Few-Shot Audio-Visual Learning of Environment Acoustics Supplementary Material

Anonymous Author(s)
Affiliation
Address
email

In this supplementary material we provide additional details about:

- Video (with audio) for qualitative illustration of our task and qualitative evaluation of our model predictions (Sec. 1).
- Potential societal impact of our work (Sec. 2), as noted in L557 of the main paper.
- Evaluation of the impact of the query source location on our model's prediction quality for a fixed receiver (Sec. 3).
- Audio dataset details (Sec. 4), as mentioned in L245 and 571-2 of the main paper.
- Model architectures details for RIR prediction (Sec. 5.1) and downstream tasks (Sec. 5.2), as noted in 245 of the main paper.
- Training hyperparameters (Sec. 5.3), as referenced in L245 and 571-2 of the main paper.

1 Supplementary Video

The supplementary video shows the perceptually realistic SoundSpaces [2] audio simulation platform that we use for our experiments, and provides a qualitative illustration of our task, Few-Shot Audio-Visual Learning of Environment Acoustics. Moreover, we qualitatively demonstrate our model's prediction quality by comparing the predictions with the ground truths, both at the RIR level and in terms of perceptual similarity when the RIRs are convolved with real-world monaural sounds, like speech and music. We also analyze common failure cases for our model (L290 in main) and qualitatively show how our model predictions can be used to successfully localize an audio source in a 3D environment. Please use headphones to hear the spatial audio correctly.

2 Potential Societal Impact

Our model enables modeling the acoustics in a 3D scene using only few observations. This has multiple applications with positive impact. For example, accurate modeling of the scene acoustics enables a robot to locate a sounding object more efficiently (like finding a crying baby, or locating a broken vase). Additionally, this allows for a truly immersive experience for the user in augmented and virtual reality applications. However, RIR generative models allow the user to match the acoustic reverberation in their speech to an arbitrary scene type, hence disguise their true location type from the receiver which may have both positive and negative implications. Finally, our model uses visual samples from the environment for more accurate modeling of the acoustic properties of the scene. However, the dataset used in our experiments contains mainly indoor spaces that are of western designs, and with certain object distribution that is common to such spaces. This may bias models trained on such data toward similar types of scenes and reduce generalization to scenes from other cultures. More innovations in the model design to handle strong shifts in scene layout and object

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

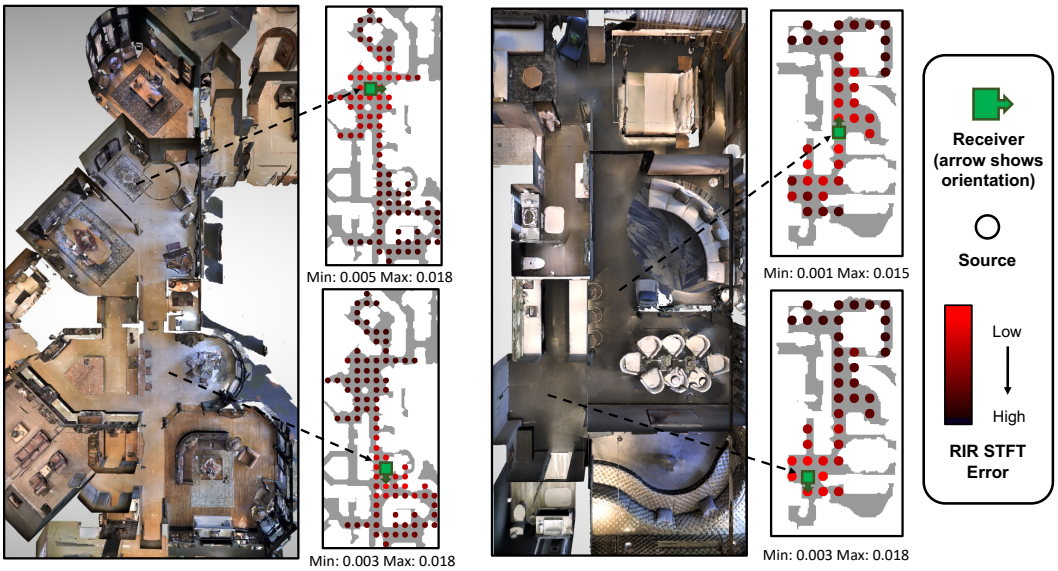

Figure 1: RIR prediction STFT error as a function of varying source locations (filled circles) for a given receiver (a green square with an arrow). We show two scenes and two examples per scene. The color of the circle at the source location indicates the STFT error in the RIR prediction associated with that source and receiver pair. The error in each example is normalized between the min and max values shown underneath the map.

distribtutions, as well as more diverse datasets are needed to mitigate the impact of such possible biases.

3 Impact of the Source Location on the Prediction Error

In Fig. 1, we show the RIR prediction error as a function of different source locations for a fixed receiver location. As we can see, predictions error tend to be small when the source is relatively close to the receiver, or there are no major obstacles along the path connecting them. This indicates that the model leverages the local geometry of the scene and the acoustic information captured from echoes for better predictions. However, the error increases when there are large distances between the source and receiver (L290 in main), and especially when there are major obstacles for audio propagation in between (e.g., walls, narrow corridors). Modeling how audio gets transformed on such a long path becomes very challenging due to the limited observations available to the model and the larger scene area that contributes to transforming the audio.

4 Audio Dataset

For computing the mean opinion score error (MOSE) [1] (L269-71 in main), we sample 5 second long speech clips from the LibriSpeech [9] dataset, which comprise both male and female speakers. For every test query, we randomly choose one of the sampled clips and convolve with the true RIR or a model's prediction for that query to estimate the corresponding mean opinion score (MOS) [7] and, subsequently, the error in MOS for a model's prediction relative to the true RIR. We use a 5-second-long temporal window for all model predictions and true RIRs when estimating their MOS.

For our experiment with ambient environment sounds (L331-5 in main), we use ambient sounds from the ESC-50 [10] dataset (e.g., dog barking, running water). For every test query, we randomly sample a location in the 3D scene for an ambient sound and play a randomly chosen 1 second long clip from the ESC-50 dataset at that location. To retrieve the observed binaural echo response A_i (L111-4 and 150-3 in main) in this setting, first we convolve the clean echo RIR for each observation O_i with the sinusoidal sweep sound, then mix it with the binaural the ambient sound for its pose P_i, and finally deconvolve using the inverse sweep (L150-3 in main).

We will release our datasets upon acceptance.

5 Architecture and Training

Here, we provide our architecture and additional training details for reproducibility. We will release our code upon acceptance.

5.1 Model Architectures for RIR Prediction

Visual Encoder. Our visual encoder f^V is a ResNet-18 [4] model (L147-9 in main) that takes egocentric RGB and depth images from the observation set, which are concatenated channel-wise, as input and produces a 512-dimensional feature.

Acoustic Encoder. Our acoustic encoder f^A is another ResNet-18 [4] (L156-7 in main) that separately encodes the binaural log magnitude spectrogram for an echo RIR A_i into a 512-dimensional feature.

Pose Encoder. To embed an observation pose P_i or a query source-receiver pair Q (L114-7 in main), we use sinusoidal positional encodings [14] (L158-60 and 179-81 in main) with 8 frequencies, which generate a 16-dimensional feature vector (the positional encodings comprise both sine and cosine components with 8 features per component) for every attribute of an observation pose or a query (i.e., x, y, and θ).

Modality Encoder. For our modality embedding m (L161-8 in main), we maintain a sparse lookup table of 8-dimensional learnable embeddings, which we index with 0 to retrieve the visual modality embedding (m_V) and 1 to retrieve the acoustic modality embedding (m_A).

Fusion Layer. To generate the multimodal memory S (L169-73 in main) for our context encoder (L169-77 in main), we separately concatenate the modality features (produced by f^V for vision and f^A for echo responses) for an observation, the corresponding sinusoidal pose embedding, and the modality embedding (m_V for visual features and m_A for acoustic features), and project using a single linear layer to 1024-dimensional embedding space. Similarly, to generate the query encoding q for our conditional RIR predictor (L181-2 in main), we use another linear layer to project the query's sinusoidal positional encodings to a 1024-dimensional feature vector. Furthermore, we don't use bias in any fusion layer.

Context Encoder. Our context encoder (L179-88) is a transformer encoder [14] with 6 layers, 8 attention heads, a hidden size of 2048 and ReLU [13, 8] activations. Additionally, we use a dropout [12] of 0.1 in our context encoder.

Conditional RIR Predictor. Our conditional RIR predictor (L179-88 in main) has 2 components: 1) a transformer decoder [14] to perform cross-attention on the implicit representation C (L173-7 in main), which is produced by the previously described context encoder, using the query encoding q (L182-4 in main), and 2) a multi-layer transpose convolution network U (L186-7 in main) to upsample the decoder output d^Q and predict the magnitude spectrogram for the query Q in log space.

The transformer decoder [14] has the same architecture as our context encoder.

The transpose convolution network U comprises 7 layers in total. The first 6 layers are transpose convolutions with a kernel size of 4, stride of 2, input padding of 1, ReLU [13, 8] activations and BatchNorm [5]. The number of input channels for the transpose convolutions are 128, 512, 256, 128, 64 and 32, respectively. The last layer of U is a convolution layer with a kernel size of 3, stride of 1, padding of 1 along the height dimension and 2 along the width dimension, and 16 input channels. Finally, we switch off bias in all layers of U.

5.2 Model Architectures for Downstream Tasks

Sound Source Localization. We use a ResNet-18 [4] feature encoder that takes the log magnitude spectrogram of an RIR (predicted or ground truth as input). We take the encoded features and feed them to a single linear layer that predicts the location coordinates of a query's source relative to the query's receiver pose.

Depth Estimation. Following VisualEchoes [3], we use a U-net [11] that takes the log magnitude spectrogram of an echo as input and predicts the depth map (L347-58) as seen from the echo's pose. The encoder of our U-net has 6 layers. The first layer is a convolution with a kernel size of 3, stride of 2, padding of 1 along the height dimension, and 2 input channels. The remaining 5 layers are convolutions with a kernel size of 4, padding of 1 and stride of 2. These 5 layers have 64, 64, 128, 256 and 512 input channels, respectively. Each convolution is followed by a ReLU activation [13, 8] and a BatchNorm [5].

The decoder of the U-net has 5 transpose convolution layers. Each transpose convolution has a kernel size of 4, stride of 2 and input padding of 1. Except for the last layer that uses a sigmoid activation function to generate depth maps, which are normalized such that all pixels are in the range of $[0, 1]$, each transpose convolution has a ReLU activation [13, 8] and a BatchNorm [5]. The decoder layers have 512, 1024, 512, 256 and 128 channels, respectively. We use skip connections between the encoder and the decoder starting with their second layer.

5.3 Training Hyperparameters

In addition to the training details specified in main (L215-6), we use a batch size of 24 during training. Furthermore, for every entry of the batch, we query our model with 60 arbitrary source-receiver pairs for the same observation set, which effectively increases the batch size further and improves training speed. Other training hyperparameters specific to our Adam [6] optimizer include $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-5}$.

References

[1] Changan Chen, Ruohan Gao, Paul Calamia, and Kristen Grauman. Visual acoustic matching. *arXiv preprint arXiv:2202.06875*, 2022.

[2] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *European Conference on Computer Vision*, pages 17–36. Springer, 2020.

[3] Ruohan Gao, Changan Chen, Ziad Al-Halab, Carl Schissler, and Kristen Grauman. Visualechoes: Spatial image representation learning through echolocation. In *ECCV*, 2020.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456, Lille, France, 07–09 Jul 2015. PMLR.

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

[7] Chen-Chou Lo, Szu-Wei Fu, Wen-Chin Huang, Xin Wang, Junichi Yamagishi, Yu Tsao, and Hsin-Min Wang. Mosnet: Deep learning based objective assessment for voice conversion. In *Proc. Interspeech 2019*, 2019.

[8] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010.

[9] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

[10] Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia*, pages 1015–1018. ACM Press.

[11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[12] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[13] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2892–2900, 2015.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.