from the interaural axis. By then rotating the KEMAR on the turntable to the desired azimuth angles and lifting the boom to reach the desired elevation angles measurements were then taken one by one. A total of 710 measurements and placements were recorded in total. (Gardner & Martin, 1994)

Implementation

The function hrtfmodel.m draws upon raw data attained from the KEMAR measurements. The provided folder ‘*HRIRData*’ contains the recorded data at elevations ranging in 10 degree intervals from -40 through to 90 degrees, each sorted into respective folders. Within these folders then features data for 360 degrees on the horizontal plane, or the azimuth, although each of the elevations have varying degrees of azimuth due to testing capabilities. Table 1 represents the azimuth increments, and number of measurements of each of the elevations.

Elevation	Number of Measurements	Azimuth Increment
-40	56	6.43
-30	60	6.00
-20	72	5.00
-10	72	5.00
0	72	5.00
10	72	5.00
20	72	5.00
30	60	6.00
40	56	6.43
50	45	8.00
60	36	10.00
70	24	15.00
80	12	30.00
90	1	x.xx

Table 1: Number of measurements and azimuth increment at each elevation. (Gardner & Martin, 1994)

The first section of the Matlab code specifies the specific data file to be called depending on the azimuth and elevation variables. The Matlab internal function of num2str has been used to convert the user defined variables of *Azi* and *Ele* into numbers to suit the file name.
“If” statements have then been used for the housekeeping of the conversion from the users input variable into the way the file is expecting to see the data.

```
filenameL = strcat('./HRIRData/elev',num2str(Ele),'/L',num2str(Ele),'e',num2str
(Azi),'a.dat');

if Azi == 0
   filenameL = strcat('./HRIRData/elev',num2str(Ele),'/L',num2str
(Ele),'e000a.dat');

elseif Azi > 0 && Azi < 10;
    filenameL = strcat('./HRIRData/elev',num2str(Ele),'/L',num2str
(Ele),'e00',num2str(Azi),'a.dat');
elseif Azi >= 10 && Azi < 99;
     filenameL = strcat('./HRIRData/elev',num2str(Ele),'/L',num2str
(Ele),'e0',num2str(Azi),'a.dat');
else
    filenameL = strcat('./HRIRData/elev',num2str(Ele),'/L',num2str
(Ele),'e',num2str(Azi),'a.dat');
end
```

After the data file is specified, the function then reads the required HRIR data from the file. This part of the code was provided by Gardner and Martin alongside their database to help extract the data into Matlab. It uses the fopen function, followed by fread to open the file, reads the data from the file, then closes the file again. The code used looks like that seen below, through the FAQ page associated with the MIT media lab data page. (MIT, 1994)

```
fpL = fopen(filenameL,'r','ieee-be');
    dataL = fread(fpL, 256, 'short');
    fclose(fpL);

    leftimp = dataL(1:2:256);
```

This data is then used to filter the user defined *input* variable. To do this, the function uses the simple *filter* function which filters the input variable through the impulse response. The resulting signal is then divided by the absolute maximum of itself to prevent any clipping.

```
left = filter(leftimp,1,input);
left = left./max(abs(left));
```

The process is done for both the left and right channels of the final signal, and summed into a final output stereo array.
Once the outputs from the left and right have been summed the function then uses the inbuilt *wavwrite* function to produces a wav file of the output. Seen below, the file outputs a copy of the output, at the sample rate (44100) at 32 bit audio. 32-bit was chosen because any lower bit rate produced errors or the “data clipped” warning which was due to quantization error. The strcat and num2str functions are then used again, but this time to label the output file based on the original Azi and Ele values.

```
wavwrite(output, fs, 32, strcat('HRTFOutput',num2str(Ele),'e',num2str
(Azi),'a'));
```

Following the creation of the output file, the function plays back the output using the inbuilt *sound* function.

```
sound(output, fs);
```

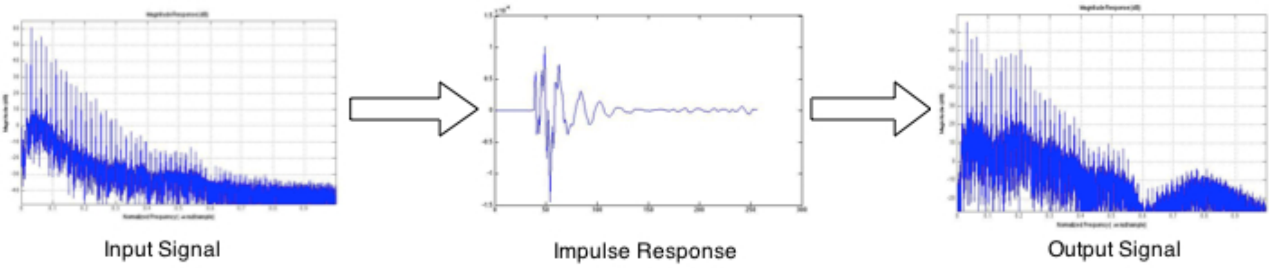

Diagram 1: *Trombone.wav* passing through the 90 degree impulse response in the contralateral (left) ear, and the resulting combination signal of the output.

The above diagram visually represents how the input signal of the trombone.wav file is effected by the impulse response. Showing just how the HRIR manipulates the audio signal to produce what we perceive to be a directivity change.

<u>Results</u>

As with any use of a HRTF based on another persons ear shape, it won't always sound correct. For instance for my own ear, a 10 degree elevation change was needed for me to perceive sound to be at ear level. These sorts of inconsistencies are expected when working with a set of HRIR's that are not based on the individuals measurements. Apart from movement detail in the higher elevations, and around the cone of confusion, the function operations as expected, with clear perceived movement around the head.

The major drawback of this particular data set was that the frequency content dramatically decreases below the 300Hz range which for many audio samples, such as speech, means it alters the original sound source so dramatically it no longer sounds like a natural recreation of the file. But despite the content missing content, basic directivity is not effected.

For further production of this function, user based variables could be included in order to customize the outputting signal to be more accurate. A more extensive database could also be utilized where multiple pinna's have been measured to provide a different option for the listen.

A sample calling script has been provided where an input and elevation have been set, then the function is called multiple times as the input moves from left to right across the azimuth plane, with the pause function used as it moves around.

Bibliography

Brown and Duda, 1998 C.P. Brown and R.O. Duda, Sept. 1998, A structural model for binaural sound syn- thesis. *IEEE Trans. Speech and Audio Processing,* 6(5):476-488, accessed at <*http://www.zaeron.com/projet7/IMG/pdf/Brown_Duda98.pdf*>

CIPIC, 2002, The CIPIC HRTF Database, accessed online at: <http://interface.cipic.ucdavis.edu/sound/hrtf.html>

George & Martin, 1994, HRTF Measurements of a KEMAR Dummy-Head Microphone
MIT Media Lab Perceptual Computing - Technical Report #280, accessed online at: <http://sound.media.mit.edu/resources/KEMAR/hrtfdoc.txt>

Mathworks, 2013, document database - wavwrite, acessed online at: <http://www.mathworks.com.au/help/matlab/ref/wavwrite.html>

MIT Media Lab, 2000, HRTF Measurements of a KEMAR Dummy-Head Microphone, accessed online at: <http://sound.media.mit.edu/resources/KEMAR.html>

Week 6 Class Tutorials, 2013, Week 6 – Head Related Transfer Functions and Impulse Responses
Provided by Luis Alejandro Miranda Jofre

Week 9 Class Tutorial, 2013, Week 9 – System identification by digital filter approximation
Provided by Luis Alejandro Miranda Jofre

Zolzer, U. 2002, DAFX: Digital Audio Effects Chapter 6: Spatial Effects, John Wiley & Sons, Ltd, New York, NY