SEMI-SUPERVISED LEARNING WITH DEEP NEURAL NETWORKS FOR RELATIVE TRANSFER FUNCTION INVERSE REGRESSION

*Ziteng Wang**, *Yonghong Yan*

University of Chinese Academy of Sciences
Institute of Acoustics
Beijing, China

Emmanuel Vincent

Inria
Villers-lès-Nancy, France

ABSTRACT

The prior knowledge of relative transfer function (RTF) is preferable in many applications but remains less studied. In this paper, we propose a semi-supervised learning algorithm based on deep neural networks (DNNs) for RTF inverse regression, that is to generate the full-band RTF vector directly from the source-receiver *pose* (position and orientation). Two typical scenarios are discussed: training on labeled RTFs with or without unlabeled ones. Both setups utilize the low-dimensional manifold property of RTF in stationary environments. With this property as an additional regularization term, a smooth mapping solution with respect to the manifold is obtained. Experimental simulations show that the proposed method achieves lower mean prediction errors than the free field model with few labeled RTFs, and the unlabeled ones are essential in improving the inverse regression performance.

Index Terms— relative transfer function, semi-supervised learning, deep neural network, manifold regularization

1. INTRODUCTION

Relative transfer function (RTF) [1, 2] represents the coupling between a pair of microphones in response to the source signal. The estimation of RTF has been an essential task in many applications, such as beamforming [3], source separation [4] and source localization [5]. Generally the estimation is based merely on the microphone observations, and the prior knowledge of RTF given the source-microphone pose remains less studied. Such knowledge could bring additional performance benefits [6, 7].

The RTF relates two acoustic transfer functions (ATFs) and hence depends on the properties of the acoustic environment and on the poses of the source and microphones. Conventional room acoustic simulation methods, such as the image-source method [8], rely on explicit physical models to simulate the ATF. Recent practices perform transfer function interpolation based on models derived from the wave propagation equation [9, 10]. In [11], we proposed to predict the full-band RTF vector from a distinctive data-driven perspective, which was defined as RTF inverse regression. A deep neural network (DNN) model was trained to learn the mapping from the low-dimensional source pose to the high-dimensional RTF based on pairwise data collected in advance. It turned out that with enough dense training samples, a simple linear interpolation [12] could also achieve low mean prediction errors.

Imagine the scenario that a smart device collects data automatically in the environment, as what could happen, e.g., in conference rooms or cars. Labeling the data to obtain enough pairwise training samples can be a cumbersome task, and this motivates the semi-supervised learning setup in this paper. To train on the unlabeled RTFs, we propose an *encoder-decoder* framework and utilize the low-dimensional manifold property of RTF that was discussed recently in the source localization tasks [13, 14, 15, 16]. A smoothness constraint on the manifold is introduced to regularize the encoder network, which first gives the RTF noisy labels. The function and necessity of the unlabeled RTFs are further investigated in experimental simulations. The results show that the performance degrades with few labeled data compared to the fully labeled case, but the unlabeled samples are indispensable in the inverse regression performance.

2. DEFINITIONS

In a standard enclosure, an unknown signal S is emitted from the source and measured by a pair of microphones. For simplicity, the microphones are assumed to be fixed and the source pose is given by $\mathbf{p} = [\rho, \theta, \phi]$, the spherical coordinates. Under the narrowband approximation, the observations in the short-time Fourier transform (STFT) domain are written as

$$A_m(n,k) = H_m(\mathbf{p},k)S(n,k) + V_m(n,k), \;\; m = 1,2 \quad (1)$$

where n denotes the time index and k denotes the frequency index, H_m is the acoustic transfer function relating the source and the mth microphone and V_m is the additive noise. The RTF is defined as

$$H(\mathbf{p},k) = \frac{H_2(\mathbf{p},k)}{H_1(\mathbf{p},k)}. \quad (2)$$

The RTF is first independent of the source signal, and in stationary environments the source pose is the only factor that controls its variations. For instance, this association is given by

$$H_d(\mathbf{p},k) = \exp(j \cdot 2\pi k \frac{|r_2(\mathbf{p}) - r_1(\mathbf{p})|}{c}) \quad (3)$$

under the free field assumption, where j is the complex unit, c denotes the speed of sound and r_2, r_1 are the Euclidean distances between the source and the microphones. In reverberant conditions, the association becomes complex due to the effect of multi-path propagation. The RTF inverse regression is then to predict the full-band RTF from the source-microphone pose based on pre-collected training examples. For clearness, the RTF is referred as *sample* and the pose as *label* in the following.

*Thanks to CSC (201604910623) for funding. The work was performed while the author was at Inria.

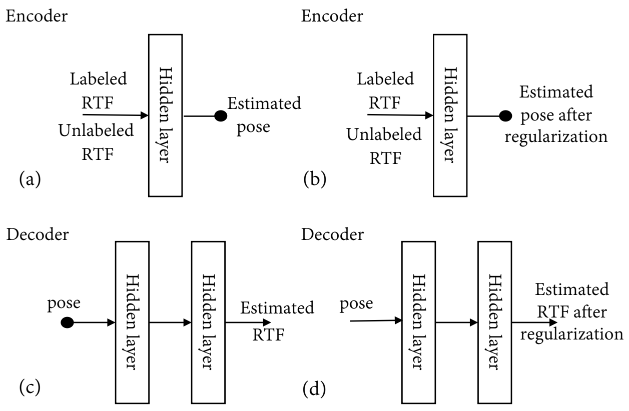

Fig. 1. The encoder and decoder networks proposed for semi-supervised RTF inverse regression.

3. RELATION TO PRIOR WORK

In [15], probabilistic piecewise affine mapping (PPAM) realized a bijective mapping between the interchannel level difference (ILD) and interchannel phase difference (IPD) vector and the source-receiver pose. ILD and IPD are defined by $\text{ILD}(\mathbf{p}, k) = 20\text{log}_{10}|H(\mathbf{p}, k)|$ and $\text{IPD}(\mathbf{p}, k) = \arg(H(\mathbf{p}, k))$, respectively, that are equivalent representations of RTF. Nevertheless, this method was only discussed for source localization. In [11], we evaluated PPAM in terms of RTF inverse regression performance together with linear interpolation and our proposed DNN model, which were all supervised setups. We briefly review this work as follows.

Given pairwise training RTFs and poses, the DNN model is optimized to minimize the mean squared prediction error:

$$\text{MSE} = \sum_i ||\mathcal{D}(\mathbf{p}_i) - \mathbf{h}_i||^2 \tag{4}$$

where $\mathcal{D}(\mathbf{p}_i)$ is the model output from input pose $\mathbf{p}_i$. Since the RTF is complex value, the target vector $\mathbf{h}_i$ is defined as a concatenation of $\text{ILD}(\mathbf{p}, k)$ and the sine and cosine of $\text{IPD}(\mathbf{p}, k)$ in all frequencies. The output nodes corresponding to the sine part $o_{k,s}$ and cosine part $o_{k,c}$ are thus normalized by

$$o_{k,s/c} = \frac{o_{k,s/c}}{\sqrt{o_{k,s}^2 + o_{k,c}^2}} \tag{5}$$

that ensures their squared sum to be one. This model is illustrated by the *decoder* network in Fig. 1(c). In the test phase, the trained decoder is used to generate RTF of a new pose $\mathbf{p}_t$ as $\widehat{\mathbf{h}_t} = \mathcal{D}(\mathbf{p}_t)$.

4. SEMI-SUPERVISED RTF INVERSE REGRESSION

The semi-supervised setup considers the case that there are only L labeled samples $\{\mathbf{h}_{1:L}, \mathbf{p}_{1:L}\}$ and the remaining samples $\{\mathbf{h}_{L+1:L+U}\}$ are unlabeled. To utilize the unlabeled samples in the training process, an intuitive encoder-decoder architecture is first discussed in Section 4.1. The idea is refined in Section 4.2 by optimizing the encoder network with manifold regularization, which turns out in latter experiments as giving the unlabeled RTFs noisy pose labels. An illustration of the involved architectures is given in Fig. 1. The network training details are discussed in Section 4.3.

4.1. Encoder-decoder

An auxiliary encoder network E is introduced to help optimize the decoder network under the new loss function:

$$\sum_{i=1}^{L} ||\mathcal{D}(\mathbf{p}_i) - \mathbf{h}_i||^2 + \alpha \sum_{j=1}^{L+U} ||\mathcal{D}(E(\mathbf{h}_j)) - \mathbf{h}_j||^2 \tag{6}$$

where α is a constant scaling factor, and the second term is the reconstruction error on the training samples. This network architecture is given in Fig. 1 by (a)+(c). The encoder can be interpreted as a localization network, that maps the RTF space to the pose space, and the encoder is expected to learn the pose labels implicitly. In the training phase, the encoder and decoder parameters are updated simultaneously.

4.2. Encoder with manifold regularization

The work of manifold regularization for localization (MRL) [16] shows that, the RTFs collected in a specific environment are confined to a low-dimensional manifold with local linearity. A simple validation of this concept is shown in Fig. 2, that plots the Euclidean distances between the RTFs with respect to the pose distances to an arbitrary chosen reference source pose. In a local range of the reference pose (about 1.6° in azimuth and elevation angles as in this case), the distance changes between the corresponding RTFs are approximately linear and uniform in different directions. In other words, a small shift in the source pose only lead to small changes in the RTF and vice versa. The detailed setup can be found in the experiment section.

Inspired by this property, the encoder is further refined from a localization perspective under the loss function:

$$\sum_{i=1}^{L} ||E(\mathbf{h}_i) - \mathbf{p}_i||^2 + \beta \sum_{i,j}^{L+U} W_{ij} ||E(\mathbf{h}_i) - E(\mathbf{h}_j)||^2 \tag{7}$$

that incurs a weighted penalty when similar inputs have different outputs. β is a scaling factor and the second term is commonly known as graph Laplacian regularization [17], that imposes a smoothness constraint on the final mapping solution. W_{ij} is the weight that reflects the adjacency between encoder inputs $\mathbf{h}_i$ and $\mathbf{h}_j$, and it is close to 0 when the samples are far away in distance. The standard Gaussian kernel function is then used for weight calculation as in [16], since it is symmetric positive semi-definite and meets the locality requirements:

$$W_{ij} = \exp(-\frac{||\mathbf{h}_i - \mathbf{h}_j||^2}{\varepsilon^2}) \tag{8}$$

where the variance ε^2 controls how fast the value decays with distance. Accordingly, the new encoder-decoder architecture is given in Fig. 1 by (b)+(c). The model is optimized under losses (6) and (7).

4.3. Network training

During training, the networks are initialized following a standard procedure, i.e., the weights are initialized with Gaussian distributed samples and the biases with zeros. ReLU is used as the activation function for all the hidden layers and layer normalization [18] is applied to regularize the parameters. The Adam method [19] is chosen to update the model using an adaptive learning rate.

One issue with the mini-batch based gradient descent is that the regularization in (7) will tend to fail after random shuffling because of the sparse affinity inside the mini-batches, especially with large

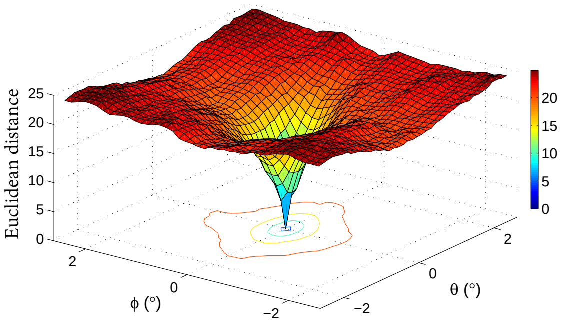

Fig. 2. The Euclidean distances between RTFs with respect to the pose distances to a reference pose.

amount of training data. In [20], one nearest-neighbor graph based solution was discussed to sample the data efficiently. We here, however, adopt a simplified technique. The training samples are first randomly shuffled. We then start from one sample, collect all its neighbors, move on to the next remaining sample and repeat until the mini-batch size is reached. The regularization is found to gradually take place against all the adjacent samples after several training epoches.

For the refined encoder, the network is first trained and then kept fixed when the decoder is optimized under loss (6), while further joint tuning doesn't bring additional performance gains in our case.

5. ALTERNATIVE DECODER WITH REGULARIZATION

Given the locality property on the RTF manifold as discussed in Section 4.2, we are also motivated to apply regularization directly to the decoder. Since the source poses come free in the environment, there would be even no need to collect the unlabeled RTFs in this case. Similar to (7), the loss function for the new decoder network can be defined by

$$\sum_{i=1}^{L} ||\mathcal{D}(\mathbf{p}_i) - \mathbf{h}_i||^2 + \gamma \sum_{i,j}^{L+U} w_{ij} ||\mathcal{D}(\mathbf{p}_i) - \mathcal{D}(\mathbf{p}_j)||^2 \qquad (9)$$

where γ is a scaling factor, and the weight is decided by the pose distances

$$w_{ij} = \exp(-\frac{||\mathbf{p}_i - \mathbf{p}_j||^2}{\epsilon^2}) \qquad (10)$$

with variance ϵ^2. This decoder is illustrated by Fig. 1(d).

6. EXPERIMENTS AND ANALYSIS

6.1. Setup

The experiments are conducted in a simulated room with dimension 6×6.2×3 m and moderate reverberation time 300 ms. Two microphones are positioned at (3,3,1) m and (3.2,3,1) m respectively. The source pose is confined to a spatial area of distance $\rho = 2$ m, azimuth angle $\theta \in [10^\circ, 60^\circ]$, and elevation angle $\phi \in [0^\circ, 30^\circ]$. Within this source space, 8 poses are sampled uniformly per degree. For each pose, a one-second white noise signal is emitted from the source and captured by the microphones. The sampling rate is 16 kHz. The corresponding acoustic impulses responses are simulated using an efficient implementation of the image-source method [21]. STFT is applied with frame length 64ms and 87.5% overlap. The target RTF is then estimated by $\widehat{H}(\mathbf{p}, k) = \frac{\sum_n A_2(n,k)A_1^*(n,k)}{\sum_n |A_1(n,k)|^2}$ with $*$ denoting complex conjugate. Note that the approach here serves as one possible way to measure the RTF in advance, and the white noise signal is special here to provide a reliable estimation in all frequencies.

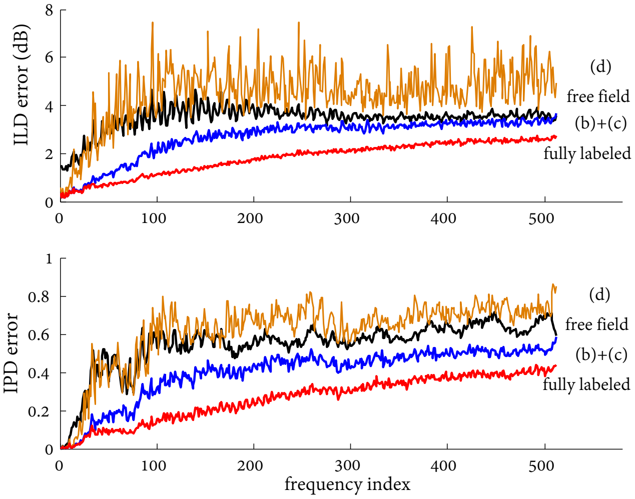

Fig. 3. ILD/IPD mean prediction errors along frequency for network (d), free field model, network (b)+(c) and network trained with fully labeled data.

In total, 400×240 training samples are genearated, of which only 24 RTFs are labeled with their true source poses, creating a grid of 10° in angle distance. The input to the decoder network is the source pose $\mathbf{p}$, and the training target vector is the concatenation of ILD and IPD as discussed in Section 3. We use two hidden layers each with 1024 nodes for the decoder and one hidden layer with 1024 nodes for the auxiliary encoder. The scaling factors α, β, γ are heuristically set to be 0.01. The variances ε^2, ϵ^2 are decided such that the weights of samples farther than 2° away are close to 0.

For evaluation, 1000 extra poses are picked randomly in the specified source area. As far as we know, there is no established methods to measure closeness of two high-dimensional vectors, and a mean absolute error metric is chosen here to measure the performance, which shows the prediction errors in each frequency:

$$\mu_k = \frac{1}{T} \sum_{t=1}^{T} |\widehat{\mathbf{h}}_t(k) - \mathbf{h}_t(k)| \qquad (11)$$

Still the metric makes some sense as shown in latter experiments when applying the generated RTF to specific applications. The 95% confidence interval (CI) of μ_k is also calculated to give an idea of the error distribution. The ILD and IPD prediction errors are treated separately.

6.2. RTF Inverse regression performance

The prediction performance for each frequency bin is shown in Fig. 3 and the frequency averaged results are summarized in Table 1. For comparison, the free field model (3) and a decoder trained with all the data labeled, are included as baselines.

The prediction errors clearly increase with frequency. The network trained with fully labeled data achieves the lowest mean prediction errors as expected, with 1.80±0.09 dB in ILD and 0.26±0.02

Table 1. Summarization of frequency averaged ILD/IPD prediction errors (mean±CI bound) and SBF (dB).

	ILD error	IPD error	SBF
(a)+(c)	4.05±0.23	0.64±0.03	-0.60
(d)	3.94±0.20	0.61±0.03	-0.35
free field model	3.47±0.16	0.55±0.03	0.69
(b)+(c)	2.67±0.13	0.39±0.02	2.04
fully labeled	1.80±0.09	0.26±0.02	3.19

in IPD. It is significantly better than other setups where only few training data are labeled. The unlabeled RTFs function differently in different architectures. The (b)+(c) network performs much better than the free field model, while the intuitive encoder-decoder setup (a)+(c), that relies on the same training data, fails to give good predictions. To see how the unlabeled samples help, we investigate the encoder networks, that indeed realize pose localization. The root mean squared error (RMSE) between the encoder outputs and the true source poses are computed. For encoder (a), the localization RMSE is 13.08° in azimuth and 8.74° in elevation. For the encoder (b) with manifold regularization, the results are 2.94° and 2.08°, respectively. It achieves quite accurate pose predictions considering that the labeled samples are 10° away in distance. The encoders provide the unlabeled RTFs with noisy pose labels. Additional experiments show that the manifold regularized encoder even slightly outperforms MRL in the same localization task.

The decoder network (d) with regularization fails to compete with the free field model. One reason could be due to the l_2 norm used in (9) to measure the affinity between RTFs, that ignores the angle of the high-dimensional vector. Clearly, the l_2 norm makes more sense for measuring the poses.

6.3. Further analysis

It is commonly acknowledged that learning based methods would perform better with more labeled training data and suffer performance loss in mismatched test conditions. So these aspects are not further investigated here. Meanwhile, we evaluate the generated RTFs in a specific application, the generalized sidelobe canceller (GSC) [1], that requires RTF in the implementation of a blocking matrix to provide the reference noise signal. A frequency domain signal blocking factor (SBF) is first defined as:

$$\text{SBF} = 10\log_{10} \frac{\sum_{n,k} ||A_1(n,k)||^2}{\sum_{n,k} ||A_2(n,k) - H(\mathbf{p},k)A_1(n,k)||^2} \qquad (12)$$

where the denominator denotes the energy of the leakage signal. SBF indicates the ability to block the first-channel source image in the second microphone and thus correlates with signal distortion.

The SBF scores of different methods are given in Table 1. The results are consistent with the mean prediction errors, with the fully labeled setup scoring the best (3.19 dB), the (b)+(c) network outperforming the free field model and the simple (a)+(c) the scoring the worst (-0.60 dB). A negative score means that the generated RTF is not helpful at all.

Note that the generated RTF is prior information obtained from the source pose only and can be incorporated with the observations to achieve maximum a posterior RTF estimation.

7. CONCLUSIONS

We considered the RTF inverse regression task from a practical perspective and introduced the semi-supervised learning method that requires few pairwise training data. Several architectures were discussed. The proposed encoder with manifold regularization and decoder architecture gave low prediction errors and outperformed the free field model. Regularization on the high dimensional RTF vector didn't work well, which indicated that the unlabeled RTFs were necessary in the final performance. Incorporating the generated RTFs in GSC and other applications is some work worth further investigation.

8. REFERENCES

[1] Sharon Gannot, David Burshtein, and Ehud Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Transactions on Signal Processing*, vol. 49, no. 8, pp. 1614–1626, 2001.

[2] Israel Cohen, "Relative transfer function identification using speech signals," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 451–459, 2004.

[3] Ronen Talmon, Israel Cohen, and Sharon Gannot, "Relative transfer function identification using convolutive transfer function approximation," *IEEE Transactions on audio, speech, and language processing*, vol. 17, no. 4, pp. 546–555, 2009.

[4] Maja Taseska and Emanuel AP Habets, "Relative transfer function estimation exploiting instantaneous signals and the signal subspace," in *23rd European Signal Processing Conference (EUSIPCO)*. IEEE, 2015, pp. 404–408.

[5] Xiaofei Li, Laurent Girin, Radu Horaud, and Sharon Gannot, "Estimation of relative transfer function in the presence of stationary noise based on segmental power spectral density matrix subtraction," in *IEEE International Conference onAcoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 320–324.

[6] Ronen Talmon and Sharon Gannot, "Relative transfer function identification on manifolds for supervised GSC beamformers," in *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*. IEEE, 2013, pp. 1–5.

[7] Zbyněk Koldovskỳ, Petr Tichavskỳ, Francesco Nesta, et al., "Semi-blind noise extraction using partially known position of the target source," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2029–2041, 2013.

[8] Jont B Allen and David A Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[9] Prasanga Samarasinghe, Thushara Abhayapala, Mark Poletti, and Terence Betlehem, "An efficient parameterization of the room transfer function," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 12, pp. 2217–2227, 2015.

[10] Remi Mignot, Gilles Chardon, and Laurent Daudet, "Low frequency interpolation of room impulse responses using compressed sensing," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 1, pp. 205–216, 2014.

[11] Ziteng Wang, Emmanuel Vincent, and Yonghong Yan, “Relative transfer function inverse regression from low dimensional manifold,” *arXiv preprint*, 2017.

[12] Emmanuel Vincent, Jon Barker, Shinji Watanabe, Jonathan Le Roux, Francesco Nesta, and Marco Matassoni, “The second CHiME speech separation and recognition challenge: Datasets, tasks and baselines,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 126–130.

[13] Antoine Deleforge and Radu Horaud, “2D sound-source localization on the binaural manifold,” in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2012, pp. 1–6.

[14] Bracha Laufer, Ronen Talmon, and Sharon Gannot, “Relative transfer function modeling for supervised source localization,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2013, pp. 1–4.

[15] Antoine Deleforge, Florence Forbes, and Radu Horaud, “Acoustic space learning for sound-source separation and localization on binaural manifolds,” *International Journal of Neural Systems*, vol. 25, no. 01, 2015.

[16] Bracha Laufer, Ronen Talmon, and Sharon Gannot, “Semi-supervised sound source localization based on manifold regularization,” *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 8, pp. 1393–1407, 2016.

[17] Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov, “Revisiting semi-supervised learning with graph embeddings,” in *International Conference on Machine Learning*, 2016, pp. 40–48.

[18] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.

[19] Diederik Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[20] Sunil Thulasidasan and Jeffrey Bilmes, “Acoustic classification using semi-supervised deep neural networks and stochastic entropy-regularization over nearest-neighbor graphs,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2731–2735.

[21] Emanuel AP Habets, “Room impulse response (rir) generator,” *Available online: https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator*, 2014.