HRTF Individualization using Deep Learning

Riccardo Miccini
Aalborg University
rmicci18@student.aau.dk

***Abstract*—The research presented in this paper focuses on investigating HRTF individualization techniques using deep learning approaches, with an emphasis on HRTF estimation and synthesizing. (also mention FABIAN dataset, autoencoders, depthmaps etc, see slides)**

***Index Terms*—Spatial Audio; HRTF; Deep Learning**

I. Introduction

Virtual reality (VR) and augmented reality (AR) research has made substantial progress over the last decades, and virtual environments created using binaural sound rendering technologies find applications in a wide array of areas, such as aids for the visually impaired, tools for audio professionals, and VR-based entertainment systems.

These techniques are based on the application of a particular filter called *head-related transfer function* (HRTF), which colors a sound according to its location in the virtual environment. However, HRTFs derived from standard anthropometric pinnae, such as those in dummy heads, often results in localization errors and wrong spatial perception [1]. In fact, while generic HRTFs may successfully approximate the interaural time difference (ITD) and interaural level difference (ILD) cues which are used to perceive the horizontal direction of a sound source, the monaural cues needed to discern its vertical direction are highly dependent on the anthropometric characteristics of each ear.

In order to provide the most realistic and immersive experience possible, it is necessary for users to have their custom set of HRTFs measured, which can prove quite impractical due to the need for dedicated facilities and the overall invasiveness of the procedure. Recently, attempts have been performed at synthesizing or customizing HRTFs using various data from users such as anthropometric measurements, 3D scans, or perceptual feedback.

This paper investigates methods for generating individualized HRTFs, in particular using newly-developed deep learning algorithms, and further expands on the topic by documenting the replication attempts and experiments conducted as part of the research. In the next section, current relevant contributions to the field are introduced. The Methods section details the computational techniques used in selected works from the literature as well as in the research carried out as part of the seminar, with particular focus on deep learning methods. In Results, the applications and outcomes of the aforementioned techniques for replications and other experiments are discussed, with the purpose of assessing their effectiveness. Finally, closing remarks as well as pointers for future research are stated in the Conclusions section.

II. State of the art

Over the past decades, several strategies have been devised, in order to avoid the burden of conducting strenuous acoustical measurements with human subjects. In a recent review, Guezenoc [2] divides such alternative approaches into *numerical simulation*, *anthropometrics*-based, and *perceptual feedback*-based.

The former method consists in simulating the propagation of acoustic waves around the subject, using 3D scans; the most common simulation schemes include Fast-Multipole-accelerated Boundary Element Method (FM-BEM) [3] and Finite Difference Time Domain (FDTD) [4] for frequency and time domain respectively.

With the help of databases of publicly available HRTFs and machine learning techniques, anthropometric measurements can be used to choose, adapt, or estimate a subject's HRTF set. In 2010, Zeng [5] implements an hybrid model based on principal component analysis (PCA) and multiple linear regression, which uses anthropometric parameters to select the most suitable HRTF set for the given user. Similarly, user feedback on perceptual tests can be used to inform regression models for tasks such as those listed above.

In more recent times, there has been an interest in solving the aforementioned tasks using deep learning techniques. In 2017, Yao [6] uses anthropometric measurements to select the most suitable HRTF sets from a larger database. In 2018, Lee [7] develops a double-branched neural network that processes anthropometric data with a multi-layer perceptron and edge-detected pictures of the ear with convolutional layers, combining the outputs of the two into a third network to estimate HRTF sets. Again in 2017, Yamamoto [8] trains a variational autoencoder on HRTF data, and devises a perceptual calibration procedure to fine-tune the latent variable used as input by the generative part of the model. Finally, in 2019, Chen et al. [9] train an autoencoder to reconstruct HRTFs along the horizontal plane, and subsequently uses the resulting latent representations as targets for a multilayer perceptron which feeds on anthropometric data and azimuth angle, allowing users to synthesize new HRTFs using the MLP and decoder.

III. Methods

This section presents some of the most relevant computational methods found in the relevant literature on HRTF individualization. The aspects covered in the following subsections include the encoding of generated HRTFs, the extraction and choice of predictors, and the deep neural network architectures adopted.

A. HRTF representation

A single HRTF is defined as the the far-field frequency response of a given ear, measured from a point in the free field to a point in the ear canal [10]. An HRTF set is composed of the HRTFs of both left and right ears, measured at a fixed radius from the head, and across several elevations and azimuths. According to Kulkarni et al. [11], HRTFs specified as minimum-phase FIR filters have been empirically proved to be perceptually acceptable. Thus, HRTFs can be stripped of the ITD information and stored as real-valued log-magnitude response.

While this is the preferred way of storing, exchanging, and using HRTF sets, neural networks have different requirements that call for ad-hoc formats. In particular, Yamamoto [8] uses different representations for the input and output of his autoencoder. The input data format, which is dubbed *HRTF patch*, consists of a 4-dimensional tensor of shape $(5 \times 5 \times 128 \times 4)$. The first two dimension describe the HRTF under investigation and its neighbors along the elevation and azimuth directions, for a total of 25 HRTFs in each given patch. The remaining ones describe the content of each HRTF in the patch: the last dimension, also called *channel*, encodes frequency power spectrum or time-domain signal for either left or right ear, where 128 is their length. This data representation provides a substantial amount of contextual information, which can be learnt by 3D-convolutional layers.

The output of the autoencoder does not contain any neighbor HRTF, but instead of encoding the frequency power spectrum or time-domain information as a continuous signals, it uses a quantized format where each sample can have one of 256 possible discrete values that are then mapped to another dimension using one-hot encoding. The continuos signals can be reconstructed by taking the index of the value with highest magnitude and passing it to a µ-law algorithm. This strategy makes sure to retain some of the high-frequency details of the continuous signals, which are often lost when reconstructing data with autoencoders, and can be found in certain WaveNet implementations [12].

Further formats which have been investigated include mappings where HRTFs sharing the same azimuth or elevation are combined in a 2-dimensional image-like representation with either elevation or azimuth along one axis and frequency along the other; the color of each pixel would then represent the log-magnitude of the spectrum. The structure expressed by adjacent HRTFs could therefore be learnt using 2D-convolutional layers. However, the downside of combining data in this way is the reduction of available data points to use for training.

Finally, a compact representation for individual HRTFs has been proposed, consisting of the first N principal components of the HRTF. While it has been observed that as little as 10 components are enough to reconstruct the original HRTF with maximum $1dB$ of spectral distortion, the loadings of the PCA must be learned, and become an essential part of the representation.

B. User data extraction

A fundamental aspect of HRTF individualization is the kind of data used to personalize the frequency response. Most often, acquiring data about a subject is faster and less strenuous than collecting an entire HRTF set, as well as having looser requirements in terms of external conditions and tools. The kind of data that can be collected comprises anthropometric measurements, other anthropometric data, and perceptual feedback.

The CIPIC dataset [13] released in 2001 sets a convention for anthropometric data collection and reporting, which has been adopted by later datasets too [14]. Its format specifies 17 anthropometric parameters for head and torso, and 10 for the pinna. This data has the disadvantage of having loosely defined measurement points, which translate into systematic biases that make merging different datasets particularly prone to errors. Moreover, anthropometric features are only unique for each given subject and as such, may not have enough predictive power to be used for the regression of several HRTFs per subject. Spagnol et al. address this shortcoming by introducing elevation-dependent anthropometric measurements as predictors [15]. These features consists in the lengths of the segments spanning from the tragus to each of the three contours outlined by the helix and concha, and oriented according to the elevation angle under investigation.

Another source of useful predictors for regression and prediction tasks may be found in 3-dimensional representation of the subjects' pinnae and head. Recent HRTF datasets include digital scans of the heads and pinnae [14, p. @spagnol_viking_2019] stored as 3D models, which can be used for feature extraction. In particular, 2-dimensional projections such as digital renderings or depth maps can be conveniently processed in neural networks using convolutional and pooling layers. As mentioned in the following sections, convolutional autoencoders can be used to extract salient features from ear images, which can then be used as predictors.

Finally, perceptual feedback can be useful when collecting manual measurements or complex user data is not viable. Furthermore, this kind of data is more resilient to systematic errors and biases introduced by measurement procedures [2]. Perceptual feedback often consists in letting a user evaluate and rate the performance of a given HRTF set, and is most commonly adopted in HRTF selection or adaptation tasks. Nevertheless, Yamamoto uses perceptual feedback to navigate the latent space in order to synthesize suitable HRTFs for a given user [8]. It is worth noting how, while validating models based on anthropometric data is quite trivial, models using perceptual data requires a user study or the implementation of a virtual agent.

C. Autoencoder

Most conventional neural networks are employed to predict a target y from an input x in a supervised manner. On the other hand, autoencoders learn a compressed representation z of the input data x called *latent representation*, which is then used to generate a reconstructed version $\hat{x}$. The purpose of autoencoders is to learn useful features from the input data in an unsupervised manner [17].

Autoencoders usually consists of a feed-forward neural network composed of two sub-nets: an encoder network $f()$ and a decoder network $g()$ such that $g(f(x)) = g(z) = \hat{x}$. Training an autoencoder usually involves iteratively updating the weights and biases of the two networks through backpropagation, in order to minimize a cost function representing the mean squared error (MSE) between x and $\hat{x}$.

Over time, several variants of the autoencoder have been developed. Each variant extends on the conventional autoencoder architecture by promoting different properties of the latent space, thereby catering to different tasks such as denoising, classification, or — as it is this case here — generative applications. Two common generative models based on the autoencoder are described below.

1) Variational autoencoder (VAE): Variational autoencoders are a class of generative models, extending the classic autoencoder. A VAE is a probabilistic model where the encoder $q_\theta(z|x)$ maps the probability distribution a certain latent representation given a data point, and the decoder $p_\phi(x|z)$ outputs the probability distribution of the data, given a point in the latent space. In VAEs, it is often desireable to model the latent space as an isotropic multivariate Gaussian distribution. This constraint is enforced by introducing a measure of distance between the aforementioned prior distribution $p_\theta(z) \sim \mathcal{N}(0, 1)$ and the encoder distribution, called Kullback-Leibler divergence. This probabilistic framework proves useful when synthesizing HRTFs, because it can learn causal factors of variations in the data [18]. However, there exists no way of generating a specific HRTF — i.e. one for a given combination of azimuth and elevation angles; while the points in latent space are likely to generate plausible new data, one can only obtain random instances of said data. The class of autoencoders described below aims at addressing this shortcoming.

2) Conditional variational autoencoder (CVAE): CVAEs are an extension of variational autoencoders, where an input data labels c modulate the prior distribution of the latent variables that generate the output [19]. Thus, the encoder is formulated as $q_\theta(z|x, c)$, meaning that the encoding process is conditioned by an attribute c, instead of the data content alone. Furthermore, the decoder is also conditioned by the label, so that it models $p_\phi(x|z, c)$. The influence of the label c is incorporated into the VAE structure by means of concatenating its value to the input data x before feeding it into the encoder, as well as to the latent variables z before feeding them into the decoder. Yamamoto [8] uses a customized deep CVAE, where labels consisting of a subject's ID and a spatial orientation, both provided as one-hot encoded vectors, are used to condition each layer of the encoder and decoder.

D. Architectures and models

IV. Results

The research conducted as part of this work can be grouped into three main areas, which are related to feature extraction to use as predictors, unsupervised learning of HRTF data based on the deep autoencoder network in [8], and synthesis of HRTFs from anthropometric data using deep multilayer perceptrons and principal component analysis, inspired by the work of Chen et al. [9]. The following subsections elaborate on the aforementioned topics.

A. Autoencoding ear images

The limitations of anthropometric measurements mentioned in section III-B, together with with the limits of their predicting power highlighted by a previous work [20], prompted the exploration of alternative features to be used in HRTF prediction tasks. Features from pinna images have been previously extracted using convolutional neural networks, for the purpose of biometrics-based identification [21]. Thus, convolutional layers have been employed in a variational autoencoder in order to derive salient features from its compact representation in an unsupervised manner.

Digital renderings of the z-channel (also known as depth maps) of the 3D models in the HUTUBS dataset [14] have been extracted, using the `pyrender` and `trimesh` packages for Python, and converted into 8-bit grayscale images. For each of the 55 unique head meshes, the point of view has been placed on either side of the head, so as to show each pinna separately.

In order to increase the amount of images used for training, several data augmentation techniques have been adopted. Firstly, variations of the point of view have been introduced, by tilting the camera along both elevation and azimuth. Secondly, slight vertical and horizontal offsets have also been introduced. Lastly, each of the images thus generated has been duplicated and processed with sparse, discrete noise. A tool for filtering out the images, based on the augmentation parameters, has been developed, allowing the size of the dataset used for training to be anywhere from a few pictures to well over a million.

The model used in this experiment is a convolutional variational autoencoder. This architecture is similar to the one described in section III-C1, except it uses 2D-convolutional layers. In the encoder, several dimensionality reduction techniques have been tested, such as *max pooling* layers or *strides* in the convolutional kernels, with no discernible difference in performances. Similarly, the decoder part has been originally implemented using transpose convolution with strides, which caused noticeable artifacts in the output images. In order to address this, a combination of *upsampling* layers and regular convolution has instead been used [22]. The hyperparameters of the architecture included the number of stacked layers, the number of convolutional filters for each layer, the number of latent dimensions, and batch normalization.

Multiple experiments have been conducted, using different combinations of input data and model hyperparameters. The main criterion for evaluating the model effectiveness is the quality of the reconstruction. Indeed, images that have been faithfully reconstructed are indicative of a meaningful compact representation that can effectively encode the characteristics of the pinnae; conversely, ear pictures that fail to be distinguishable from those from different subjects are not satisfactory. Furthermore, it is expected that latent variables show some degree of correlation with the anthropometric measurements related to the pinna, since — given an effectively trained model — they would both encode similar information.

The experiments have shown how, while the elevation and azimuth dependent data augmentations expose or hide different parts of the pinnae thereby affecting their appearance, most of the variance in the data occurs in the surrounding area of the head, with is of no interest. This resulted in latent variables encoding mostly features related to the gradient in the background areas, while pinnae appear blurry and undistinguishable. Using data rendered from the same point of view and only augmenting the data using noise seems to alleviate the problem, but the decrease in available training data negatively affects generalization, causing severe artifacts when sampling the latent space. Furthermore, the compact representation of the data points exhibited little to no correlation with the anthropometric measurements.

B. *Autoencoding HRTFs*

Synthesizing HRTFs from a set of given parameters is a fundamental aspect of the individualization task. Since it is not yet fully understood what these parameters could be, it might be interesting to let a neural network derive its own by means of autoencoding the HRTFs. These parameters can then be the target of another prediction task [9] or adjusted through perceptual feedback from the user [8]. Therefore, several VAE models have been developed and trained, using different data formats and architectures.

The architecture of the networks employed in this series of experiments in heavily dependant on the data layout.

- 3d, 2d, 1d
- choice of content in 2d
- ResNet

- similarity

C. *Predicting principal components*

- inspired by [9]
- general principles

- pass angles to each layer
- hyperparams

- spectral distortion
- k-fold validation

V. Conclusions

This work presented some of the most promising advances in HRTF individualization, introduced the deep learning technologies associated with them, and detailed the results of the replication efforts and further experiments based on the underlying knowledge base.

While none of the three domains of investigation has shown outstanding result, the knowledge acquired throughout the development and troubleshooting phases highlighted areas of improvement which can pave the way to further research.

[23]

Bibliography

[1] H. Møller, M. F. Sørensen, C. B. Jensen, and D. Hammershøi, “Binaural Technique: Do We Need Individual Recordings?” *J. Audio Eng. Soc*, vol. 44, no. 6, pp. 451–469, 1996.

[2] C. Guezenoc and R. Seguier, “HRTF Individualization: A Survey,” in *Audio Engineering Society Convention 145*, 2018.

[3] N. A. Gumerov, R. Duraiswami, and D. N. Zotkin, “Fast Multipole Accelerated Boundary Elements for Numerical Computation of the Head Related Transfer Function,” in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP ’07*, 2007, vol. 1, pp. I–165–I–168.

[4] P. Mokhtari, H. Takemoto, R. Nishimura, and H. Kato, “Comparison of Simulated and Measured HRTFs: FDTD Simulation Using MRI Head Data,” in *Audio Engineering Society Convention 123*, 2007.

[5] X.-Y. Zeng, S.-G. Wang, and L.-P. Gao, “A hybrid algorithm for selecting head-related transfer function based on similarity of anthropometric structures,” *Journal of Sound and Vibration*, vol. 329, no. 19, pp. 4093–4106, Sep. 2010.

[6] S.-N. Yao, T. Collins, and C. Liang, “Head-Related Transfer Function Selection Using Neural Networks,” *Archives of Acoustics*, vol. 42, no. 3, pp. 365–373, Sep. 2017.

[7] G. Lee and H. Kim, “Personalized HRTF Modeling Based on Deep Neural Network Using Anthropometric Measurements and Images of the Ear,” *Applied Sciences*, vol. 8, no. 11, p. 2180, Nov. 2018.

[8] K. Yamamoto and T. Igarashi, “Fully perceptual-based 3D spatial sound individualization with an adaptive variational autoencoder,” *ACM Transactions on Graphics*, vol. 36, no. 6, pp. 1–13, Nov. 2017.

[9] T.-Y. Chen, T.-H. Kuo, and T.-S. Chi, “Autoencoding HRTFS for DNN Based HRTF Personalization Using Anthropometric Features,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 271–275.

[10] C. I. Cheng, “Introduction to Head-Related Transfer Functions (HRTFs): Representations of HRTFs in Time, Frequency, and Space,” *J Audio Eng Soc*, vol. 49, no. 4, p. 19, 2001.

[11] A. Kulkarni, S. K. Isabelle, and H. S. Colburn, “On the minimum-phase approximation of head-related transfer functions,” in *Proceedings of 1995 Workshop on Applications of Signal Processing to Audio and Accoustics*, 1995, pp. 84–87.

[12] A. van den Oord *et al.*, “WaveNet: A Generative Model for Raw Audio,” *arXiv:1609.03499 [cs]*, Sep. 2016.

[13] V. R. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano, “The CIPIC HRTF database,” in *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*, 2001, pp. 99–102.

[14] B. Fabian *et al.*, “The HUTUBS head-related transfer function (HRTF) database,” 2019.

[15] S. Spagnol and F. Avanzini, “Frequency estimation of the first pinna notch in head-related transfer functions with a linear anthropometric model,” p. 6, 2015.

[16] S. Spagnol, K. B. Purkhús, R. Unnthórsson, and S. K. Björnsson, “THE VIKING HRTF DATASET,” p. 6, 2019.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[18] D. P. Kingma and M. Welling, “An Introduction to Variational Autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[19] K. Sohn, H. Lee, and X. Yan, “Learning Structured Output Representation using Deep Conditional Generative Models,” in *NIPS*, 2015.

[20] R. Miccini and S. Spagnol, “Estimation of pinna notch frequency from anthropometry: an improved linear model based on principal component analysis and feature selection,” in *Combined proceedings of the Nordic Sound and Music Computing Conference 2019 and the Interactive Sonification Workshop 2019*, 2019, p. 4.

[21] H. Sinha, R. Manekar, Y. Sinha, and P. K. Ajmera, “Convolutional Neural Network-Based Human Identification Using Outer Ear Images,” in *Soft Computing for Problem Solving*, vol. 817, J. C. Bansal, K. N. Das, A. Nagar, K. Deep, and A. K. Ojha, Eds. Singapore: Springer Singapore, 2019, pp. 707–719.

[22] A. Odena, V. Dumoulin, and C. Olah, “Deconvolution and checkerboard artifacts,” *Distill*, 2016.

[23] G. D. Romigh, D. S. Brungart, R. M. Stern, and B. D. Simpson, “Efficient Real Spherical Harmonic Representation of Head-Related Transfer Functions,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 5, pp. 921–930, Aug. 2015.