Supplementary Material for Omnidirectional Information Gathering for Knowledge Transfer-based Audio-Visual Navigation

Jinyu Chen[1], Wenguan Wang[2*], Si Liu[1*], Hongsheng Li[3], Yi Yang[2]

[1] Institute of Artificial Intelligence, Beihang University [2] ReLER, CCAI, Zhejiang University [3] The Chinese University of Hong Kong

In this supplementary material, we provide details about i) implementation in §1, ii) model ensemble in §3, iii) environment & metric in §4.

1. Implementation Details

AudioGoal model. Following [4], we set output dimensions of CNNs for top-down map G_t, audio intensity map A_t and binaural S_t as 512. The CNNs for G_t and A_t have three convolution layers each, with kernel sizes of $(8, 4, 3)$, strides of $(4, 2, 1)$ respectively. The CNN for S_t has three convolution layers with kernel sizes of $(5, 3, 3)$ and strides of $(2, 1, 1)$. Each layer's output channel size is $(32, 64, 32)$, which is the same for all CNNs. We select ReLU as the activation function. The GRU is of 1 layer with 512-dimension hidden units. The size of the action map M_t is 9×9.

PointGoal model. We initialize the PointGoal policy by the weight provided in [7], which is trained in Matterport3D and iGibson environment. We modified two parts of the model: we i) replace the encoder for the previous action with the CNN encoding G_t, and the output dimension keeps the same, which is 32; ii) replace the FC layer predicting low-level actions with the one predicting M_t with the shape of 9×9. Then the model is finetuned under the PointGoal task on the Soundspaces training split by PPO [6], and the depth encoder is fixed. The hyperparameters of PPO [6] are the same with [4].

2. Model for Comparison Details

We compare our method with following existing methods and model baselines in Table 1 of the main paper:

- **Random Agent**: an agent randomly selects action and stops when it reaches the audio goal.
- **Direction Follower [4]**: a hierarchical model with a model predicting the audio goal direction and one model deciding the when the agent stops.
- **Frontier Waypoints [2, 4]**: a hierarchical model that intersects the predicted direction with the frontiers of the explored area and selects that point as the next waypoint.
- **Supervised Waypoints [1, 4]**: a hierarchical model that uses the RGB frame and audio input to predict waypoints in its field of view (FoV) with supervised learning.
- **Gan et al. [5]**: AudioGoal agent that predicts the audio goal location from audio alone and then navigates with an path planner on an occupancy map. The accuracy on Soundspaces is provided by [4].
- **AV-Nav [3]**: An end-to-end RL agent predicting low-level actions via visual-audio observations.
- **AV-WaN [4]**: An end-to-end RL agent with geometric and acoustic maps predicts the intermediate goal and generates trajectory with an analytical path planner.

3. Model Ensemble Details

Table 1 summarizes the properties of the fused models in §4.4 of the main paper. We elaborate on the implementation details of the strategies as follows:

- `Re.Loss`: spectrogram reconstruction loss. We add a decoder at the 3rd CNN layer of S_t's encoder. The decoder is one layer CNN with kernel size $(1, 1)$, stride $(1, 1)$, and output channel 2. We then interpolate the decoder's output to the same shape as the input spectrogram to compute the L1 loss.
- `Spec.Aug`: spectrogram data augmentation, which is composed of the time mask and the frequency mask. We start the data augmentation at 1M steps when the time and frequency mask's size is 1, and then the mask size increases by 1 every 1M steps.
- `Sample.Aug`: the data augmentation for training samples. We double the training set by sampling more trajectories from the environment and changing the target sound of episodes from the origin training set.
- `MFCC`: an extra CNN which encodes the binaural MFCC (Mel-frequency cepstral coefficients). The CNN is of the same structure as the encoder for S_t. The output of this CNN is concatenated with the other three encoders' as the input of GRU.

4. Environment & Metric

SoundSpaces [3]. SoundSpaces provides room impulse response (RIR) to simulate realistic sound that comes from

*Corresponding author: *Wenguan Wang, Si Liu*.

name	*Strategies*				*Matterport3D Unheard*			
	Re.Loss	Spec.Aug	Sample.Aug	MFCC	SPL↑	SR↑	SNA↑	SoftSPL↑
A					40.9	56.7	30.6	46.3
B	✓	✓			43.5	60.4	31.9	47.7
C		✓	✓		42.4	57.5	32.1	48.2
D		✓		✓	43.1	60.1	30.2	47.5

Table 1: Summary of fused models. A is the baseline model from [4]. See details in §3.

the target at each position in the scene. The RIR's spatial resolution of Replica is 0.5m, and Matterport3D is 1m. SoundSpaces maintains a navigability grid graph of the environment, which is of the same resolution as RIR. The agent can only move from the current node to a neighbour one on the graph. So the action space of Soundspaces is $\mathcal{A}$ ={*MoveForward*, *TurnLeft*, *TurnRight*, *Stop*.}

Metric. The definition of the metrics we used in the paper are as follows:

- **SR:** Success Rate, the proportion of success episodes, i.e., where the agent takes the *Stop* action exactly at the goal location within 500 action steps.
- **SPL:** Success weighted by Path Length, where success is weighted by path efficiency. Let S_i be a binary indicator of success, p_i be the length of agent's path, l_i be the shortest path:

$$\text{SPL} = \frac{1}{n}\sum_{i=1}^{N}\frac{S_i \cdot l_i}{\max(l_i, p_i)} \tag{1}$$

- **SoftSPL:** the patched SPL, where the binary success is replaced by progress toward the goal. let d_i be the initial distance to the goal and e_i be the distance to the goal at the end of episodes:

$$\text{SoftSPL} = \frac{1}{n}\sum_{i=1}^{N}(1-\frac{e_i}{d_i})\frac{l_i}{\max(l_i, p_i)} \tag{2}$$

- **SNA:** Success weighted by the Number of Actions. Let S_i be a binary indicator of success, p_i^a be the number of agent's actions and l_i^a be the number of actions taken for the shortest path:

$$\text{SNA} = \frac{1}{n}\sum_{i=1}^{N}\frac{S_i \cdot l_i^a}{\max(l_i^a, p_i^a)} \tag{3}$$

References

[1] Somil Bansal, Varun Tolani, Saurabh Gupta, Jitendra Malik, and Claire Tomlin. Combining optimal control and learning for visual navigation in novel environments. In *CoRL*, 2020. 1

[2] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. In *ICLR*, 2020. 1

[3] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *ECCV*, 2020. 1

[4] Changan Chen, Sagnik Majumder, Al-Halah Ziad, Ruohan Gao, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Learning to set waypoints for audio-visual navigation. In *ICLR*, 2021. 1, 2

[5] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B. Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *ICRA*, 2020. 1

[6] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1

[7] 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*, 2019. 1