

# ANAVI: Audio Noise Awareness by Visual Interaction

## Supplementary Material

### 1 A Real-world Experiments

#### 2 A.1 Real-world Audio Data Collection

3 **Setup** We compare the model’s prediction with how loud the robot would be in the real world in  
4 Section 4. We describe how we collected the data and provide real-world acoustic measurements in  
5 Table 1. In this experiment, we fix the robot at a location and vary the listener locations.

6 We want accessible commodity hardware to ensure easy replicability for researchers so they can use  
7 existing resources. We also want to ensure that an easy setup on a mobile robot for future research.  
8 To this end, we use mobile phones to capture images, record audio and measure distances.

9 In ideal settings, we would move the robot to multiple different rooms/apartments and then record  
10 the audio. However, practically, robots can be hard to move across apartments and this limits data  
11 collection. Thus, an easier method to gather audio responses is to record the robot sound onto a  
12 device (e.g. phone or laptop). Then instead of moving the robot to other locations, we can just take  
13 our device and play back the sound. An added benefit of using recorded audio is that we reduce the  
14 variability of the source sound across acoustic measurements for the listener at different locations.

15 Here we are the steps that we followed after we had recorded our the Stretch and Unitree audio onto  
16 our laptop. Our laptop acts as a proxy for the two robots in our experiments, and we place it at the  
17 location where we want to pretend the robot is at.

- 18 1. Decide an origin (robot location), and take a panorama picture. We select indoor locations like  
19 bedrooms, living area, dining area, open and closed working spaces, corridors, stairs, auditori-  
20 ums and more.
- 21 2. Place speaker device with the recorded robot audio at the origin. We used our laptop and set the  
22 volume to its maximum. The speaker volume is fixed across all acoustic measurements.
- 23 3. Measure distance and direction for a listener location. We used the ‘Measure’ app to record the  
24 relative distance between the sound source and the listener location.
- 25 4. Place a microphone for recording at the listener location. We used an iPhone for recording.
- 26 5. Start the recording on the device and walk back to the origin to play the robot action sound.  
27 We note that the walk to the origin and back needs to be trimmed from the audio file so that it  
28 only contains the main audio. We thus found it helpful to first say ‘Action’, then play the audio  
29 file for robot action (at max volume) and then say, ‘Cut’. After ‘Cut’ we walked back to the  
30 microphone (iPhone in our case) and stopped the recording. Saying ‘Action’ and ‘Cut’ helps  
31 to trim the audio faster by looking at the audio waveform to trim and only saving the audio in  
32 between. Researchers working in pairs can skip this complication and just use hand signals to  
33 denote starting and stopping the audio recording and speaker sound.
- 34 6. Extract the max decibels from the recorded audio files.

35 To estimate the max decibels at the origin (sound source), we need to know the volume gain adjust-  
36 ment on the original waveform. We guesstimate it to be a factor of 10, that is 20 dB. Note, the model  
37 is likely to have sim-to-real gap and our estimated value also accounts for the systematic error in  
38 the model’s predictions. To obtain the model’s prediction for the max decibel level by each robot’s  
39 action, we use a linear approximation, that is, multiplying the normalized max dB output with the  
40 estimated max decibels at the origin (sound source). We report the error between this value obtained  
41 with linear approximation and that from the recorded audio files in the main paper.



Figure 1: Panorama taken from the sound source location. Note the directions west, north are in the room, facing east is a wall and the south direction leads out of the room into the hallway.

Table 1: Real Audio for a Bedroom in a Single Family House.

Distance-Direction	Line-of-sight	Outside the room	Stretch Real dB	Unitree Go2 Real dB
1m-west	yes	no	54.2	69.7
1m-south	yes	yes	52.1	67.1
1m-south-1m-east	no	yes	49.7	66.1
2m-north	yes	no	51.7	67.3
2m-south	yes	yes	50.5	66.7
2m-north-2m-west	no	no	45.3	61.1
3m-west	yes	no	48.6	65.4
3m-south	yes	yes	48.2	65.4

**Discussion** Table 1 shows the real audio recorded for Stretch at the fastest forward velocity and Unitree Go2 in running mode using our set-up. We keep the speaker at the origin corresponding to Fig. 1 and vary our listener location. As we can see in the figure, moving west or north stays in the room while moving south or east corresponds to moving outside the room. We first compare the dB between “1m-west” and “1m-south” and observe how moving the sensor out the room decreases dB (as expected). We see this difference consistently although with smaller values as the distance from the source increases. This highlights how architectural geometries can have non-trivial impacts on audio (as expected). Additionally, we observe how line-of-sight between the source and listener affects the dB values. In case of obstacles like the bed or wall, the dB values decrease faster as compared to longer distances with line-of-sight (e.g. “1m-south-1m-east” is lower than “2m-north”).

## A.2 Zero-Shot Sim-to-Real

We collect real-world panorama of indoor environments to visualize the model’s predictions. We focus on sim-to-real performance of the audio prediction, as once accurately estimated, these values can be weighted against distance for audio-informed planners.

**Setup** To collect real world panoramas, we requested graduate students to contribute these panoramic images, by capturing their surrounding indoor environment for acoustic profiling. Contributors used their mobile phones at zoom level 1x and took a single panoramic image which we then resized to  $256 \times 1024$ . The images are then fed into a neural network that predicts the maximum decibel level at a given distance and direction from the robot. The ANP neural network is trained on a dataset of simulated Matterport renderings, and we qualitatively evaluate its performance on a dataset of real-world panoramas. Below we use Figure 2 to explain our visualizations.

The first row shows the real-world panorama. The leftmost and rightmost edges correspond to  $45^\circ$ , and we change angles in clockwise direction from left-to-right. The reason for non-increasing order of angles on x-axis is that panorama’s are taken left-to-right, that is, clockwise, whereas angles are measured anti-clockwise. Note that the cardinal directions indicated on the plots does not imply that the panorama starting direction and are only for illustration purposes.



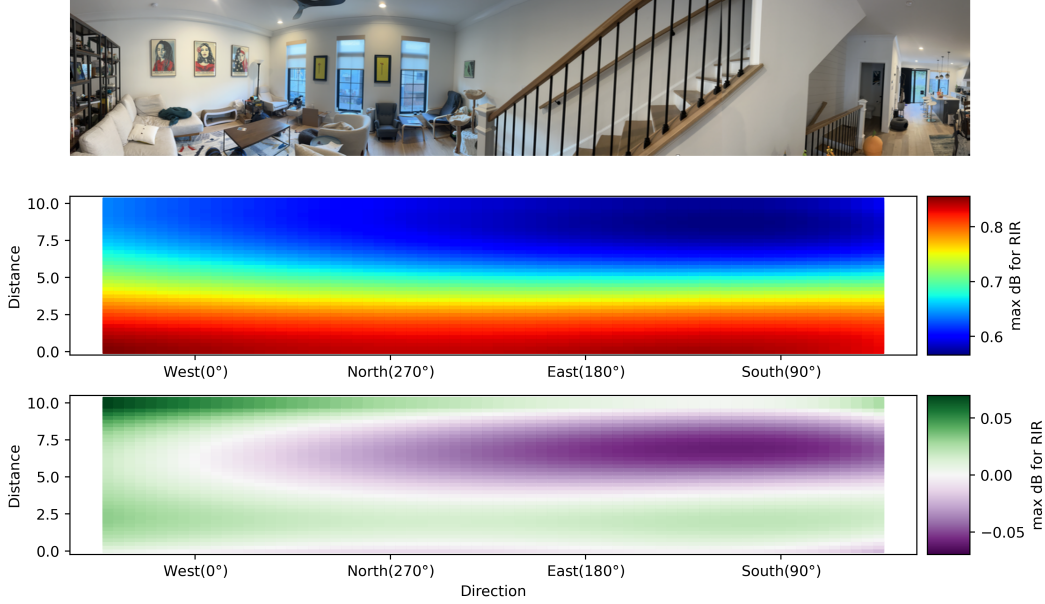


Figure 2: **Setup for ANP Predictions on Real World Images.** This image visualizes ANP model predictions similar to Figure 7 from the main paper. The top row depicts a real-world panoramic image. The second row plots the maximum decibels of room impulse response  $\hat{y}_{ANP}$  (max dB RIR). Note again that max dB RIR is normalized between 0 to 1. The third row plots the difference between the model’s prediction and the linear regression prediction, i.e.  $\hat{y}_{ANP} - \hat{y}_{linreg}$ .

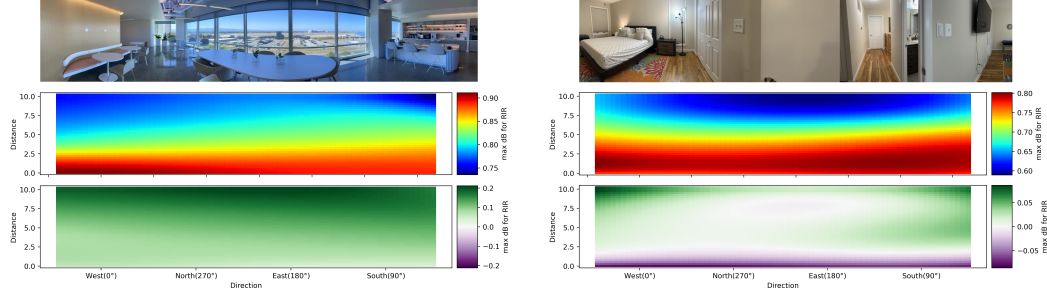
68 The second row shows heatmap plot with the model’s prediction for different distance and direction  
69 values. The direction is shown on x-axis, covering 360°, and the distances on y-axis range from 0 to  
70 10 meters. Note that the direction is aligned with the visual image, i.e. it starts from -45° to 0, then  
71 reduces 270°, 180° 90°, and finally to 45°.

72 In the third row, we plot the difference from the distance based baseline. Since the distance based  
73 baseline doesn’t depend on the image, we visualize the variation in our model’s prediction from  
74 the baseline. Here the green indicates where the model predicts significantly higher normalized  
75 max dB values than distance-based linear regression model, and purple indicates significantly lower  
76 normalized max dB values than naive baseline. Note that the dB values are normalized from 0 to 1.

77 **Discussion** We see that the model mostly predicts that the max dB values monotonically decrease  
78 with distance, though it may vary with direction. This is an important initial sign of life, as im-  
79 age features are very high-dimensional as compared to a scalar number used for relative distance.  
80 Nevertheless, the model largely learns to attend to the distance as input.

81 *Does our model understand acoustic features beyond distance?* In Figure 3, we show cherry-picked  
82 examples of (3a) an open office area, (3b) a bedroom in a single family house. In Fig. 3a, we  
83 observe that the model predicts high dB values at larger distances in open spaces, as in north-east  
84 direction. In terms of the difference with the baseline, we observe that the model predicts higher  
85 sound intensity than the baseline at larger distances. In Fig. 3b, we observe that the model predicts  
86 low values at larger distance for some areas where there is a wall. Walls acts as an obstacle for  
87 sound intensity (see east direction). Additionally, we observe that the model predicts higher values  
88 at larger distances in open spaces, within the room (see west and north directions), than in the outside  
89 corridor (see south direction). In the second subplot, we note that our model predicts higher max dB  
90 RIR than the baseline at larger distances, and lower values near the sound source.

91 Through the qualitative real-world data evaluation, we note several sim-to-real gaps. First, the real  
92 images are higher quality than the Habitat environment’s Matterport3D renderings. This includes  
93 differences in how visually environment attributes as inferred, such as depth, material texture, and



(a) **Open office area.** High dB values predicted at larger distances for open spaces (see east), and at shorter distances in the corners, near walls (see west and south-west). ANP predicts higher sound intensity than the baseline, especially at larger distances, as shown in green.

(b) **Bedroom in a single family house** Low dB values predicts near wall as an obstacle (see east) and outside the room (see south). High dB values at larger distances in open spaces (see west). ANP predicts higher values than the baseline at larger distances (green), and lower values near source (purple).

Figure 3: We show cherry-picked examples of model’s prediction in (a) an open office area, and (b) a bedroom in a single family house.

architectural geometry. Second, lighting changes and variations are more drastic in real-world than the static Matterport renderings in simulation. This has implications on acoustic noise prediction models deployed on the home robots, and we need to improve the robustness to lighting variations for the same scene. Third, the height perspective of our images and the relative angle to the ground differed between contributors, while this was a fixed constant in simulation. Differences in camera height, pan and tilt requires slightly different interpretation of the distances from visual features.

We observe that the current model performs poorly on these diverse set of real-world panoramic images. The first row (Figures 4a and 4b) shows two panoramas of corridors where we expect corridors to have high predicted db regardless of sensor distance while the corresponding wall area have small values when the distance is large (as then the sensor would be behind the wall). Instead, we did not see any noticeable correlation to the corridors. Upon closer inspection on predictions, the model seems to struggle with identifying walls and their impact on sound. We expect high dB predictions in rooms when the distance falls in the room, but then a large drop-off to low dB prediction outside rooms. However in Figures 4c, 4e and 4g, we notice no drop-off in predicted values.

Our results show that zero-shot generalization to real panoramas is not easy. It is unclear if this issue lies in simulation inaccuracies, an inadequate ANP model, or the distribution shift to real-world panoramas. Future work should address these sim-to-real differences to enable using an ANP model effectively in real environments. In future, we hope to bridge the sim2real gap with (1) effective domain randomization for camera poses, lighting and robot’s viewing angles, (2) improving model training with auxiliary losses, and (3) fine-tuning model with real-world measurements for acoustics, distances and visuals, collected from diverse indoor environments.

## B Details of Training with SoundSpaces 2.0

### B.1 Training Data Generation

We generate training data for ANP in simulation using SoundSpaces 2.0. Here we provide additional details for the data generation process:

To ensure uniform sampling across the map, we first divide the map into grids to get 100 points. These points serve as circle centers to sample a suitable navigable point within 20 meters radius for the robot’s location as the source. We then use the robot’s location as the circle center and sample a listener location within 10 meters radius.

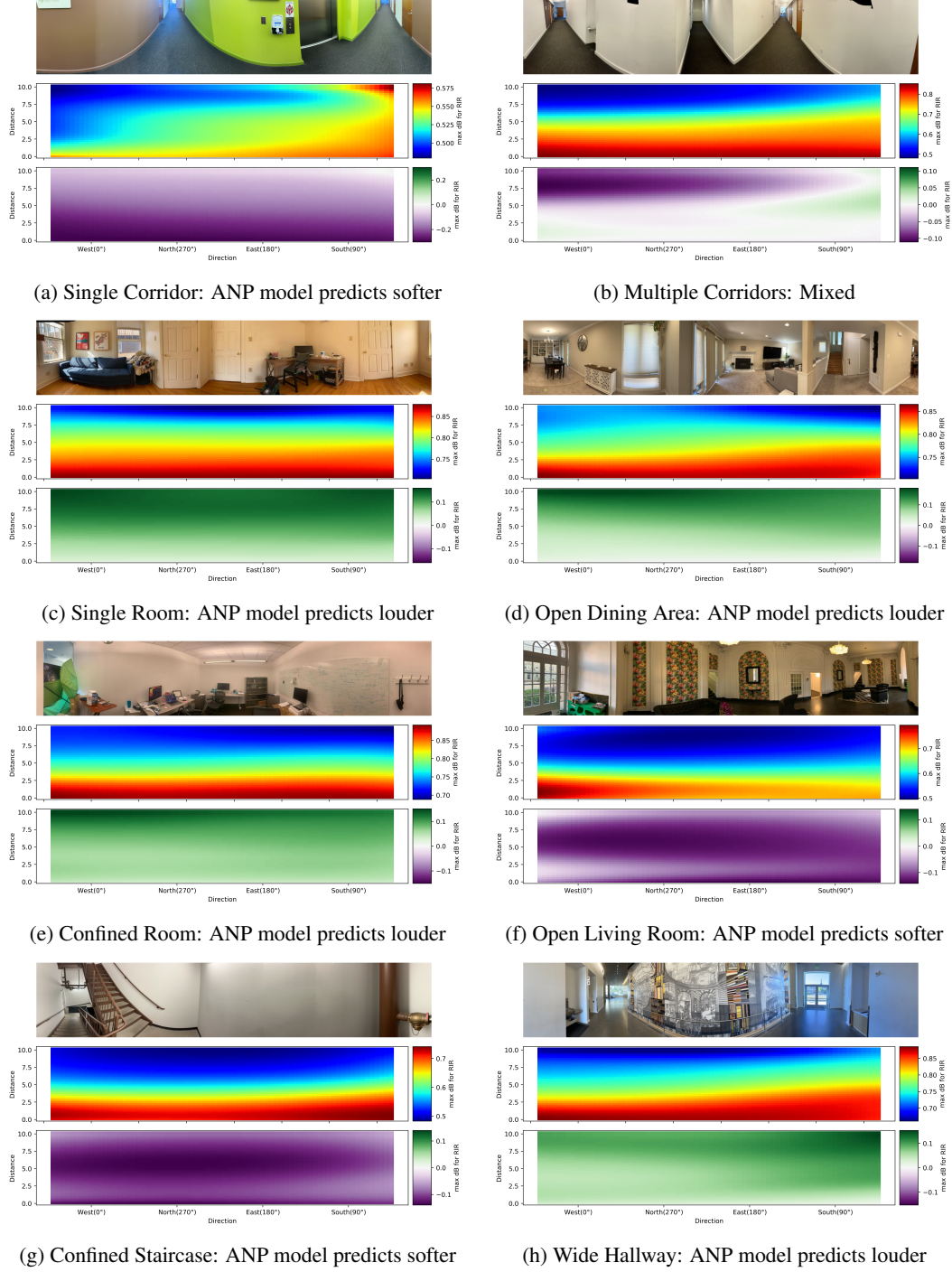


Figure 4: We see a severe sim-to-real gap when applying the ANP model to real world panoramas. For each plot, the top row shows a real world panorama, second row the ANP predictions depending on the distance and angle to the sensor location, and the third row the difference compared a basic linear regression estimate (green = ANP predicts louder). Unfortunately, the model is unable to consistently capture meaningful visual features like walls or corridors that would effect dB.

124 To capture the visual panorama for the robot's source location, we sample RGB camera observations  
 125 4 times with rotations of 90 degrees at  $256 \times 256$  resolution and 90 degrees field-of-view.

## 126 B.2 Obtaining max decibels target from impulse response

127 SoundScapes 2.0 gives the impulse response (how the audio would echo given a single short burst),  
 128 which we must then convert into a max decibel level target given the robot’s audio. From a high  
 129 level, we do this by first convolving the impulse response with the robot audio and then computing  
 130 the corresponding decibel level of the loudest part of the convolved audio. Concretely, we do the  
 131 following steps:

$$w(t) = \text{SimulateIR}(\text{from} = p_{\text{source}}, \text{at} = p_{\text{receiver}}) \quad (1)$$

$$W(f) = \text{Fast Fourier Transform}(w(t)) \quad (2)$$

$$I(f) = (W(f)^2)/(\rho * c) \quad (3)$$

$$I_{\text{max}} = \max I(f) \quad (4)$$

$$I_{\text{max}} = \text{clip}(I_{\text{max}}, \min = 10^{-12}, \max = 10^{0.8}) \quad (5)$$

$$dB_{\text{max}} = 10 \log_{10} I_{\text{max}} + 120 \quad (6)$$

$$y = dB_{\text{max}}/128 \quad (7)$$

132 Here  $w(t)$  is the time-domain waveform generated at the receiver’s location,  $W(f)$  is the frequency  
 133 domain waveform,  $I(f)$  is the frequency domain sound intensity through air computed by root mean  
 134 square. The  $\rho$  is air density and  $c$  is speed of sound in air. We use the maximum sound intensity  
 135 and convert it to decibels. To ensure that the decibel values are normalized for training, we assume  
 136 the highest value of decibel as 128 and compute target labels  $y$ . Since the faintest sound human ears  
 137 can hear is considered  $I_o = 10^{-12} \text{W/m}^2$ , we convert the max sound intensity into to decibels by  
 138  $dB_{\text{max}} = 10 \log_{10}(I_{\text{max}}/I_o) = 10 \log_{10} I_{\text{max}} + 120$ .