SoundSpaces 2.0: A Simulation Platform for Visual-Acoustic Learning Supplemental Material

Changan Chen[1,4*] **Carl Schissler**[2*] **Sanchit Garg**[2*] **Philip Kobernik**[2] **Alexander Clegg**[4]
Paul Calamia[2] **Dhruv Batra**[3,4] **Philip Robinson**[2] **Kristen Grauman**[1,4]
[1]UT Austin [2] Reality Labs at Meta [3]Georgia Tech [4]Meta AI

8 Supplementary

In this supplementary material, we provide additional details about:

1. Supplementary video for qualitative examples (referenced in Sec. 1 of the main paper).
2. Details of RLR-Audio-Propagation (referenced in Sec. 3.1).
3. Details of APIs and interface (referenced in Sec. 3.3).
4. Details of material configuration (referenced in Sec. 3.3).
5. Statistics of the PanoIR dataset (referenced in Sec. 4).
6. RT60 comparison with real measurements (referenced in Sec. 5.2).
7. Training and implementation details of the navigation benchmark.
8. Training and implementation details of the ASR benchmark.
9. Discussion on societal impacts of this work.

8.1 Supplementary video

In this video, we provide several demos of the simulation, comparison with real measurements, impact of acoustic continuity, examples of the PanoIR dataset, navigation videos of the trained policy and far-field ASR example. Wear your headphone for spatial effects.

8.2 Details of RLR-Audio-Propagation

SoundSpaces 2.0 models indirect sound propagation using an energy-based bidirectional path tracing algorithm. The simulation begins by emitting rays from each sound source in the scene, where each ray carries a spectrum of energy in log-spaced frequency bands. These rays are then propagated through the scene via reflection, diffraction, and transmission, until they reach a maximum number of bounces. Reflections use a Phong BRDF [8], where the Phong exponent is determined from the material scattering coefficient. Diffraction of rays occurs probabilistically when they hit specially-constructed edge diffraction geometry [14]. Transmission of rays occurs with probability proportional to the material transmission coefficients. The vertices along each source ray path are retained for the listener ray tracing step.

After tracing source rays, rays are then emitted from the listener and propagated through the scene in a similar way. At each listener path vertex, a connection is attempted to a random point on each sound source, as well as to a randomly selected vertex on that source's ray paths. When a complete path

[*]Equal contribution

Submitted to the 36th Conference on Neural Information Processing Systems (NeurIPS 2022) Track on Datasets and Benchmarks. Do not distribute.

from source to listener is formed, the energy for that path is calculated using multiple importance sampling [5]. The path energy for each frequency band is then added to a histogram of energy with respect to propagation delay time. Spherical harmonic coefficients representing the sound directivity at each histogram bin are constructed as an energy-weighted sum of ray directions arriving in that bin. Early reflection (ER) paths (≤ 2 bounces) and direct sound are handled differently. Individual ER paths are clustered together into discrete reflections based on the plane equations of the reflecting surfaces.

The resulting energy-based impulse response representation is then converted to a pressure impulse response so that it can be convolved with source audio. This is done by converting the energy histogram to a pressure envelope by taking the square root, and then synthesizing random phase by multiplying each frequency band's envelope with pre-filtered white noise and summing frequency bands [7]. Early reflections and direct sound are added to the pressure IR as positive impulses with frequency-dependent amplitude. Spatialization is achieved by multiplying the omnidirectional pressure impulse response with the spherical harmonic coefficients at each IR time sample. The result is an ambisonic pressure IR which can also be converted to a binaural IR by convolving with an ambisonic representation of the head-related transfer function (HRTF) [20]. The spatial fidelity of the direct sound is preserved by barycentric interpolation of the original HRTF data, rather than using the ambisonic HRTF.

8.3 Details of APIs and Interface

See `https://github.com/facebookresearch/habitat-sim/blob/main/docs/AUDIO.md` for the detailed API and interface of RLR-Audio-Propagation. Below we briefly summarize the complete list of exposed parameters.

Simulation parameters. The complete list of the exposed simulation parameters includes sampling rate, number of frequency bands, the spherical harmonic order used for calculating direct sound spatialization, the spherical harmonic order used for calculating indirect sound spatialization, number of CPU threads, simulation time step, maximum IR length, unit scale for the scene, initial pressure value, number of direct rays, number and maximum depth of source indirect rays, number and maximum depth of listener indirect rays, whether direct/indirect/diffraction/transmission is enabled, whether mesh is simplified for faster computation, whether temporal coherence is enabled for faster computation, and whether custom material properties are used.

Microphone configurations. We provide seven built-in microphone types, including mono, stereo, binaural, quad, surround_5_1, surround_7_1 and ambisonics. The channel layout API takes the channel type and number of channels as input. Users can also configure their own microphone array by provide an array of mono microphones.

8.4 Details of Material Configuration

The material configuration consist of two parts. First, we need to define an acoustic material database, which has different materials and their corresponding acoustic parameters. Secondly, we define a mapping function that maps objects to their corresponding materials.

In the material folder, we provide four configuration files: default_material_coefficients.py, default_material_to_category.py, category_to_material_mapping.py and mp3d_material_config.json. default_material_coefficients.py defines material database and the coefficients for each acoustic material. default_material_to_category.py defines the default mapping between material and categories. category_to_material_mapping.py defines the one-to-many mapping for the acoustic randomization strategy, where each object category has several plausible acoustic materials. mp3d_material_config.json is the combined configuration file that the simulation takes as input.

8.5 Details of PanoIR Dataset

In this dataset, we provide rendered panoramic images (RGB/Depth) and IRs for Matterport3D [1], Gibson [19] and HM3D [12] datasets. Each panoramic image is stitched from 18 images with 20 field

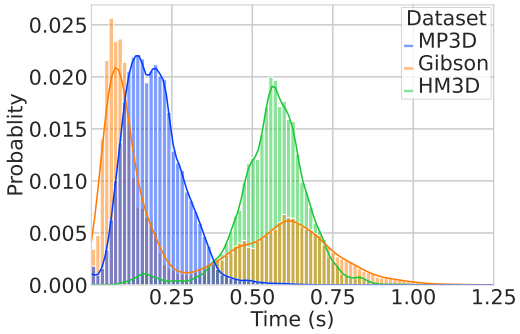

Figure 4: RT60 distribution in the PanoIR dataset for different scene datasets.

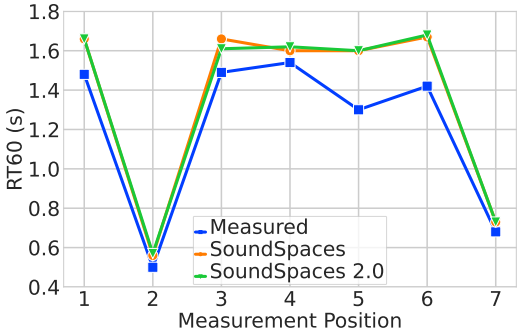

Figure 5: RT60 comparison with real measurements.

of view (FoV) each, resulting in a size of 1152×384. Both the source and receiver are at height of 1.5m from the ground. The coordinates of the source is provided as polar coordinates in the format of (θ, d), where θ defines the rotation from the center of the panorama image anti-clockwise and d is the distance between the receiver and the source. Both the receiver and listener are randomly sampled from mesh environments and the distance is limited to be within 5m such that the local geometry is captured in the panorama. We use high-quality mode to render IRs for this dataset. The PanoIR dataset is available at `https://github.com/facebookresearch/sound-spaces/blob/main/PanoIR/README.md`.

Fig. 4 shows the distribution of RT60s in the PanoIR dataset for different scene datasets. As we can see, the RT60 distribution varies from dataset to dataset. This difference comes from several factors: quality of the mesh, distribution of scene types and material properties. The main reason for the Matterport3D's RT60 distribution skewing towards left is because there are lots of broken meshes in that dataset, which results in ray leaking from holes and smaller reverberation in general. On the contrary, Gibson and HM3D have higher quality mesh and have larger RT60s on average.

8.6 RT60 Comparison with Real IRs

Fig. 5 shows the RT60 comparison between real measurements and simulations in the Replica apartment [16] for 7 measurement positions and the 250Hz to 4000Hz frequency band. Both the original SoundSpaces and SoundSpaces 2.0 have an average relative RT60 error of 12.4%. Altogether with the DRR comparison in Fig.3(b), we show that SoundSpaces 2.0 has higher acoustic realism than the original SoundSpaces.

These real measurements are included in the supplementary file as well, and are available at `http://dl.fbaipublicfiles.com/SoundSpaces/real_measurements.zip`. Open sourcing these will allow researchers to test their own models against this real data.

8.7 Training and Implementation Details of the Navigation Benchmark

In this continuous audio-visual navigation benchmark, we use the AV-Nav agent [3] with the decentralized distributed proximal policy optimization (DD-PPO) [18]. We train the navigation policy for 80 million steps on the same AudioGoal navigation dataset [3] except that the movement and audio are continuous. For continuous navigation, we define success as the agent issuing a stop action within 1m of the goal location. We train the policy on 32 GPUs for 46 hours to converge.

8.8 Training and Implementation Details of the ASR Benchmark

For finetuning the pretrained ASR model on speech augmented with IRs, we first generate the same amount of IRs for the train-clean-100 split in LibriSpeech [10] (28539 IRs) with randomly sampled configurations (source, receiver and environment). For training, we convolve a given speech clip with a random IR in the training set. The batch size for finetuning is 24 and we finetune the ASR model on

8 GPUs for 60 epochs. The number of warmup steps for the transformer is set to 1000. After training, we test the finetuned ASR model on the test-clean split of LibriSpeech [10] convolved with real IRs from BUT ReverbDB [17] for sim2real evaluation. For fintuning on Pyroomacoustics [13], we also generate 28539 IRs by varying the room dimension configurations.

8.9 Discussion on Societal Impacts

We are not aware of any negative societal impact. We do believe this work will open up many possibilities for visual-acoustic learning research [2, 4, 9, 6, 11, 15]., which has many applications in robotics and AR/VR.

References

[1] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *3DV*, 2017. MatterPort3D dataset license available at: `http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf`.

[2] Changan Chen, Ziad Al-Halah, and Kristen Grauman. Semantic audio-visual navigation. In *CVPR*, 2021.

[3] 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 *ECCV*, 2020.

[4] Victoria Dean, Shubham Tulsiani, and Abhinav Gupta. See, hear, explore: Curiosity via audio-visual association. In *NeurIPS*, 2020.

[5] Iliyan Georgiev. Implementing vertex connection and merging. *Technical Report. Saarland University*, 2012.

[6] Teerapat Jenrungrot, Vivek Jayaram, Steve Seitz, and Ira Kemelmacher-Shlizerman. The cone of silence: Speech separation by localization. In *Advances in Neural Information Processing Systems*, 2020.

[7] K Heinrich Kuttruff. Auralization of impulse responses modeled on the basis of ray-tracing results. *Journal of the Audio Engineering Society*, 41(11):876–880, 1993.

[8] Eric P Lafortune and Yves D Willems. Using the modified phong reflectance model for physically based rendering. 1994.

[9] Andrew Luo, Yilun Du, Michael J Tarr, Joshua B Tenenbaum, Antonio Torralba, and Chuang Gan. Learning neural acoustic fields. *arXiv preprint arXiv:2204.00628*, 2022.

[10] 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. IEEE, 2015.

[11] Senthil Purushwalkam, Sebastian Vicenc Amengual Gari, Vamsi Krishna Ithapu, Carl Schissler, Philip Robinson, Abhinav Gupta, and Kristen Grauman. Audio-visual floorplan reconstruction. In *ICCV*, 2021.

[12] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[13] Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. Pyroomacoustics: A python package for audio room simulations and array processing algorithms. *arXiv*, 2017.

[14] Carl Schissler, Gregor Mückl, and Paul Calamia. Fast diffraction pathfinding for dynamic sound propagation. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021.

[15] Nikhil Singh, Jeff Mentch, Jerry Ng, Matthew Beveridge, and Iddo Drori. Image2reverb: Cross-modal reverb impulse response synthesis. In *ICCV*, 2021.

[16] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.

[17] MIgor Szoke, Miroslav Skacel, Ladislav Mosner, Jakub Paliesek, and Jan "Honza" Cernocky. Building and evaluation of a real room impulse response dataset. *arXiv*, 2018.

[18] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. DD-PPO: Learning near-perfect PointGoal navigators from 2.5 billion frames. In *ICLR*, 2020.

[19] Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: real-world perception for embodied agents. In *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*. IEEE, 2018.

[20] Markus Zaunschirm, Christian Schörkhuber, and Robert Höldrich. Binaural rendering of ambisonic signals by head-related impulse response time alignment and a diffuseness constraint. *The Journal of the Acoustical Society of America*, 143(6):3616–3627, 2018.