HEAD RELATED IMPULSE RESPONSE INTERPOLATION AND EXTRAPOLATION USING DEEP BELIEF NETWORKS

Grady Kestler[1], Shahrokh Yadegari[2], and David Nahamoo[3]

Qualcomm Institute at UC San Diego[1,2], Pryon[3]
Department of Electrical Engineering[1], Department of Music[2]
{gkestler[1], sdy[2]}@ucsd.edu, davidnahamoo@pryoninc.com[3]

ABSTRACT

This paper presents a machine learning Deep Belief Network technique for interpolation and extrapolation of HRTF (Head Related Transfer Function) databases. In this technique, we decouple the stereo pair of HRTFs for each ear. Furthermore, we remove the inter-aural time differences (ITD) and distance attenuation from the recorded HRTF measurements such that the DBNs only interpolate and extrapolate the spectrum filtering of the audio for the two ears. Testing on the CIPIC and SCUT databases produces results of an average log spectral distortion less than 3 dB for all points around the head.

Index Terms— HRTF Interpolation, Deep Belief Networks, Binaural Audio

1. INTRODUCTION

One of the most common forms of synthesizing spatial audio on headphones is to use Head Related Transfer Functions (HRTF), usually measured around the head of a subject. These are stored as a stereo pair (left and right ear) of time domain FIR filters known as a Head Related Impulse Responses (HRIRs). Selecting a left and right HRIR pair at a specific location, we can synthesize binaural audio with the impression that the sound is emanating from the chosen location. In order to recreate a spatial audio impression at any arbitrary location, we can only expect to interpolate or extrapolate the necessary HRIR pair from an existing set of measured HRIRs at regularly spaced discrete locations.

An individual HRIR encodes spectrum filtering due to anthropometry of a subject as well as travel time and distance attenuation dependent on the distance from the source to the corresponding ear. A pair of HRIRs represents relative information as ITD - the inter-aural time differences and ILD - inter-aural level differences. All of these are related to the desired perceived location of the sound. The interpolation technique proposed in this paper is designed to work in conjunction with a ray-tracing algorithm for audio spatialization called Space3d [1, 2, 3]. This algorithm calculates the ITD and distance attenuation for each ear separately and uses the HRTFs only for the spectrum filtering of the audio. Thus, we first remove the ITD and distance attenuation from the recorded HRTFs and train our neural network to learn the resulting filters for each desired location. Then the predicted results can be used as binaural filters for spatial synthesis using Space3d.

In section 2, we discuss the preprocessing techniques used to prepare the HRTF data for the network. Section 3 describes the training methods and architecture of our Deep Belief Network [4]. Lastly, sections 4 and 5 discuss our results in terms of the log spectral distortion of predicted HRTFs and possible avenues to pursue in the future.

2. DATA

The two datasets used in these experiments were the CIPIC database [5] and the publicly available SCUT database [6]. The CIPIC database consists of 35 subjects with anthropometric data and 1250×128 HRIRs measured at various positions around the head at 1.0 meter distance from the subject. The SCUT database consists of a single subject, with 10 spheres of 493×256 HRIRs meaured at varying distances from .2 to 1.0 meters. We used the SCUT database to test the networks ability to extrapolate over the distance parameter. The preprocessing steps were applied equally to both databases.

2.1. Magnitude Response

From [7], it is possible to reconstruct the minimum phase response of the HRTF based solely on the magnitude response. We chose to learn only the magnitude of the HRTFs to avoid dealing with discontinuities which are introduced by unwrapping the phase, especially at the higher end of the spectrum as we get close to Nyquist.

It is best to do the measurement of the HRIRs in anechoic conditions; however, that is not usually possible and HRIRs can contain reflections from the physical environment in which the measurement is done. The presence of these reflections in the HRIRs will not only make the synthesis results less accurate, but will also complicate the learning process for the network as the two physical effects (the effect of the head, and the effect of reflections) will be combined.

Based on our subjective listening tests, we found that the first 64 samples of the HRIRs, after accounting for the travel time from the speaker to the ear, contained the majority of the information needed for creating spatial impressions. Truncating the HRIRs to 64 samples before taking a 64 point FFT allowed us to remove many of the reflections. Since real data has a symmetric fourier domain representation, we used the 32 positive frequency bins and converted the results to decibels; $20\log_{10}|HRTF|$.

In order to account for the travel time of the sound, we traced the sound ray along its shortest path from the speaker to the ear. Modelling the head as a sphere, we assume that once the audio reaches the head, it travels along its surface. Using this distance, the speed of sound, and the sampling rate, we were able to remove an appropriate number of samples from the beginning of each HRIR. A result of this technique is the removal of ITD from HRIR pairs. The time delay, and thus ITD, is re-introduced during the Space3d synthesis in order to maintain accurate spatial impressions.

Additionally, sound pressure level is attenuated by a factor of $\frac{1}{d}$ where d is the distance from the ear to the sound source. Thus, we perform an inverse scaling by d during preprocessing. Like ITD, the effect of the distance attenuation is later introduced back in the synthesis process based on the location of the virtual audio source being synthesized.

2.2. Position and Anthropometric

Each of the 1250 positions in the CIPIC HRTF database is a 3×1 spherical coordinate vector. However, this introduced a number of discontinuities at 0° and 360° in the azimuthal direction leading us to use cartesian coordinates instead. The anthropometric data for a single subject from CIPIC is provided as a 17×1 dimensional vector of head and torso measurements and a 10×2 dimensional vector of ear measurements; both left and right ears. When using the SCUT database and testing the extrapolation capability of the system, the network was not provided with any anthropometric data.

3. NETWORK

3.1. Data management

The input to our network consists of a single 3×1 Cartesian coordinate vector, the 17×1 head and torso measurements, and the 10×2 left and right ear measurements from [5]. The output of the network is a 32×1 log magnitude response. When training the network, we remove 10% of the data as test data and 20% of the remaining data for validation. Because the structure of our network is divided into multiple sub-networks, different components of the input data are used at different levels. For example, since the left ear measurements should not contribute to the HRTF of the right ear but the head measurements effect both, some sub-networks have layers to which the input was position and head measurements only, later splitting into two separate paths which took ear measurements as auxiliary input before outputting either the left or the right ear's HRTF. The HRTFs data from the CIPIC database consists of 1250×32 dimensional data which we preprocessed following the description above. Each of the 1250 rows corresponds to a different position. The same removal of 10% and 20% for testing and validation were applied.

Because 1250×32 dimensional training data is not necessarily large for deep neural network training, we introduce a mixing scheme between the testing and validation data to increase performance but avoid over-training. For a single *iteration*, i, the training and validation data are set for a number of epochs, n_e. Once the network is trained for n_e epochs, on the next *iteration*, $i+1$, a new 20% of the data is removed for validation. The network continued to train like this for a set number of iterations. After experimenting with the balance of epochs and iterations, we found that after training for $n_e = 20$ epochs and $n_i = 20$ iterations for a total of 400 cycles, the network performed well, but was not over trained.

3.2. Architecture

The network architecture shown in Figure 1, applies only to the left ear. However, the right ear network is identical and in fact, most of the sub-networks are shared, outputting predictions for both the left and right ears. That being said, for the $MagL_{\bar{M},\sigma_M}$ network, an equivalent $MagR_{\bar{M},\sigma_M}$ was developed for the right ear with identical architecture.

Within the full architecture, we introduce a number of sub-networks utilizing the physics of HRTFs to accommodate the small amount of data we have. The three types of sub-networks we designed were TypeA networks in red, TypeB networks in blue, and a TypeC network in green. The TypeA networks ($Mag_n, Real_n, Imag_n$), take as input a 20×1 vector for the position and head information and later a 10×2 vector of the ear measurements for the left and right ears. The subscript, $_n$, denotes the cost function produces a zero mean, unit variance prediction of the HRTF (see Section 3.3). The TypeB networks, $Real_{\bar{R},\sigma_R}$ and $Imag_{\bar{I},\sigma_I}$, predict the mean value and standard deviation of the real or imaginary part of the HRTF ($\bar{R}, \sigma_R$ or $\bar{I}, \sigma_I$). These predicted values are used to un-normalize the prediction from $Real_n$ and $Imag_n$ in order to generate the magnitude using real and imaginary values, $MagRI$. From here, the normalized predictions of Mag_n and $MagRI_n$, along with the position, head, and ear measurements, are passed to a mixing network, $Mag2_n$, part of which is structured identically to Mag_n and initialized to the same weights as Mag_n.

At this point, we have three separate, normalized predictions of the HRTF magnitude; Mag_n, $MagRI_n$, and $Mag2_n$. In order to un-normalize them, we generate two separated left and right networks, $MagL$ and $MagR$, whose purpose is to

predict the mean and standard deviation of the un-normalized magnitude response $(\bar{M}, \sigma_M)$. The un-normalized Mag, $MagRI$, and $Mag2$, are passed to a 96×32 mixing matrix, W, initialized to

$$W = \begin{bmatrix} \frac{1}{3}, & 0, & \dots, & \frac{1}{3}, & 0, & \dots, & \frac{1}{3}, & 0, & \dots \\ 0, & \frac{1}{3}, & \dots, & 0, & \frac{1}{3}, & \dots, & 0, & \frac{1}{3}, & \dots \\ \vdots, & \vdots, & \ddots, & \vdots, & \vdots, & \ddots, & \vdots, & \vdots, & \ddots \end{bmatrix}$$

producing an average of the three networks. This averaged result is then passed to $MagFinal$ whose layers are initialized to an identity matrix based on the presumption that the average of Mag, $MagRI$, and $Mag2$ would be close to the true output.

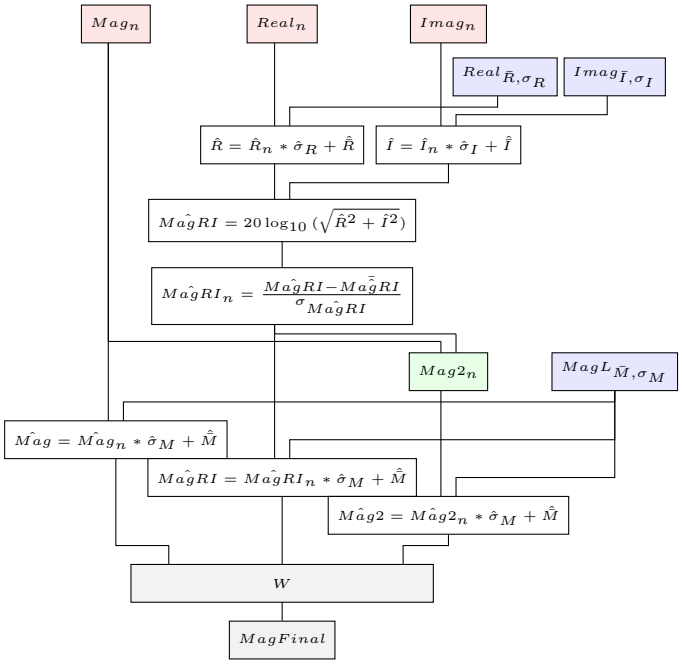


Fig. 1. TypeA and TypeC networks predict zero mean, unit variance normalized functions. TypeB networks predict the mean and standard deviation of the raw value to be used for un-normalization of the TypeA and TypeC predictions later. The white blocks indicate mathematical operations that were not trained in the network. The mixing matrix W and the $MagFinal$ network produce the final magnitude output.

3.3. Objective Functions

Using standard notation, let the prediction from the network be denoted $\hat{y}$ and the true data be y. For TypeA and TypeC networks, the objective function is given by

$$mse(y_{norm}, \hat{y}_{norm}) = \sum_{i=0}^{31} (y_{norm}[i] - \hat{y}_{norm}[i])^2$$

where

$$y_{norm} = \frac{y - \bar{y}}{\sigma_y} \qquad \hat{y}_{norm} = \frac{\hat{y} - \hat{\bar{y}}}{\sigma_{\hat{y}}}$$

and $(\bar{y}, \sigma_y)$ denote the sample mean and standard deviation.

TypeB networks used $mse(\bar{y}, \hat{\bar{y}})$ and $mse(\sigma_y, \hat{\sigma_y})$ and the training of $MagFinal$ used a cost function given by

$$mse(y, \hat{y}) + mse(\bar{y}, \hat{\bar{y}}) + mse(\sigma_y, \sigma_{\hat{y}})$$

4. RESULTS

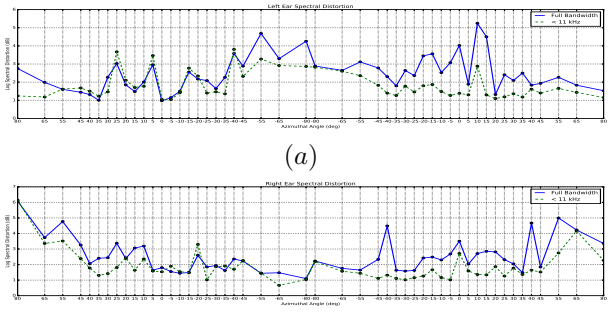
The log spectral distortion (LSD) is a common metric used to compare generated HRTFs to true HRTFs [8, 9, 10], and the errors in HRTFs with spectral distortion around 4 dB are often imperceivable in listening tests [11, 12, 13]. Because our network predicts log data, the $mse(y, \hat{y})$ is the same as $LSD(y, \hat{y})$ up to a scaling factor. Let $x = 20\log_{10} y$ be the output of the network, then

$$\begin{aligned} mse(x, \hat{x}) &= \sum_{i=0}^{N-1} (20\log_{10} y[i] - 20\log_{10} \hat{y}[i])^2 \\ &= \sum_{i=0}^{N-1} (20\log_{10} \frac{y[i]}{\hat{y}[i]})^2 \\ LSD(y, \hat{y}) &= \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (20\log_{10} \frac{y[i]}{\hat{y}[i]})^2} \end{aligned}$$

where $N = 32$ is the number of FFT bins used.

Our results are presented in three separate experiments comparing the LSD at various positions of the data. The first experiment predicts HRTFs at 0° elevation around the head. The second compares HRTFs at all locations around the head, and the third attempts to extrapolate HRTFs at different distances. The results of the first two experiments are presented using the CIPIC database subject 003, with similar results for alternative subjects. Due to the limited radial distances in the CIPIC database, the third experiment was performed on data from the SCUT database which has measurements at distances of $1.0, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.25$, and 0.2 meters. For experiments 1 and 2, the network was trained on 800 of the 1250 data points, validated on 225 data points, and tested on 125 data points. The plots shown consider all 1250 data points. Specific results for the test data are mentioned in the text.

Figure 2 illustrates our results at 0° elevation around the head from the left ear to the right ear in front of the head followed from the right ear to the left ear in the back of the head. We show the LSD below 11 kHz to illustrate the networks ability to predict within a more meaningful audible range. The erratic peaks that appear in the full bandwidth plot, but not in the narrow bandwidth plot (< 11 kHz), can be

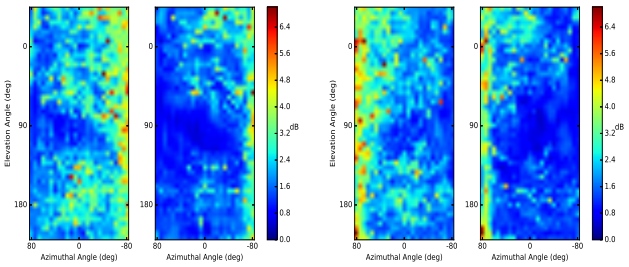


(a)

(b)

Fig. 2. The plots correspond to the $0°$ elevation ring tranversing from the left ear in front of the head from $80°$ to $-80°$ where $0°$ corresponds to directly in front of the head. The second set of azimuthal angles happens from traversing from the right ear to the left ear behind the head $(180°)$ elevation). Blue solid line indicates LSD over the full spectrum. Green dotted line indicates LSD up to 11 kHz. (a) Left ear (b) Right ear

attributed to errors in the high frequency components often at, or around, the nyquist frequency. Of the points plotted, positions at $0°$ and $-80°$ were removed as two of the 125 test data points that the network had never seen. Still, the network is able to interpolate these locations relative to the surrounding locations with comparable LSD.

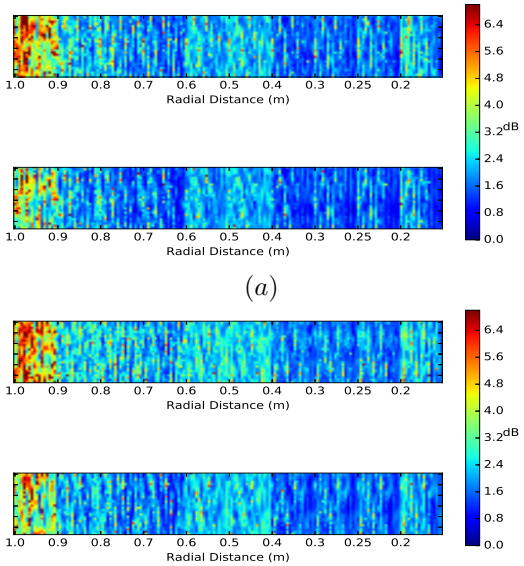


(a) (b)

Fig. 3. (a) Left ear. (b) Right ear. LSD for $(a, right)$, $(b, right)$ full bandwidth and $(a, left)$, $(b, left)$ up to 11 kHz.

Figure 3 expands the results of $0°$ elevation to all elevations for the left and right ear. It is clear that our network predicts best at positions above our head, but does comparably well at all positions, not only at the $0°$ elevation. The average LSD for all 1250 data points for the left ear over all positions for both the full bandwidth and half bandwidth LSD are $2.48, 1.73$ dB respectively and $2.35, 1.73$ dB for the right ear. On the 125 points the network was not trained with, these numbers were $2.76, 1.96$ dB (left) and $2.42, 1.77$ dB (right).

The extrapolation experiment was performed on the SCUT database by training the network on all data points between 0.9 and 0.25 meters in the hopes of extrapolating the HRTFs at 1.0 meters and 0.2 meters for all positions around the head. Figure 4 shows the LSD for each distance. From these results, it is clear that the network is able to extrapolate HRTFs at the closer position, but has difficulty extrapolating at further distances. As we move farther from the microphones, reflections from the torso can be more easily mixed with reflections from nearby surfaces [14]. Additionally, the signal-to-noise ratio at further distances for recording audio is decreased which increases the error from the predictions to the actual HRTFs.



(a)

(b)

Fig. 4. (a) Left ear. (b) Right ear. LSD for (top) full bandwidth and $(bottom)$ up to 11 kHz. The average LSD for each radial position are (a, top) $4.81, 2.58, 2.27, 1.94, 2.31, 2.39, 1.68, 1.75, 1.82, 2.26$ (a, bot) $3.71, 2.21, 1.88, 1.62, 2.12, 2.19, 1.50, 1.56, 1.58, 1.78$ (b, top) $5.01, 2.66, 2.52, 2.24, 2.59, 2.56, 1.88, 1.86, 1.92, 2.41$ (b, bot) $4.07, 2.26, 2.08, 1.82, 2.36, 2.39, 1.63, 1.69, 1.74, 2.08$

5. CONCLUSION

In this paper, we presented a neural network based technique for interpolation and extrapolation of HRTF databases. We presented results on two different databases; CIPIC and SCUT. We demonstrated that these neural networks can interpolate the HRTF at all locations around the head with an average spectral distortion less the 3 dB. In addition, the technique can also extrapolate near-field HRTFs with low spectral distortion. The introduced methodology holds promise for predicting individualized HRTFs, i.e. for any anthropometric data, when a larger dataset with more subjects becomes available.

6. REFERENCES

[1] F Richard Moore, "A general model for spatial processing of sounds," *Computer Music Journal*, vol. 7, no. 3, pp. 6–15, 1983.

[2] Shahrokh Yadegari, F Richard Moore, Harry D Castle, Anthony Burr, and Ted Apel, "Real-time implementation of a general model for spatial processing of sounds.," in *ICMC*, 2002.

[3] Shahrokh Yadegari, "Inner room extension of a general model for spatial processing of sounds.," in *ICMC*, 2005.

[4] Geoffrey E Hinton and Ruslan R Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[5] V Ralph Algazi, Richard O Duda, Dennis M Thompson, and Carlos Avendano, "The cipic hrtf database," in *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*. IEEE, 2001, pp. 99–102.

[6] BoSun Xie, XiaoLi Zhong, Dan Rao, and ZhiQiang Liang, "Head-related transfer function database and its analyses," *Science in China Series G: Physics, Mechanics and Astronomy*, vol. 50, no. 3, pp. 267–280, 2007.

[7] T Quatieri and Alan Oppenheim, "Iterative techniques for minimum phase signal reconstruction from phase or magnitude," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1187–1193, 1981.

[8] Piotr Bilinski, Jens Ahrens, Mark RP Thomas, Ivan J Tashev, and John C Platt, "Hrtf magnitude synthesis via sparse representation of anthropometric features," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4468–4472.

[9] Hongmei Hu, Lin Zhou, Hao Ma, and Zhenyang Wu, "Hrtf personalization based on artificial neural network in individual virtual auditory space," *Applied Acoustics*, vol. 69, no. 2, pp. 163–172, 2008.

[10] Lin Li and Qinghua Huang, "Hrtf personalization modeling based on rbf neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3707–3710.

[11] T Nishino, Y Nakai, K Takeda, and F Itakura, "Estimating head related transfer function using multiple regression analysis," *IEICE Trans. A*, vol. 84, pp. 260–268, 2001.

[12] Takanori Nishino, Naoya Inoue, Kazuya Takeda, and Fumitada Itakura, "Estimation of hrtfs on the horizontal plane using physical features," *Applied Acoustics*, vol. 68, no. 8, pp. 897–908, 2007.

[13] Hongmei Hu, Lin Zhou, Jie Zhang, Hao Ma, and Zhenyang Wu, "Head related transfer function personalization based on multiple regression analysis," in *Computational Intelligence and Security, 2006 International Conference on*. IEEE, 2006, vol. 2, pp. 1829–1832.

[14] Jens Blauert, *The technology of binaural listening*, Springer, 2013.