

---

# SoundSpaces 2.0: A Simulation Platform for Visual-Acoustic Learning

---

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

## Abstract

We introduce SoundSpaces 2.0, a platform for on-the-fly geometry-based audio rendering for 3D environments. Given a 3D mesh of a real-world environment, SoundSpaces can generate highly realistic acoustics for arbitrary sounds captured from arbitrary microphone locations. Together with existing 3D visual assets, it supports an array of audio-visual research tasks, such as audio-visual navigation, mapping, source localization and separation, and acoustic matching. Compared to existing resources, SoundSpaces 2.0 has the advantages of allowing continuous spatial sampling, generalization to novel environments, and configurable microphone and material properties. To our best knowledge, this is the first geometry-based acoustic simulation that offers high fidelity and realism while also being fast enough to use for embodied learning. We showcase the simulator’s properties and benchmark its performance against real-world audio measurements. In addition, through two downstream tasks covering embodied navigation and far-field automatic speech recognition, highlighting sim2real performance for the latter. SoundSpaces 2.0 is publicly available to facilitate wider research for perceptual systems that can both see and hear.<sup>2</sup>

## 1 Introduction

What we see and hear dominates our perceptual experience, and there is often a strong relationship between the two modalities. At the object level, we can anticipate the sounds an object makes based on how it looks, and vice versa (a dog barks, a door slams, a baby cries). At the environment level, materials and geometry of the surrounding 3D space that we see transform the sounds that reach our ears. For example, a person speaking in a marble-floored, high-ceiling museum sounds distinct from one speaking in a cozy carpeted bookshop.

Modeling the correspondence between visuals and acoustics in 3D spaces is of vital importance for many applications in embodied AI and augmented/virtual reality (AR/VR). For instance, a rescue robot needs to localize the person who is calling for help; a service robot needs to look and listen to know if the espresso machine is running properly; an AR system needs to generate sounds that are consistent with the user’s acoustical environment for an immersive experience.

Realistic simulations of the first-person perceptual experience are a valuable resource for AI research. They allow training and evaluating models at scale and in a replicable manner. On the visual side, fast visual simulators [58, 68] coupled with 3D assets from scanned real-world environments [11, 76, 66, 54] have facilitated substantial work in visual navigation and related tasks in recent years [75, 12, 6, 30, 55, 33], enabling rigorous benchmarks [1] and even successful “sim2real” transfer to agents that move in the real world [74, 71, 35]. On the audio side, acoustic simulation has been traditionally

---

<sup>\*</sup>Equal contribution

<sup>2</sup><https://github.com/facebookresearch/sound-spaces>

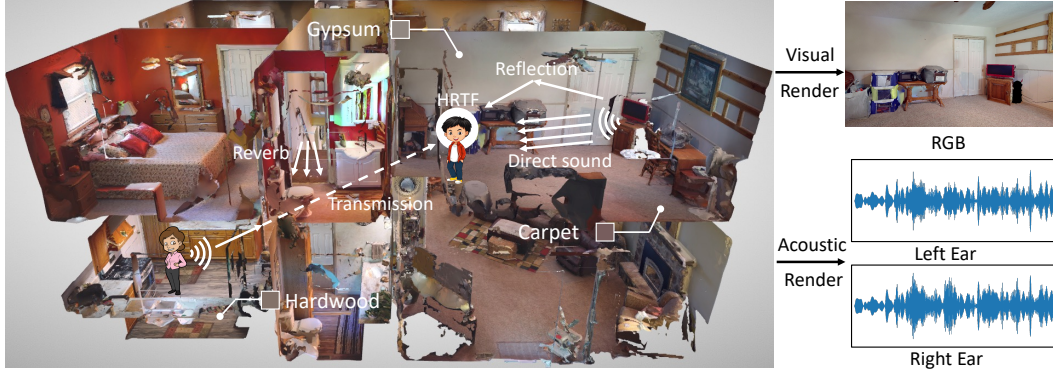


Figure 1: Illustration of SoundSpaces 2.0 rendering in a multi-room multi-floor HM3D [54] environment. In this scenario, a boy is watching TV in the living room while his mom calls him to have dinner from the kitchen downstairs. We model various frequency-dependent acoustic phenomenon for sound propagation from all sources (TV and mom) to him, including direct sound, reflection, reverb, transmission, diffraction and air absorption. The sound propagation is based on a bidirectional path-tracing algorithm that takes the geometry of the scene as well as materials of objects in the space as input. The received sound is spatialized to binaural with the head-related transfer function (HRTF). As a result, SoundSpaces 2.0 renders the visual and audio observations with *spatial and acoustic correspondence*. For example, the TV being situated more towards the right results in right-ear signals that are stronger than those in the left ear.

pursued for physical models [9], gaming [43] and auralization for architectural design [73], typically restricted to simple parametric geometries and in isolation from visual context.

Towards bringing the two modalities together in joint audio-visual simulations, recent work offers initial steps [14, 21]. SoundSpaces [14] provides highly realistic room impulse responses (RIRs)<sup>3</sup> obtained via bidirectional path tracing for 100 real multi-room environment meshes from Replica [66] and Matterport3D [11], while ThreeDWorld [21] uses physics simulations in the Unity3D video game platform to model object collisions, impact sounds, and environment reverberation. These tools support an array of new research in navigation [22, 17, 14, 15, 12, 45], floorplan reconstruction [53], feature learning [23], audio-visual (de)reverberation [16, 13], and audio-visual collision detection [20].

Though inspiring, these early platforms have several limitations. SoundSpaces’s foremost limitation is its pre-computed, discretized nature. The provided RIRs are pre-computed for all source and receiver pairs on a 0.5m grid, and for a fixed list of 100 total environments. This prevents sampling data at new locations. This in turn means that 1) an agent in the simulator can only move or hop between discrete grid points in the space, which which abstracts away some difficult parts of the navigation task; 2) the simulations do not generalize to novel environments—just the 100 provided; and 3) the pre-computed data itself is on the order of TBs, impeding the ability to change configurations, e.g., of the microphone types or materials. ThreeDWorld offers continuous-space rendering, yet it only supports audio rendering for simple 3D environment geometry, namely an oversimplified “shoebox” (rectangular parallelepiped) model, thus is not applicable to real-scan datasets [76, 54]. In sum, today’s audio-visual rendering platforms fall short in accuracy, speed, and flexibility, which in turn constrains the scope of research tasks they can support, such as audio-visual embodied learning [12, 17] or visual-acoustic learning [65, 44, 13].

In this work, we introduce SoundSpaces 2.0, which performs on-the-fly geometry-based audio rendering for arbitrary environments. It allows highly realistic rendering of arbitrary camera views and arbitrary microphone placements for waveforms of the user’s choosing, accounting for all major real-world acoustic factors: direct sounds, early specular/diffuse reflections, reverberation, binaural spatialization, and frequency-dependent effects from materials and air absorption (illustrated in Fig. 1). Furthermore, SoundSpaces 2.0 generalizes audio simulation to *any* input mesh, making it possible for

<sup>3</sup>A room impulse response (RIR) is the transfer function that defines how sound is transformed by the environment for a given source and receiver (microphone) location pair.

Table 1: Comparison with existing non-commercial datasets/simulation platforms. *Geometric* refers to acoustic simulation that is based on geometry of the objects and the space. *Configurable* means ability to alter simulation parameters, material and microphone properties. *Arbitrary Env* refers to the ability to render for an arbitrary new mesh environment, including point clouds generated in the wild.

Platform	Audio-Visual	Geometric	Configurable	Arbitrary Env
SoundSpaces [14]	✓	✓	✗	✗
GWA [69]	✗	✓	✗	✗
ThreeDWorld [21]	✓	✗	✓	✗
Pyroomacoustics [59]	✗	✗	✓	✗
SoundSpaces 2.0 (Ours)	✓	✓	✓	✓

the first time to import sound into well-used environment assets like Gibson [76], HM3D [54], and Matterport3D [11], as well as any future or emerging one like Ego4D [25]. In addition, SoundSpaces 2.0 allows users to configure various properties of the simulation such as source-receiver locations, simulation parameters, material properties, and the microphone configuration. The rendering platform and associated research codebase will be made publicly available.

In this paper, we describe the new platform and its functionality, and we illustrate its flexibility with various concrete examples (please see also the Supplementary video). In addition, we perform systematic experiments to answer two questions: 1) how accurate are the audio-visual simulations? and 2) how well can machine learning models trained in SoundSpaces 2.0 generalize to real world data? For this purpose, we collect real-world audio RIR measurements for a public scene dataset Replica [66] and benchmark the simulation accuracy. We also benchmark two downstream tasks: continuous audio-visual navigation and far-field speech recognition. For speech recognition, we show the machine-learning models trained on our synthetic data can generalize when tested on real data. We propose an acoustic randomization technique that models the real-world distribution of materials’ acoustic properties, and we show that this strategy leads to better sim2real generalization. Finally, aside from the rendering engine itself, which is readily integrated with Habitat [58], we also release SoundSpaces-PanoIR: a large-scale dataset of images paired with RIRs computed in SoundSpaces 2.0; this prepared dataset can facilitate future research on visual-acoustic learning in a stand-alone manner (without interfacing with the simulators themselves).

## 2 Related Work

We overview related work on simulations, audio(-visual) learning, and sim2real transfer.

**Acoustic simulation.** Sounds are first produced by vibrating objects and then propagate in space before reaching human ears. Modeling sound propagation has a long history in the literature, the goal of which is to simulate realistic high-fidelity audio that is consistent with the given environment specification. Interactive acoustic simulation systems have been extensively used in games and AR/VR applications. Sound propagation algorithms typically fall into two main categories: wave-based [3, 34, 49] and geometric [19, 40, 62]. Wave-based methods aim to solve the wave equation numerically, resulting in high computation expense. In the geometric method family, the Image-Source Methods [4] solve the specular reflection of sounds deterministically but have low accuracy for late reverb, while path-tracing based approaches offer both high accuracy and efficiency [57].

Our work builds on the SoundSpaces [14] dataset in that we use their bidirectional path-tracing algorithm and simulation framework as a starting point; however, as discussed above, we overcome its core limitations by enabling on-the-fly rendering, and we also augment the propagation algorithm by adding diffraction and improving reverberation level accuracy. Compared to existing public platforms, SoundSpaces 2.0 adds significant generality and flexibility—accepting arbitrary scene geometry, generalizing to new 3D meshes on-the-fly, rendering in real-time, and allowing configuration of materials and microphones—all of which we demonstrate. See Table 1 for comparisons.

**Audio-visual learning.** Recent advances in audio-visual learning include self-supervised cross-modal feature learning from video [5, 38, 48], object localization [28], and audio-visual speech enhancement and source separation [18, 51, 78, 2, 47, 79, 27]. Besides learning from video, audio-visual simulation supports the study of embodied tasks like navigation [14, 22, 17, 45], where an agent moves intelligently based on the visual and auditory observations. Another line of research facilitated by simulation is visual-acoustic learning [65, 16, 13, 44], where the goal is to either match or remove

the room acoustics implied by the image. Acoustic rendering also facilitates audio-only research, such as far-field speech recognition [36, 46, 69], sound source separation [31, 7], localization [26, 31] and sound synthesis [32, 70]. Our work revisits multiple audio(-visual) learning tasks to showcase the advantages of having a continuous, configurable, and generalizable simulation.

**Simulation-to-reality transfer.** Several large-scale datasets of real-world 3D scans of buildings have been released in the past few years [76, 11, 54]. In parallel, multiple simulation environments [76, 58, 37] have been created in order to simulate embodied motion in these 3D scans. These advances allow large-scale training, fast experimentation, consistent benchmarking, and replicable research compared to physical experimentation. Transferring the model trained in simulation to the real world is thus of great interest. The mostly widely used approaches are domain randomization [71, 72], system identification [35, 39], and transfer learning and domain adaptation [80]. Most sim2real transfer research studies transferring a policy from simulation to the real world based on visual input. While some works leverage synthetic audio data for speech tasks [69, 31], transferring models trained on acoustic simulation has been understudied due to the lack of real-world benchmarks. To the best of our knowledge, this is the first work to both benchmark simulation performance with real measurements (Sec. 5.2) as well conduct sim2real transfer for machine learning models (Sec. 5.4).

### 3 SoundSpaces 2.0 Audio-Visual Rendering Platform

In this section, we detail the audio-visual rendering pipeline for SoundSpaces 2.0.

#### 3.1 Rendering Pipeline and Simulation Enhancements

The core of SoundSpaces 2.0 is the audio propagation engine (RLR-Audio-Propagation) we are releasing for research purposes.<sup>4</sup> We integrate this engine into the existing visual simulator Habitat-Sim [58], which offers fast visual rendering.<sup>5</sup> In addition, we provide high-level APIs for various downstream tasks (e.g., navigation) and training scripts at the SoundSpaces repo.<sup>6</sup>

Fig. 1 illustrates the propagation pipeline. RLR-Audio-Propagation takes the scene mesh data processed by Habitat, together with source and receiver locations specified by the user, and computes a room impulse response (RIR) using a bidirectional path-tracing algorithm [10]. This module models various acoustic phenomena, including reflection, transmission, and diffraction, as well as spatialization. The simulation operates in  $M$  logarithmically-spaced frequency bands (configurable), where it computes an energy-time histogram at the audio sampling rate. This histogram incorporates spatial information using spherical harmonics for each time sample that represents the directional distribution of arriving sound energy. This representation is then spatialized to either an ambisonic or binaural pressure impulse response [64], which can be convolved with the source audio signals to generate the sound at the receiver position. See Supp. for more details.

Compared to the original SoundSpaces, we have improved the simulation in a few ways. SoundSpaces did not include any simulation of acoustic diffraction, and thus exhibited abrupt occlusion of sources. We have removed this limitation using the fast diffraction approach from [63], which is able to efficiently compute smooth diffraction effects for occluded sources. We also improved the accuracy of the direct-to-reverberant ratio (DRR), the ratio of the sound pressure level of a direct sound from a directional source to the reverberant sound pressure level, by fixing a bias of  $\sqrt{4\pi}$  that was present in the indirect sound pressure of the original SoundSpaces.

In the following, we overview modeling advances in SoundSpaces 2.0 that promote continuity, configurability, generalizability, and performance.

#### 3.2 Continuity

**Spatial continuity.** Humans move around in the real world continuously while hearing. Given an arbitrary source location  $s$ , receiver location  $r$ , and receiver’s heading direction  $\theta$  in a given mesh environment, we render the impulse response between the source and receiver as  $R(s, r, \theta)$ . The sound received by the receiver is computed as  $A^r = A^s * R(s, r, \theta)$ , where  $A^s$  is the sound emitted from the source and  $*$  denotes convolution. Whereas SoundSpaces [14] restricts the  $s$  and  $r$  locations to a 0.5m discrete grid due to its pre-computed approach and hefty storage requirements, SoundSpaces 2.0 allows arbitrary placements.

<sup>4</sup><https://github.com/facebookresearch/rlr-audio-propagation>

<sup>5</sup><https://github.com/facebookresearch/habitat-sim/blob/main/docs/AUDIO.md>

<sup>6</sup><https://github.com/facebookresearch/sound-spaces>

**Acoustic continuity.** While an agent moves in the environment, it moves smoothly from point A to point B (even with a small step size). With the spatial continuity property, we can render  $R(s, r_A, \theta_A)$  and  $R(s, r_B, \theta_B)$  for these two locations respectively. However, the original SoundSpaces takes the rendered IR for each location and convolves it with the source sound directly as the audio observation. This calculation implicitly assumes the source does not emit sound continuously, i.e., it starts to emit when the agent moves to a new location, stops after one second, and resumes at agent’s next location.

In SoundSpaces 2.0, we introduce acoustic continuity for both the source sound and listener. More specifically, given a sampling rate  $F$  and the time between two steps  $\Delta t$ , the number of received audio samples is  $N = F\Delta t$  per step. Assuming a listener is at location  $x_i$  at time  $t_i$ , the audio signal received by the listener at time  $t_i$  emitted from the source at time  $t_p = t_i - R(s, x_i, \theta_{x_i}) + 1$ . We take the corresponding source sound segment  $A^s[t_p : t_p + N]$  and convolve it with  $R(s, x_i, \theta_{x_i})$  without zero padding to compute  $A_{t_i}^{x_i}$ . Following the common practice [50], we apply linear crossfading between  $A_{t_i}^{x_{i-1}}$  and  $A_{t_i}^{x_i}$  to smooth out the transition from  $x_{i-1}$  to  $x_i$  with an overlap time window of  $T$  seconds. See Supp. video for the impact on perceptual quality.

### 3.3 Configurability

Due to its pre-computed nature, it is impossible to change any simulation setup (parameters, microphones, or materials) for the original SoundSpaces. All are configurable in SoundSpaces 2.0, as summarized below and in more detail in Supp.

**Simulation parameters.** We expose many useful parameters for users to configure, including the sampling rate, the number of frequency bands, number of rays for direct/indirect sounds, whether reflection, transmission or diffraction is enabled, etc.

**Microphone types.** We provide several types of built-in microphone configurations, including monaural single-channel audio, binaural (modeling a human listener), and ambisonics (full sphere surround sound). In addition, users are also able to configure their own microphone array by specifying an array of monaural microphone locations.

**Material modeling.** Materials of objects/surfaces have a big impact on how humans perceive the sound in an environment. Consider the difference between sound in a recording studio versus a living room of the same size. Due to the absorptive materials in the recording studio, the sound will consist primarily of direct sound without reverberation, whereas in the living room, the sound will consist of a mixture of direct sound and reverberation.

Existing real-scan datasets have semantic annotations at the level of object categories, e.g., chair, table, couch and floor, while lacking material annotations of what these objects are made of, e.g., wood or steel for tables. SoundSpaces coped with this issue by defining a fixed mapping from object categories to acoustic materials, e.g., floors are always mapped to the carpet material, which is very absorptive. However, this fixed mapping fails to reflect the fact in the real world, different instances of the same object category could have very different acoustic properties, e.g., a floor could be carpet or wood or concrete materials depending on home types.

To account for this variation, we expose an API to let users define their own acoustic material configurations. We provide 29 built-in acoustic materials, e.g., wood, concrete, curtain, soil, water. Every acoustic material has a list of candidate object categories to be mapped from. It also has a set of coefficients for absorption, scattering, and transmission in the following format:  $[f_1, c_1, f_2, c_2, \dots, f_n, c_n]$ , where  $f_i$  is a frequency and  $c_i$  is the coefficient for a certain acoustic phenomenon at frequency  $f_i$ . This allows modeling the frequency-dependent acoustic properties of different acoustic materials. For example, high-frequency waves are absorbed more compared to low frequencies when reflecting from carpets. See Supp. for details.

We also model distance-dependent damping of the sound propagation media. This includes air absorption as well as transmission losses through materials. Air absorption is calculated using an analytical model [8]. Users can specify the frequency-dependent damping coefficients for each material, expressed as dB per meter, in a similar format to the other material properties.

### 3.4 Generalizability

**Generalization to scene datasets.** Our new simulator accommodates arbitrary 3D meshes as input. This makes it compatible with all available scene datasets (e.g., Gibson [76], HM3D [54], Ego4D [25], Matterport3D [11], Replica [66]), as well as any future assets that become available, such as if a user



Figure 2: An example of the SoundSpaces-PanoIR dataset rendered on Gibson [76]. We render panoramic RGB and depth images to capture the scene geometry. We provide the images, impulse responses, and the coordinates of the point source location with respect to the current camera pose.

scans their own lab or home environment. This is an important advance over SoundSpaces, which was restricted to Replica and Matterport3D alone. See Supp. for videos of generated examples.

**Generalization to the real world.** The fidelity and flexibility of our simulation platform also supports generalization to the real world. In Sec. 5, we score the simulator output against real-world RIRs and show how machine learning models trained on SoundSpaces 2.0 can generalize to real data.

### 3.5 Rendering Modes and Rendering Performance

Our simulation generates high-quality audio rendering based on mesh and materials, and this fidelity can be instrumental for certain research areas. On the other hand, in tasks like embodied navigation with reinforcement learning, which typically require millions (or even billions [75]) of training iterations, rendering speed is of vital importance. Thus, we offer two built-in rendering modes: *high-speed* and *high-quality*.

In high-speed mode, we reduce the number of rays and improve the accuracy by leveraging previously computed impulse responses [61], under the assumption that movements are spatially continuous. Our algorithms use information computed on previous simulation frames, such as sound propagation paths and RIRs, to reduce the number of rays and ray bounces that are needed on each frame for sufficient sound quality (see Sec.5.1). In high-quality mode, we set all rendering parameters to max and turn off the temporal coherence feature to ensure that every impulse response is accurate without temporal blurring. Our engine is multi-threaded and users can set the number of threads when using either mode. See Sec. 5.1 for analysis of the simulation performance in terms of speed and accuracy.

## 4 Large-scale SoundSpaces-PanoIR Dataset

We are releasing SoundSpaces 2.0 as a general-purpose platform with which a user can generate observations on-the-fly (particularly relevant for embodied AI models), or populate a new offline dataset of their own design. As an example of the latter, and to ease adoption for researchers who wish to work with visual-acoustic scene data without a layer of agent interaction, we next use SoundSpaces 2.0 to compose a large-scale dataset called *SoundSpaces-PanoIR* that couples IRs with images.

For visual-acoustic learning tasks, such as audio-visual dereverberation [16] and synthesizing acoustics based on visuals [13, 65, 44], there are no existing large-scale accurate image-IR datasets due to the high expense and complexity of data collection. Our SoundSpaces-PanoIR dataset has 1M panoramic image and IR pairs rendered from 750 environments across the Matterport3D, Gibson, and HM3D datasets. We provide the data in the following format: panorama (RGB/Depth), IR, polar coordinates of the source with respect of the center of the panorama. Fig. 2 shows one example in Gibson. See Supp. for more examples and statistics. We will also share the data generation scripts.

## 5 Evaluation and Benchmarks

Next we evaluate both the simulation quality and its value for downstream tasks with two machine learning benchmarks. Fig. 3a illustrates these two tasks.

### 5.1 Simulation Speed vs. Quality Tradeoff

To understand the tradeoff between the quality versus speed of rendering, we report the accuracy and speed of different modes by rendering RIRs along random trajectories with an average length of 15m across 20 Matterport3D environments. We profile the speed on a Xeon(R) Gold 6230 CPU with 2.10GHz. See Table 2. For accuracy, we measure the relative RT60 error of RIRs generated in high-speed mode compared to RIRs generated in high-quality mode. RT60 is a standard acoustic

Table 2: Simulation speed vs. quality tradeoff. We report mean and standard deviation over 5 runs.

	Relative RT60 Error (%)	1 Thread (FPS)	5 Threads (FPS)
High-quality	$0.0 \pm 0.0$	$0.9 \pm 0.0$	$4.0 \pm 0.1$
High-speed	$9.5 \pm 0.2$	$7.7 \pm 0.2$	$33.5 \pm 0.4$

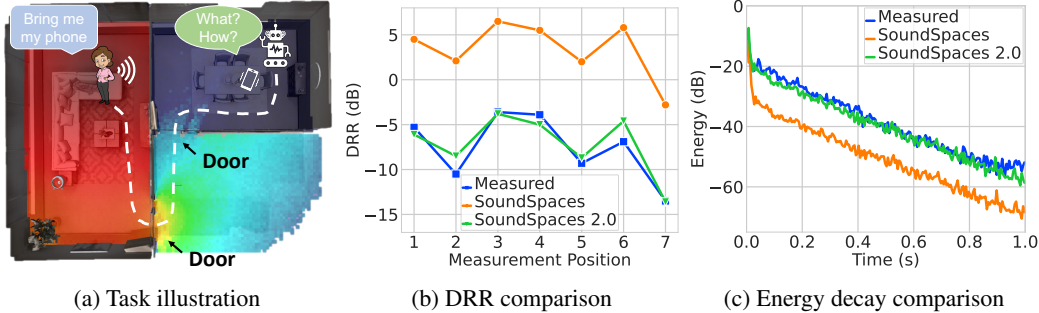


Figure 3: **(a)** In this example, a person’s phone rings in the dining room while she is in the living room and she asks the robot to bring her the phone. Upon receiving the audio signal with the binaural microphone, the robot needs to figure out two things: 1) what she is saying (far-field automatic speech recognition) and 2) how to navigate to her and the phone (audio-visual navigation). Note that far-field ASR is not limited to robotics; it has various applications such as video captioning. **(c)** Comparing real measurements and simulations in the Replica apartment [66] for 7 measurement positions and the 250Hz to 4000Hz frequency band. SoundSpaces 2.0 has much lower error for the direct-to-reverberant ratio (DRR) compared to SoundSpaces. **(c)** Energy decay curve comparisons. The energy decay curve of SoundSpaces 2.0 is much closer to the real measurements than SoundSpaces.

measurement that is defined as the time it takes for the sound pressure level to reduce by 60 dB [29]. We see high-speed greatly improves efficiency over the high-quality mode, by  $8\times$  with single thread and  $33\times$  with 5 threads, while only losing 9.5% accuracy despite RT60 calculation being noisy. When coupled with distributed training, it meets the requirement of today’s RL agent training. In addition, we test the navigation model trained in high-speed mode on high-quality mode; the performance difference is smaller than 1% compared to the test performance in high-speed mode in Table 3. We treat high-quality mode as the gold-standard and benchmark its quality against real-world IRs next.

## 5.2 Validating Simulation Accuracy with Real IRs

How realistic are our audio simulations? To quantify this, we collect real acoustic measurements of the FRL apartment from the Replica dataset [66] and compare them to SoundSpaces 2.0 outputs. IR measurements were captured at seven different source/receiver positions throughout the real-world apartment using an omnidirectional B&K Type 4295 speaker (100Hz to 8kHz frequency response) and Earthworks M30 microphone with the exponential sine sweep method. These measurements will be released as well to assist future research.

Figure 3b compares the measurements to the corresponding simulations at the same source/receiver positions, for both the original SoundSpaces and the proposed SoundSpaces 2.0 (high-quality mode).<sup>7</sup> We report the direct-to-reverberant ratio (DRR) acoustic parameters derived from the impulse responses [29] in Figure 3b. SoundSpaces 2.0 has a better match of direct-to-reverberant ratio, where the error compared to measurements is reduced from 11.0 dB to 0.98 dB on average, while preserving the same relative RT60 error of 12.4% (see Supp.). Figure 3c reinforces that advantage, plotting the energy-time curves of the simulations versus the real measurements from 250Hz to 4000Hz. Overall, the proposed new features and improvements lead to higher realism for the acoustic simulation.

## 5.3 Benchmark 1: Continuous Audio-Visual Navigation

Navigating to localize the sound source in an unmapped environment has many real world applications, such as rescue robot or service robot (e.g., find the person calling for help or the ringing phone). The audio-visual navigation task (AV-Nav), originally introduced in [14], is gaining attention from the

<sup>7</sup>The measurements were scaled to match the direct sound level of the simulations. The acoustic material properties of the mesh were optimized to match the measurements following [60].



Table 3: Continuous audio-visual navigation benchmark. DTG stands for distance to goal. We report the mean and standard deviation by training on 1 random seed, and evaluating on 3 random seeds.

Train	Test	Success (%)	SPL (%)	DTG (m)
SoundSpaces [14]	Continuous space	$64.2 \pm 0.8$	$27.5 \pm 0.4$	$5.6 \pm 0.2$
SoundSpaces [14]	Continuous space & continuous sound	$0.9 \pm 0.2$	$0.3 \pm 0.1$	$12.9 \pm 0.1$
SoundSpaces 2.0	Continuous space & continuous sound	$64.7 \pm 3.9$	$49.3 \pm 3.0$	$5.9 \pm 0.5$

Table 4: Far-field automatic speech recognition benchmark.

	Word Error Rate (%)
Pretrained	29.10
Finetuned on Pyroomaoustics [59]	16.24
Finetuned on SoundSpaces 2.0	12.48

broader community via public competitions at CVPR 2021 and CVPR 2022.<sup>8</sup> However, due to its reliance on SoundSpaces, AV-Nav thus far must assume the agent travels along the discrete grid. The navigation task is thus easier due to the lack of collisions and implied perfect localization.

Here we introduce the *continuous* AV-Nav task, enabled by SoundSpaces 2.0 simulation. In this task, the agent can either move forward 0.15 m per step at a speed of 1m/s or turn left/right 10 degrees. If the agent issues a stop action within 1m radius of the goal, the episode is regarded as successful. Importantly, the agent not only moves in continuous space but also receives acoustically continuous audio signals (cf. Sec. 3.2). We use the high-speed rendering mode.

We generalize the existing audio-visual navigation (AV-Nav) agent [14] to a distributed audio-visual navigation (DAV-Nav) agent equipped with DD-PPO [75] to speed up the training process. We train and test on the AudioGoal navigation dataset [14]. To ablate the simulation improvement as detailed in Sec. 3.1, for the SoundSpaces baseline, we train DAV-Nav on SoundSpaces’ discrete setup with data rendered from the enhanced simulation; the action space is the same as during training.

Table 3 shows the results using the standard metrics of success rate, success rate normalized by path length (SPL), and distance to goal. If only the space is continuous, the DAV-Nav agent trained on SoundSpaces has 64.2% success rate and 27.5% SPL on average compared to 64.7% success rate and 49.3% SPL of the agent trained on SoundSpaces 2.0. This shows spatial continuity mostly harms the agent’s efficiency rather than its success rate; the agent can still navigate to the source despite having more collisions. However, when the sound is acoustically continuous, the baseline’s performance drops. This is likely because the agent relies on the direct-sound cue that is (inaccurately) always present in the audio, while in the continuous-sound rendering, direct sound is always mixed with the reverberation in the environment, making navigation more difficult. In comparison, the agent trained on SoundSpaces 2.0 achieves a much higher success rate and is much closer to the goal location on average. This shows it is essential to model both spatial and acoustic continuity for AV-Nav, which SoundSpaces 2.0 enables. Furthermore, recall that SoundSpaces 2.0 opens up any other 3D scene dataset for exploring AV-Nav, whereas previously only Replica or Matterport3D were applicable.

## 5.4 Benchmark 2: Far-Field Automatic Speech Recognition

Speech recognition is critical for many applications, including far-field scenarios where the speaker is far from the microphone (e.g., speaking to a smart home assistant device). When speech recognition models are trained on a clean speech corpus, such as LibriSpeech [52], they generalize poorly to far-field cases with unanticipated reverberation. Due to the high expense of collecting real IRs, synthetic impulse responses are thus often used to augment speech for far-field ASR [36, 46, 69]. Here, we propose to benchmark far-field ASR systems augmented by our generated impulse responses.

We take the pretrained transformer-based ASR system from SpeechBrain [56], an open-sourced speech toolkit, as the base model. For finetuning, we augment speech in the train-clean-100 split of LibriSpeech [52] with IRs generated in different systems and finetune 60 epochs. For testing, we augment speech from a real RIR dataset [67], where IRs are recorded in real environments, e.g., home, conference rooms, auditoriums. In this way, we test the sim2real generalization for models

<sup>8</sup><https://soundspaces.org/challenge>



trained on the synthetic data. We compare the pretrained model with the ASR model finetuned on IRs generated with Pyroomacoustics and SoundSpaces 2.0 (high-quality mode).

Table 4 shows the results. As we can see, the pretrained model generalizes poorly to far-field speech with word error rate (WER) of 29.1%, compared to 2.4% WER on a clean test set lacking any reverberation. Finetuning with synthetic IRs leads to a dramatic improvement. Comparing Pyroomacoustics to SoundSpaces 2.0, our generated IRs lead to much lower WER. Our simulation generates more realistic IRs that help machine learning models generalize better to reality.

**Acoustic randomization.** In the real world, instances of a given object category need not share identical material profiles. While existing simulations do not model such nuances, in SoundSpaces 2.0 we can manipulate the materials in a more subtle way. Inspired by domain randomization techniques [71, 72] that randomize simulation parameters for better sim2real generalization, we explore if acoustic randomization offers similar benefits. Specifically, we define a set of possible acoustic materials for each object category. When rendering, a random material is picked for a category to simulate the category-level variation. In addition, to model the differences of acoustic materials, we add  $\mathcal{N}(0, 0.1)$  Gaussian noise to each coefficient. Altogether, this strategy models both the category-level and instance-level material nuances.

When we use the proposed acoustic randomization technique to generate the same amount of data for finetuning, the ASR model has even lower WER on the test set, reduced from 12.48% to 12.04%, while uniform randomization, i.e., uniformly sampling coefficients between 0 and 1, leads to a higher WER of 12.58%. This not only shows the benefit of acoustic randomization but also how SoundSpaces 2.0’s configurability facilitates research on acoustic sim2real.

## 6 Conclusion

We introduced SoundSpaces 2.0, a platform for on-the-fly geometry-based audio rendering for 3D environments. We collected real measurements in public 3D scenes to validate the simulation accuracy. We benchmarked two tasks and showed that systems trained on this simulation generalize to the real world. Notably, beyond those two tasks, our platform will support an upgrade many other tasks currently explored in the literature [12, 17, 44, 31, 53, 65]. Lastly, we are releasing a large-scale SoundSpaces-PanoIR dataset that is ready to used for research. We will make the simulation and real measurements publicly available to facilitate research on visual-acoustic learning.

## References

- [1] Embodied AI workshop challenges. <https://embodied-ai.org>.
- [2] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. In *INTERSPEECH*, 2018.
- [3] Andrew Allen and Nikunj Raghuvanshi. Aerophones in flatland: Interactive wave simulation of wind instruments. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–11, 2015.
- [4] Jont B. Allen and David A. Berkley. Image method for efficiently simulating small-room acoustics. *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, 1979.
- [5] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.
- [6] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [7] Rohith Aralikatti, Anton Ratnarajah, Zhenyu Tang, and Dinesh Manocha. Improving reverberant speech separation with synthetic room impulse responses. *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 900–906, 2021.
- [8] HE Bass, LC Sutherland, and AJ Zuckerwar. Atmospheric absorption of sound: Update. *The Journal of the Acoustical Society of America*, 88(4):2019–2021, 1990.
- [9] Stefan Bilbao, Charlotte Desvages, Michele Ducceschi, Brian Hamilton, Reginald Harrison-Harsley, Alberto Torin, and Craig Webb. Physical Modeling, Algorithms, and Sound Synthesis: The NESS Project. *MIT Press*, 2020.
- [10] Chunxiao Cao, Zhong Ren, Carl Schissler, Dinesh Manocha, and Kun Zhou. Interactive sound propagation with bidirectional path tracing. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016.
- [11] 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](http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf).
- [12] Changan Chen, Ziad Al-Halah, and Kristen Grauman. Semantic audio-visual navigation. In *CVPR*, 2021.
- [13] Changan Chen, Ruohan Gao, Paul Calamia, and Kristen Grauman. Visual acoustic matching. In *CVPR*, 2022.
- [14] 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.
- [15] Changan Chen, Sagnik Majumder, Ziad Al-Halah, Ruohan Gao, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Learning to set waypoints for audio-visual navigation. In *ICLR*, 2021.
- [16] Changan Chen, Wei Sun, David Harwath, and Kristen Grauman. Learning audio-visual dereverberation. *arXiv:2106.07732*, 2021.
- [17] Victoria Dean, Shubham Tulsiani, and Abhinav Gupta. See, hear, explore: Curiosity via audio-visual association. In *NeurIPS*, 2020.
- [18] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. In *SIGGRAPH*, 2018.
- [19] Thomas Funkhouser, Ingrid Carlbom, Gary Elko, Gopal Pingali, Mohan Sondhi, and Jim West. A beam tracing approach to acoustic modeling for interactive virtual environments. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 1998.
- [20] Chuang Gan\*, Yi Gu\*, Siyuan Zhou, Jeremy Schwartz, Seth Alter, James Traer, Dan Gutfreund, Joshua B. Tenenbaum, Josh McDermott\*, and Antonio Torralba\*. Finding fallen objects via asynchronous audio-visual integration. In *CVPR*, 2022.
- [21] Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, Kuno Kim, Elias Wang, Damian Mrowca, Michael Lingelbach, Aidan Curtis, Kevin T. Feiglis, Daniel M. Bear, Dan Gutfreund, David D. Cox, James J. DiCarlo, Josh H. McDermott, Joshua B. Tenenbaum, and Daniel L. K. Yamins. Threedworld: A platform for interactive multi-modal physical simulation. In *NeurIPS Track on Datasets and Benchmarks*, 2021.
- [22] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *ICRA*, 2020.

- [23] Ruohan Gao, Changan Chen, Ziad Al-Halah, Carl Schissler, and Kristen Grauman. Visualechoes: Spatial image representation learning through echolocation. In *ECCV*, 2020.
- [24] Iliyan Georgiev. Implementing vertex connection and merging. *Technical Report. Saarland University*, 2012.
- [25] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abriham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3, 000 hours of egocentric video. In *CVPR*, 2022.
- [26] Pierre-Amaury Grumiaux, Srdan Kitic, Laurent Girin, , and Alexandre Guérin. A survey of sound source localization with deep learning methods. *CoRR abs/2109.03465 (2021)*, 2021.
- [27] Jen-Cheng Hou, Syu-Siang Wang, Ying-Hui Lai, Yu Tsao, Hsiu-Wen Chang, and Hsin-Min Wang. Audio-visual speech enhancement using multimodal deep convolutional neural networks. In *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2017.
- [28] Di Hu, Rui Qian, Minyue Jiang, Xiao Tan, Shilei Wen, Errui Ding, Weiyao Lin, and Dejing Dou. Discriminative sounding objects localization via self-supervised audiovisual matching. In *NeurIPS*, 2020.
- [29] ISO. ISO 3382, Acoustics—Measurement of room acoustic parameters. *International Standards Organization*, (3382), 2012.
- [30] Unnat Jain, Luca Weihs, Eric Kolve, Mohammad Rastegari, Svetlana Lazebnik, Ali Farhadi, Alexander G. Schwing, and Aniruddha Kembhavi. Two body problem: Collaborative visual task completion. In *CVPR*, 2019. first two authors contributed equally.
- [31] 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.
- [32] Shulei Ji, Jing Luo, , and Xinyu Yang. A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions. *arXiv preprint arXiv:2011.06801*, 2020.
- [33] Mohit Bansal Jialu Li, Hao Tan. Envedit: Environment editing for vision-and-language navigation. In *CVPR*, 2022.
- [34] III Julius O. Smith. Physical modeling using digital waveguides. *Computer Music Journal Vol. 16, No. 4 (Winter, 1992)*, pp. 74-91 (18 pages), 1992.
- [35] Abhishek Kadian, Joanne Truong, Aaron Gokaslan, Alexander Clegg, Erik Wijmans, Stefan Lee, Manolis Savva, Sonia Chernova, and Dhruv Batra. Sim2real predictivity: Does evaluation in simulation predict real-world performance? In *RA-L*, 2020.
- [36] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L. Seltzer, and Sanjeev Khudanpur. A study on data augmentation of reverberant speech for robust speech recognition. In *ICASSP*, 2017.
- [37] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*, 2017.
- [38] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.
- [39] Kristinn Kristinsson and Guy Albert Dumont. System identification and control using genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(5), 1992.
- [40] Asbjørn Krokstad, S Strom, and Svein Sørsdal. Calculating the acoustical room response by the use of a ray tracing technique. *Journal of Sound and Vibration* 8, 1 (1968), 118–125., 1968.
- [41] 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.
- [42] Eric P Lafortune and Yves D Willems. Using the modified phong reflectance model for physically based rendering. 1994.
- [43] Shiguang Liu and Dinesh Manocha. Sound synthesis, propagation, and rendering: A survey. *arxiv*, 2020.

- [44] 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.
- [45] Sagnik Majumder, Ziad Al-Halah, and Kristen Grauman. Move2hear: Active audio-visual source separation. In *ICCV*, 2021.
- [46] Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood, and Imran Makhdoom. Automatic speech recognition: a survey. 2021.
- [47] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. An overview of deep-learning-based audio-visual speech enhancement and separation. In *arXiv*, 2020.
- [48] Pedro Morgado, Yi Li, and Nuno Vasconcelos. Learning representations from audio-visual spatial alignment. In *NeurIPS*, 2020.
- [49] D.T. Murphy, Antti Kelloniemi, Jack Mullen, and Simon Shelley. Acoustic modeling using the digital waveguide mesh. *Signal Processing Magazine, IEEE*, 24:55 – 66, 04 2007.
- [50] Christian Müller-Tomfelde. Time-varying filter in non-uniform block convolution. In *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-01)*, 2001.
- [51] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
- [52] 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.
- [53] 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.
- [54] 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.
- [55] Ram Ramrakhya, Eric Undersander, Dhruv Batra, and Abhishek Das. Habitat-web: Learning embodied object-search strategies from human demonstrations at scale. In *CVPR*, 2022.
- [56] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. SpeechBrain: A general-purpose speech toolkit, 2021. arXiv:2106.04624.
- [57] Lauri Savioja and U Peter Svensson. Overview of geometrical room acoustic modeling techniques. *The Journal of the Acoustical Society of America*, 138(2):708–730, 2015.
- [58] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *ICCV*, 2019.
- [59] Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. Pyroomacoustics: A python package for audio room simulations and array processing algorithms. *arXiv*, 2017.
- [60] Carl Schissler, Christian Loftin, and Dinesh Manocha. Acoustic classification and optimization for multi-modal rendering of real-world scenes. *IEEE transactions on visualization and computer graphics*, 24(3):1246–1259, 2017.
- [61] Carl Schissler and Dinesh Manocha. Adaptive impulse response modeling for interactive sound propagation. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 71–78, 2016.
- [62] Carl Schissler, Ravish Mehra, and Dinesh Manocha. High-order diffraction and diffuse reflections for interactive sound propagation in large environments. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 39., 2014.
- [63] Carl Schissler, Gregor Mückl, and Paul Calamia. Fast diffraction pathfinding for dynamic sound propagation. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021.
- [64] Carl Schissler, Peter Stirling, and Ravish Mehra. Efficient construction of the spatial room impulse response. In *2017 IEEE Virtual Reality (VR)*, pages 122–130. IEEE, 2017.
- [65] Nikhil Singh, Jeff Mentch, Jerry Ng, Matthew Beveridge, and Iddo Drori. Image2reverb: Cross-modal reverb impulse response synthesis. In *ICCV*, 2021.

- [66] 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.
- [67] Migor Szoke, Miroslav Skacel, Ladislav Mosner, Jakub Paliesek, and Jan "Honza" Cernocky. Building and evaluation of a real room impulse response dataset. *arXiv*, 2018.
- [68] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- [69] Zhenyu Tang, Rohith Aralikatti, Anton Ratnarajah, and Dinesh Manocha. Gwa: A large high-quality acoustic dataset for audio processing. In *Advances in Neural Information Processing Systems*, 2020.
- [70] Zhenyu Tang, Nicholas J Bryan, Dingzeyu Li, Timothy R Langlois, and Dinesh Manocha. Scene-aware audio rendering via deep acoustic analysis. *IEEE Transactions on Visualization and Computer Graphics* (2020), 2020.
- [71] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IROS*, 2017.
- [72] Julianio Vacaro, Guilherme Marques, Bruna Oliveira, Gabriel Paz, Thomas Paula, Wagston Staehler, and David Murphy. Sim-to-real in reinforcement learning for everyone. *LARS-SBR-WRE*, 2019.
- [73] Michael Vorländer. *Auralization*. Germany: Springer International Publishing, 2020.
- [74] Tomi Westerlund Wenshuai Zhao, Jorge Peña Queralta. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. *arxiv*, 2020.
- [75] 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.
- [76] 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.
- [77] 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.
- [78] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.
- [79] Hang Zhou, Ziwei Liu, Xudong Xu, Ping Luo, and Xiaogang Wang. Vision-infused deep audio inpainting. In *ICCV*, 2019.
- [80] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2020.

## 7 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 limitations of this work.

### 7.1 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.

### 7.2 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 [42], where the Phong exponent is determined from the material scattering coefficient. Diffraction of rays occurs probabilistically when they hit specially-constructed edge diffraction geometry [63]. 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 from source to listener is formed, the energy for that path is calculated using multiple importance sampling [24]. 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 ( $\leq 2$  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 [41]. 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) [77]. The spatial fidelity of the direct sound is preserved by barycentric interpolation of the original HRTF data, rather than using the ambisonic HRTF.

### 7.3 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.

### 7.4 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.

### 7.5 Details of PanoIR Dataset

In this dataset, we provide rendered panoramic images (RGB/Depth) and IRs for Matterport3D [11], Gibson [76] and HM3D [54] datasets. Each panoramic image is stitched from 18 images with 20 field of view (FoV) each, resulting in a size of  $1152 \times 384$ . 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  $(\theta, d)$ , where  $\theta$  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 can be downloaded at <http://dl.fbaipublicfiles.com/SoundSpaces/PanoIR.zip>.

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.

### 7.6 RT60 Comparison with Real IRs

Fig. 5 shows the RT60 comparison between real measurements and simulations in the Replica apartment [66] 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.



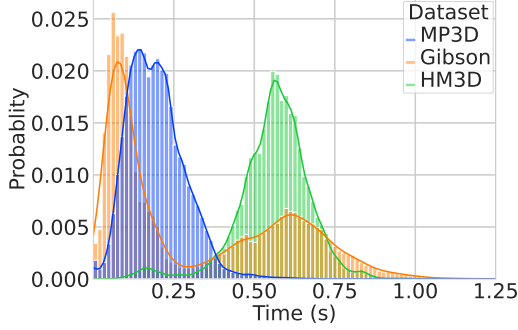


Figure 4: RT60 distribution in the PanoIR dataset for different scene datasets.

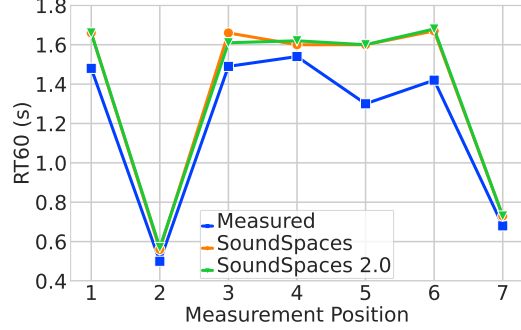


Figure 5: RT60 comparison with real measurements.

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

### 7.7 Training and Implementation Details of the Navigation Benchmark

In this continuous audio-visual navigation benchmark, we use the AV-Nav agent [14] with the decentralized distributed proximal policy optimization (DD-PPO) [75]. We train the navigation policy for 80 million steps on the same AudioGoal navigation dataset [14] 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.

### 7.8 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 [52] (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 [52] convolved with real IRs from BUT ReverbDB [67] for sim2real evaluation. For finetuning on Pyroomacoustics [59], we also generate 28539 IRs by varying the room dimension configurations.

### 7.9 Discussion on Limitations

Our simulation platform supports audio rendering for arbitrary environments. However, for the rendered audio to be highly realistic, we require the scene meshes to have high quality, i.e., no large open holes on the mesh, otherwise the rays will leak from the holes, resulting in inaccurate simulation. For example, as mentioned in Sec. 7.5, Matteport3D has lower RT60s compared to other datasets on average due to some of its broken mesh environments.