HRTF SPATIAL UPSAMPLING IN THE SPHERICAL HARMONICS DOMAIN EMPLOYING A GENERATIVE ADVERSARIAL NETWORK

Xuyi Hu[*], *Jian Li*[*], *Lorenzo Picinali* [*] *and Aidan O. T. Hogg* [†]

[*] *Audio Experience Design - www.axdesign.co.uk, Dyson School of Design Engineering, Imperial College London, UK*
[†] *Centre for Digital Music, School of Electronic Engineering and Computer Science, Queen Mary University of London, UK*
xuyi.hu22@imperial.ac.uk

ABSTRACT

A Head-Related Transfer Function (HRTF) is able to capture alterations a sound wave undergoes from its source before it reaches the entrances of a listener's left and right ear canals, and is imperative for creating immersive experiences in virtual and augmented reality (VR/AR). Nevertheless, creating personalized HRTFs demands sophisticated equipment and is hindered by time-consuming data acquisition processes. To counteract these challenges, various techniques for HRTF interpolation and up-sampling have been proposed. This paper illustrates how Generative Adversarial Networks (GANs) can be applied to HRTF data upsampling in the spherical harmonics domain. We propose using Autoencoding Generative Adversarial Networks (AE-GAN) to upsample low-degree spherical harmonics coefficients and get a more accurate representation of the full HRTF set. The proposed method is benchmarked against two baselines: barycentric interpolation and HRTF selection. Results from log-spectral distortion (LSD) evaluation suggest that the proposed AE-GAN has significant potential for upsampling very sparse HRTFs, achieving 17% improvement over baseline methods.

1. INTRODUCTION

Recent advancements in the Metaverse underscore a paradigm shift in augmented and virtual reality (AR/VR) technologies, also thanks to the incorporation of immersive audio interactions to augment user experiences. Binaural audio is a recording and synthesis technique designed to create an immersive, three-dimensional sound experience for the listener using only two channels. To achieve a truly accurate binaural audio experience, it is essential to consider the Head-Related Transfer Function (HRTF). It is standard practice to refer to the HRTF when discussing the impulse response (IR) in the frequency domain, and the Head-Related Impulse Response (HRIR) when referring to it in the time domain. HRTF embeds the alterations sound waves undergo from their source to a listener's ears, shaped by anatomical and environmental variables. The fidelity of HRTF measurements, capturing diverse spatial orientations, is critical for replicating auditory spatial accuracy, thereby augmenting the realism perceived in AR/VR simulations.

This study was made possible by support from SONICOM (www.sonicom.eu), a project that has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No.101017743

Ambisonic is a comprehensive method for capturing, processing, and reproducing spatial sound. Binaural rendering of Ambisonic signals via magnitude least squares is particularly beneficial in advanced audio systems applications, where spatial audio cues are crucial for enhancing user immersion [1]. This integration proves pivotal across various domains, including interactive design [2], gaming [3], and educational applications [4], enhancing the intuitiveness and engagement of virtual environments. However, Ambisonics-based binaural methods are fundamentally dependent on HRTFs.

Humans utilize both binaural cues (i.e. involving the two ears) - interaural time differences (ITD) and interaural level differences (ILD), as well as monaural cues (spectral cues) to localize sound sources around them. More specifically, above 5-6kHz spectral cues play a dominant role in front-back discrimination, as well as vertical localization [5, 6]. However, spectral cues are highly dependent on the listener's anatomy, particularly the shape of their pinnae [7]. In addition, studies have demonstrated that utilizing non-individualized HRTFs can lead to poor performance in sound localization [8, 9]. Hence, to achieve accurate sound localization, besides having spatially dense HRTF measurements, the individualization of HRTFs for each listener is also advantageous.

Researchers have developed various approaches to HRTF individualization [10]. One such approach involves acoustic measurements [11], where sine sweeps are emitted from specific source points and then recorded when they reach the listener's ears, followed by a deconvolution to extract the impulse responses. However, the measurements require sophisticated equipment and the data collection process is rather time-consuming [12]. Spatial upsampling refers to the process of increasing the spatial resolution of audio. By employing spatial upsampling through directional equalization, it is possible to substantially improve spatial audio quality even with limited HRTF datasets [13]. But this technique is limited by its dependency on initial data quality and potential inaccuracies in directional cues. An alternative is numerical calculation approaches [14, 15], which utilize anatomical structural information to compute individualized HRTFs. Nevertheless, obtaining an accurate 3D representation of the listener's anatomical structure that contains pinnae itself is a challenging problem, which may involve a costly setup such as CT scans [16] and MRI [17].

An initial pilot study on machine learning-based HRTF upsampling was conducted by [18], followed by a comprehensive study employing a GAN-based model for HRTF upsampling in the frequency domain [19]. In this paper, the authors aim to develop and assess an innovative approach for HRTF up-sampling using Spherical Harmonics Transformation(SHT) based Generative Adversarial Network(GAN) models. This paper will first delve into the current GAN-based up-sampling technique [19] to pinpoint its drawbacks, particularly regarding the transition to cubic space

representations. Following this, the authors intend to implement and refine the SHT-based GAN up-sampling process, transitioning from low-resolution HRTFs to SHT coefficients, and subsequently upsample these using AE-GAN. The performance of our proposed method will be gauged against two baselines (Barycentric Interpolation and HRTF Selection) with LSD as the evaluation metric.

2. RELATED WORK

2.1. Spatial Upsampling of HRTFs

In order to improve the efficiency of HRTF personalization and enable scalability, the method of spatial up-sampling has been introduced. This approach involves taking low-resolution HRTF data, which may have only a few measurements from limited directions, and up-sampling it to generate high-resolution HRTF data that includes many more measurements from a wider range of directions. Barycentric interpolation [20, 21] and spherical harmonics interpolation [22, 23] are two common methods for HRTF up-sampling. Within the context of barycentric interpolation, the HRTF at the desired direction is computed by taking a weighted average of the nearest three or four measurements. Barycentric interpolation effectively yields accurate interpolation results when the input HRTF data exhibits relatively dense spatial coverage [24].

Considering spherical harmonics interpolation, HRTFs are first converted into the spherical harmonic domain through a spherical harmonics transformation (SHT). This transformation represents the HRTFs as a collection of spherical harmonics along with their corresponding weights, known as spherical harmonics coefficients. Spherical harmonics capture the spatial distribution of sound energy from various directions. Each spherical harmonics captures a unique pattern of variation in sound intensity across different angles around the listener's head.

However, when the low-resolution HRTF is spatially sparse, both barycentric and spherical harmonics interpolations may yield poor reconstruction results that deviate significantly from the measured ones. This is because distant neighbors are unsuitable as correlated references for barycentric interpolation and a lack of sufficient HRTF data hinders the generation of comprehensive harmonics for accurate sampling.

2.2. Machine Learning based HRTFs Generation

Recently, there has been a growing number of machine learning methods proposed to tackle the task of HRTF up-sampling. Ito et al. [25] explored the similarity between regularized linear regression and an autoencoder and found that measured HRTFs can be broken down into source-position-dependent and source-position-independent factors. Building upon this finding, they devised an encoder and a decoder such that their weights and biases are related to the source position. Their loss function incorporates cosine distances between the latent variables of each subject at various source positions. Such a model structure and loss function effectively encouraged the latent variables to capture the personalized characteristics of the HRTF of each individual. The work presented in [26] designed a deep belief network to perform HRTF interpolation and extrapolation. In addition to the measured HRTFs for the left and right ears, the position and anthropometric information are also utilized as part of the input data to the network. They evaluated their network using a test set with 125 data points to generate the full 1250 data points and promising results were obtained. However, using 125 data points is still relatively dense. Ziran et al. [27] developed a dual U-net network architecture to up-sample low-resolution HRTFs. In their dual U-net network, one is responsible for producing high-resolution HRTF magnitude spectra from low-resolution magnitude spectra, and the other one is used to generate high-resolution ITD from sparse measurements. By combining the magnitude spectra and ITD estimates, HRTFs with a high spatial resolution can be obtained. Notably, their evaluation demonstrated the model's efficacy in reconstructing HRTF at 1250 directions from only 23 directions as input measurements. Nevertheless, they simplified the HRTF data in a two-dimensional space instead of considering the entire spherical domain.

3. PROPOSED METHOD

3.1. Data Pre-processing

Convolutional Neural Networks (CNNs) are a useful tool for extracting spatial information from data. However, CNNs are more suited for data with uniform spacing, such as pixels in an image. In the context of HRTFs, data points are distributed in a non-uniform manner on the surface of a sphere. Moreover, the distribution of measurements tends to be denser around the horizontal plane, while generally, no measurements are available at lower elevations. This non-uniform distribution poses unique challenges for directly applying traditional CNN architectures to HRTF data analysis. Thuillier et al. [28] proposed spherical CNNs by using neural processes to learn and predict HRTFs at arbitrary points on a sphere, addressing the challenges of sparse and irregularly sampled HRTF data. However, Implementing neural process meta-learners can be computationally intensive. Therefore, the implementation of SHT on HRTF data offers significant advantages. It not only circumvents the challenge of adapting CNNs to the non-uniform nature of HRTF data but also enhances computational efficiency. The spherical harmonics transformation F_l^m is defined as

$$F_l^m = \int_0^{2\pi} \int_0^{\pi} f(\theta, \phi) Y_l^m(\theta, \phi) \sin(\phi) d\phi d\theta \tag{1}$$

where $Y_l^m(\theta, \phi)$ are the spherical harmonic functions, l and m are the degree and order of the spherical harmonic, respectively. θ and ϕ are the azimuth and elevation angles respectively. $f(\theta, \phi)$ is the original HRTF data function. The spherical harmonic functions can be real-valued or complex-valued functions. In acoustics, these are defined as:

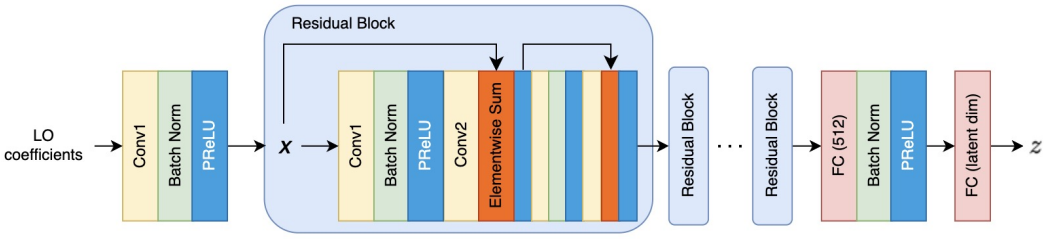
$$Y_l^m(\theta, \phi) = \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(\cos(\phi)) e^{jm\theta} \tag{2}$$

where $P_l^m(x)$ are the associated Legendre functions.

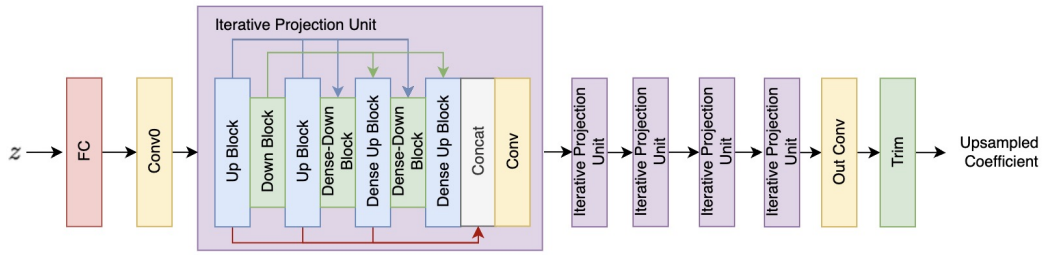
The inverse SHT, which reconstructs the function $f(\theta, \phi)$ from its spherical harmonic coefficients F_l^m, is given by the formula:

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} F_l^m Y_l^m(\theta, \phi) \tag{3}$$

Given that sound sources can be imagined as points on an idealized sphere enclosing the listener's head, this becomes especially pertinent for HRTF data. The SHT enables the decomposition of this spatial data into a number of coefficients, each of which captures a distinct spatial frequency or resolution, while processing



(a) Encoder Architecture.



(b) Decoder Architecture.

Figure 1: Autoencoder architecture. The blue and green arrows represent the dense connection for Dense Down Blocks and Dense Up Blocks respectively. The red arrow indicates the concatenation of upsampled feature maps.

HRTFs [29]. This decomposition can significantly simplify the analysis, and manipulation of HRTFs and even get information between sample points of HRTFs by making HRTFs continuous. Sparse measurements of HRTFs are first transformed into low-resolution SH coefficients via SHT. This process enables the GAN-based model to perform upsampling of SH coefficients, leveraging their uniform distribution. Subsequently, the upsampled high-resolution SH coefficients generated by the GAN are converted back into high-resolution HRTFs using inverse SHT.

3.2. Autoencoder

The autoencoder consists of two networks: an encoder and a decoder. The encoder aims to find the latent representation z of the low-degree spherical harmonics coefficients, capturing the most salient feature to assist the decoder in generating high-degree coefficients.

The encoder's architecture is shown in Fig.1a. An initial convolutional layer extracts low-level features from the input low-degree coefficients. This is followed by a sequence of residual blocks, which enable the learning of higher-level features. Lastly, two fully connected layers are utilized to compress the feature map into the latent space, obtaining the latent representation z. Batch normalization is applied to stabilize the training process and serves as a regularization technique. The activation function used is a parametric rectified linear unit (PReLU), defined as:

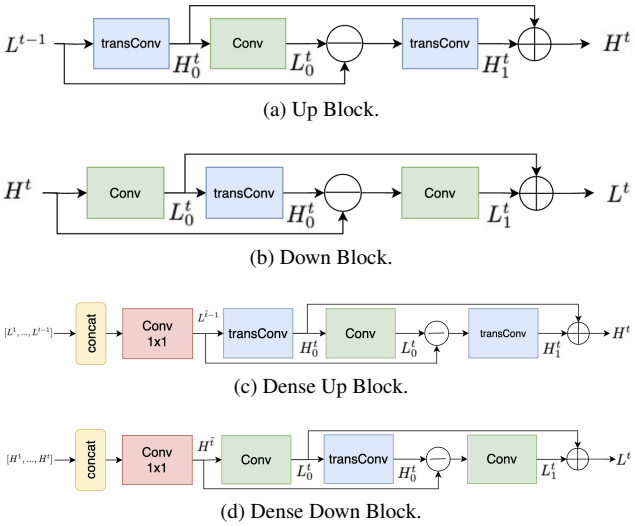
$$\textbf{PReLU}(x) = \max(0, x) + a \times \min(0, x) \ , \tag{4}$$

where a is a learnable parameter. Given a non-zero slop a, it can effectively alleviate the 'dying ReLU' problem. This activation function is also used in the decoder network.

The design of the decoder incorporates the concept of iterative up and downsampling proposed by [30]. This architecture introduces an error feedback mechanism where reconstruction error is computed at each stage, thereby enhancing the ability to capture the intrinsic connection between low-degree and high-degree coefficient pairs. The network is built based on four fundamental blocks illustrated in Fig.2. The up block, shown in Fig.2a, upsamples the low-resolution feature map L^{t-1} to an intermediate high-resolution output H_0^t, which is then mapped back to a low-resolution feature map L_0^t. The difference between L^{t-1} and L_0^t is upsampled to get H_1^t which is then added to H_0^t to obtain the final output of this block, H^t. The workflow of a down block is similar to that of an up block but operates in reverse, as depicted in Fig.2b. The inter-layer connection in dense up and down blocks effectively mitigates the vanishing gradient problem, and yields enhanced features. As illustrated in Fig.2c, the low-resolution feature maps from previous layers are concatenated along the channel dimension before being passed into the dense up block.

In this work, an iterative projection unit is designed for progressive upsampling. As illustrated in Fig.1b, it is composed of the four fundamental blocks mentioned earlier. Within this unit, the dense down blocks take concatenated high-resolution feature maps from all previous upsampling blocks as input, and these skip connections are indicated by the blue arrows. Conversely, the dense up blocks utilize concatenated low-resolution feature maps from all previous downsample blocks, with green arrows denoting the associated skip connections. Finally, the high-resolution feature maps are concatenated and fed into the last output convolutional layer, which serves for dimension reduction. After going through this iterative projection unit, the input feature map will be upscaled by a factor of 2.

The overview of the designed decoder is presented in Fig.1b. The latent representation z obtained from the encoder first goes through a fully connected layer. The resultant 1D vector is then reshaped so that it is suitable for the convolution operation. Five



(a) Up Block.

(b) Down Block.

(c) Dense Up Block.

(d) Dense Down Block.

Figure 2: Basic Blocks in Decoder.

iterative projection units are cascaded to upsample the feature map progressively. And lastly excessive coefficients are trimmed off.

3.3. Discriminator Network

The discriminator aims to validate the authenticity of the input of spherical harmonics coefficients, determining whether they are genuine or created by the generator. The discriminator is trained using a supervised learning approach with real and generated HRTF data. It assigns a probability to each input, indicating whether it is real or fake, using a sigmoid activation function. The discriminator learns to classify real and fake data by updating its weights through backpropagation to minimize classification error. This iterative process improves its accuracy in distinguishing between real and generated HRTF coefficients, effectively learning the underlying probability distribution of the real data. As shown in Fig.3, the discriminator consists of 9 convolutional layers for feature extraction. The triplet enclosed in the bracket denotes the kernel size, the output channel, and the stride, respectively. Notably, starting from the second layer, the feature map is downsampled by a factor of 2 every two layers. Each convolutional layer is followed by a batch normalization layer for the purpose of training stability. An exception is made for the initial layer, where the application of batch normalization is avoided due to potential issues such as sample oscillation and model instability, as indicated in [31]. The last two layers are fully connected layers and a sigmoid activation function as it is suitable for binary prediction. The activation function employed throughout the architecture is the leaky rectified linear unit (ReLU), except for the last layer. The leaky ReLU activation function is a variation of ReLU, defined as:

$$\textbf{LeakyReLU}(x) = \max(0, x) + \text{negative slop} \times \min(0, x)\ , \quad (5)$$

where the negative slop is set to 0.2.

3.4. Cost Functions

The loss function $\mathcal{L}^{\mathrm{G}}$ for the autoencoder has three components: the cosine similarity loss $\mathcal{L}^{\mathrm{G}}_{\mathrm{cos}}$ for coefficients, the content loss $\lambda\mathcal{L}^{\mathrm{G}}_{\mathrm{C}}$, and the adversarial loss $\mathcal{L}^{\mathrm{G}}_{\mathrm{A}}$. Mathematically, it can be expressed as follows:

$$\mathcal{L}^{\mathrm{G}} = \mathcal{L}^{\mathrm{G}}_{\mathrm{cos}} + \lambda\mathcal{L}^{\mathrm{G}}_{\mathrm{C}} + \mathcal{L}^{\mathrm{G}}_{\mathrm{A}}\ , \quad (6)$$

where λ is a weight applied to the content loss term, ensuring that it will not become too large and disrupt the training process.

3.5. Cosine Similarity Loss

Typically within the autoencoder framework, the focal point of the loss function lies in the reconstruction process. This commonly involves employing the mean squared error (MSE) loss to quantify the disparity between the reconstructed output and the intended target. For instance, in the realm of image generation, the MSE loss is computed by evaluating the difference in pixel values. In this work, a modification is introduced to the MSE loss, incorporating the cosine similarity. This adjustment, denoted as $\mathcal{L}^{\mathrm{G}}_{\mathrm{cos}}$, measures the likeness between the extrapolated coefficients from the autoencoder and the original high-degree coefficients for each frequency bin. The resulting similarity measurements are then averaged over the total number of frequency bins present in the HRTF data. Therefore, $\mathcal{L}^{G}_{cos}$ is defined as:

$$\mathcal{L}^{\mathrm{G}}_{\mathrm{cos}} = \sqrt{\frac{1}{W}\sum_{w=1}^{W}\left(1 - \frac{\mathbf{c}_{\mathrm{G}}^{f_w}\cdot\mathbf{c}_{\mathrm{H}}^{f_w}}{\|\mathbf{c}_{G}^{f_w}\|\|\mathbf{c}_{H}^{f_w}\|}\right)^2}\ , \quad (7)$$

where $\mathbf{c}_{\mathrm{G}}^{f_w}$ and $\mathbf{c}_{\mathrm{H}}^{f_w}$ represent upsampled coefficients and target high-degree coefficients respectively, W is the number of frequency bins in the HRTF, and f_w denotes a specific frequency.

3.6. Content Loss

Given that the primary goal of this project is to upsample spatially sparse HRTFs, the task of extrapolating spherical harmonics coefficients serves as an intermediate step. To effectively guide the autoencoder to create meaningful coefficients and eventually to produce realistic HRTFs, the content loss introduced in [19] has been adopted in the autoencoder loss. The content loss is the sum of the log-spectral distortion (LSD) metric and the interaural level difference (ILD) metric:

$$\mathcal{L}^{\mathrm{G}}_{\mathrm{C}} = \mathrm{LSD} + \mathrm{ILD}. \quad (8)$$

The LSD [32] is a metric used to measure the quality of a synthesized audio signal compared to a reference audio signal. The LSD loss quantifies this comparison by evaluating the discrepancy between the target magnitude spectrum H_{HR} and the generated spectrum H_{G} at frequency f_w and position x_n. This computation can be expressed using the following mathematical formula:

$$\mathrm{LSD} = \frac{1}{N}\sum_{n=1}^{N}\sqrt{\frac{1}{W}\sum_{w=1}^{W}\left(20\log_{10}\frac{|H_{\mathrm{HR}}(f_w, x_n)|}{|H_{\mathrm{G}}(f_w, x_n)|}\right)^2}\ , \quad (9)$$

where N represents the overall count of positions, f_w is a certain frequency, and x_n corresponds to a specific position.

The ILD [23] refers to the difference in magnitudes perceived by each ear due to the spatial location of a sound source. It is

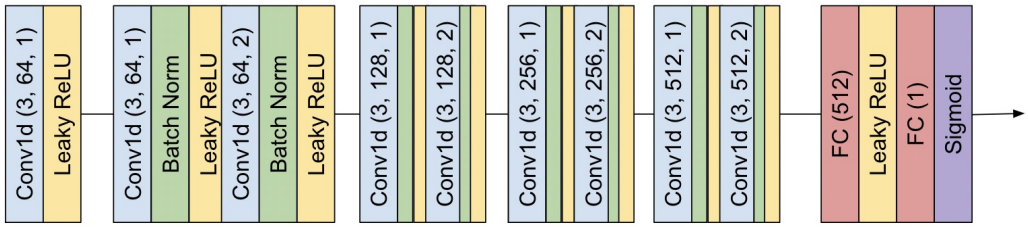


Figure 3: Discriminator architecture.

defined as:

$$\text{ILD} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{W}\sum_{w=1}^{W}\left|\left(20\log_{10}\frac{|H_{\text{HR}}^{\text{Left}}(f_w,x_n)|}{|H_{\text{HR}}^{\text{Right}}(f_w,x_n)|}\right) - \left(20\log_{10}\frac{|H_{\text{G}}^{\text{Left}}(f_w,x_n)|}{|H_{\text{G}}^{\text{Right}}(f_w,x_n)|}\right)\right|, \quad (10)$$

where $H^{\text{Left}}(f_w, x_n)$ and $H^{\text{Right}}(f_w, x_n)$ denote the left and right ear magnitude responses at frequency f_w and position x_n.

3.7. Adversarial Loss

Lastly, since the overall model structure operates as a generative adversarial network with the autoencoder acting as the generator, an adversarial loss is incorporated to measure how realistic the upsampled coefficients are to fool the discriminator. The adversarial loss is defined as the binary cross-entropy loss over M training samples, expressed as:

$$\mathcal{L}_{\text{A}}^{\text{G}} = -\frac{1}{M}\sum_{m=1}^{M}\log(1 - D(G(\mathbf{c}_{\text{L}}^{m}))), \quad (11)$$

where $G(\mathbf{c}_{\text{L}}^{m})$ corresponds to the upsampled spherical harmonics coefficients from the generator, and the term $D(G(\mathbf{c}_{\text{L}}^{m}))$ can be interpreted as the probability assigned by the discriminator, signifying the possibility of the given sample being an authentic high-degree coefficient. Similarly, the loss function for the discriminator is defined as:

$$\mathcal{L}^{\text{D}} = -\frac{1}{M}\sum_{m=1}^{M}[\log D(\mathbf{c}_{\text{H}}^{m}) + \log(1 - D(G(\mathbf{c}_{\text{L}}^{m})))], \quad (12)$$

where $\mathbf{c}_{\text{H}}^{m}$ represents a sample of high-degree coefficients.

4. EXPERIMENTAL EVALUATION

4.1. Baselines

The proposed AE-GAN is evaluated on a test set of 41 subjects from the SONICOM HRTF dataset [11]. Each HRTF in the SONICOM data set contains a total of 793 different positions for each individual. Each of these measurements was taken around the subject's head at a distance of 1.5m, where the azimuth was sampled every 5° and the elevation ranged from –45° to 90° (sampled every 10° between –30° and 30° and every 15° otherwise). These measurements are available in 44kHz, 48kHz, and 96kHz. To accurately represent these HRTFs in the SH domain within the audible spectrum (up to 20 kHz), a truncation order of approximately 32 would be needed [33]. That is to say, truncating the order to below 32 could potentially introduce audible artefacts in any binaural signal rendered with it. The model result is compared against two baselines:

4.1.1. Baseline-1 - Barycentric interpolation

In this work, the implementation of barycentric interpolation follows the approach outlined in [19]. Barycentric interpolation is a powerful technique particularly suited for interpolating values within a simplex, leveraging weighted averages of the function's values at the vertices of the simplex. It is used as one of the baselines to compare with our model result. It utilizes the concept of three barycentric coordinates, which are weights assigned to these data points. By using these weights, the method provides a way to interpolate or find an unknown value within the data points based on the known values around it.

For barycentric interpolation with HRTFs, we first need to identify the three nearest known points $P1, P2, P3$ around the target point Pi on a spherical surface. For each given point (azimuth and elevation), we determine the optimal triangle for barycentric interpolation. The triangle that encloses the target point and has the smallest total distance between its vertices and target point is called the *best triangle*. Then, calculate the barycentric coordinates α, β and γ. These represent how much each of the three points $P1, P2, P3$ contributes to Pi, ensuring their sum is always 1. Using these weights, we can determine the interpolated HRTF for Pi. Elevation and azimuth are considered as Cartesian coordinates. The ratio of the areas is calculated using the following formulas:

$$\alpha = \frac{(\phi^{P2}-\phi^{P3})(\theta^{Pi}-\theta^{P3}) + (\theta^{P3}-\theta^{P2})(\phi^{Pi}-\phi^{P3})}{(\phi^{P2}-\phi^{P3})(\theta^{P1}-\theta^{P3}) + (\theta^{P3}-\theta^{P2})(\phi^{P1}-\phi^{P3})}, \quad (13)$$

$$\beta = \frac{(\phi^{P3}-\phi^{P1})(\theta^{Pi}-\theta^{P3}) + (\theta^{P1}-\theta^{P3})(\phi^{Pi}-\phi^{P3})}{(\phi^{P2}-\phi^{P3})(\theta^{P1}-\theta^{P3}) + (\theta^{P3}-\theta^{P2})(\phi^{P1}-\phi^{P3})}, \quad (14)$$

$$\gamma = 1 - \alpha - \beta. \quad (15)$$

The following adjustments were made to represent elevation and azimuth as spherical coordinates. The spherical triangle's surface area is denoted by $A = r^2E$, with r being the sphere's radius and E the excess angle. This angle, E, is the surplus of the triangle's interior angles over the total interior angles of a flat triangle, which is π radians. The connection between the excess angle E and the triangle side lengths a, b, c is elucidated by L'Huilier's

Table 1: A comparison of the mean LSD error (Standard Deviation) for different upsampling factors. The 'Best' result of each upsampling factor has been highlighted

Method	Upsampling (Scale Factor) [No. intial → No. upsampled]			
	27 → 864 (32)	18 → 864 (48)	12 → 864 (72)	8 → 864 (108)
AE-GAN	5.01 (0.58)	5.11 (0.59)	5.17 (0.56)	6.05 (0.94)
Barycentric	4.89 (0.24)	5.46 (0.27)	6.28 (0.29)	7.22 (0.35)
Selection-1	6.31 (0.59)			
Selection-2	8.33 (0.47)			

Theorem [34], with s representing the semiperimeter:

$$\tan(\tfrac{1}{4}E) = \sqrt{\tan(\tfrac{1}{2}s)\tan(\tfrac{1}{2}(s-a)) \times \tan(\tfrac{1}{2}(s-b))\tan(\tfrac{1}{2}(s-c))}. \quad (16)$$

Consequently, the weight coefficients are determined as:

$$\alpha = \frac{E^{PiP2P3}}{E^{P1P2P3}}, \quad \beta = \frac{E^{P1PiP3}}{E^{P1P2P3}}, \quad \text{and} \quad \gamma = 1-\alpha-\beta. \quad (17)$$

The target HRIR is computed as the weighted linear combination of the three vertices with weights alpha, beta and gamma:

$$\text{HRIR}^{P_i} = \alpha\text{HRIR}^{P_1} + \beta\text{HRIR}^{P_2} + \gamma\text{HRIR}^{P_3}\ . \quad (18)$$

The discrete Fourier transform (DFT) converts the HRIR into the HRTF after interpolation.

4.1.2. Baseline-2 - non-individual HRTF selection

Additionally, the suggested AE-GAN methodology is compared to the following HRTF selection method as another baseline. In this baseline, two HRTFs are chosen from the test set based on their average LSD error when compared to all other HRTFs in the test set, as opposed to choosing a HRTF at random. Selection-1 corresponds to the subject whose HRTF yields the lowest average LSD error, which can be seen as the most generic one. On the other hand, Selection-2 represents the subject whose HRTF produces the highest average LSD error, giving some indication to the most.

4.2. LSD metric evaluation

The LSD metric, as defined in equation (9), is calculated for every measurement and subsequently averaged across all source positions. The average LSD evaluation results for 41 subjects in the test set are presented in Table 1. The best performance of each upscaling factor is highlighted in blue.

From the visual representation of the evaluation outcomes, as depicted in Fig.4, it can be seen that AE-GAN produces a result with an LSD of 5.01 that is comparable to that of the barycentric interpolation (4.89) when the upscale factor is 32. Moreover, the proposed model surpasses the baseline approach when the initial HRTF is sparser. These findings are aligned with the outcomes reported in [19] even though a different HRTF dataset (ARI) was used. As shown in Table 2, their proposed SRGAN achieves lower errors when the initial HRTF is more spatially sparse.

As for the HRTF selection, the utilization of Selection-1 and Selection-2 results in LSD errors of 6.31 and 8.33, respectively. However, this approach demonstrates inferior performance compared to the proposed AE-GAN in all cases. Notably, Selection-1

Table 2: The LSD evaluation results from [19] using a frequency-domain GAN approach. The 'Best' result of each upsampling factor has been highlighted

Method	Upsampling (Scale Factor) [No. intial → No. upsampled]			
	320 → 1280 (4)	80 → 1280 (16)	20 → 1280 (64)	5 → 1280 (256)
SRGAN	3.28 (0.13)	4.86 (0.24)	4.99 (0.27)	5.30 (0.35)
SH	3.54 (0.15)	4.94 (0.20)	5.90 (0.25)	10.36 (0.74)
Barycentric	2.50 (0.20)	3.71 (0.22)	5.18 (0.23)	7.30 (0.33)
Selection-1	6.96 (0.47)			
Selection-2	8.20 (0.61)			

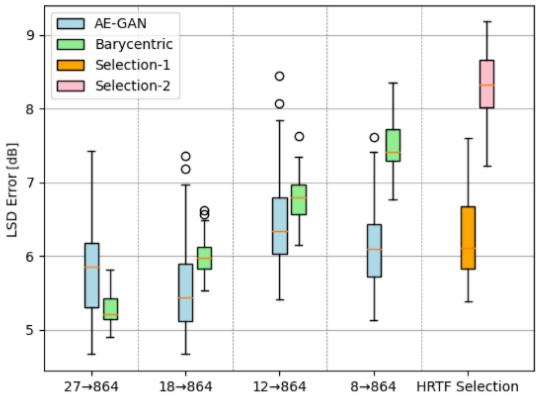


Figure 4: LSD error comparison.

excels the barycentric interpolation when the upscale factor is 108 (where the mean LSD error is 7.22 for barycentric interpolation) by a small margin.

In order to gain a deeper comprehension of the locations where these errors arise for the proposed AE-GAN and barycentric interpolation, the visualization of these errors across all source positions under different upsampling conditions for SubjectID 10 is provided in Fig.5. The original data points present in the low-resolution HRTF are indicated by spikes pointing to the azimuth-elevation plane. The z-axis represents the average LSD error.

From Fig.5a, it can be seen that when upsampling from 27 to 864, most LSD errors come from -40° to 0° elevation range for AE-GAN. Conversely, barycentric interpolation is doing slightly better within this region. However, at higher elevations, considerably high LSD error (above 10) occurs in the case of barycentric interpolation, whereas AE-GAN manages to keep the errors at a low level of around 4 decibels. Moreover, with the increasing sparsity of the initial low-resolution HRTF, the area of the darker region grows rapidly when using barycentric interpolation, as shown in the high-elevation part in 5b and Fig.5c. This is because the barycentric interpolation could not accurately estimate the value at the target position when the neighboring measured ones are far away from the desired location. In contrast, the proposed AE-GAN has learned from low and high-order coefficient pairs, enabling it to reconstruct the spherical harmonics that closely represent the whole set of HRTF measurements. Therefore, irrespective of the upscale factor applied and the separation between the mea-

sured points and the target position, the AE-GAN is able to predict the values decently. This is evident when comparing the upper plot in Fig.5b with the upper plot in Fig.5c. The LSD errors spanning all positions exhibit minimal variation even though 75% of initial points are removed.

5. CONCLUSIONS

In this paper, the proposed AE-GAN shows great potential for upsampling very sparse HRTFs, when more traditional interpolation methods start failing. Instead of applying deep learning techniques directly on the unevenly distributed HRTF data, the HRTF data in the frequency domain is first transformed into the spherical harmonics domain, where the original spatial and frequency information is represented by a set of spherical harmonics and associated coefficients. The LSD evaluation result suggests that the proposed deep-learning model is capable of achieving superior results compared to those obtained by the barycentric interpolation when the HRTFs are extremely sparse, with fewer than 12 measurements.

Future investigations could focus on further reducing the number of needed measured positions or exploring which are the best 8 positions to get optimal results. This is motivated by the fact that point position selection is highly related to the quality of representation of the original HRTFs. Conventional HRTF interpolation methods, not based on machine learning, merit consideration as perceptually motivated optimizations. These optimizations, such as separately interpolating ITDs and HRTF magnitude responses, then reintroducing interaural phase differences (IPDs) only at low frequencies (e.g., <1.5 kHz according to the Duplex theory), can substantially enhance interpolation accuracy. These approaches could serve as effective baselines, given their likely use in existing binaural panning tools over naive direct complex-HRTF interpolation methods.

6. REFERENCES

[1] C. Schörkhuber, M. Zaunschirm, and R. Höldrich, "Binaural rendering of ambisonic signals via magnitude least squares," 03 2018.

[2] V. Bauer *et al.*, "Designing an interactive and collaborative experience in audio augmented reality," in *Virtual Reality and Augmented Reality: 16th EuroVR International Conference, EuroVR 2019, Tallinn, Estonia, October 23–25, 2019, Proceedings*, vol. 16. Springer International Publishing, 2019.

[3] E. M. Raybourn *et al.*, "Adaptive thinking & leadership simulation game training for special forces officers," in *The Interservice, Industry Training, Simulation & Education Conference (ITSEC)*, 2005.

[4] E. Rovithis *et al.*, "Bridging audio and augmented reality towards a new generation of serious audio-only games," 2019.

[5] J. Blauert, *Spatial hearing: the psychophysics of human sound localization*. MIT press, 1997.

[6] B. Xie, *Head-related transfer function and virtual auditory display*. J. Ross Publishing, 2013.

[7] Y. Kahana and P. A. Nelson, "Numerical modelling of the spatial acoustic response of the human pinna," *Journal of Sound and Vibration*, vol. 292, no. 1-2, pp. 148–178, 2006.

[8] P. Stitt, L. Picinali, and B. F. Katz, "Auditory accommodation to poorly matched non-individual spectral localization cues through active learning," *Scientific reports*, vol. 9, no. 1, p. 1063, 2019.

[9] E. M. Wenzel, M. Arruda, D. J. Kistler, and F. L. Wightman, "Localization using nonindividualized head-related transfer functions," *The Journal of the Acoustical Society of America*, vol. 94, no. 1, pp. 111–123, 1993.

[10] L. Picinali and B. F. Katz, "System-to-user and user-to-system adaptations in binaural audio," in *Sonic Interactions in Virtual Environments*. Springer International Publishing Cham, 2022, pp. 115–143.

[11] I. Engel, R. Daugintis, T. Vicente, A. O. Hogg, J. Pauwels, A. J. Tournier, and L. Picinali, "The sonicom HRTF dataset," *Journal of the Audio Engineering Society*, vol. 71, no. 5, pp. 241–253, 2023.

[12] S. Li and J. Peissig, "Measurement of head-related transfer functions: A review," *Applied Sciences*, vol. 10, no. 14, p. 5014, 2020.

[13] C. Pörschmann, J. M. Arend, and F. Brinkmann, "Directional equalization of sparse head-related transfer function sets for spatial upsampling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 6, pp. 1060–1071, 2019.

[14] B. F. Katz, "Boundary element method calculation of individual head-related transfer function. i. rigid model calculation," *The Journal of the Acoustical Society of America*, vol. 110, no. 5, pp. 2440–2448, 2001.

[15] S. Harder, R. R. Paulsen, M. Larsen, S. Laugesen, M. Mihocic, and P. Majdak, "A framework for geometry acquisition, 3-d printing, simulation, and measurement of head-related transfer functions with a focus on hearing-assistive devices," *Computer-Aided Design*, vol. 75, pp. 39–46, 2016.

[16] H. Ziegelwanger, A. Reichinger, and P. Majdak, "Calculation of listener-specific head-related transfer functions: Effect of mesh quality," in *Proceedings of Meetings on Acoustics ICA2013*, vol. 19, no. 1. Acoustical Society of America, 2013, p. 050017.

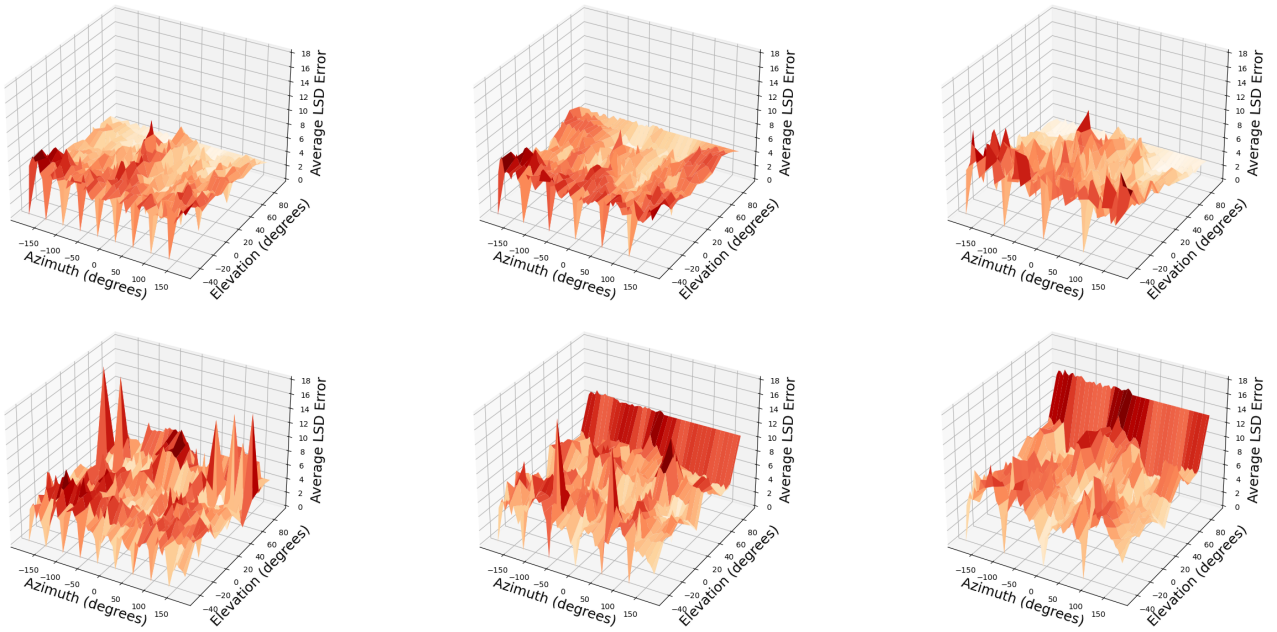
[17] R. Greff and B. F. Katz, "Round robin comparison of HRTF simulation systems: Preliminary results," in *Audio Engineering Society Convention 123*. Audio Engineering Society, 2007.

[18] P. Siripornpitak, I. Engel, I. Squires, S. J. Cooper, and L. Picinali, "Spatial up-sampling of HRTF sets using generative adversarial networks: A pilot study," *Frontiers in Signal Processing*, p. 54, 2022.

[19] A. O. Hogg, M. Jenkins, H. Liu, I. Squires, S. J. Cooper, and L. Picinali, "HRTF upsampling with a generative adversarial network using a gnomonic equiangular projection," *IEEE/ACM Trans. Audio, Speech, Language Process.*, 2024, in press.

[20] D. Poirier-Quinot and B. F. Katz, "The anaglyph binaural audio engine," in *Audio Engineering Society Convention 144*. Audio Engineering Society, 2018.

[21] M. Cuevas-Rodríguez, L. Picinali, D. Gonzalez-Toledo, C. Garre, E. de la Rubia-Cuestas, L. Molina-Tanco, and A. Reyes-Lecuona, "3d tune-in toolkit: An open-source



(a) 27 → 864. (b) 12 → 864. (c) 8 → 864.

Figure 5: Comparison of the proposed AE-GAN (top row) and barycentric interpolation (bottom row) in terms of LSD errors at different upscale factors for SubjectID 10. The initial source positions before interpolation are shown by spikes on the azimuth-elevation plane.

library for real-time binaural spatialisation," *PLOS ONE*, vol. 14, no. 3, p. e0211899, 2019.

[22] M. J. Evans, J. A. Angus, and A. I. Tew, "Analyzing head-related transfer function measurements using surface spherical harmonics," *The Journal of the Acoustical Society of America*, vol. 104, no. 4, pp. 2400–2411, 1998.

[23] I. Engel, D. F. Goodman, and L. Picinali, "Assessing HRTF preprocessing methods for ambisonics rendering through perceptual models," *Acta Acustica*, vol. 6, p. 4, 2022.

[24] H. Gamper, "Head-related transfer function interpolation in azimuth, elevation, and distance," *The Journal of the Acoustical Society of America*, vol. 134, no. 6, pp. EL547–EL553, 2013.

[25] Y. Ito, T. Nakamura, S. Koyama, and H. Saruwatari, "Head-related transfer function interpolation from spatially sparse measurements using autoencoder with source position conditioning," in *2022 International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2022, pp. 1–5.

[26] G. Kestler, S. Yadegari, and D. Nahamoo, "Head related impulse response interpolation and extrapolation using deep belief networks," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 266–270.

[27] Z. Jiang, J. Sang, C. Zheng, A. Li, and X. Li, "Modeling individual head-related transfer functions from sparse measurements using a convolutional neural network," *The Journal of the Acoustical Society of America*, vol. 153, no. 1, pp. 248–259, 2023.

[28] E. Thuillier, C. Jin, and V. Valimaki, "HRTF interpolation using a spherical neural process meta-learner," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 32, pp. 1790–1802, 2024, publisher Copyright: IEEE.

[29] A. Zonca *et al.*, "healpy: equal area pixelization and spherical harmonics transforms for data on the sphere in python," *Journal of Open Source Software*, vol. 4, no. 35, p. 1298, 2019.

[30] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1664–1673.

[31] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016.

[32] P. Gutierrez-Parera, J. J. Lopez, J. M. Mora-Merchan, and D. F. Larios, "Interaural time difference individualization in HRTF by scaling through anthropometric parameters," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2022, no. 1, pp. 1–19, 2022.

[33] Z. Ben-Hur, D. L. Alon, R. Mehra, and B. Rafaely, "Efficient representation and sparse sampling of head-related transfer functions using phase-correction based on ear alignment," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2249–2262, 2019.

[34] E. D. Zwillinger, *CRC Standard Mathematical Tables and Formulas*, 33rd ed. Boca Raton: Chapman and Hall/CRC, Jan 2018.