SoundSpaces: Audio-Visual Navigation in 3D Environments Supplementary Materials

Changan Chen*[1,4], Unnat Jain*†[2,4], Carl Schissler[3], Sebastia Vicenc Amengual Gari[3], Ziad Al-Halah[1], Vamsi Krishna Ithapu[3], Philip Robinson[3], and Kristen Grauman[1,4]

[1]UT Austin, [2]UIUC, [3]Facebook Reality Labs, [4]Facebook AI Research

In this supplementary material we provide additional details about:

1. Details of the audio simulation—including grid construction, mesh upgrades, acoustic simulation technique, and connectivity graph (as referenced in Sec. 3 of the main paper).
2. Additional illustrations of pressure fields from the audio simulation and the sampled grid.
3. Reinforcement learning training utilized in the network description (as referenced in Sec. 5 of the main paper.)
4. Audio intensity baselines, as referenced in Sec. 6 of the main paper.
5. Heard/unheard sounds, as referenced in Sec. 6, Tab. 2, and Tab. 3.
6. Additional navigation trajectory examples, similar to Fig. 4 in the main paper.

8 Audio Simulation Details

Grid construction. We use an automatic point placement algorithm to determine the locations where the simulated sound sources and listeners are placed in a two-step procedure: adding points on a regular grid and then pruning. For adding points on a regular grid, first, we compute an axis-aligned 3D bounding box of a scene. Within this box we sample points from a regular 2D square grid with resolution 0.5m (Replica) or 1m (Matterport) that slices the bounding box in the horizontal plane at a distance of 1.5m from the floor (representing the height of a humanoid robot).

The second step prunes grid points in inaccessible locations. To prune, we compute how *closed* the region surrounding a particular point is. This entails tracing R uniformly-distributed random rays in all directions from the point, then letting them diffusely reflect through the scene up to B bounces using a path tracing algorithm. Simultaneously, we compute the total number of "hits" H: the number of rays that intersect the scene. After all rays are traced, the *closed-ness* $C \in [0, 1]$ of a point is given by $C = \frac{H}{R \cdot B}$. A point is declared outside the scene if $C < C_{min}$. the value of C for a particular point is below a threshold C_{min}. Finally, we remove points that are within a certain distance d_{min} from

*CC and UJ contributed equally; †work done as an intern at Facebook AI Research

the nearest geometry, as identified using the shortest length of the initial rays traced from the point in the previous pruning step.

For all scenes we use $R = 1000$, $B = 10$ and $d_{min} = 5$cm. This value of d_{min} was chosen to avoid placement of points inside walls or in small inaccessible areas. We find $C_{min} = 0.5$ works for most scenes. The exceptions are scenes with open patio areas, where we found $C_{min} = 0.1$ works best to provide a sufficient number of points on the patio.

Materials and transmission model. In addition to its geometry, a room's *materials* affect the RIR, as discussed in the main paper. To capture this aspect, we use the semantic labels provided in Replica to determine the acoustic material properties of the geometry. For each semantic class that was deemed to be acoustically relevant, we provide a mapping to an equivalent acoustic material from an existing material database [3]. For the *floor*, *wall*, and *ceiling* classes, we assume acoustic materials of carpet, gypsum board, and acoustic tile, respectively. This helps simulate more realistic sounds than if a single material were assumed for all surfaces. In addition, we add a ceiling to those Replica scenes that lack one, which is necessary to simulate the acoustics accurately.

The simulation also includes a path-tracing simulation through walls according to their material properties. Each material has absorption, scattering, and transmission coefficients. We use a transmission model similar to that used in graphics rendering. While this is modeled to ensure precision of the simulation, the impact of transmission is generally small compared to the propagation of sound through open doors [6].

Acoustic simulation technique. During the simulations, we compute the room impulse responses between all pairs of points, producing N^2 RIRs. The simulation technique stems from the theory of geometric acoustics (GA), which supposes sound can be treated as a particle or ray rather than a wave [8]. This class of simulation methods is capable of accurately predicting the behavior of sound at high frequencies, but requires special modeling of wave phenomena (e.g., diffraction) that occur at lower frequencies.Specifically, our acoustic simulation is based on a bidirectional path tracing algorithm [11] modified for room acoustics applications [1]. Additionally, it uses a recursive formulation of multiple importance sampling (MIS) to improve the convergence of the simulation [4].

The simulation begins by tracing rays from each source location in $\mathcal{S}$. These source rays are propagated through the scene up to a maximum number of bounces (200). At each ray-scene intersection of a source path, information about the intersected geometry, incoming and outgoing ray directions, and probabilities are cached. After all source rays are traced, the simulation traces rays from a listener location in $\mathcal{L}$. These rays are again propagated through the scene up to a maximum number of bounces. At each ray-scene intersection of a listener path, rays are traced to connect the current path vertex to the path vertices previously generated from all sources. If a connection ray is not blocked by scene geometry, a path from the source to listener has been found. The energy throughput along that path is multiplied by a MIS weight and is accumulated

to the impulse response for that source-listener pair. After all rays have been traced, the simulation is finished.

We perform the simulation in parallel for four logarithmically-distributed frequency bands.* These bands cover the human hearing range and are uniform in their distribution from a perceptual standpoint. For each band, the simulation output is a histogram of sound energy with respect to propagation delay time at audio sample rate (44.1kHz for Replica and 16kHz for Matterport). Spatial information is also accumulated in the form of low-order spherical harmonics for each histogram bin. After ray tracing, these energy histograms are converted to pressure IR envelopes by applying the square root, and the envelopes are multiplied by bandpass-filtered white noise and summed to generate the frequency-dependent reverberant part of the monaural room impulse response [5].

Ambisonic signals (roughly speaking, the audio equivalent of a 360° image) are generated by decomposing a sound field into a set of spherical harmonic basis. We generate ambisonics by multiplying the monaural RIR by the spherical harmonic coefficients for each time sample. Early reflections (ER, paths of order ≤ 2) are handled specially to ensure they are properly reproduced. ER are not accumulated to the main energy histogram, but are instead clustered together based on the plane equation of the geometry involved in the reflection(s). Then, each ER cluster is added to the final pressure IR with frequency-dependent filtering corresponding to the ER energy and its spherical harmonic coefficients.

The result of this process is 2nd-order ambisonic pressure impulse responses that can be convolved with arbitrary new monaural source audios to generate the ambisonic audio heard at a particular listener location. We convert the ambisonics to binaural audio [12] in order to represent an agent with two human-like ears, for whom perceived sound depends on the body's relative orientation in the scene.

9 Visualizing Audio Simulations

Next we illustrate the pressure field visualization of two other scenes in the Replica dataset. In Fig. 7, we display another big scene (apartment_2) with four rooms, with the audio source inside one of the rooms. Notice how the pressure decreases from the source along geodesic paths, which leads to doors serving as secondary sources or intermediate goals that lead the agent in the right direction.

Fig. 8 displays a second-order ambisonics representation showing the direction and intensity of the incoming direct sound. Particularly, it demonstrates the spatial properties of the audio simulation at two receiver locations. Recall that we render impulse responses for source and receiver positions sampled from a grid in each scene. These impulse responses are stored in ambisonics and converted to binaural to mimic the signals received by a human at the entrance of the ear canal. We create Fig. 8 by evaluating the incoming energy of the direct

*[0Hz,176Hz], [176Hz,775Hz], [775Hz,3409Hz], [3409Hz,20kHz]

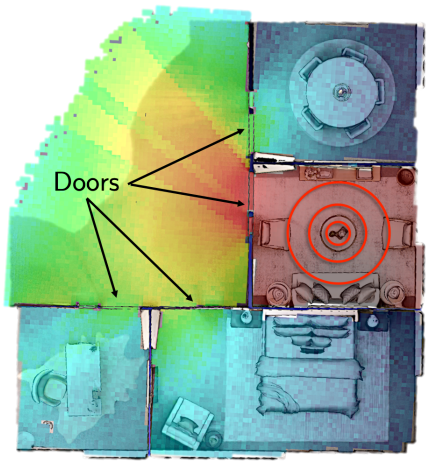

Fig. 7: **Pressure field of audio simulation** overlaid on the top-down map of apartment_2 from Replica [10]. Our audio-enabled agent gets rich directional information about the goal, since the pressure field variation is correlated with the shortest distance. Notice the discontinuities across walls and the gradient of the field along the *geodesic* path an agent must use to reach the goal (different from shortest Euclidean path). As a result, to an agent standing in the top right or bottom rooms, the audio reveals the door as a good intermediate goal. In other words, the audio stream signals to the agent that it must leave the current room to get to the target. In contrast, the GPS displacement vector would point through the wall and to the goal, which is a path the agent would discover it cannot traverse. Note that the visual stream is essential to couple with the audio stream in order to navigate around obstacles.

sound (excluding reflections and reverberation) at the horizontal plane.[†] The greater the energy the bigger the size, and the orientation depicts the angular distribution of energy. In Location 1 energy comes predominantly from its right. Since it is closer to the audio source, the directional sound field has more energy than Location 2.

10 Reinforcement Learning Training Details

In the following, we provide details of our reinforcement learning (RL) formulation for navigation tasks. This notation links to Sec. 4 and Fig. 3 in the main paper.

An agent embedded in an environment must take actions from an action space $\mathcal{A}$ to accomplish an end goal. For our tasks, the actions are navigation motions: $\mathcal{A} = \{MoveForward, TurnLeft, TurnRight, Stop\}$. At every time step

[†]The minor side lobes pointing in directions other than the source are a result of representing the sound field as a 2^{nd} order ambisonics signal, thus using only 9 spherical harmonics. We refer the reader to [2, 7, 13] for more details on ambisonics sound field representation.

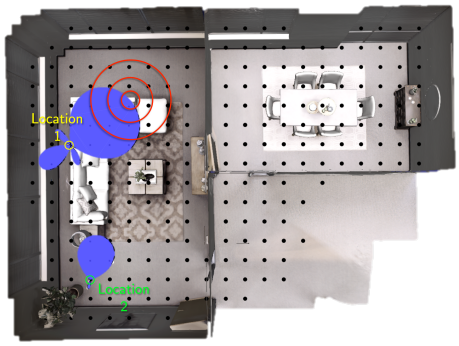

Fig. 8: **Visualizing ambisonics.** We visualize the ambisonics components (blue *lobes*) of the impulse response. Notice that the ambisonics sound fields characterize direction and intensity of the incoming energy.

$t = \{0, 1, 2, \ldots, T-1\}$ the environment is in some state $s_t \in \mathcal{S}$, but the agent obtains only a partial observation of it in the form of o_t. Here T is a maximal time horizon, which corresponds to 500 actions for our task. The observation o_t is a combination of the audio, visual, and displacement vector inputs.

Using information about the previous time steps h_{t-1} and current observation o_t, the agent develops a policy $\pi_{t,\theta} : \mathcal{A} \rightarrow [0, 1]$, where $\pi_{t,\theta}(a|o_t, h_{t-1})$ is the probability that the agent chooses to take action $a \in \mathcal{A}$ at time t. We use the shorthand of $\pi_{t,\theta}(o_t, h_{t-1})$ to show the feed-forward nature of the actor head. After the agent acts, the environment goes into a new state s_{t+1} and the agent receives individual rewards $r_t \in \mathbb{R}$.

The agent optimizes its *return*, *i.e.* the expected discounted, cumulative rewards

$$G_{\gamma,t} = \sum_{t=0}^{T-1} \gamma^t r_t, \tag{1}$$

where $\gamma \in [0, 1]$ is the discount factor to modulate the emphasis on recent or long term rewards. The value function $V_{t,\theta}(o_t, h_{t-1})$ is the expected return. The particular reinforcement learning objective we optimize directly follows from Proximal Policy Optimization. We refer the readers to [9] for additional details on optimization.

11 Audio Intensity Baseline

In the main paper, we presented an audio intensity baseline in Sec **??**. It is an ablation of our model where the policy is learned directly from the intensity of the left and right waveforms together with the depth-based visual stream. We compute the intensity of audio using the Root-Mean-Square (RMS) of channel's waveform, which produces two real numbers as the audio feature. We showed that it is inferior to our approach, meaning that our model is able to learn

Table 4: Intensity only versus spectrograms as audio input for our model and with different visual inputs for AudioGoal agents (blind / RGB / depth).

Audio Features	Replica	MP3D
Intensity only	0.276 / 0.177 / 0.291	0.173 / 0.003 / 0.014
Spectrograms	0.673 / 0.626 / 0.756	0.438 / 0.479 / 0.552

additional environment information from the full spectrograms. Here we provide the parallel results for the blind and RGB visual streams (Tab. 4).

We see a significant drop in performance when using audio intensity only compared to spectrograms. This demonstrates that our model extracts useful acoustic features for navigation (*e.g.* relative angle to goal, major obstacles) that go beyond just intensity.

12 Heard/Unheard Dataset Splits

In the following we provide details about the sounds used in Sec. 6. We utilize 102 copy-free natural sounds across a wide variety of categories: air conditioner, bell, door opening, music, computer beeps, fan, people speaking, telephone, and etc. We divide these 102 sounds in to non-overlapping 73/11/18 splits for train, validation and test.

For Tab. 2 and the *same sound* experiment in Tab. 3 of the main paper, we use the sound source of 'telephone'. In Tab. 3, for the *varied heard sounds* experiment we train using the 78 sounds and test on unseen scenes with the same sounds. Recall that the audio observations vary not only according to the audio file but also the 3D environment. For the *varied unheard sounds* experiment, we use the 78 sounds for training scenes, and generalize to unseen scenes as well as unheard sounds. Particularly, we utilize the 11 sounds for validation scenes, and the rest 18 sounds for test scenes.

13 Additional Navigation Trajectory Examples

Fig. 9 shows four additional trajectory examples of three agents in different test environments of Replica and Matterport3D. These trajectories show the AudioGoal agent and AudioPointGoal agent navigate to goals more efficiently compared to PointGoal.

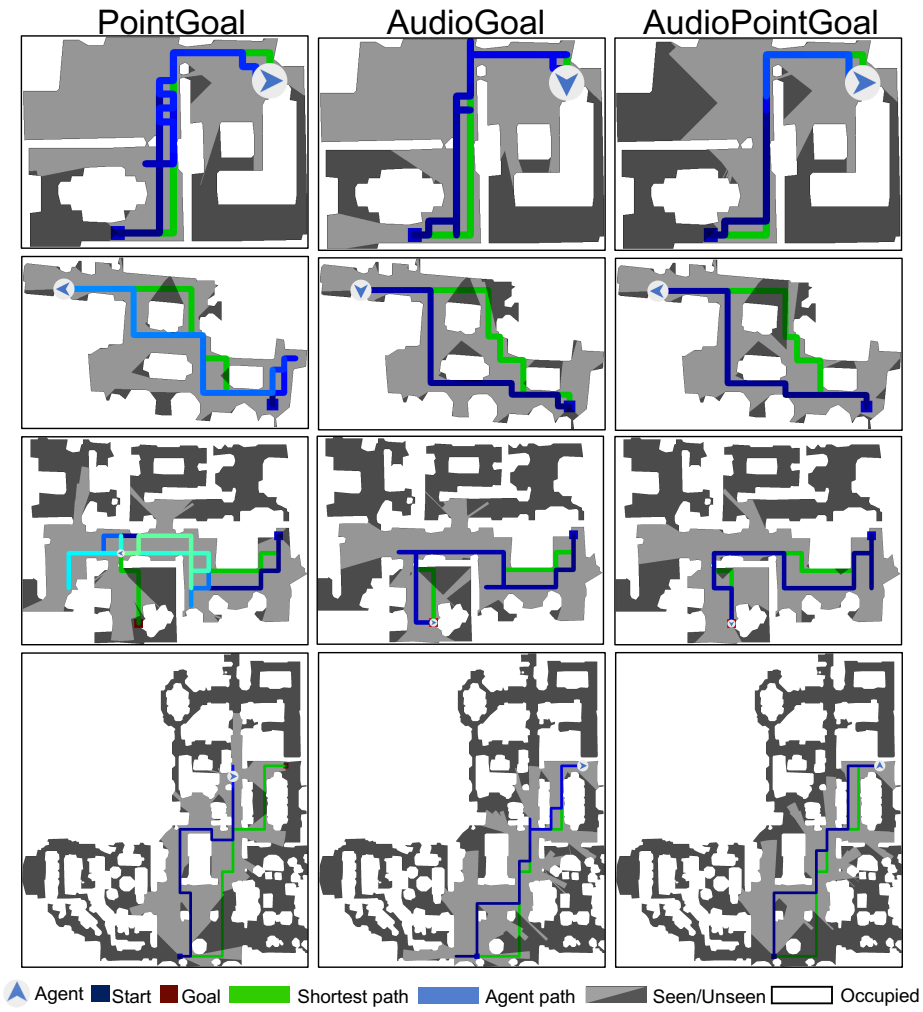

Fig. 9: **Navigation trajectories on top-down maps.** The top two and bottom two rows are environments in Replica and Matterport3D, respectively. Agent path color fades from dark blue to light blue as time goes by. Green path indicates the shortest geodesic path. In this figure, we show navigation trajectories of three agents in varied test environments. The AudioGoal agent and AudioPointGoal agent navigate more efficiently compared to PointGoal agent. Best viewed in color.

References

1. Cao, C., Ren, Z., Schissler, C., Manocha, D., Zhou, K.: Interactive sound propagation with bidirectional path tracing. ACM Transactions on Graphics (TOG) (2016)
2. Daniel, J.: Spatial sound encoding including near field effect: Introducing distance coding filters and a viable, new ambisonic format. In: Audio Engineering Society Conference: 23rd International Conference: Signal Processing in Audio Recording and Reproduction. Audio Engineering Society (2003)
3. Egan, M.D., Quirt, J., Rousseau, M.: Architectural acoustics (1989)
4. Georgiev, I.: Implementing vertex connection and merging. Technical Re-port. Saarland University. (2012)
5. Kuttruff, K.H.: Auralization of impulse responses modeled on the basis of ray-tracing results. Journal of the Audio Engineering Society (1993)
6. Locher, B., Piquerez, A., Habermacher, M., Ragettli, M., Röösli, M., Brink, M., Cajochen, C., Vienneau, D., Foraster, M., Müller, U., et al.: Differences between outdoor and indoor sound levels for open, tilted, and closed windows. International journal of environmental research and public health (2018)
7. Rafaely, B.: Fundamentals of Spherical Array Processing. Springer (2015)
8. Savioja, L., Svensson, U.P.: Overview of geometrical room acoustic modeling techniques. The Journal of the Acoustical Society of America (2015)
9. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
10. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., et al.: The replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
11. Veach, E., Guibas, L.: Bidirectional estimators for light transport. In: Photorealistic Rendering Techniques (1995)
12. Zaunschirm, M., Schörkhuber, C., Höldrich, R.: Binaural rendering of ambisonic signals by head-related impulse response time alignment and a diffuseness constraint. The Journal of the Acoustical Society of America (2018)
13. Zotter, F., Frank, M.: Ambisonics: A Practical 3D Audio Theory for Recording, Studio Production, Sound Reinforcement and Virtual Reality. Springer (2019)