HyperSound: Generating Implicit Neural Representations of Audio Signals with Hypernetworks

Filip Szatkowski^{1,2*} **Karol J. Piczak**^{4*} **Przemysław Spurek**^{4}
Jacek Tabor^{4} **Tomasz Trzciński**^{1,2,3,4}

^{1}Warsaw University of Technology ^{2}IDEAS NCBR ^{3}Tooploox
^{4}Faculty of Mathematics and Computer Science, Jagiellonian University

{filip.szatkowski.dokt,tomasz.trzcinski}@pw.edu.pl
{karol.piczak,przemyslaw.spurek,jacek.tabor}@uj.edu.pl

Abstract

Implicit neural representations (INRs) are a rapidly growing research field, which provides alternative ways to represent multimedia signals. Recent applications of INRs include image super-resolution, compression of high-dimensional signals, or 3D rendering. However, these solutions usually focus on visual data, and adapting them to the audio domain is not trivial. Moreover, it requires a separately trained model for every data sample. To address this limitation, we propose HyperSound, a meta-learning method leveraging hypernetworks to produce INRs for audio signals unseen at training time. We show that our approach can reconstruct sound waves with quality comparable to other state-of-the-art models.

1 Introduction

Implicit neural representations (INRs) are coordinate-based representations of multimedia signals, where the signal is modeled with a neural network. Such representations are decoupled from the spatial resolution, so the signal can be resampled at any arbitrary frequency. At the same time, the memory requirements for its storage remain constant. The field of INRs is rapidly growing, and their applications include super-resolution [1, 2], compression [1, 2] or 3D rendering [3]. However, in the audio domain, the evaluation of these approaches was so far limited to learning individual INRs of particular input recordings [1], which is highly ineffective.

Meta-learning methods, such as hypernetworks [4], can generate INRs for any arbitrary signal with a single model. Hypernetworks learn to generate weights for smaller target networks, which can serve as INRs. Hypernetworks were successfully applied in obtaining INRs for images [2, 5], point clouds [2, 6] and videos [2]. However, creating INRs for high-dimensional and high-variance data such as audio is difficult, and training hypernetworks for this task can be unstable.

In this work, we propose a meta-learning approach based on hypernetworks, where we learn a general recipe for creating INRs of arbitrary audio samples from outside of the training dataset. To our knowledge, our model is the first application of hypernetwork-based INRs to the audio domain.

2 Related Works

Implicit neural representations (INRs) are applied across many multimedia domains. **NeRF** [3]-based INRs are currently state-of-the-art for image-based rendering and view synthesis. **SIREN** [1] shows how to produce good quality INRs for various signals, such as image, video, and sound signals, using neural networks with periodic activation functions and dedicated weight initialization

6th Workshop on Meta-Learning at NeurIPS 2022, New Orleans.

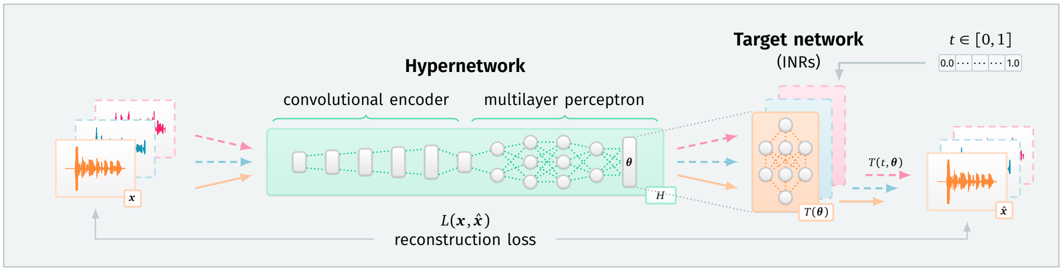

Figure 1: Overview of the HyperSound framework. We use a single hypernetwork model to produce distinct INRs based on arbitrary audio signals provided as input.

scheme. **Modulated Periodic Activations** [2] further improve the quality of SIREN-based INRs for high-resolution signals by introducing modulation and synthesis sub-networks. However, in the audio domain, the evaluation of these approaches was so far limited to learning individual INRs of particular input recordings.

Hypernetworks [4] are a meta-learning framework, where one network (hypernetwork) generates the weights for another network (target network). Hypernetworks can be used for a variety of tasks such as model compression [7], continual learning [8] or generating **INRs**. In particular, hypernetworks were used to generate INRs for images [5], shapes [6] and videos [2]. To our knowledge, our work is the first application of hypernetworks for audio INRs generation.

First successful attempts at **raw waveform processing with deep neural networks** were models such as **WaveNet** [9] and **SampleRNN** [10], but their autoregressive nature makes them slow and prone to accumulation of errors. Later architectures such as **ParallelWaveNet** [11], **NSynth** [12], **MelGAN** [13] or **SING** [14] proposed non-autoregressive architectures for audio generation. Recent autoencoder-based models such as **RAVE** [15] or **SoundStream** [16] are able to process high-resolution signals in an end-to-end fashion, producing audio of very good perceptual quality.

3 Model overview

Sound waves are traditionally represented digitally as a collection of amplitude values sampled at regular intervals, which approximates a continuous real function $x(t)$. Our goal is to obtain a meta-recipe for generating audio INRs replicating such functions. While creating INRs for particular audio samples can be quite easily done with gradient descent, finding a general solution is much harder due to the inherent complexity of audio time series. Therefore, similar to [5], we model such functions with neural networks T (target networks), parameterized by weights generated by another neural network H (hypernetwork). Our framework, shown in Fig. 1, can be described as

$$\boldsymbol{\theta}_{\boldsymbol{x}} = H(\boldsymbol{x}), \tag{1}$$

$$\hat{x}(t) = T(t, \boldsymbol{\theta}_{\boldsymbol{x}}). \tag{2}$$

3.1 Hypernetwork architecture

Typical audio recordings contain several thousands of samples, so the hypernetwork is composed of a convolutional encoder that produces a latent representation of a lower dimensionality, and fully connected layers that transform this representation to weights $\boldsymbol{\theta}$ of the target network.

We use an encoder based on SoundStream [16], and the fully connected part of the hypernetwork is composed of six fully-connected layers with biases and ELU [17] activation, where the last layer produces the flattened weights of the target network.

3.2 Approximating sound waves with neural networks

A target network should be as small as possible to avoid overfitting and explosion of weights in the hypernetwork, but must be expressive enough to represent a wide variety of audio recordings. Our target network has one input and one output, and consists of a positional embedding layer followed by four fully-connected layers of 256 neurons with biases and ReLU activation. Inspired by NeRF [3],

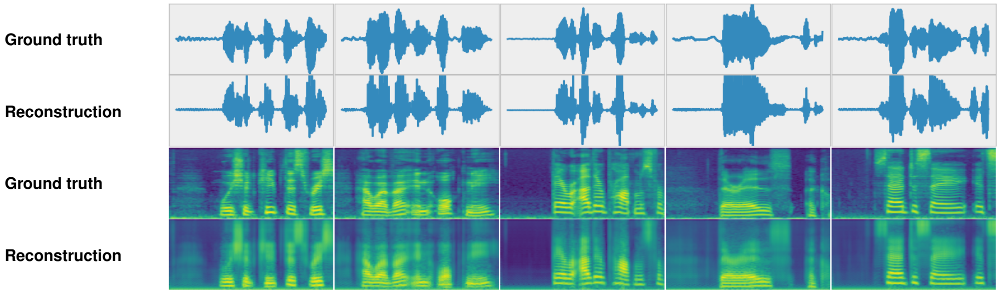

Figure 2: Examples of VCTK validation samples reconstructed with HyperSound.

we define embedding vectors $\boldsymbol{\gamma}$ as

$$\boldsymbol{\gamma}(t) = \left[\sin(2^0\pi t),\, \cos(2^0\pi t),\, \sin(2^1\pi t),\, \cos(2^1\pi t),\, \ldots,\, \sin(2^{L-1}\pi t),\, \cos(2^{L-1}\pi t)\right], \quad (3)$$

where t is the time coordinate and L denotes the embedding size. We rescale input coordinates to a range of $[0, 1]$ and use $L = 16$.

3.3 Optimization

We train the hypernetwork in a supervised fashion using backpropagation. To obtain audio results that are more perceptually pleasant, we use a loss function that penalizes the reconstruction error both in the time and frequency domains. Given an original recording $\boldsymbol{x}$ and its reconstruction $\hat{\boldsymbol{x}}$ generated with a target network, we compute the loss function as

$$L(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \lambda_{SL1} * L_{SL1}(\boldsymbol{x}, \hat{\boldsymbol{x}}) + \lambda_{STFT} * L_{STFT}(\boldsymbol{x}, \hat{\boldsymbol{x}}), \quad (4)$$

where L_{SL1} is a smooth L1 loss [18] with $\beta = 0.1$, L_{STFT} is a multi-resolution mel-scale STFT loss introduced in ParallelWaveGAN [19] and λ_{SL1}, λ_{STFT} are the weights of these two losses. We use $\lambda_{SL1} = 1$ and $\lambda_{STFT} = 1$. For STFT loss, we use 128 mel bins and FFT sizes of $[512, 1024, 2048]$ with matching window sizes and an overlap of 87.5%.

4 Experiments

We test the reconstruction quality of our model on the VCTK dataset downsampled to $f = 22\,050$ Hz, with the recordings of the last 10 speakers retained as a validation set. We set the recording length to 32 768 samples, and use data augmentations such as random crop, phase mangle, or dequantization proposed in RAVE [15]. We train the models for 1.25M steps using the AdamW optimizer [20] with a learning rate of $5\mathrm{e}{-5}$ and a batch size of 16. In Fig. 2, we show waveforms and spectrograms of the reconstructions obtained with our model on samples selected from the validation set.

Since there is no consensus on a single approach for quantitative evaluation of audio quality, we assess the reconstruction results with multiple metrics such as MSE, Log-Spectral Distance (LSD) [21], SI-SNR [22], PESQ [23], STOI [24] and CDPAM [25]. We also compare the reconstruction quality of our model with the RAVE baseline. Moreover, we test the quality of resampling performed by our model. Finally, we investigate the impact of the target network size and the employed loss function on the reconstruction quality. The results of our experiments are shown in Tab. 1. Unless explicitly mentioned, model hyperparameters in experiments are as described in Sec. 3.

Comparision with RAVE. We compare the reconstruction error of HyperSound with RAVE [15] trained for 3M steps on the same dataset, with an identical sampling rate of 22 050 Hz. We find that our model produces reconstructions closer to the originals in the spectral domain and obtains better perceptual scores of PESQ and STOI. However, RAVE reconstructions achieve better MSE, SI-SNR, and perceptual CDPAM scores, with slightly less perceptible noise and robotic artifacts.

Reconstruction quality while resampling. We also test the reconstruction error of the model trained with a sampling rate of 22 050 Hz when used for downsampling and upsampling 22 050 Hz recordings to other commonly used frequencies: 8000 Hz, 16 000 Hz and 44 100 Hz. We obtain the ground truth by downsampling the original VCTK 48 kHz recordings to the desired sampling frequency using *soxr* high-quality setting. As shown by the LSD levels, the hypernetwork approach works best when

Table 1: HyperSound evaluation based on reconstructions of the VCTK validation set

		MSE	**LSD**	**SI-SNR**	**PESQ**	**STOI**	**CDPAM**
	Ideal metric behavior	→ 0	→ 0	→ 100	→ 4.5	→ 1	→ 0
Comparison with RAVE	**Model**						
	*HyperSound**	0.049	**0.99**	-23.50	**1.75**	**0.87**	0.35
	RAVE	**0.040**	1.19	**-22.42**	1.22	0.74	**0.19**
Reconstruction quality while resampling	**Target SR [Hz]**						
	8000	0.048	1.24	-23.76	1.67	0.83	0.37
	16 000	**0.047**	1.19	-23.80	1.66	0.86	**0.35**
	*22 050**	0.049	**0.99**	**-23.50**	**1.75**	**0.87**	**0.35**
	44 100	**0.047**	1.38	-24.16	1.72	**0.87**	0.38
Impact of target network size	**Network size**						
	Small [4 x 64]	0.048	1.07	-24.56	1.56	0.84	**0.35**
	*Base [4 x 256]**	0.049	0.99	**-23.50**	1.75	0.87	**0.35**
	Large [6 x 384]	**0.047**	**0.96**	-23.70	**1.86**	**0.89**	0.36
Impact of the loss function	**Loss function**						
	L1 + STFT	0.052	**0.94**	-25.05	1.35	0.81	0.25
	Only STFT	0.055	**0.94**	-25.63	1.34	0.81	**0.23**
	*L1 + MelSTFT**	**0.049**	0.99	-23.50	**1.75**	**0.87**	0.35
	Only MelSTFT	0.050	1.02	**-23.31**	1.70	**0.87**	0.29

**default HyperSound hyperparameters*

reconstructing signals close to the sampling rate used for training. Nevertheless, it does not collapse when queried with time coordinates from outside the training domain. However, we hypothesize that for proper super-resolution capabilities, our learning regime would require an introduction of an additional loss term directed specifically at the quality of representation in higher frequency bands.

Impact of target network architecture. We compare our model which uses the baseline target network (4 layers of 256 neurons each, 206K parameters) with target networks that use 4 layers of 64 neurons (14K parameters) and 6 layers of 384 neurons (752K parameters). We find that the base version presents an optimal trade-off between the reconstruction quality and computational requirements, as the size of the last hypernetwork layer scales linearly with the number of parameters in the target network. The smaller variant of the target network achieves comparable results to the base using fewer parameters than the number of samples in the original signal.

Impact of the loss function. We compare HyperSound trained with the loss function as described in Sec. 3.3 with models trained using the STFT loss where we do not apply the mel-scale, but use a wider array of FFT sizes of $[128, 256, 512, 1024, 2048]$. We also try training without Smooth L1 loss ($\lambda_{SL1} = 0, \lambda_{STFT} = 1$). Moreover, we find that STFT is vital for stable training and obtaining perceptually plausible reconstructions, as our training runs with $\lambda_{SL1} = 1.$ and $\lambda_{STFT} = [0.1, 0.01]$ collapsed. As evidenced by the results, the L1 part of the loss function leads to slightly better results and faster training, as it enables the model to correctly learn the DC offset.

5 Conclusion

We demonstrate the possibility of applying hypernetworks to the generation of implicit neural representations for audio signals. Reconstructions generated with our model are quantitatively comparable to the state-of-the-art model RAVE. However, we find that the perceptual quality of our reconstructions is still slightly lacking. We hope that our work can be further improved by optimizing the hypernetwork architecture and designing target networks better suited to the audio domain. Initial results for signal compression are also promising but require further investigation.

Acknowledgments and Disclosure of Funding

This work was supported by Foundation for Polish Science (grant no POIR.04.04.00-00-14DE/18-00) carried out within the Team-Net program co-financed by the European Union under the European Regional Development Fund, as well as the National Centre of Science (Poland) Grant No. 2020/39/B/ST6/01511. Przemysław Spurek is supported by the National Centre of Science (Poland) Grant No. 2021/43/B/ST6/01456.

References

[1] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit Neural Representations with Periodic Activation Functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020. 1

[2] Ishit Mehta, Michaël Gharbi, Connelly Barnes, Eli Shechtman, Ravi Ramamoorthi, and Manmohan Chandraker. Modulated Periodic Activations for Generalizable Local Functional Representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14214–14223, 2021. 1, 2

[3] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[4] David Ha, Andrew Dai, and Quoc V Le. HyperNetworks. *arXiv preprint arXiv:1609.09106*, 2016. 1, 2

[5] Sylwester Klocek, Łukasz Maziarka, Maciej Wołczyk, Jacek Tabor, Jakub Nowak, and Marek Śmieja. Hypernetwork functional image representation. In *International Conference on Artificial Neural Networks*, pages 496–510. Springer, 2019. 1, 2

[6] Przemysław Spurek, Sebastian Winczowski, Jacek Tabor, Maciej Zamorski, Maciej Zięba, and Tomasz Trzciński. Hypernetwork approach to generating point clouds. *arXiv preprint arXiv:2003.00802*, 2020. 1, 2

[7] Dominic Zhao, Johannes von Oswald, Seijin Kobayashi, João Sacramento, and Benjamin F Grewe. Meta-Learning via Hypernetworks. 2020. 2

[8] Johannes Von Oswald, Christian Henning, João Sacramento, and Benjamin F Grewe. Continual learning with hypernetworks. *arXiv preprint arXiv:1906.00695*, 2019. 2

[9] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. *arXiv preprint arXiv:1609.03499*, 2016. 2

[10] Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. SampleRNN: An Unconditional End-to-End Neural Audio Generation Model. *arXiv preprint arXiv:1612.07837*, 2016. 2

[11] Aaron Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George Driessche, Edward Lockhart, Luis Cobo, Florian Stimberg, et al. Parallel WaveNet: Fast High-Fidelity Speech Synthesis. In *International conference on machine learning*, pages 3918–3926. PMLR, 2018. 2

[12] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders. In *International Conference on Machine Learning*, pages 1068–1077. PMLR, 2017. 2

[13] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis. *Advances in neural information processing systems*, 32, 2019. 2

[14] Alexandre Défossez, Neil Zeghidour, Nicolas Usunier, Léon Bottou, and Francis Bach. SING: Symbol-to-Instrument Neural Generator. *Advances in neural information processing systems*, 31, 2018. 2

[15] Antoine Caillon and Philippe Esling. RAVE: A variational autoencoder for fast and high-quality neural audio synthesis. *arXiv preprint arXiv:2111.05011*, 2021. 2, 3

[16] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. SoundStream: An End-to-End Neural Audio Codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021. 2

[17] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv preprint arXiv:1511.07289*, 2015. 2

[18] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 3

[19] Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6199–6203. IEEE, 2020. 3

[20] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*, 2017. 3

[21] Haohe Liu, Woosung Choi, Xubo Liu, Qiuqiang Kong, Qiao Tian, and DeLiang Wang. Neural Vocoder is All You Need for Speech Super-resolution. *arXiv preprint arXiv:2203.14941*, 2022. 3

[22] Yi Luo and Nima Mesgarani. TasNet: time-domain audio separation network for real-time, single-channel speech separation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 696–700. IEEE, 2018. 3

[23] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 01CH37221)*, volume 2, pages 749–752. IEEE, 2001. 3

[24] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In *2010 IEEE international conference on acoustics, speech and signal processing*, pages 4214–4217. IEEE, 2010. 3

[25] Pranay Manocha, Zeyu Jin, Richard Zhang, and Adam Finkelstein. CDPAM: Contrastive learning for perceptual audio similarity. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 196–200. IEEE, 2021. 3