Robust offline trained neural network for TDOA based sound source localization

Srikanth Raj Chetupalli
Dept. of ECE
Indian Institute of Science
Bangalore, India
sraj@iisc.ac.in

Ashwin Ram
Dept. of ECE
National Institute of Technology, Thiruchirapalli
Tamilnadu, India
ashwinram10@gmail.com

Sreenivas Thippur V.
Dept. of ECE
Indian Institute of Science
Bangalore, India
tvsree@iisc.ac.in

***Abstract*—Passive sound source localization (SSL) using time-difference-of-arrival (TDOA) measurements is a non-linear inversion problem. In this paper, a data-driven approach to SSL using TDOA measurements is considered. A neural network (NN) is viewed as an architecture constrained non-linear function, with its parameters learnt from the training data. We consider a three layer neural network with TDOA measurements between pairs of microphones as input features and source location in the Cartesian coordinate system as output. Experimentally, we show that, NN trained even on noise-less TDOA measurements can achieve good performance for noisy TDOA inputs also. These performances are better than the traditional spherical interpolation (SI) method. We show that the NN trained offline using simulated TDOA measurements, performs better than the SI method, on real-life speech signals in a simulated enclosure.**

***Index Terms*—Sound source localization, Time difference of arrival, Neural network regression, Spherical interpolation.**

I. Introduction

Passive sound source localization (SSL) refers to the problem of identifying the location of a sound source given the signals received at one or more microphones. The problem has applications in several areas such as signal enhancement, automatic speech recognition, and acoustic scene analysis [1]–[3]. Time-difference-of-arrival (TDOA) is a non-intrusive measurement that can be obtained from pairs of microphone signals. SSL problem using TDOA measurements involves (i) accurate estimation of TDOA, and (ii) estimation of source location using the TDOA measurements. Generalized cross correlation with phase transform (GCC-PHAT) [4], [5] method is widely used for TDOA estimation from the received signals, because of its robustness to different source signals and room reverberation. However, given TDOA measurements, solution to SSL using least squares error criterion, is a non-linear least squares minimization problem. Spherical intersection [6], and spherical interpolation [7] are two example methods that can solve the problem by linearization. Correct source position can be estimated using the above methods for noise-free TDOA measurements. However, for noisy measurements, source position estimates become erroneous because of the non-linearity of the problem.

Neural networks (NN) have been used for source localization tasks in narrow-band antenna array scenario [8] for estimating the source direction of arrival (DoA). In speech processing, SSL using NN for a humanoid robot with two microphones has been considered in [9]. Here, a short segment of difference signal computed from the two microphone signals is used as input feature with the DoA as the output. In [10], an approach based on cross power spectrum between adjacent microphone signals in a linear microphone array is considered. Normalized cross power spectrum, which carries the phase information, is extracted from pairs of adjacent microphones in the array, and its real and imaginary parts are input to a 3 layer NN with DoA of the source as the target output. A similar approach is considered in [11] for a two microphone array, however with complex-valued input features instead of the real and imaginary parts, and the complex phase corresponding to the desired DoA as the target output. Similar efforts have been made in binaural localization using ILD, IPD and ITD cues [12], [13]. Recently, a discriminative training approach using deep neural networks (DNN) with short-time spectral features is proposed for single source [14], and multi-source scenarios [15]. For each frequency bin, a DNN is learnt as a directional activator, which is similar to a collection of steering vectors. The outputs of bin-wise activators are integrated to give posterior probability over position labels (discretized DoA in the azimuth plane). In all these approaches, SSL is confined to the estimation of DoA using features which capture the time and intensity difference cues between pairs of microphones.

In this paper, we consider SSL using only TDOA features without the need of any specific array geometry. Since TDOA estimation is well established, we can exploit the TDOA between pairs of mics, even arbitrarily placed, for location estimation and not just DOA. Here, a NN is viewed as an architecture constrained non-linear function mapping scheme and SSL is posed as a function approximation problem using NNs. We explore a 3 layer (single hidden layer) NN with TDOA measurements between pairs of microphones as input features and the location of the source in Cartesian coordinates as output. This results in a compact NN, since the number of unique TDOA measurements is one less than the number of mics in the array; compared to a short-time spectrum based system, where number of inputs is a function of the number of mics and the size of fft. We found that, responses of learnt neurons in the hidden layer show spatial selectivity with directional and near-field/far-field distinctive properties.

978-1-5386-1224-8/18/$31.00 ©2018 IEEE

In terms of localization performance, we show that, NN approach performs better than the conventional approaches even for noisy TDOA measurements. The errors in TDOA can arise due to, (i) signal sampling frequency and interpolation, (ii) reverberant convolutive distortion or additive noise at the microphones, (iii) broadband and time-varying nature of speech, and (iv) difficulty of specific source position itself. Interestingly, we found that NN approach can provide effective solution even when trained on simulated TDOA data and tested on real-life noisy TDOA obtained under reverberant condition of real speech signals.

II. TDOA Source localization

Let $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_M\}$ be the set of microphone position vectors and let $\mathbf{s}$ be the position vector of the source. The Time-Difference-Of-Arrival (TDOA) of the signal received at the positions of two microphones $\mathbf{m}_i, \mathbf{m}_j$, due to source at $\mathbf{s}$ is,

$$t_{ij} = \frac{1}{c}(\|\mathbf{m}_i - \mathbf{s}\|_2 - \|\mathbf{m}_j - \mathbf{s}\|_2), \tag{1}$$

where c is the speed of sound propagation. For source localization, the goal is to estimate the position vector $\mathbf{s}$ of the source, given the position vectors $\{\mathbf{m}_i\}_{i=1}^{M}$ of the microphones and a set of TDOA measurements between all/subset of the pairs of microphones. Thus, the inverse problem of determining $\mathbf{s}$ becomes non-linear with respect to $\{t_{ij}\}$, as intersection of hyperboloids.

A. NN based approach

The localization problem can be thought of as a non-linear function (f),

$$\hat{\mathbf{s}} = f(\mathcal{M}, \mathcal{T}), \tag{2}$$

with the microphone positions ($\mathcal{M}$) and the set of TDOA values ($\mathcal{T}$) as inputs and the source position as output; i.e., we can pose it as a non-linear "regression problem" with the goal of estimating the function f with measurement set $\{\mathcal{M}, \mathcal{T}\}$ as input and source position $\mathbf{s}$ as output, such that the Euclidean distance between the estimated and the known source position is minimized. Given a set of N training examples, $\{\mathcal{M}, \mathcal{T}_n, \mathbf{s}_n\}_{n=1}^{N}$, the regression problem is posed as,

$$\min_{f} \sum_{n=1}^{N} \|\mathbf{s}_n - f(\mathcal{M}, \mathcal{T}_n)\|_2^2 \triangleq J(f). \tag{3}$$

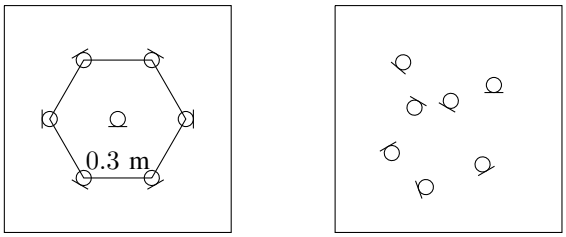
We know that a neural network can learn a non-linear mapping function between the input and output variables; the learning can be interpreted as a network constrained non-linear function optimization. For a chosen network architecture, we determine the network parameters such that the objective function $J(f)$ is minimum. It is also known that a neural network with one hidden layer can approximate any arbitrary mapping function between input and output, depending on the number of units in the hidden layer [16], [17]. Hence, we explore a 3 layer (one hidden layer) neural network to approximate the function f. Of course the non-linear mapping is with respect to fixed, but arbitrary mic positions and source position anywhere within the enclosure (the network can be retrained for a different mic configuration). Interestingly, we found that the hidden neurons (nodes) learn to segregate the enclosure in a structured manner so that the collective output of all the hidden neurons, leads to a correct output estimation. Since the microphone positions are fixed (and arbitrary), only TDOA values are considered as input features to the neural network. For a fixed mic configuration $\mathcal{M}$ and choosing one mic as the reference (say $\mathbf{m}_1$), source position $\mathbf{s}$ is defined w.r.t. $\mathbf{m}_1$. Let $\mathbf{x}_n = \mathcal{T}_n$ denote the input variable, source position $\hat{\mathbf{s}}_n$ be the output variable and $\mathbf{W_1}, \mathbf{W_2}$ and $\mathbf{b_1}, \mathbf{b_2}$ be the weight and bias variables respectively, of the two layers. We choose a linear activation function for the output layer, and let $\sigma()$ denote the activation function for the hidden layer. Thus, the output of the neural network is given by,

$$\hat{\mathbf{s}}_n = f(\mathbf{x}_n) = \mathbf{W}_2\left[\sigma(\mathbf{W}_1\mathbf{x}_n + \mathbf{b}_1)\right] + \mathbf{b}_2. \tag{4}$$

III. Experiments

We first explore the neural network model for localization in 2D space. Consider a microphone array with six microphones in a hexagonal configuration with a side dimension of $0.3\ m$, and one microphone at the center as origin (Fig. 1(a)). The array is assumed to be placed at the center of a planar enclosure of size $3\ m \times 3\ m$. TDOA between the microphone at the origin and the six microphones on the vertices of the hexagon are chosen as inputs; the number of outputs is two for the case of 2D localization. Simulated data is used for training the NN. $10,000$ Source positions, chosen randomly from a uniform distribution over the range of $[-1.5\ m, 1.5\ m]$ along each dimension are used for training the NN. For each source position, exact TDOA values for the six microphone pairs are computed using the forward relation of eqn. (1). TDOA inputs are normalized using the maximum possible TDOA (corresponds to the maximum distance between microphone pairs). A separate test set with 1000 source positions (different from the training source positions) is generated similarly for the evaluation. In all the experiments, speed of sound is assumed to be $340\ m/sec$ and a sampling rate of $F_s = 48$ KHz. We used TensorFlow [18] neural network library to conduct the NN experiments. As indicated in eqn. (3), MSE cost function is used for NN training. Rectified linear unit (ReLU) is used as the activation function (unless otherwise stated). Momentum optimizer with a momentum value of 0.8 and initial learning rate of 0.15, and a mini batch size of 64 is used for training. Training is performed for 100 epochs.

Fig. 2 shows the results of one experiment of $2D$ localization showing the activation value at the output of 48 neurons in the hidden layer. Each image depicts the source locations sampled from a $3\ m \times 3\ m$ area and the image pixel values correspond to the output of neurons showing its activation sensitivity. Microphones are located at the center of the sampling plane. We observe that a subset of the neurons learn to respond to different directions, often with less activation for locations near the center of the array (locations in the far-field



(a) (b)

Fig. 1. (a) Hexagonal, and (b) Random microphone configurations.

of the array). Another subset of the neurons learn to respond to the locations near the array center (comprising mostly of the near-field locations), and a few other neurons respond to locations close to boundary of the sampled surface. This may be because, the TDOA measurements are sensitive to the actual source location in the vicinity of the array, and become direction sensitive as the distance from array increases.

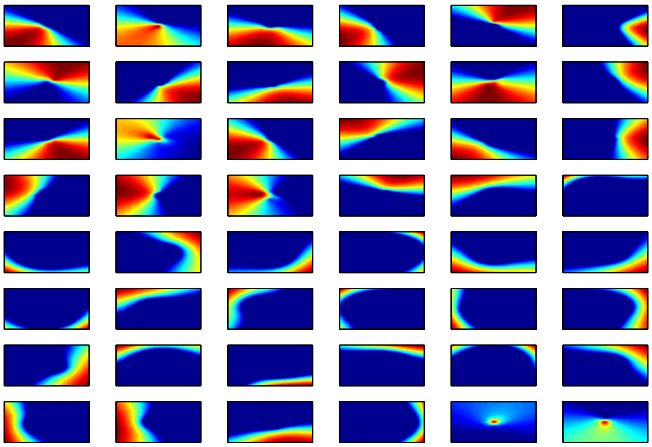
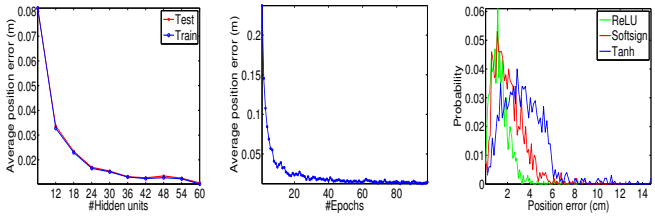


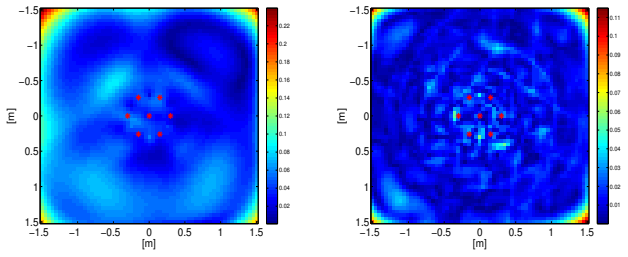
Fig. 2. 2D spatial selection response of each of the 48 neurons in the hidden layer to source locations in 2D. Each neuron specializes to a 2D region. Color gradient from blue to red shows increasing activation value.



(a) (b) (c)

Fig. 3. Average position error using ReLU activation, as a function of (a) the number of hidden units and (b) the number of epochs for 48 hidden units. (c) Histogram of the position error for softsign, ReLU and tanh activation functions.

Fig. 3(a) shows the average of the position error for training and test data sets, separately for a varying number of hidden units (thus, we explored a minimal compact NN for the localization problem). We see that average position error decreases with increasing number of hidden units. Fig. 3(b) shows the average position error as a function of the number of training epochs, with hidden units fixed at 48; this also shows a monotonic trend, and saturation beyond ≈ 40 epochs. We also explored alternate activation functions for improved NN performance. The error with ReLU activation function is found to be smaller compared to the softsign and tanh activation functions. This can be seen in the histogram of the position estimation error shown in Fig. 3(c). ReLU activation function performs better than the remaining two with lesser mean error and also smaller variance. Location estimation error of the NN



(a) (b)

Fig. 4. Spatial distribution of error for (a) softsign and (b) ReLU activation functions. The microphone array is shown as red dots.

on training data is found to be nearly uniform at all spatial locations; however, this may not be guaranteed for the test set. Fig. 4 shows the test position estimation error for source locations spread over a $3\ m \times 3\ m$ area. We can see that the error is uniformly small across most sampled locations, except near boundary corners. From Figs. 3,4, we conclude that, a compact neural network with 48 hidden units using ReLU activation functions, can estimate source locations with $\approx 1\ cm$ accuracy uniformly across the target range of source locations.

A. 3D Localization and Random Mics

In this experiment, we consider source localization in $3D$ space using an array of randomly placed microphones. The microphone positions are chosen within a cube of side length $0.6\ m$, and fixed for the experiment. The source positions are then sampled uniformly inside a cube of side $3\ m$. This is done so that mics are closer to the center, although random. The number of source locations used for training is 10000. Clean TDOA measurements are obtained using the forward computation of eqn. (1). Noisy TDOA measurements are obtained using U samples of error are generated by adding uniform random noise distributed $[-U\ \ U]$ to the clean TDOA measurements and rounding to nearest integer sample (corresponding to $F_s = 48$ KHz). We evaluate the NN performance on noisy TDOA measurements, under clean and noisy training conditions of the network. We compare this performance with the spherical interpolation (SI) method of [7]. For noisy training, we consider TDOA measurements with $U = 10$ samples of error.

Fig. 5(a) shows the average position error as a function of the number of microphones, using a NN with ReLU activation functions and 64 hidden units. Noise-less TDOA measurements are used for training and testing the NN. We see that, position error decreases with increase in the number

of microphones, Fig. 5(b,c,d) show the performance of NN for noisy TDOA inputs. Examples of zero error in the figure correspond to clean TDOA measurements, rounded to nearest integer sample TDOA. Fig. 5(b,c) show probability distribution of the position error for $U = 1$ and $U = 5$ distribution of error in TDOA measurements. We can see that, for $U = 1$, SI method performs better with more probability mass near zero error compared to the NN approach; also NN trained with clean data performs better than noisy data training for the small error distribution case. However, when the error in TDOA measurements is increased up to $U = 5$ samples, NN method is found to out perform the SI method. Further, NN trained with noisy TDOA measurements performs better than that trained with clean TDOA data as expected. Fig. 5(d) shows the probability that the position error is $\leq 0.5\ m$ w.r.t. TDOA error distribution. The NN trained with noisy data shows uniformly good performance compared to the NN trained with clean TDOA, also, with respect to the SI method. However, for small TDOA errors, clean data trained NN does show superior performance. It may be noted that SI method does guarantee exact location estimation for clean TDOA measurements, which is not guaranteed using the NN approach.

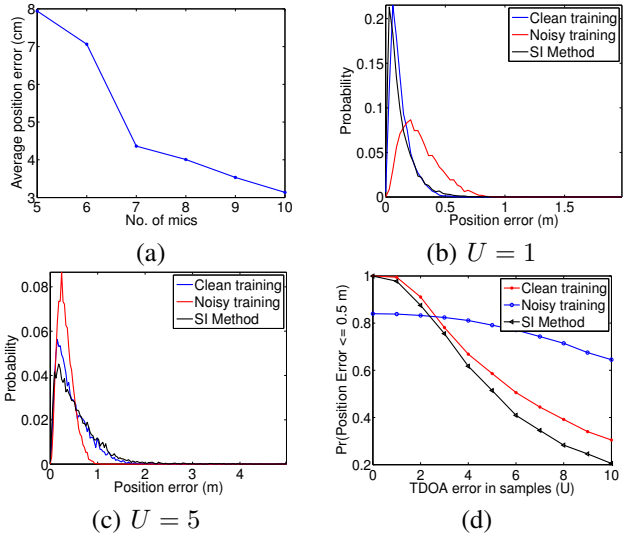


Fig. 5. (a) SSL error as a function of the number of microphones. (b,c) Histogram of SSL error for random errors in TDOA measurements. (d) Probability of position error $\leq 0.5\ m$ as a function of TDOA error indicating uniformly robust performance of NN trained using noisy TDOA.

B. Reverb speech TDOA errors

To examine the NN performance for TDOA errors caused by real speech signals and room reverberation, we consider two microphone configurations, (i) hexagonal array configuration of Fig. 1(a), and (ii) array with random microphone placement (example illustration shown in Fig. 1(b)). In each case, the array is placed at the center of a simulated room of size $3\ m \times 3\ m \times 3\ m$, and the neural network (48 hidden units with ReLU activation functions) is trained offline similar to the experiments in previous section, on simulated training data.

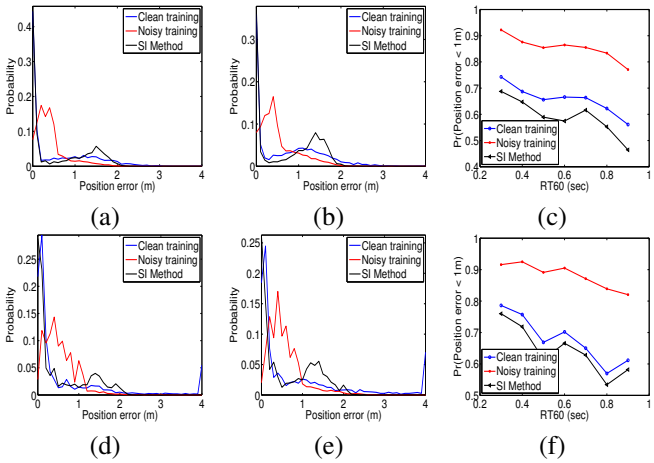


Fig. 6. Performance on TDOA of simulated microphone signals. Histogram of the position error for, (a,d) $RT60 = 0.3$s and (b,e) $RT60 = 0.6$s. (c,f) Probability of position error $\leq 1\ m$ vs. $RT60$. The two rows correspond to hexagonal and random microphone configurations in Fig. 1.

For testing the NN, source positions are chosen on the same plane as the $2D$ microphone array, but distributed randomly on the place. For each source position, a speech signal with sampling rate $F_s = 48$ KHz is used to simulate the microphone signals by convolving it with the room impulse response generated using the image-method [19], [20]. The well established GCC-PHAT method is used to estimate TDOA values from the microphone signals using a signal window size of 4096 samples ($85\ ms$). 100 random source positions are considered, which resulted in $\approx$ 5000 frame test instances of speech. TDOA estimates are input to the neural network trained offline earlier to predict the speech source position.

Fig. 6(a,d) show histograms of the position error for $RT60 = 0.3s$, and (b,e) for $RT60 = 0.6s$, using the SI method, along with the two NN train conditions. The histograms show performance similar to Fig. 5; we see that NN training with noisy TDOA does not result in small errors for clean data, but it performs better on the noisy data. Fig. 6(c,f) show the fraction of source location estimates with a position error $\leq 1\ m$. NN trained with simulated TDOA gives performance significantly better than the clean trained NN and also the SI method. The superior performance is consistently better for RT60 ranging upto 1 sec. We note that, here NN has been trained offline, and it has not seen the real TDOA errors due to reverb speech; the SSL during test requires only evaluation through the trained network. The evaluation involves only two matrix multiplications and application of non-linear function, which makes the NN approach very fast and suitable for real-time applications, unlike other localization methods.

IV. Conclusions

We develop a neural network based approach to acoustic source localization using TDOA measurements. We show that the network can be trained offline using simulated source positions and their corresponding TDOA values. We find that a single hidden layer neural network with $40 - 60$ units can

achieve good performance better than analytical solutions in the literature, even when the network is trained using only noise free TDOAs. Further, training the network using simulated noisy TDOAs improves performance robustness. Results for real-life speech signals in a simulated enclosure also show improved performance compared to traditional methods.

REFERENCES

[1] J. Benesty, M. M. Sondhi, and Y. Huang, Springer handbook of speech processing. Springer Science & Business Media, 2007.

[2] P. A. Naylor and N. D. Gaubitch, Speech dereverberation. Springer Science & Business Media, 2010.

[3] M. Brandstein and D. Ward, Microphone arrays: signal processing techniques and applications. Springer Science & Business Media, 2013.

[4] C. Knapp and G. Carter, The generalized correlation method for estimation of time delay, IEEE Trans. Acoust., Speech, Signal Process., vol. 24, no. 4, pp. 320327, Aug 1976.

[5] M. S. Brandstein and H. F. Silverman, A robust method for speech signal time-delay estimation in reverberant rooms, in Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP), vol. 1, Apr 1997, pp. 375378.

[6] H. Schau and A. Robinson, Passive source localization employing intersecting spherical surfaces from time-of-arrival differences, IEEE Trans. Acoust., Speech, Signal Process., vol. 35, no. 8, pp. 12231225, Aug 1987.

[7] J. Abel and J. Smith, The spherical interpolation method for closed-form passive source localization using range difference measurements, in Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP), vol. 12, Apr 1987, pp. 471474.

[8] K.-L. Du, A. Lai, K. Cheng, and M. Swamy, Neural methods for antenna array signal processing: a review, Signal Processing, vol. 82, no. 4, pp. 547561, 2002.

[9] Y. Geng, J. Jung, and D. Seol, Sound-source localization system based on neural network for mobile robots, in 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), June 2008, pp. 31263130.

[10] G. Arslan and F. A. Sakarya, A unified neural-network-based speaker localization technique, IEEE Transactions on Neural Networks, vol. 11, no. 4, pp. 9971002, Jul 2000.

[11] H. Tsuzuki, M. Kugler, S. Kuroyanagi, and A. Iwata, An approach for sound source localization by complex-valued neural network, IEICE Transactions on Information and Systems, vol. 96, no. 10, pp. 22572265, 2013.

[12] K. Youssef, S. Argentieri, and J.-L. Zarader, A binaural sound source localization method using auditive cues and vision, in Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP). IEEE, 2012, pp. 217220.

[13] N. Ma, G. Brown, and T. May, Exploiting deep neural networks and head movements for binaural localisation of multiple speakers in reverberant conditions, in Interspeech, vol. 2015. International Speech Communication Association, 2015, pp. 160164.

[14] R. Takeda and K. Komatani, Sound source localization based on deep neural networks with directional activate function exploiting phase information, March 2016, pp. 405409.

[15] R. Takeda and K. Komatani, Discriminative multiple sound source localization based on deep neural networks using independent location model, in 2016 IEEE Spoken Language Technology Workshop (SLT), Dec 2016, pp. 603609.

[16] G. Cybenko, Approximation by superpositions of a sigmoidal function, Mathematics of control, signals and systems, vol. 2, no. 4, pp. 303314, 1989.

[17] K. Hornik, Approximation capabilities of multilayer feedforward networks, Neural networks, vol. 4, no. 2, pp. 251257, 1991.

[18] TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, [Online] Software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[19] J. B. Allen and D. A. Berkley, Image method for efficiently simulating small-room acoustics, The Journal of the Acoustical Society of America, vol. 65, no. 4, pp. 943950, 1979.

[20] E. A. P. Habets, Room impulse response (rir) generator, 2014, [Online] software available from https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator.