applied sciences

MDPI

Article

Low-Order Spherical Harmonic HRTF Restoration Using a Neural Network Approach

Benjamin Tsui [1,*], William A. P. Smith [2] and Gavin Kearney [1]

[1] AudioLab, Communications Technologies Research Group, Department of Electronic Engineering, University of York, York YO10 5DD, UK; gavin.kearney@york.ac.uk

[2] Computer Vision and Pattern Recognition (CVPR) Research Group in the Department of Computer Science, University of York, York YO10 5GH, UK; william.smith@york.ac.uk

* Correspondence: bt712@york.ac.uk

Received: 8 July 2020; Accepted: 15 August 2020; Published: 20 August 2020

Abstract: Spherical harmonic (SH) interpolation is a commonly used method to spatially up-sample sparse head related transfer function (HRTF) datasets to denser HRTF datasets. However, depending on the number of sparse HRTF measurements and SH order, this process can introduce distortions into high frequency representations of the HRTFs. This paper investigates whether it is possible to restore some of the distorted high frequency HRTF components using machine learning algorithms. A combination of convolutional auto-encoder (CAE) and denoising auto-encoder (DAE) models is proposed to restore the high frequency distortion in SH-interpolated HRTFs. Results were evaluated using both perceptual spectral difference (PSD) and localisation prediction models, both of which demonstrated significant improvement after the restoration process.

Keywords: deep learning; head related transfer function (HRTF); restoration; ambisonics; spatial audio; spherical harmonic; audio signal processing; denoising; auto-encoder; neural network

1. Introduction

Virtual reality (VR) and augmented reality (AR) technologies are on the rise, through the advent of commercially available and affordable VR/AR headsets, with applications in gaming, education, therapy, social media and digital culture, amongst others. To provide a high fidelity immersive experience in virtual environments requires good quality spatial audio. To achieve this, the VR/AR technology must be able to deliver to the ears the same binaural cues as would be experienced in real life [1,2]. These are Interaural Time Difference (ITD), Interaural Level Difference (ILD) and spectral cues introduced by the ear pinnae and torso [3,4]. Head related transfer functions (HRTFs) are sets of binaural filters that encapsulate these cues from different angles relative to the head in three dimensions. Sound sources can be spatialised by direct convolution with the a given HRTF pair representing the intended sound source direction. Alternatively, a virtual loudspeaker framework can be employed, wherein methods such as vector base amplitude panning (VBAP) [5] or Ambisonics [6] are used to render sources between virtual loudspeaker points formed from the HRTFs [7]. Both methods typically require a high number of HRTF measurements to ensure good spatial resolution in the rendered audio [8].

However, HRTFs are highly personalised due to different head and ear shapes. Using mismatched HRTFs can affect timbre quality, localisation performance and externalisation. Currently, the main way to obtain personalised HRTFs is through physical measurements, where microphones are placed at the ear canal of a subject and the loudspeakers are positioned at different angles relative to the head to measure the transfer functions [9,10]. This measurement process is often tedious and requires substantial setup and calibration. Recent developments have been made in HRTF-based selection

based on anthropomorphic data extracted from photographs of the ear [11] and HRTF simulation using 3D head models [12]. However, simulations are computationally expensive and usually require a lot of processing time [13,14].

To simplify the measurement process, different HRTF interpolation methods have been proposed to acquire dense HRTF sets from sparse HRTF measurements. The current state of the art method is spherical harmonic (SH) interpolation, which leverages the spatial continuity in the spherical harmonic (SH) domain and uses it as a bridge to spatially up-sample a sparse HRTF measurement set to a denser one [15,16]. Depending on the number of sparse HRTF measurement points and SH order, important high frequency information can be lost in this process. Consequently, listeners may perceive timbre difference and weakened localisation performance in practical use.

Recently, developments in machine learning have shown great improvement in neural style transfer and data restoration, especially in the image domain [17,18]. This paper investigates whether similar models can be used to restore the distorted high frequency data in SH-interpolated HRTFs.

This paper is organised as follows: Section 2 covers relevant background information on SH HRTF interpolation. It also discusses the motivation for using a machine learning approach along with identification of some candidate models. Section 3 discusses the method used in this study, including the data pre-processing workflow, a baseline model and different techniques investigated on top of the baseline model. Section 4 describes the evaluation of the performance of the model based on perceptual spectral difference and localisation performance. Section 5 discusses the results and potential directions for the work. Section 6 concludes the paper.

2. Spherical Harmonic HRTF Interpolation

HRTF interpolation can be done in many different ways, including using inverse-distance weighting and spherical splines in the time or frequency domains or manifold learning [19–23]. However, SH interpolation is one of the more elegant methods which has a more standardised procedure and shows promising results [23].

HRTF sets are recorded in the time domain as Head Related Impulse Responses (HRIRs). These HRIR sets are commonly described in the SH domain due to simplicity and ease of use in Ambisonics for spatial audio reproduction [24]. Since the SH domain describes a continuous spatial representation of the HRIRs, interpolation can be readily achieved. A given HRIR $\mathrm{H}(\theta, \phi)$ can be converted to the spherical harmonic domain at a given spherical harmonic order M by using a re-encoding matrix C with K rows and L columns, where K is the number of SH channels calculated as $K = (M+1)^2$ and L is the number of HRTF measurements from different angles, where $L \geq K$. The coefficients Y_{mn}^{σ} in the re-encoding matrix C with SH order m and degree n are calculated by

$$Y_{mn}^{\sigma}(\theta, \phi) = \sqrt{(2 - \delta_{n,0}) \frac{(m-n)!}{(m+n)!}} P_{mn}(\sin\phi) \times \begin{cases} \cos(n\theta), \text{ if } \sigma = +1 \\ \sin(n\theta), \text{ if } \sigma = -1 \end{cases} \quad (1)$$

where $\sigma = \pm 1$, $P_{mn}(\sin\phi)$ are the Legendre functions of order m and degree n; $\delta_{n,0}$ is the Kronecker delta function:

$$\delta_{n,0} \equiv \begin{cases} 1, \text{ for } n = 0 \\ 0, \text{ for } n \neq 0 \end{cases} \quad (2)$$

This paper uses Schmidt semi-normalisation (SN3D) in the computation of Y_{mn}^{σ}. For normal SH HRIR use, a mode matching decoding matrix D can be calculated from C with the following pseudo-inverse equation:

$$\mathrm{D} = \mathrm{C}^{-1} = \mathrm{C}^{T}\left(\mathrm{C}\mathrm{C}^{T}\right)^{-1} \tag{3}$$

which can be used to inverse the SH HRIR back into the original HRIR [25,26].

However, for HRTF interpolation, another re-encoding matrix $\hat{\mathrm{C}}$ and decoding matrix $\hat{\mathrm{D}}$ with the desired target angles can be calculated with Equations (1)–(3), where $\hat{L}$ is replaced as the number of HRTF measurements to be interpolated from the SH HRIRs whilst K remains the same.

The interpolated HRIRs $\hat{\mathrm{H}}(\theta, \phi)$ then can be calculated with the following equation:

$$\hat{\mathrm{H}} = \hat{\mathrm{D}}(\mathrm{C}(\mathrm{H})) \tag{4}$$

To challenge the full potential of the use of machine learning, this study used 1st order SH interpolation as it requires the least number of HRIR measurements. An octahedron configuration with six measurements was selected, which has a more stable energy distribution than other arrays for 1st order SH [8]. Interpolation with 1st order SH up to the original number of measurements must be undertaken, which, depending on the dataset can be >2000 measurements. More specifically, the six selected sparse HRIRs from the HRIR database were first converted to SH HRIR 1st order SH which has four channels) by using the re-encoding matrix C with four rows and six columns. Then a decoding matrix $\hat{\mathrm{D}}$ was computed using a different re-encoding matrix $\hat{\mathrm{C}}$ with the same number of rows and a different number of columns based on the desired number of HRIRs to interpolate. A brief overview of the concept can be found in Figure 1.

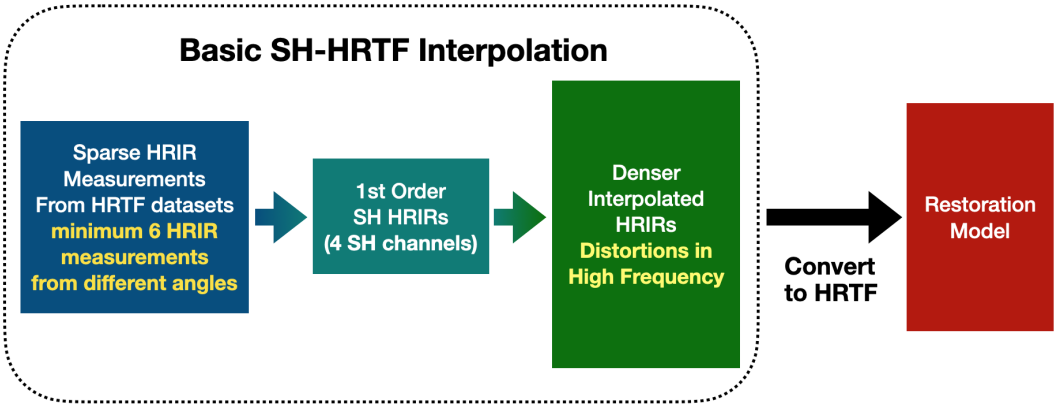


Figure 1. An overview of the proposed method wherein a model was trained to reconstruct the distorted high frequency information in the SH-interpolated HRTFs.

The main issue caused by SH HRTF interpolation is that the interpolated HRTFs are only accurate up to the spatial aliasing frequency f_{lim}, approximated by

$$f_{lim} \approx \frac{\mathrm{c}M}{4r(M+1)\sin(\pi/(2M+2))} \approx \frac{\mathrm{c}M}{2\pi r} \tag{5}$$

where c is the speed of sound, approximated as 343 m/s at 20 °C in air, M is the spherical harmonics order and r is the radius of the human head [27]. For 1st order, the spatial aliasing frequency is around 700 Hz.

The spectral distortions will not only affect the timbre, but will also degrade the localisation performance since the important cues for identification of source elevation are changed, as shown in Figure 2.

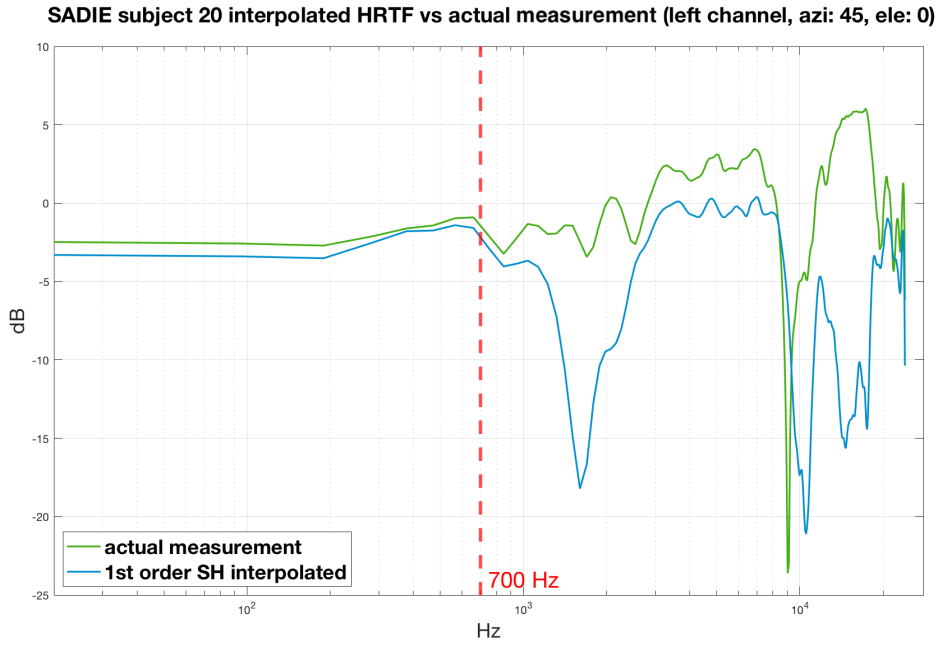


Figure 2. Actual ipsilateral HRTF measurement vs. SH-interpolated HRTF (green: Actual HRTF measurement, blue: SH-interpolated HRTF), $M = 1$.

To combat this, we also employed dual-band Time Alignment (TA) in the encoding of the HRTFs [28,29]. Given ITD is only effective at low frequencies, we can time-align the high frequencies of the HRTFs when undertaking SH encoding, thereby removing high frequency ITD. By doing this, lower order SH HRTFs are more effective at preserving high frequency information, which improves interpolation results. In this study, the crossover frequency was set at 2.5 kHz, as suggested in [29].

3. Machine Learning HRTF Restoration

Research domains such as speech recognition, natural language processing and computer vision have demonstrated that a more general data-driven method often beats traditional knowledge-based signal processing methods in the long run, as the data can keep growing in the future [30]. In image processing, recent developments in machine learning have shown great potential in noise reduction in images, image inpainting, colorising old photos or videos and neural style transfer [17,18,31–37]. These tasks can be considered to be quite similar to restoring distorted SH-interpolated HRTFs. Some examples include variants of fully connected neural networks (NN), auto-encoder (AE) convolution auto-encoder (CAE) [36], residual network (ResNet) [38] and generative adversarial networks (GANs) [39].

Most machine learning models require large amounts of labelled data to produce excellent results. However, currently the number of available HRTF databases is very limited. There are only a total of 233 HRTF datasets freely available combined in (Spatially Oriented Format for Acoustics) SOFA format at the time of conducting this research [40,41]. Compared to the data size used to train image processing machine learning models, which can be in the region of hundreds of thousands of images, HRTF data are far too few to generalise well or to train sophisticated models. However even with such limited data, SH-interpolated HRTF restoration is potentially achievable using machine learning algorithms and is investigated herein. The advantage in using a machine learning model is that the result can be improved in the long run when more labelled data are available in the future. The disadvantage is that current methods are somewhat tedious in actual implementation as they require an extra step after interpolation and can take up significant processing time if the number of interpolated HRTFs is large.

An overview of the proposed method is shown in Figure 1. A subset of HRTFs are selected from a database to represent a sparse HRTF measurement set. These HRTFs are then interpolated in a

traditional SH HRTF interpolation manner. After the interpolation, each HRTF measurement is fed into the machine learning model for restoration. The output size for this study was chosen for the interpolation process based on the number of HRTF measurements of the original dataset, which is typically over 2000 measurements (depending on the dataset). The restored HRTFs output from the machine learning model can then be easily compared with the true HRTF measurements. In this section, we first discuss the data preparation and format, and then introduce the baseline model used in this study before improving it with different enhancement techniques, including weight decay, dropout, early stopping, etc.

3.1. Data Pre-Processing

The training and testing data were extracted from different HRTF databases, including ARI [42], ITA [43], RIEC [44], SADIE I [45], SADIE II [9], IRCAM Listen [46] and the Bernschutz KU100 [47].

The SH interpolation process takes place in the time domain, as HRIRs which are then converted to frequency domain HRTFs after the interpolation process for input to the restoration model. The phase is discarded after the conversion because using complex numbers in conventional NN and CNN can be problematic for certain functions. Whilst alternative methods for using complex numbers have been proposed, it is questionable whether these nontrivial methods are necessary for this project [48–51]. Note that due to the randomness of machine learning models, there is a chance that a model could incorrectly give negative amplitude spectra in the output. To avoid this, the input and output data were scaled to decibels. The output data were then rescaled before converting back to the time domain.

3.1.1. Data Selection

We used 6 measurements with an octahedron configuration for the sparse dataset, which is one of the more challenging cases for SH HRTF interpolation, as it involves the second lowest number of measurements in common configurations for 1st order SH [8]. There were two sets of configurations used, one used in both training and testing, the another one used only in training for data augmentation purpose. The angles are shown in Tables 1 and 2 and Figure 3.

Table 1. Angle selection for training, validation and testing.

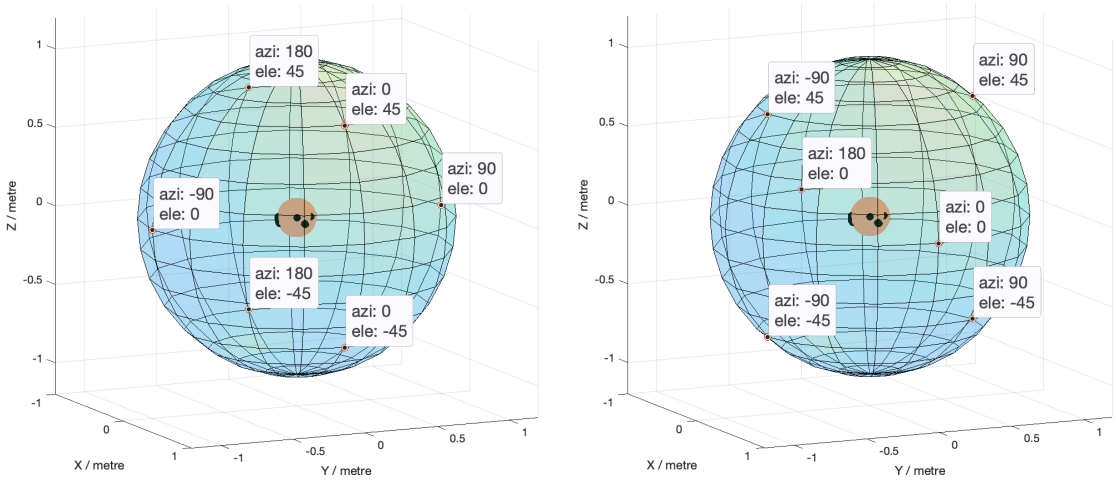
	Azimuth	Elevation
1	90.0	0.0
2	270.0 (or −90.0)	0.0
3	0.0	45.0
4	0.0	−45.0
5	180.0	45.0
6	180.0	−45.0

Table 2. Angle selection for training and validation only.

	Azimuth	Elevation
1	0.0	0.0
2	180.0	0.0
3	90.0	45.0
4	90.0	−45.0
5	270.0 (or −90.0)	45.0
6	270.0 (or −90.0)	−45.0

Amongst all the popular HRTF datasets, only the SADIE I [45], SADIE II [9], IRCAM Listen [46] and Bernschutz KU100 [47] databases can provide the measurements from these angles. Subjects 19

and 20 from the SADIE II database were held out for testing and evaluation of the model and were not used in the training. SADIE II Subject 20 was tracked during the training process. This design tries to show how well the model copes with unforeseen HRTF measurements. Furthermore, the Bernschutz HRTFs, measured from a Neumann KU100 dummy head, were also excluded from the training and validation sets but tracked during the training process. This allowed us to study the effect of alternate HRTF measurement methods of expected near-match datasets to the existing KU100 measurements in the training data. By tracking the loss of the test data during training, we could also observe whether there is any over-fitting or under-fitting with different unforeseen measurements (SADIE II test data) or different measurement methods (Bernschutz KU100) separately.



(**a**) (**b**)

Figure 3. Selected HRTF data points (**a**) Angle selection for training, validation and testing. (**b**) Angle selection for training and validation only.

Since only the SADIE, IRCAM and Bernschutz datasets had the required measurements for training and validation, it was challenging to produce an accurate model with such a limited variation of HRTF data. Consequently, data augmentation of other HRTF datasets was undertaken to provide some extra data for training and validation. The ARI [42], ITA [43] and RIEC [44] datasets were included with modified angles, i.e., positions with an elevation angle at −45° were changed to −30°. This modification was also undertaken for the RIEC dataset, and positions with an elevation angle at 45° had that changed to 50°. The effect of this data augmentation is demonstrated in Section 3.2. A summary of the data selection is shown in Table 3.

Table 3. A summary of data selection.

HRTF Dataset	Training and Validation	Testing
SADIE I	✓	
SADIE II (besides subject 19 and 20)	✓	
IRCAM Listen	✓	
ARI	modified [a]	
ITA	modified [a]	
RIEC	modified [b]	
SADIE II (subject 19 and 20)		✓
Bernschutz KU100		✓

[a] Positions with an elevation angle at −45° were changed to −30°. [b] Positions with an elevation angle at −45° were changed to −30°, elevation angle at 45° changed to 50°.

Once all the training and validation HRTFs were concatenated, 50,000 measurements were randomly selected for the training and validation set with an 80:20 ratio, considering practical training time and the limitations of available computer memory (see Section 3.2).

To improve the speed and stability in the training process, it is considered good practice to standardise the input data before feeding it into the machine learning model. The standardisation equation is as follows:

$$z = \frac{x - \mu}{\sigma} \tag{6}$$

where x is the input data; μ and σ are the mean and standard deviation of the training and validation data, given by:

$$\mu = \frac{\sum x}{N} \tag{7}$$

$$\sigma = \sqrt{\frac{\sum (x - \mu)^2}{N}} \tag{8}$$

where N is the total number of training and validation data. Note that the same μ and σ should be used for test data.

To summarise, in total there were 230 subjects used in the training, from SADIE I (18 subjects), SADIE II (18 subjects not counting the 2 hold out datasets), IRCAM Listen (51 subjects), ARI (60 subjects), ITA (45 subjects) and RIEC (38 subjects). Subjects 19 and 20 from SADIE II and the KU100 measurement from Bernschutz were held out as test sets. Data were standardised before training. The data can be found in: https://github.com/Benjamin-Tsui/SH_HRTF_Restoration (Supplementary material).

3.2. Baseline Model

As mentioned in Section 3, there are numerous ML (machine learning) models throughout the literature that have the potential for SH-interpolated HRTF restoration as they have shown some promising results in similar tasks in the visual domain. We aimed to find a model that has a simple architecture whilst being able to produce viable results. The reason for using a simple model is based on the consideration of the limited number of HRTF datasets, i.e., a simpler model is less likely to over-fit the training data. For comparison, all the models in this paper were trained with 500 epochs. The majority were trained with an NVIDIA Quadro P4000M GPU with 32 GB of computer RAM. For the models with extensive amount of data in Section 3.3.6, a NVIDIA GeForce RTX2080 Ti with 40 GB of computer RAM (random-access memory) was used.

The proposed model can be seen as a simplified version of an inception network [52]. Separate models for left and right channels are trained individually, whilst the input of the model takes both channels to provide additional information which improves the results as shown in Table 4. Here, the overall mean is the average MSE loss across all training, validation and test results and the test mean is the average MSE loss of SADIE subject 20 and Bernschutz KU100 test data. The phase difference between two channels was handled with dual-band time alignment (TA), as discussed in Section 2. The model input size was 2×256 (left and right channels of the interpolated HRTFs with the length of 256 samples per channel) and the output was 256 (either left or right channel).

Table 4. Comparison between stereo input and mono input with the baseline model demonstrating that stereo input performs better than mono input.

Comparison of Results Between Mono and Stereo Inputs (Lower Is Better)						
Model	**Overall Mean**	**Training**	**Validation**	**SADIE 20**	**Bernschutz**	**Test Mean**
Baseline (mono)	50.66	31.40	33.94	62.27	75.04	68.65
Baseline (stereo)	44.46	28.17	30.17	52.34	67.17	59.75

The model proposed here uses a combination of a convolutional auto-encoder (CAE) and a denoising auto-encoder (DAE) [53]. Preliminary test results showed that the DAE is better with the main contour of the frequency response (Figure 4) and CAE is better with the finer details (Figure 5). The combination of the two yields positive results. Similar results were also observed in image research [54].

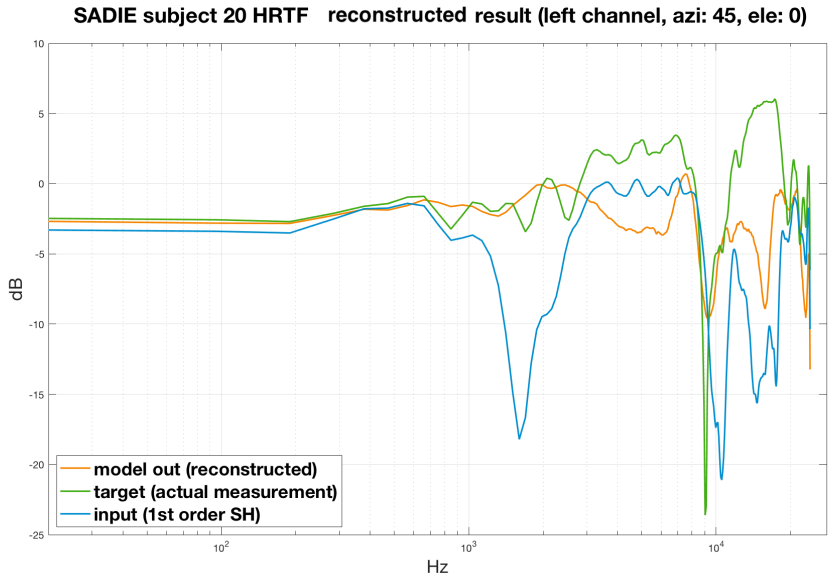


Figure 4. Restoration result example for ipsilateral HRTF response with DAE only (orange: model restored output, green: actual HRTF measurement, blue: SH-interpolated HRTF.)

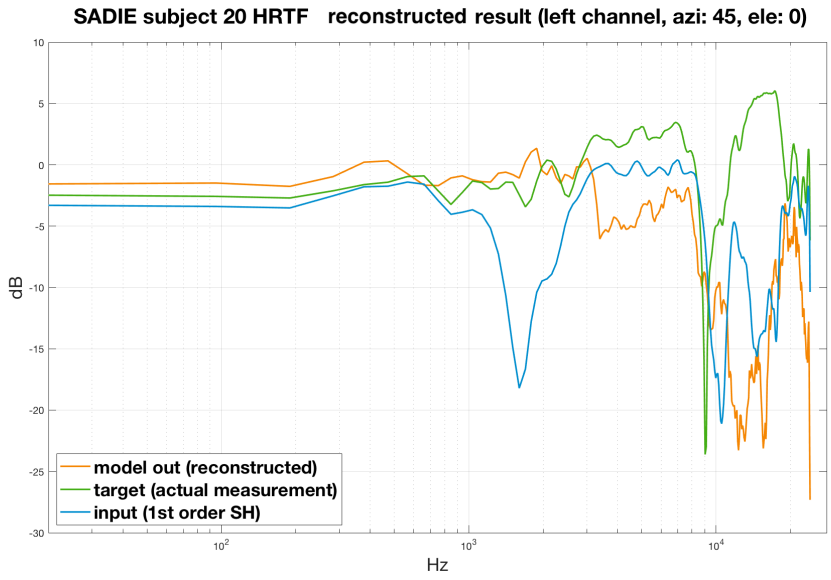


Figure 5. Restoration result example for ipsilateral HRTF response with CAE only (orange: model restored output, green: actual HRTF measurement, blue: SH-interpolated HRTF).

The results from the convolutional CAE and DAE are concatenated and passed through a fully connected layer for the voting process.

The complete model is shown in Figure 6. Note that batch normalisation is performed after each convolution layer and transposed convolution layer, with the exception of the very last transposed convolution layer so the output of the CAE should have similar magnitude with the DAE. The models are built and trained with PyTorch [55,56] using smooth L1 loss with Adam optimiser (learning rate: 0.000001, beta 1:0.9, beta 2:0.999).

Different loss functions were compared and the results are shown in Table 5. Note that all the losses were calculated with the HRTF values in dB. The mean square error (MSE) loss, also known as the L2 loss, is given by the following equation:

$$\text{L2 Loss} = \sum_{i=1}^{n} \left(y_{\text{label}} - y_{\text{predicted}} \right)^2 \tag{9}$$

where y_{label} is the target value of the output and $y_{\text{predicted}}$ is the prediction from the model. The MSE loss performs worse in the test data but slightly better in the training and validation data. L1 loss, also known as mean absolute error (MAE), is given by the following equation:

$$\text{L1 Loss} = \sum_{i=1}^{n} \left| y_{\text{label}} - y_{\text{predicted}} \right| \tag{10}$$

MAE showed key improvements with the SADIE Subject 20 dataset and slight improvements with the Bernschutz KU100 test data. One reason L1 loss out-performs MSE loss might be that L1 loss is usually less sensitive to outliers, such as those in the case here, wherein there are HRTFs from different databases. Smooth L1 loss is a combination of L1 loss and MSE loss expressed with the following equation:

$$\text{Smooth L1 Loss} = \begin{cases} 0.5(y_{\text{label}} - y_{\text{predicted}})^2, & \text{if } |y_{\text{label}} - y_{\text{predicted}}| < 1 \\ |y_{\text{label}} - y_{\text{predicted}}| - 0.5, & \text{otherwise} \end{cases} \tag{11}$$

For an error below 1, Smooth L1 loss performs as the MSE loss function; and for above 1 it performs as L1 loss. Compared to L1 loss, this method has a continuous derivative at zero so it provides a smoother gradient when the error gets smaller than 1. The result of Smooth L1 loss further improved the SADIE 20 dataset but there was a trade-off with the Bernschutz dataset. Considering the real world application, it is more practical to optimise for unforeseen HRTF measurements of different human subjects instead of different measurement methods with the same artificial head model. The same principle holds in the further optimisation techniques in the upcoming tests. Therefore, the models in this paper use Smooth L1 loss as the loss function. Note that these loss functions only compare the difference between the model output and target. Whilst they can indicate a model's performance to some extent, they may not represent human perceptual response, although they are easier to back-propagate than perceptual models. Nonetheless, the proposed model from this paper is further evaluated with perceptual models in Section 4.

Table 5. Comparison between different loss functions demonstrating that Smooth L1 loss performs the best in SADIE II test data.

MSE With Different Loss Functions (Lower the Better)						
Model	**Overall Mean**	**Training**	**Validation**	**SADIE 20**	**Bernschutz**	**Test Mean**
Baseline (MSE loss)	44.97	27.38	29.43	58.14	64.92	61.53
Baseline (L1 loss)	44.04	28.16	30.14	53.66	64.18	58.92
Baseline (Smooth L1 loss)	44.46	28.17	30.17	52.34	67.17	59.75

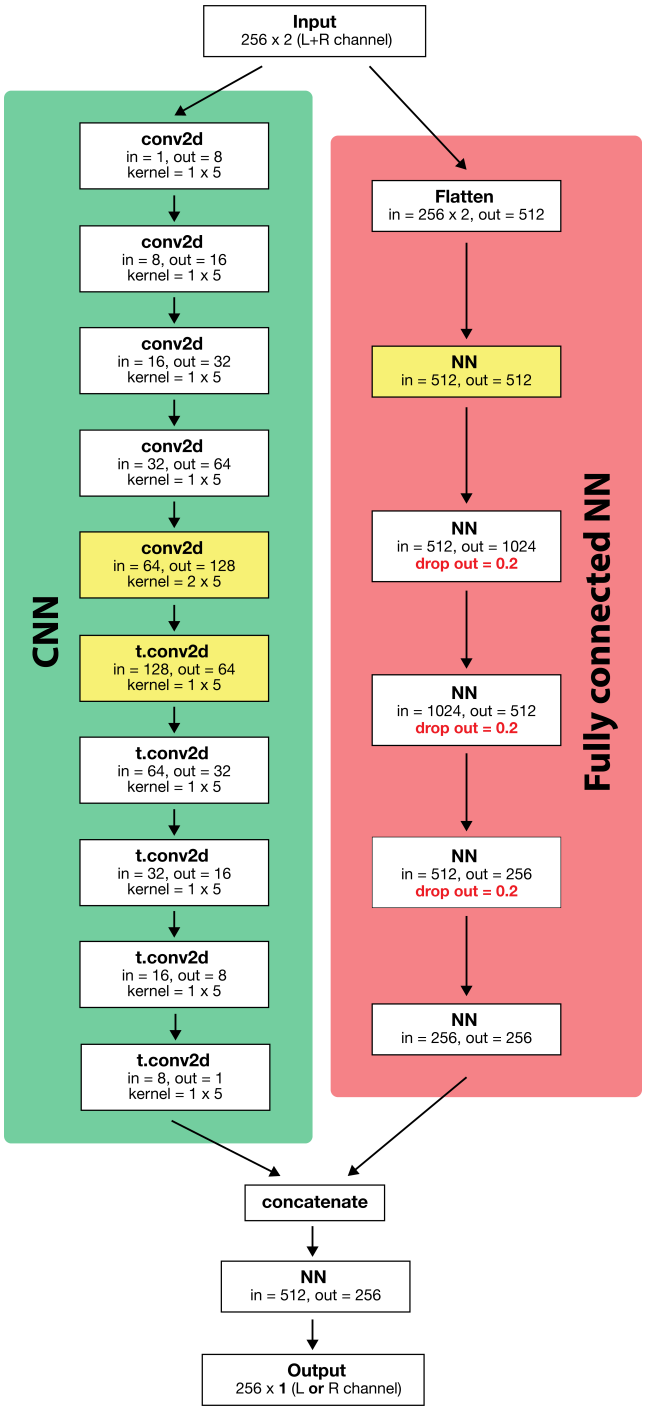


Figure 6. Baseline model, smaller model (smaller model removes the layers highlighted in yellow) and proposed model (proposed model uses drop out in some NN layers, shown in red, amongst other techniques discussed in Section 3.3.4).

The baseline model was trained for 500 epochs with a batch size of 8. A small batch size was used since prior research showed that a larger batch size may produce worse performance [57,58].

Figures 7 and 8 show the MSE changes during the training. The blue and orange lines are the training and validation results respectively. The green and red lines are the test sets of Subject 20 from SADIE II database and Bernschutz KU100 measurements accordingly. The results show that the training and validation results trend downwards while the two test sets flatten out after the first 20 epochs. This indicates that the models over-fit the training data. Over-fitting is normal in this case considering the limited number of HRTF datasets in the training data (230 in total).

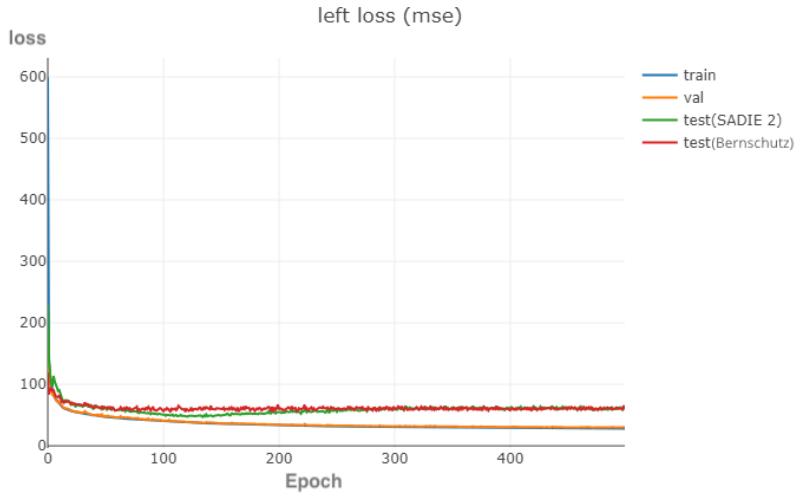


Figure 7. Left MSE during training (blue: training loss, orange: validation loss, green: hold out test data Subject 20 from SADIE II, red: Bernschutz KU100), which shows that the model over-fit the training data.

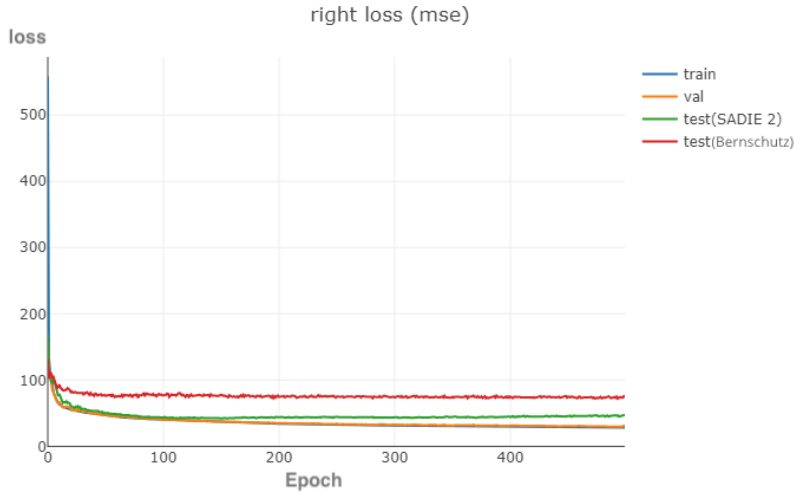


Figure 8. Right MSE during training (blue: training loss, orange: validation loss, green: hold out test data Subject 20 from SADIE II, red: Bernschutz KU100), demonstrating that the model over-fit the training data.

The effects of data augmentation are shown in Table 6, where it can be seen that there are some drawbacks with the training and validation data, and a minor drawback with the SADIE Subject 20 test data. However, the extra data provided a significant improvement with the Bernschutz interpolation. This demonstrates that the extra variety of measurements helps the model generalise better across different measurement methods.

Table 6. Mean squared error (MSE) with and without data from ARI, ITA and RIEC demonstrating that using additional data improves the result with the Bernschutz test data significantly.

Comparison of Results with and without Data from ARI, ITA and RIEC (Lower is Better).						
Model	**Overall Mean**	**Training**	**Validation**	**SADIE 20**	**Bernschutz**	**Test Mean**
Baseline (without extra data)	45.28	18.29	20.55	51.33	90.93	71.13
Baseline (with extra data)	44.46	28.17	30.17	52.34	67.17	59.75

The most ideal way reduce over-fitting is to train with more data. However, given the limited HRTF measurements available, different regularisation techniques can be utilised to improve the baseline result, which will be discussed in this section.

Figures 7 and 8 show there were some differences between the SADIE hold out test data and Bernschutz KU100 data, although the difference was less on the left channel. However, the average MSE test error of the two channels was approximately the same (left: 59.874, right: 59.633). Further investigations were required to establish the cause of the differences in the left and right channels.

On the other hand, according to Figures 7 and 8, it is interesting to see that the results with the Bernschutz KU100 data also suffered from over-fitting. As there were different KU100 HRTFs in the training data, and the KU100 measurements in the SADIE II database were very similar to the Bernschutz KU100 measurements, it was unexpected to see the model perform quite poorly when comparing the training and validation results. More oddly, in the later sessions, it showed that the Bernschutz KU100 measurements did not seem to benefit from any regularisation methods. One plausible hypothesis is that the current model only trained with 6 HRTF databases which represents 6 different measurement setups. The model requires a larger variety of inputs to be able to generalise across different measurement setups and methods.

3.3. Model Enhancement

3.3.1. Smaller Model

To address the over-fitting problem, an effective method is to decrease the complexity of the model, by decreasing the number of parameters. A smaller model was trained with a convolution and transposed convolution layer pair removed in the CAE and a NN layer removed from the fully connected NN (Figure 6 with the yellow highlighted layers removed). The result does not seem to have significant improvement on the SADIE II Subject 20 dataset. However, it substantially increased the MSE with the Bernshutuz data from 67.17 to 89.35 (Table 7 Model B).

Table 7. Comparison of MSE among different models. This table shows that the bigger model without weight decay and trained with extra data (Model I) performed the best with the Bernschutz test data, and also generalised better with different measurement methods (in bold). However, for the SADIE II Subject 20 test data, the proposed model (Model E) and the early stopped proposed model (Model F) performed the best among all models (in bold).

Compare the Results From Different Models (Lower the Better)						
Model	**Overall Mean**	**Training**	**Validation**	**SADIE 20**	**Bernschutz**	**Test Mean**
A. Baseline	44.46	28.17	30.17	52.34	67.17	59.76
B. Smaller model	45.51	19.54	21.52	51.62	89.35	70.48
C. With weight decay	46.04	28.59	30.96	52.75	71.89	62.32
D. With dropout	46.47	29.14	30.08	54.52	72.15	63.33
E. With weight decay and dropout (proposed model)	45.48	29.85	30.61	**47.21**	74.23	60.72
F. With weight decay and dropout (early stopped at 111 epoch)	49.78	40.92	41.00	**47.18**	70.04	58.61
G. Baseline trained with extra data	39.36	19.74	20.09	59.87	57.72	58.80
H. With weight decay and dropout and trained with extra data	41.44	22.49	22.07	59.69	61.50	60.60
I. Bigger model without weight decay and trained with extra data	**31.38**	**7.83**	**10.61**	56.88	**50.22**	**53.55**

3.3.2. Model With Weight Decay

Weight decay is also known as L^2 parameter regularisation or ridge regression. This is a common regularisation method for reducing over-fitting in training. The idea of weight decay is to penalise the large weights in order to simplify the model and reduce over-fitting [59,60]. We used a weight decay rate of 0.001 as an experiment to see the effect of this regularisation method. In theory, a higher the weight decay rate should have a stronger regularisation effect and 0.001 is a reasonable value to start testing with weight decay [61,62]. Note that our study aims to demonstrate the concept of using ML models for SH HRTF restoration by showing some preliminary results with various methods as guidance for future research. Fine tuning each parameter in the model is beyond the scope of this paper. The result did not seem to have any positive impact on the SADIE II Subject 20 dataset and there was a main drawback with the Bernschutz KU100 data as the error increased from 67.17 to 71.89 (Table 7 Model C).

3.3.3. Model With Dropout

Dropout randomly "drops out" a percentage of nodes in the neural network during training. The idea is to avoiding co-adaption between nodes by never guaranteeing that any pair will both be used during the training process to avoid the model over-relying on a few nodes within a layer [63–65]. It can also been seen as randomly sampling from the exponential number of possible narrow sub-networks during training, and then providing an average the performance of all these combinations in test time or application. Note that a dropout layer can only apply on fully connected layers but not convolutional layers. The model uses a 20 percent dropout ratio on the second to fourth fully connected layer.

The model (Model D in Table 7) produces worse results with the test data compared to the baseline model, especially with the hold out SADIE II data, but performs slightly better with the validation data. However, such slight differences may be introduced by the randomness of machine learning training. According to the result, dropout seems to have a more negative impact on regularisation compared to weight decay. In theory it is possible to increase the dropout ratio or use a different configuration to increase the regularisation effect. However, finding the optimal architecture and hyper-parameters is beyond the scope of this paper and could be part of the future work.

3.3.4. Combining Weight Decay and Dropout

Combining weight decay and dropout showed the best result in the hold out SADIE II data despite there being a noticeable trade-off with the Bernschutz dataset. This result in Table 7 Model E indicates that by combining weight decay and dropout, the model can generalise better across different unforeseen HRTF subjects (SADIE hold out) but not the measurement method (Bernschutz). As this model performed the best with the SADIE II test data, this became the proposed model to be further analysed in Section 4.

It was interesting to find that the combination of the two different methods showed a large difference in results, but not with either method individually. It is not clear whether the improvement comes from the combination of the techniques or is from the cumulative regularisation power. This could be an individual research topic to be investigated in the future.

3.3.5. Early Stopping

Early stopping is one of the regularisation methods that sometimes is not considered as good practise in machine learning training because it breaks the principle of orthogonalisation and makes hyper-parameter tuning difficult [66]. Another reason this method is controversial is because the result can be hard to reproduce and compare across different models. However, according to the learning curve from the model combining weight decay and dropout in Figures 9 and 10, early stopping should perform slightly better with the test data, especially with the test set from SADIE II. In order to

demonstrate the effect, this paper retrained the proposed model and stopped training at 111 epochs, as it is the lowest point in Figures 9 and 10.

The result in Table 7 Model F shows that there was a very slight improvement with the SADIE II Subject 20 test set and a more noticeable improvement with the Bernschutz KU100 test data.

3.3.6. Training With More Data

To shorten the training time to compare across different methods and considering the limited size of RAM, the models discussed above were trained with 50,000 randomly sampled HRTFs from different angles of the training and validation HRTF sets. However, the best way to reduce over-fitting is to increase the size of the training set. To investigate what the model capable of with more data, a baseline model was trained with 633,000 HRTF measurements. Training this amount of data can take a lot of time per epoch. To speed up the process the batch size for training increased to 32, whilst the validation and test sets remained the same at eight for better comparison.

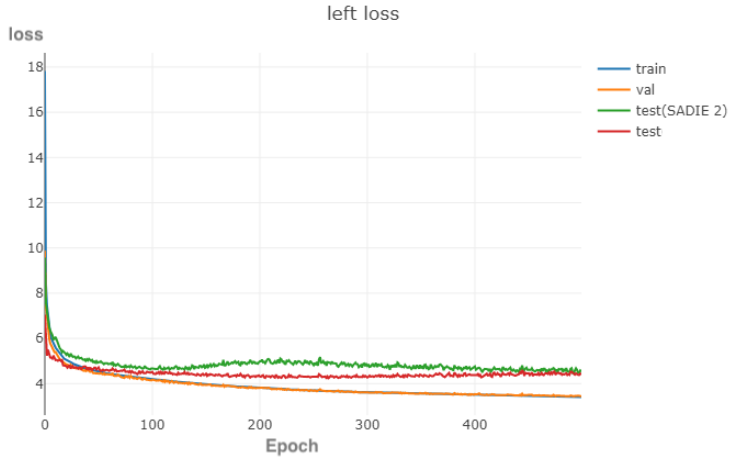


Figure 9. Left channel L1 loss during training (blue: training loss, orange: validation loss, green: hold out test data Subject 20 from SADIE II, red: Bernschutz KU100), which shows that the lowest point in the curve is at around 110 epochs.

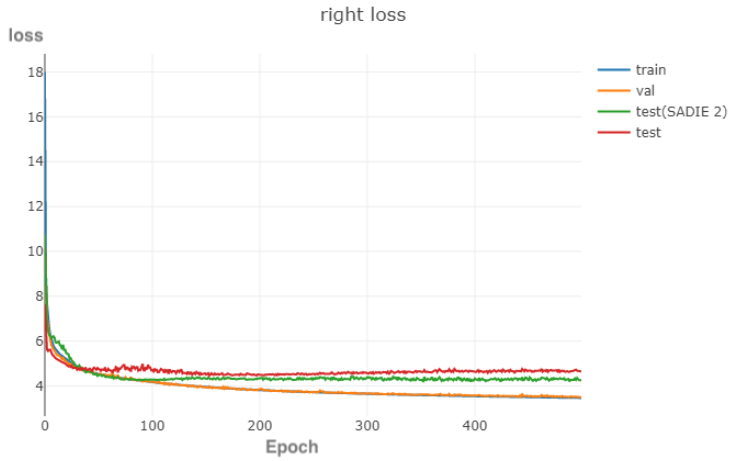


Figure 10. Right channel L1 loss during training (blue: training loss, orange: validation loss, green: hold out test data Subject 20 from SADIE II, red: Bernschutz KU100), which shows that the lowest point in the curve is at around 110 epochs.

Two models, the baseline model (Table 7 Model G) and the model with weight decay and dropout were trained with extra data (Table 7 Model H). The baseline model trained with extra data showed major improvement with the Bernschutz dataset, alongside the training and validation sets. However, there was also a noticeable trade-off with the SADIE II Subject 20 test data. It is believed that the improvement in the Bernschutz dataset is the result of the model having more examples of the KU100 HRTFs with different measurements at different angles, so it can generalise the measurement method better. The trade-off in the SADIE II Subject 20 test data may have been caused by the increased batch size, which can induce worse performance [57,58]. Nevertheless, as the extra data were within the same distribution, it is quite unlikely they could have provided any noticeable performance improvement with unforeseen HRTF measurement subjects.

3.3.7. Bigger Model

Considering the current results and the limited number of labelled data, training a bigger model is against normal machine learning practices. However, to demonstrate the potential capability of the proposed method and insight for future research, a slightly deeper model was also trained with extra data. The goal was to minimise the training and validation error as much as possible, neglecting the trade-off in test datasets' results.

To balance out the model size and training time, only the convolution neural network was changed. An extra convolution layer and transposed convolution layer pair was added in the convolution model (Figure 11).

As this model only focuses on the test and validation results, dropout and weight decay regularisation methods are lifted, which defeats the purpose of using a bigger neural network. The model was trained with 633,000 HRTFs with a batch size of 32 for training, similarly to Section 3.3.6, as bigger models usually work better with more data.

Compared to the baseline model, the training time of each epoch from the bigger model (Model I) increased from 6 min to 26 min. The model was trained with 500 epochs and the results are shown in Table 7. As expected, the model provided huge improvements in training and validation, but not much of an improvement in the SADIE II test data. On the other hand, it provided the best performance for the Bernschutz data. Comparing the results of the smaller model, baseline model and the bigger model, it seems like the bigger model has better performance with the Bernschutz data which indicates it generalises better across different measurement methods.

3.3.8. Summary

According to the results in Table 7, it is clear that there is room for improvement through some enhancements of the baseline model. The baseline model with weight decay and dropout (Model E) and baseline model with weight decay, dropout and early stopping (Model F) provide the best results with SADIE II test data, yet the bigger model with weight decay (Model I) generalises better across different measurement methods.

As mentioned in Section 3.2, this project focuses on optimising for unforeseen HRTF measurements of different human subjects instead of different measurement methods with the same artificial head model; therefore, the baseline model with weight decay and dropout (Model E) is the proposed model in this paper.

On the other hand, according to the trends seen across the smaller model (Model B), the baseline model (Model A) and the bigger model (Model I), the performance for unforeseen measurement methods, such as the Bernschutz KU100 test data, is expected to improve when the model size increases.

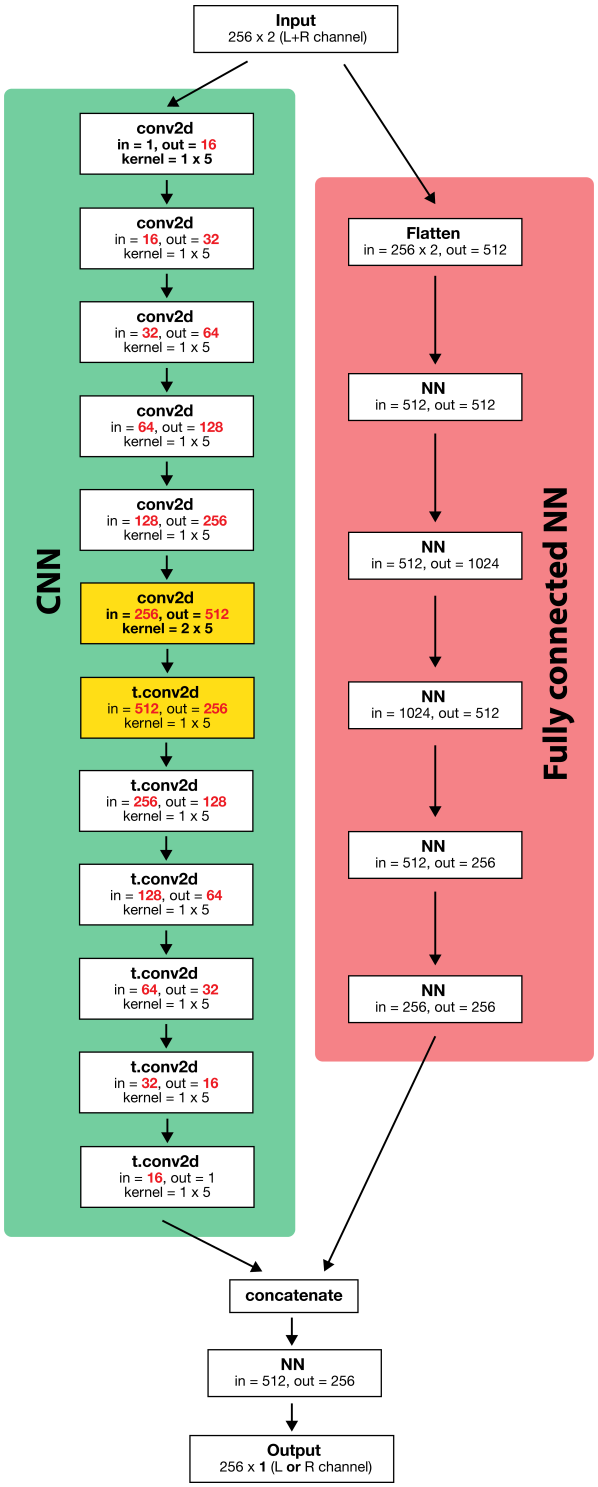


Figure 11. Wider and deeper model (highlighted: the main difference compared to the proposed model).

4. Evaluation

In this section, the results of the proposed model, including weight decay and dropout, are further analysed for perceptual difference and localisation performance. We utilised perceptual models based on these two criteria in order to provide more robust results for bench-marking.

4.1. Perceptual Spectral Difference

To formally estimate the perceptual performance, the results were further analysed with a perceptual spectral difference (PSD) model [67]. This model calculates the difference between two binaural signals or HRTFs, and presents a more accurate perceptual comparison of spectral differences as PSD. Here, we compare the difference between before and after the restoration process with the actual HRTF measurements.

The comparison between mean PSD before and after reconstruction with different HRTF datasets is shown in Table 8 and Figure 12 with the minimum and maximum PSD plotted. The results show that the model provided significant improvement in PSD across all datasets, as the mean PSD is lower. However, the minimum and maximum in Figure 12 shows that the model seems to introduce higher PSD error in some cases. As for most applications that use HRTFs, the smoothness across all angles is more crucial than the average performance. Further analysis with the box plot in Figure 13 shows that although the model may have introduced more extreme outliers with unforeseen HRTF measurement subjects (SADIE Subject 19 and 20), the model still improved the majority of the HRTFs and reduced the interquartile range (IQR) in the result. A more detailed plot across different angles is shown in Figures 14–16.

Table 8. Predicted model performance with various head related transfer function (HRTF) sets.

	SADIE 18 (Training Data)	**SADIE 19 (Hold Out)**	**SADIE 20 (Hold Out)**	**Bernschutz KU100**
PSD (sones) (SH input)	3.03	3.05	2.84	2.57
PSD (sones) (model output)	1.93	2.12	1.96	1.61
Frontal azimuth mean error (SH input)	20.81	25.36	30.00	39.67
Frontal azimuth mean error (model output)	19.29	17.47	15.84	18.98
Sagittal RMS error (deg) (SH input)	40.7	38.6	37.5	38.1
Sagittal RMS error (deg) (model output)	44.3	43.7	39.3	41.4
Sagittal quadrant errors (%) (SH input)	11.5	9.1	7.6	7.1
Sagittal quadrant errors (%) (model output)	24.8	25.2	14.7	12.4

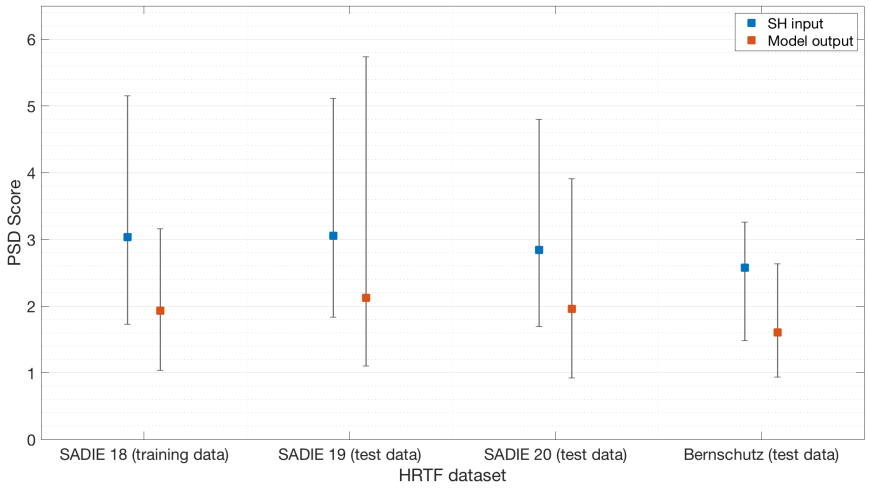


Figure 12. Minimum, maximum and average perceptual spectral difference (PSD) across different angles in different datasets which shows the model significantly improved the mean PSD from the non-reconstructed interpolated data.

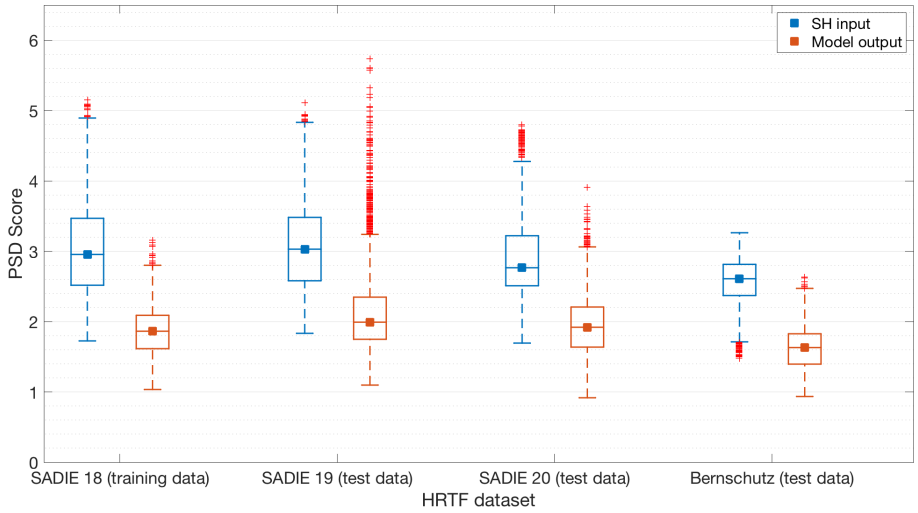
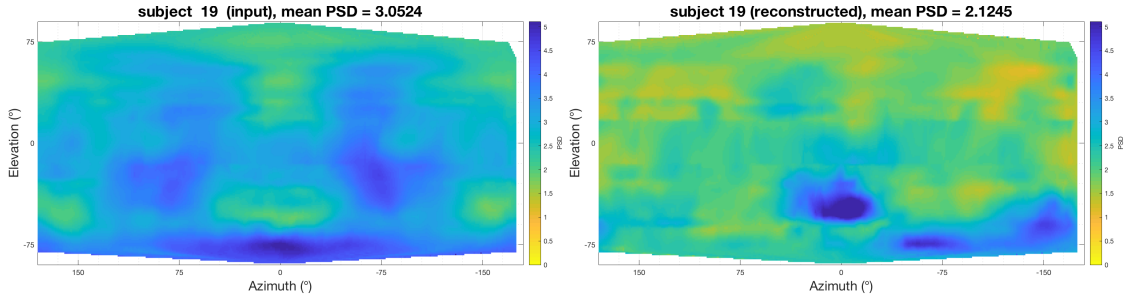
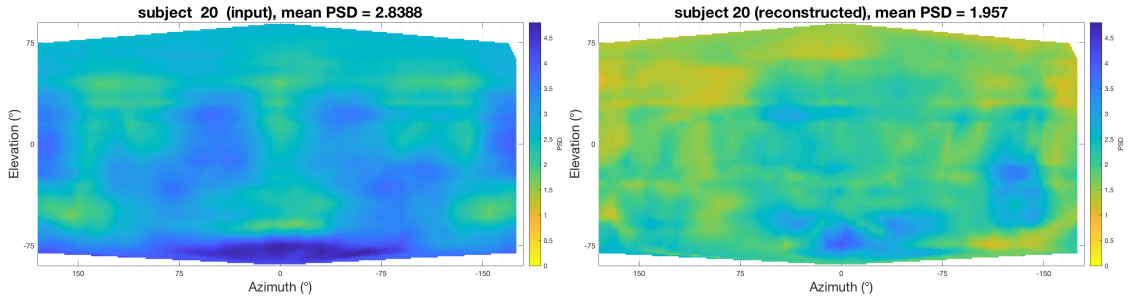


Figure 13. PSD median and box plot with whiskers with maximum 1.5 IQR which shows the model reduced the interquartile range (IQR) significantly.



(**a**) (**b**)

Figure 14. Comparing the PSD of Subject 19 from SADIE II database before and after reconstruction, which shows a noticeable improvement in most regions, besides the very low frontal region. (**a**) Before reconstruction. (**b**) After reconstruction.



(**a**) (**b**)

Figure 15. Comparing the PSD of Subject 20 from SADIE II database before and after reconstruction, which shows a noticeable improvement in most regions. (**a**) Before Reconstruction. (**b**) After Reconstruction.

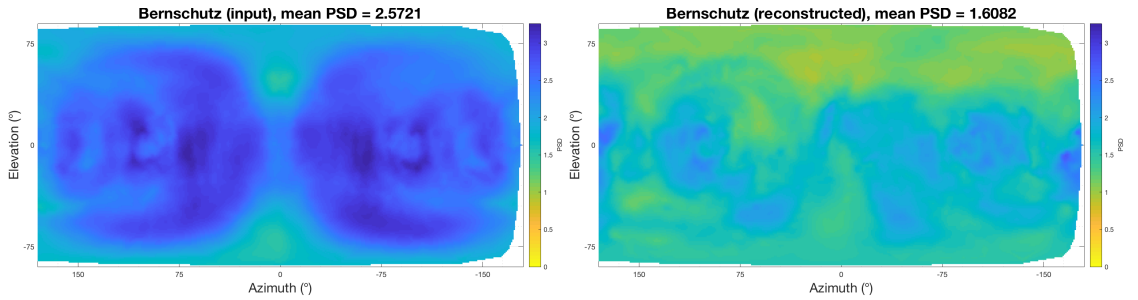


Figure 16. Comparing the PSD of Bernschutz KU100 dataset before and after reconstruction, which shows a noticeable improvement in most regions. (**a**) Before Reconstruction. (**b**) After Reconstruction.

According to Figures 12 and 13, Subject 19 from the SADIE II database had a worse maximum PSD and many outliers. To further investigate the cause of the result, Figure 14 shows the PSD before and after comparison of different angles in Subject 19 from the SADIE II database. The left shows the SH-interpolated HRTFs before restoration and the right shows the one after being processed with the ML model. The figure shows that most of the high PSD results were introduced in the lower frontal region. It is unclear what caused the increased error, but one hypothesis is that the abnormality was caused by the shadow effect from the knees, as the SADIE II along with most of the other databases were measured with subjects sitting on a chair.

However, Figure 15 shows the PSD before and after comparison with the holdout Subject 20 from SADIE II database. Besides a small area of minor PSD increment in the very low frontal region, the restored result shows there is no significant abnormality in any region. To have a deeper understanding of the cause of the abnormality in the Subject 19 holdout data, extra tests with more HRTF sets are required.

Figure 16 shows the PSD before and after comparison with the Bernschutz KU100 data. The model seemed to perform better with the unforeseen measurement method, as it showed improvements in the PSD at all angles. It is worth noticing that the lower frontal region in the figures does not have any oddly high PSD results, perhaps because KU100 is a head-only dummy-head model.

4.2. Localisation Performance

The localisation performance was analysed with May's model and Baumgatner's model in the Auditory Modelling Toolbox (AMT) [68–70]. May's model is for frontal azimuth localisation on the horizontal plane and Baumgatner's model is for the frontal sagittal plane.

Table 8 shows the localisation results of the proposed model. The frontal azimuth localisation mean error from May's model shows improvement in all HRTF datasets. However, for frontal sagittal plane with Baumgatner's model, results in RMS error and quadrant errors indicate all the reconstructed HRTFs perform worse in the frontal sagittal plane localisation.

The current model suffered in localisation tasks, perhaps because it used smooth L1 loss as the loss function. The smooth L1 loss only focuses on the magnitude difference at each frequency point. There may not be any meaningful connection between these magnitude differences and localisation performance. Therefore, it is no surprise that the model failed to optimised the HRTF restoration for the localisation performance. A future model could add some types of localisation error into the loss function to improve the localisation results.

5. Discussion and Future Work

The goal of this study was to prove the hypothesis that machine learning can be used to restore distorted interpolated HRTFs. To draw a convincing argument, this paper picked one of the more challenging situations based on 1st order SH and six measurements. Models with higher order SH

and more measurements should have better performance than the current one, as less data needs to be restored.

The results showed that a simple ML model can be used to restore distorted SH-interpolated HRTFs, although the current state of this model is far from optimised for application. It is believed that there will be significant improvements if more HRTF measurements are available for training in the future. Under the current situation, one way to improve the model is through hyper-parameter tuning, including the parameters for regularisation.

An alternative method that may reduce over-fitting is data augmentation. Currently, HRTF measurements are expensive and tedious and therefore it is not very likely that there will be a huge increase in HRTF measurement data from standard methods in the near future. To augment the current dataset, one possible way is to use more different sparse HRTF configurations or a different SH order to train the model. Table 6 shows that even if the extra data are not perfect, it is possible that it can still improve the model's performance in some cases.

Similar to data augmentation, noise injection is a different regularisation method that has be shown to work better than weight decay in some cases [71–73]. By picking the right parameters, it is believed that it could generalise better across measurement methods, as the model could focus on the general information across various HRTF measurements as opposed to the artefacts introduced by different measurement methods.

Another problem that was observed from the current model was the localisation performance. Although it shows some improvement in the horizontal plane, the sagittal error needs further improvement. As discussed in Section 4, it is believed that the smooth L1 loss function only compares the difference at each frequency and fails to capture other useful metrics of HRTFs. Potentially, some custom loss functions could be implemented to improve the model. Gatys et al. [74] trained a separate machine learning model for content loss and implemented a special function for style loss. Similar methods such as training a localisation model as a localisation loss function may be able to solve the problem. Furthermore, with the recent success in generative adversarial networks (GAN), it should be possible to build a GAN based on localisation performance [37,39,75–77]. However, as a GAN can be unstable to train and it usually requires a lot of tuning, it may not be the most effective way for SH-interpolated HRTF restoration.

This paper has shown general insights of using machine learning for HRTF reconstruction. According to Tables 7 and 8, and the discussion in Section 4, to apply the idea to real-life applications, optimising the model for some narrative tasks should yield better performance. With a more specific application in mind, not just the parameters of the model can be changed; the model can also be trained with cleaner or more particular data specialised for the task. Alternatively, using transfer learning based on the current model can provide a head-start for these applications.

6. Conclusions

HRTF interpolation in the SH domain often suffers from distortion in the high frequencies. With the recent developments in machine learning algorithms, this paper has shown that it is possible to restore the distorted SH-interpolated HRTFs with a ML model. Although the proposed method suffers from over-fitting, it still shows improvements in perceptual difference and localisation performance. It is believed that with more training data in the future, the model performance will be vastly improved. However, HRTF measurements can be difficult and time consuming to obtain. With the current amount of available data, optimising the model to a specific use case with some tuning of hyper-parameters and data augmentation should provide the best chance of making the model useful in real world applications.

Supplementary Materials: Supporting data and code are available at GitHub: https://github.com/Benjamin-Tsui/SH_HRTF_Restoration.

Author Contributions: Methodology, B.T.; writing—review and editing, B.T., W.A.P.S. and G.K.; supervision, W.A.P.S. and G.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Thanks for the technical support from Cal Armstrong and Thomas McKenzie.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AE	Auto-encoder
AMT	Auditory Modelling Toolbox
AR	Augmented Reality
CAE	Convolutional Auto-Encoder
DAE	Denoising Auto-Encoder
GANs	Generative Adversarial Networks
HRTF	Head Related Transfer Function
ILD	Interaural Level Difference
IQR	Interquartile Range
ITD	Interaural Time Difference
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Square Error
NN	Neural Networks
PSD	Perceptual Spectral Difference
RAM	Random Access Memory
ResNet	Residual Network
SH	Spherical Harmonic
SOFA	Spatially Oriented Format for Acoustics
TA	Time Alignment
VBAP	Vector Base Amplitude Panning
VR	Virtual Reality

References

1. Poirier-Quinot, D.; Katz, B.F. Impact of HRTF individualization on player performance in a VR shooter game I. In Proceedings of the AES International Conference on Spatial Reproduction, Tokyo, Japan, 7–9 August 2018; p. 7.
2. Poirier-Quinot, D.; Katz, B.F. Impact of HRTF individualization on player performance in a VR shooter game II. In Proceedings of the AES International Conference on Audio for Virtual and Augmented Reality, Redmond, WA, USA, 20–22 August 2018; p. 8.
3. Xie, B. *Head-Related Transfer Function and Virtual Auditory Display*, 2nd ed.; J. Ross Publishing: Plantation, FL, USA, 2013; p. 501.
4. Howard, D.M.D.M.; Angus, J. *Acoustics and Psychoacoustics*, 4th ed.; Focal Press: Waltham, MA, USA, 2009. [CrossRef]
5. Pulkki, V. Virtual sound source positioning using vector base amplitude panning. *J. Audio Eng. Soc.* **1997**, *45*, 456–466.
6. Gerzon, M.A. Periphony: With-height sound reproduction. *J. Audio Eng. Soc.* **1973**, *21*, 2–10.
7. Noisternig, M.; Musil, T.; Sontacchi, A.; Holdrich, R. 3D binaural sound reproduction using a virtual ambisonic approach. In Proceedings of the IEEE International Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems, VECIMS'03, Lugano, Switzerland, 27–29 July 2003; pp. 174–178.

8. Kearney, G.; Doyle, T. Height Perception in Ambisonic Based Binaural Decoding. In Proceedings of the Audio Engineering Society Convention 139, New York, NY, USA, 29 October–1 November 2015.
9. Armstrong, C.; Chadwick, A.; Thresh, L.; Murphy, D.; Kearney, G. Simultaneous HRTF Measurement of Multiple Source Configurations Utilizing Semi-Permanent Structural Mounts. In Proceedings of the Audio Engineering Society Convention 143, New York, NY, USA, 18–21 October 2017.
10. Armstrong, C.; Thresh, L.; Murphy, D.; Kearney, G. A perceptual evaluation of individual and non-individual HRTFs: A case study of the SADIE II database. *Appl. Sci.* **2018**, *8*, 2029. [CrossRef]
11. Lee, G.W.; Kim, H.K. Personalized HRTF modeling based on deep neural network using anthropometric measurements and images of the ear. *Appl. Sci.* **2018**, *8*, 2180. [CrossRef]
12. Katz, B.F. Boundary element method calculation of individual head-related transfer function. I. Rigid model calculation. *J. Acoust. Soc. Am.* **2001**, *110*, 2440–2448. [CrossRef]
13. Young, K.; Kearney, G.; Tew, A.I. Loudspeaker Positions with Sufficient Natural Channel Separation for Binaural Reproduction. In Proceedings of the Audio Engineering Society International Conference on Spatial Reproduction—Aesthetics and Science, Tokyo, Japan, 7–9 August 2018.
14. Young, K.; Tew, A.I.; Kearney, G. Boundary element method modelling of KEMAR for binaural rendering: Mesh production and validation. In Proceedings of the Interactive Audio Systems Symposium, York, UK, 23 September 2016; pp. 1–8.
15. McKeag, A.; McGrath, D.S. Sound field format to binaural decoder with head tracking. In Proceedings of the Audio Engineering Society 6th Australian Reagional Convention, Melbourne, VIC, Australia, 10–12 September 1996; pp. 1–9.
16. Noisternig, M.; Sontacchi, A.; Musil, T.; Holdrich, R. A 3D Ambisonic Based Binaural Sound Reproduction System. In Proceedings of the 24th AES International Conference on Multichannel Audio, The New Reality, Banff, AB, Canada, 26–28 June 2003.
17. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2242–2251. [CrossRef]
18. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 5967–5976. [CrossRef]
19. Gamper, H. Head-related transfer function interpolation in azimuth, elevation, and distance. *J. Acoust. Soc. Am.* **2013**, *134*, EL547–EL553. [CrossRef]
20. Grijalva, F.; Martini, L.C.; Florencio, D.; Goldenstein, S. Interpolation of Head-Related Transfer Functions Using Manifold Learning. *IEEE Signal Process. Lett.* **2017**, *24*, 221–225. [CrossRef]
21. Hartung, K.; Braasch, J.; Sterbing, S.J. Comparison of different methods for the interpolation of head-related transfer functions. In Proceedings of the AES 16th International Conference: Spatial Sound Reproduction, Rovaniemi, Finland, 10–12 April 1999; pp. 319–329.
22. Martin, R.L.; McAnally, K. *Interpolation of Head-Related Transfer Functions*; Air Operations Division Defence Science and Technology Organisation: Canberra, ACT, Australia, 2007.
23. Evans, M.J.; Angus, J.A.S.; Tew, A.I. Analyzing head-related transfer function measurements using surface spherical harmonics. *J. Acoust. Soc. Am.* **1998**, *104*, 2400–2411. [CrossRef]
24. Zotter, F.; Frank, M. *Ambisonics*, 1st ed.; Springer Topics in Signal Processing Series; Springer International Publishing: Cham, Switzerland, 2019; Volume 19. [CrossRef]
25. Chapman, M.; Ritsch, W.; Musil, T.; Zmölnig, I.; Pomberger, H.; Zotter, F.; Sontacchi, A. A Standard for Interchange of Ambisonic Signal Sets Including a file standard with metadata. In Proceedings of the Ambisonics Symposium 2009, Graz, Austria, 25–27 June 2009; pp. 25–27.
26. Daniel, J. Représentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimédia. Ph.D. Thesis, University of Paris VI, Paris, France, 2000.
27. Bertet, S.; Daniel, J.; Moreau, S. 3D Sound Field Recording with Higher Order Ambisonics—Objective Measurements and Validation of Spherical Microphone. In Proceedings of the 120th Audio Engineering Society Convention, Paris, France, 20–23 May 2006.
28. Zaunschirm, M.; Schörkhuber, C.; Höldrich, R. Binaural rendering of Ambisonic signals by head-related impulse response time alignment and a diffuseness constraint. *J. Acoust. Soc. Am.* **2018**, *143*, 3616–3627. [CrossRef]

29. Mckenzie, T.; Murphy, D.T.; Kearney, G. An Evaluation of Pre-Processing Techniques for Virtual Loudspeaker Binaural Ambisonic Rendering. In Proceedings of the EAA Spatial Audio Signal Processing symposium, Paris, France, 6–7 September 2019; pp. 149–154. [CrossRef]
30. Sutton, R. The Bitter Lesson. 2019. Available online: http://www.incompleteideas.net/IncIdeas/BitterLesson.html (accessed on 29 October 2019).
31. Yang, C.; Lu, X.; Lin, Z.; Shechtman, E.; Wang, O.; Li, H. High-resolution image inpainting using multi-scale neural patch synthesis. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 4076–4084. [CrossRef]
32. Nazeri, K.; Ng, E.; Joseph, T.; Qureshi, F.Z.; Ebrahimi, M. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv* **2019**, arXiv:1901.00212.
33. Yan, Z.; Li, X.; Li, M.; Zuo, W.; Shan, S. Shift-net: Image inpainting via deep feature rearrangement. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19. [CrossRef]
34. Liu, G.; Reda, F.A.; Shih, K.J.; Wang, T.C.; Tao, A.; Catanzaro, B. Image Inpainting for Irregular Holes Using Partial Convolutions. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 89–105. [CrossRef]
35. Antic, J. Jantic/DeOldify: A Deep Learning Based Project for Colorizing And Restoring Old Images (and Video!). 2019. Available online: https://github.com/jantic/DeOldify (accessed on 29 October 2019).
36. Mao, X.; Shen, C.; Yang, Y.B. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016; pp. 2802–2810.
37. Zhang, Y.; Tian, Y.; Kong, Y.; Zhong, B.; Fu, Y. Residual dense network for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**. [CrossRef] [PubMed]
38. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
39. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio. J. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems 27 (NIPS 2014), Montreal, QC, Canada, 8–13 December 2014; pp. 1–9. [CrossRef]
40. General Information on SOFA. 2013. Available online: https://www.sofaconventions.org/mediawiki/index.php/General_information_on_SOFA (accessed on 20 June 2019).
41. SOFA—Spatially Oriented Format for Acoustics. 2015. Available online: https://github.com/sofacoustics/API_MO (accessed on 23 June 2019).
42. Acoustics Research Institute. ARI HRTF Database. 2014 Available online: https://www.kfs.oeaw.ac.at/index.php?option=com_content&view=article&id=608&Itemid=606&lang=en#AnthropometricData (accessed on 11 July 2019).
43. Bomhardt, R.; De La, M.; Klein, F.; Fels, J. A high-resolution head-related transfer function and three-dimensional ear model database A high-resolution head-related transfer function dataset and 3D ear model database. *Proc. Meet. Acoust. J. Acoust. Soc. Am.* **2016**, *29*. [CrossRef]
44. Watanabe, K.; Iwaya, Y.; Suzuki, Y.; Takane, S.; Sato, S. Dataset of head-related transfer functions measured with a circular loudspeaker array. *Acoust. Sci. Technol.* **2014**, *35*, 159–165. [CrossRef]
45. University of York. SADIE | Spatial Audio For Domestic Interactive Entertainment. 2014. Available online: https://www.york.ac.uk/sadie-project (accessed on 11 July 2019).
46. Warusfel, O. Listen HRTF Database. 2003. Available online: http://recherche.ircam.fr/equipes/salles/listen/index.html (accessed on 11 July 2019).
47. Bernschütz, B. A Spherical Far Field HRIR/HRTF Compilation of the Neumann KU 100. In Proceedings of the AIA–DAGA 2013 Conference on Acoustics, Merano, Italy, 18–21 March 2013; pp. 592–595.
48. Sorber, L.; Barel, M.V.; Lathauwer, L.D. Unconstrained optimization of real functions in complex variables. *SIAM J. Optim.* **2012**, *22*, 879–898. [CrossRef]
49. Kim, T.; Adalı, T. Approximation by Fully Complex Multilayer Perceptrons. *Neural Comput.* **2003**, *15*, 1641–1666. [CrossRef]
50. Trabelsi, C.; Bilaniuk, O.; Zhang, Y.; Serdyuk, D.; Subramanian, S.; Santos, J.F.; Mehri, S.; Rostamzadeh, N.; Bengio, Y.; Pal, C. Deep Complex Networks. *arXiv* **2018**, arxiv:1705.09792.

51. Sarrof, A.M. Complex Neural Networks for Audio. Ph.D. Thesis, Dartmouth College, Hanover, NH, USA, 2018.
52. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Boston, MA, USA , 7–12 June 2015; pp. 1–9. [CrossRef]
53. Goodfellow, I.; Bengio, Y.; Courville, A. Deep Learning. *Nature* **2016**, *521*, 800. [CrossRef]
54. Geirhos, R.; Rubisch, P.; Michaelis, C.; Bethge, M.; Wichmann, F.A.; Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv* **2018**, arXiv:1811.12231.
55. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the NIPS 2017 Workshop Autodiff, Long Beach, CA, USA, 9 December 2017.
56. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; pp. 8024–8035.
57. Masters, D.; Luschi, C. Revisiting small batch training for deep neural networks. *arXiv* **2018**, arXiv:1804.07612.
58. Wilson, D.R.; Martinez, T.R. The general inefficiency of batch training for gradient descent learning. *Neural Netw.* **2003**, *16*, 1429–1451. [CrossRef]
59. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2015**, arXiv:1412.6980.
60. Krogh, A.; Hertz, J.A. A Simple Weight Decay Can Improve Generalization. In *Advances in Neural Information Processing Systems 4*; Moody, J.E., Hanson, S.J., Lippmann, R.P., Eds.; Morgan-Kaufmann: Burlington, MA, USA, 1992; pp. 950–957.
61. Smith, L.N. A disciplined approach to neural network hyper-parameters: Part 1—Learning rate, batch size, momentum, and weight decay. *arXiv* **2018**, arXiv:1803.09820.
62. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv* **2017**, arXiv:1711.05101.
63. Hinton, G.E.; Krizhevsky, A.; Sutskever, I. System and Method for Addressing Overfitting in a Neural Network. U.S. Patent US14/015,768, 8 February 2016.
64. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
65. Baldi, P.; Sadowski, P. Understanding dropout. *Adv. Neural Inf. Process. Syst.* **2013**, 1–9. [CrossRef]
66. Coursera. Other Regularization Methods—Practical Aspects of Deep Learning. Available online: https://www.coursera.org/lecture/deep-neural-network/other-regularization-methods-Pa53F (accessed on 1 January 2020).
67. Armstrong, C.; McKenzie, T.; Murphy, D.; Kearney, G. A perceptual spectral difference model for binaural signals. In Proceedings of the 145th Audio Engineering Society International Convention, AES 2018, New York, NY, USA, 17–19 October 2018; pp. 1–5.
68. Baumgartner, R.; Majdak, P.; Laback, B. Modeling sound-source localization in sagittal planes for human listeners. *J. Acoust. Soc. Am.* **2014**, *136*, 791–802. [CrossRef] [PubMed]
69. May, T.; Van De Par, S.; Kohlrausch, A. A probabilistic model for robust localization based on a binaural auditory front-end. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *19*, 1–13. [CrossRef]
70. Søndergaard, P.; Majdak, P. The Auditory Modeling Toolbox. In *The Technology of Binaural Listening*; Blauert, J., Ed.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 33–56.
71. Aggarwal, C.C. *Neural Networks and Deep Learning: A Textbook*; Springer International Publishing: Cham, Switzerland, 2018.
72. Zur, R.M.; Jiang, Y.; Pesce, L.L.; Drukker, K. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Med. Phys.* **2009**, *36*, 4810–4818. [CrossRef]
73. He, Z.; Rakin, A.S.; Fan, D. Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 588–597.
74. Gatys, L.; Ecker, A.; Bethge, M. A Neural Algorithm of Artistic Style. *J. Vis.* **2016**, *16*, 326. [CrossRef]
75. Zhang, M.; Zheng, Y. Hair-GANs: Recovering 3D Hair Structure from a Single Image. *arXiv* **2018**, arXiv:1811.06229.

76. Brock, A.; Donahue, J.; Simonyan, K. Large Scale GAN Training for High Fidelity Natural Image Synthesis. *Vet. Immunol. Immunopathol.* **2018**, *166*, 33–42. [CrossRef]
77. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training GANs. In Proceedings of the Neural Information Processing Systems 2016, Barcelona, Spain, 5–10 December 2016; pp. 2234–2242.