Catch Me If You Hear Me: Audio-Visual Navigation in Complex Unmapped Environments with Moving Sounds

4 authors, including:

Daniel Honerkamp
University of Freiburg
12 PUBLICATIONS   76 CITATIONS

Tim Welschehold
University of Freiburg
36 PUBLICATIONS   500 CITATIONS

Abhinav Valada
University of Freiburg
129 PUBLICATIONS   3,234 CITATIONS

Catch Me If You Hear Me: Audio-Visual Navigation in Complex Unmapped Environments with Moving Sounds

Abdelrahman Younes*   Daniel Honerkamp*   Tim Welschehold   Abhinav Valada
University of Freiburg
{younesa, honerkamp, twelsche, valada}@cs.uni-freiburg.de

Abstract

Audio-visual navigation combines sight and hearing to navigate to a sound-emitting source in an unmapped environment. While recent approaches have demonstrated the benefits of audio input to detect and find the goal, they focus on clean and static sound sources and struggle to generalize to unheard sounds. In this work, we propose the novel dynamic audio-visual navigation benchmark which requires to catch a moving sound source in an environment with noisy and distracting sounds. We introduce a reinforcement learning approach that learns a robust navigation policy for these complex settings. To achieve this, we propose an architecture that fuses audio-visual information in the spatial feature space to learn correlations of geometric information inherent in both local maps and audio signals. We demonstrate that our approach consistently outperforms the current state-of-the-art by a large margin across all tasks of moving sounds, unheard sounds, and noisy environments, on two challenging 3D scanned real-world environments, namely Matterport3D and Replica. The benchmark is available at http://dav-nav.cs.uni-freiburg.de.

1. Introduction

Humans are able to very efficiently combine their senses of hearing and sight in order to navigate unknown environments. While navigation in such environments has been an important focus of embodied AI [23, 48], existing work on navigation overwhelmingly relies on sensors such as vision and LiDAR, leaving out other core senses used by humans. Sound is a particularly unique modality as it reveals information beyond the visible walls and obstacles [44]. In particular, it has been shown to provide blind people spatial navigation capability comparable to sighted people [19].

Recent work has demonstrated the value of this signal for embodied agents in a variety of tasks. This includes audio-visual navigation, in which the agent is required to navigate

Figure 1. We introduce a novel dynamic audio-visual navigation benchmark (left). The paths of the agent and the sound source are shown in blue and red respectively, with initial poses marked as squares. The green line represents the optimal behavior to catch the moving target. Secondly, we carefully design complex audio scenarios (right). The agent needs to navigate towards the ringing phone while being confronted with a second sound source (a crying baby), and various distractor sounds such as a piano.

to the location of a sound-emitting source using audio and visual signals [9, 21], semantic audio-visual navigation [8] with coherent room and sound semantics, active perception tasks such as active audio-visual source separation [30] and audio-visual dereverberation [11], curiosity-based exploration via audio-visual association [46] as well as tasks explicitly focusing on the geometric information contained in audio such as audio-visual floor plan reconstruction [5, 35].

Navigation-centered approaches have shown that agents can successfully extract information from the audio signals. However, they have mostly focused on clean and distractor-free audio settings in which the only change to the audio signal comes from changes in the agent's position. Furthermore, they have struggled to generalize to unheard sounds [9, 10]. In this work, we take the next steps towards more challenging scenarios. First, we introduce a novel dynamic audio-visual navigation benchmark with a moving sound source. This captures common scenarios such as a robot navigating to a person issuing commands or following pets or people in the house. We argue that this strongly increases the complexity of the task through two channels: on one hand, previous observations no longer capture the current state of the environment and the agent has to learn to update its memory accordingly. On the other hand, optimal behavior now requires not just to follow the sound intensity but proactively

---
*Equal contribution.

reason about the movement of the target to catch it efficiently. Secondly, we increase the difficulty of both static and moving sound tasks by designing complex audio scenarios with augmented, noisy and distracting sounds and show the benefits of training on these scenarios for generalization to unheard sounds. Fig. 1 illustrates both the moving sound task (left) and the complex audio scenarios we construct (right). Lastly, we introduce an architecture that explicitly enables the agent to spatially fuse the geometric information inherent in obstacle maps and audio signals. We show that this leads to significant gains in the generalization to unheard sounds in both clean and complex audio scenarios. We demonstrate these results in the SoundSpaces [9] extension to the Habitat simulator [38], which allows us to generate realistic binaural sound signals for the realistic 3D environments of the Replica [42] and Matterport3D [6] datasets. In combination, these contributions achieve improvements over the previous state-of-the-art by 53% and 29% in the success rate as well as an improvement in SPL by 58% and 39% on Replica and Matterport3D respectively, on the unheard and static AudioGoal benchmark tasks [9].

To summarise, the main contributions of this work are:

- We introduce the novel dynamic audio-visual navigation benchmark that substantially increases the difficulty of the audio-visual navigation task.
- We propose complex audio scenarios with noisy and distracting sounds and demonstrate the benefits of this randomization for better generalization.
- We propose a new architecture that allows to spatially fuse sound and vision, and outperforms current state-of-the-art approaches by up to 58% on unheard sounds.
- We perform exhaustive experiments in two realistic 3D environments, and across static and dynamic audio goal tasks in both clean and complex audio scenarios.
- The code and the benchmark is made publicly available at http://dav-nav.cs.uni-freiburg.de.

2. Related Work

Embodied Navigation: The ability to navigate the physical world has been a core focus of research in embodied AI. Much progress has been achieved on tasks such as Point-Goal Navigation [13, 23, 48], ObjectGoal Navigation [16, 49], Vision-Language Navigation [2, 12, 20], Active Visual Tracking [26, 51], Visual Exploration [7, 28, 33] and Embodied Question Answering [15, 47, 50]. While most common goals are assumed to be static, some work has been conducted on moving goals. Control approaches to track and intercept moving sounds have been proposed based on explicit forecasting of the target's movement, but rely on first visually locating the target [4, 52]. Work in multi-agent systems [14, 39] and social navigation [18, 25] investigate scenarios in which multiple agents influence each other, both collaboratively

and competitively. In contrast, we focus on cases in which the target is frequently outside the field of visual sensors and assume that the agent's behavior does not impact the sound source's movements.

Sound Source Localization: The localization of sound sources based on an audio signal has been explored in robotics for both static [31, 36] and dynamic sounds [17, 22, 34, 45]. Early work in combining audio and vision focused on using Gaussian Processes [24] or canonical correlation analysis [29]. Recently, deep learning based methods have been proposed to localize sounding objects in videos [3, 43]. In contrast, audio-visual navigation problems focus on potentially out-of-sight sound sources in unmapped environments.

Audio-Visual Navigation: Recent simulators and datasets have enabled the training of learning-based systems on the combination of visually realistic scenes and varied audio signals [9, 21]. In the AudioGoal task, an agent navigates to a continuously sound-emitting target source's location using audio and visual signals [9, 21]. This task has been tackled by decomposing the problem into the prediction of the sound location and a planner [21] or by an end-to-end reinforcement learning approach with either low-level actions [9] or on a higher level, combining learned waypoints and a planner [10]. However, not much focus has been put on the fusion of sound and visual observations. In contrast, we provide the agent with an architectural prior that allows it to explicitly learn to fuse spatial information from both modalities. Semantic Audio-Visual Navigation [8] introduces a related task in which the sound is periodic and semantically consistent with the scene, allowing agents to exploit semantic understanding between sound and vision to navigate towards the goal. However, these tasks assume a single static sound-emitting source. While this setup covers a large number of potential use cases, it is still a subset of the scenarios that humans navigate. We propose a novel task where the agent tracks a moving sound source using only acoustic and visual observations.

Previous work focuses either solely on clean audio scenarios [9, 21] or provides initial but limited exploration of more complex scenarios. This includes the presence of a distractor or microphone noise on the single heard AudioGoal task [10] and evaluation of the impact of distractors in related tasks such as Semantic Audio-Visual Navigation [8] and Active Audio-Visual Source Separation [30], in which the agent needs to move in an intelligent manner to separate the input target monaural sound source from distractor sounds within a fixed time limit. In contrast, we extensively evaluate the impact of both training and testing in the presence of a variety of strong audio perturbations in audio-visual navigation and demonstrate strong benefits of these scenarios for the generalization to unheard sounds.

Augmentation and Domain Randomisation: Data augmenta-

tion [41] and domain randomization [27, 37] have shown to be very beneficial in regimes of limited data, to improve generalization to unseen data and to obtain robustness against noise. Audio-specific augmentations include transformations such as time warping, frequency masking and time masking [32]. In this work, we introduce complex audio-scenarios specific to audio-visual navigation that similarly increase the variety and diversity of the training data.

Active Visual Tracking: Distractors have been used within the visual domain, learning to keep tracking objects or people of interest within the field of view [51]. In contrast we present a setting of audio-disctractors and randomizations.

3. Problem Statement

In this work we tackle the challenge of navigating to a sound emitting goal. We use the highly photo-realistic datasets Replica [42] and Matterport3D [6], which consist of 3D scans of indoor areas such as homes, offices, hotels and rooms. Replica scenes' areas vary between $9.5\,\mathrm{m}$ to $141.5\,\mathrm{m}^2$ while Matterport3D areas vary between $53.1\,\mathrm{m}$ to $2921.3\,\mathrm{m}^2$ providing the agent with diverse experience of different real-world scenes with close and far AudioGoals. We use Habitat [38] and its Audio-compatible SoundSpaces simulator [9] to train our intelligent agents on the static and dynamic AudioGoal tasks. The SoundSpaces simulator offers binaural room impulse response (BRIR) between every two possible source and receiver locations on a grid with a spatial resolution of $0.5\,\mathrm{m}$ for Replica and $1\,\mathrm{m}$ for Matterport3D dataset. The pre-calculated BRIR can be convolved with any arbitrary sound to simulate how the receiver listens to this audio at the agent's current location. We use the same 102 copyright-free sounds used in the AudioGoal benchmark [9], available under CC-BY-4.0 licence. The BRIRs have a sampling rate of $16\,000\,\mathrm{Hz}$ for Matterport3D and $44\,100\,\mathrm{Hz}$ for Replica. The spectrograms are computed as in [9, 10]: We compute a Short-Time Fourier Transform (STFT) with a window length of 512 samples and a hop length of 160. From this we take the magnitude, downsample the axes by four and compute the logarithm. Stacking the left and right audio channel then results in a spectrogram of size (65,26,2) for Matterport3D and (65,69,2) for Replica.

We formulate the problem as a reinforcement learning task where the agent learns to navigate an unknown environment to reach a potentially previously unheard sound-emitting goal. In each step, the agent receives the current observation o_t consisting of RGB image v_t and depth image d_t as well as a binaural sound b_t in the form of a spectrogram for the right and left ear. In contrast to the common Point-Goal navigation task, the agent does not receive a displacement vector to indicate the goal position. Given the current observation and the agent's previous state s_{t-1}, the agent then produces a next action a_t from it's policy $\pi(a_t|o_t, s_{t-1})$.

The agent's aim is to maximise the expected discounted return $\mathbb{E}_\pi[\sum_{t=1}^T \gamma^t r(s_{t-1}, a_t)]$, where γ is the discount factor and $r(s_{t-1}, a_t)$ is the reward. Note that the sound goal does not have an embodiment, i.e. it is not visible in the RGB-D images and cannot collide with the agent.

We build upon the AudioGoal task as introduced by [9]. Our agent starts in a random pose in an unknown environment. It then has to navigate to the sound location and execute the *Stop* action at the exact position of the sound source. The discrete action space of the habitat simulator consists of *Move Forward, Rotate Left, Rotate Right*, and *Stop*. The parametrization of our reinforcement learning agent's action space however, follows a waypoint selection approach as [10] and is described in Sec. 4.4. In order to increase the capabilities and possible uses of such an agent, we extend the task described in [9] to include moving noise-emitting targets. We further seek to improve the agent's performance in general and in particular with regards to navigating to previously unheard sounds.

4. Technical Approach

In order to address the aforementioned challenges, we introduce the novel dynamic audio-visual navigation benchmark which largely increases the demands on both the agent's memory as well as it's policy which now has to pro-actively move and catch the agent to act optimally. We then introduce complex audio scenarios for both the moving and the existing static AudioGoal navigation task [9]. To this end, we develop audio-domain specific scenarios in which the agent is confronted with both episodic and per-step randomizations, requiring to integrate and filter the sound signals over time. We further propose a new architecture that allows the agent to directly integrate the spatial and directional information from sound and vision, and strongly increases robustness and generalization to unheard sounds as we demonstrate in Sec. 5.

4.1. Dynamic Audio-Visual Navigation

We introduce the novel task of dynamic audio-visual navigation. In this task, the agent must navigate towards a moving sound-emitting source in an unmapped complex 3D environment and output *Stop* when it catches it. The agent needs to reason about the trajectory of the moving sound depending on audio and visual observations to decide the shortest path to reach it. This can be seen as a generalization of the existing task, broadening the scope of static Audio-Goal navigation to scenarios such as navigating to a person issuing commands or following a pet. The moving target means that previous observations no longer capture the current state of the environment and the agent has to learn to update its memory accordingly. Furthermore it is not longer sufficient to follow the gradient of the sound intensity, instead optimal behavior now requires to proactively reason

about the movement of the target to catch it efficiently. To the best of our knowledge this is the first approach to investigate the use of sound and vision to catch moving sounds within unexplored environments.

Motion Model: We assume a simple, goal direct behavior of the target. The sound source starts in a random pose on the map and uniformly draws a goal from the traversable grid, excluding the agent's current position. We also ensure that there exists a traversable path from its start to goal location. The sound source then follows the shortest path towards the goal and with a probability of 30% moves to the next node. This percentage ensures the sound sources moves slightly slower than the agent, ensuring that it is possible to catch the moving sound source. Note that the moving source does not have an orientation and directly moves to the following location while the agent has to take separate rotation steps to change direction. Once the moving target reaches its goal, it draws a new random goal to navigate to.

Optimal Behavior: Dynamic Success weighted by Path Length (DSPL): The Success weighted by Path Length (SPL) [1] serves as the primary metric to evaluate the navigation performance of embodied agents. However, in the case of a moving sound source, the shortest possible path depends on the a priori unknown trajectory of the sound source. Given this knowledge, the optimal policy is to move to the earliest intersection with the target's trajectory that the agent can reach before the target passes by. Hence, we introduce the Dynamic Success weighted by Path Length (DSPL) to measure how close the agent is to this oracle optimal policy. We define the DSPL as follows: where i is the current episode count, N is the total number of episodes, S_i represents whether this episode is successful or not, g_i is the shortest geodesic distance between the agent's start location and the closest position the agent could have caught the sound source at, and p_i is the length of the path taken by the agent:

$$DSPL = \frac{1}{N} \sum_{i=1}^{N} S_i \frac{g_i}{max(p_i, g_i)}. \tag{1}$$

Note that this metric represents an oracle upper bound of the possible performance, which may not be achievable without a priori knowledge of the trajectory of the sound source. An example of the task and the optimal behavior as used in the DSPL is shown in Fig. 1.

4.2. Reward

We use the same reward definition for both the static and dynamic tasks. Upon success, the agent receives a positive reward of +10. It further receives a small dense reward of +0.25 for decreasing and -0.25 for increasing the shortest path distance to the goal. For the dynamically moving sound this is calculated with respect to the current sound source's

---

Algorithm 1: Randomization Pipeline

Require: listOfSounds: training sounds excluding current episode target sound, targetAudio: current episode target sound, agentPosition, targetPosition, computeAudio: function to compute audio observation, computeSpectrogram: function to compute spectrogram, applySpecAugment: function to apply feature augmentation, listOfNodes: traversable grid locations, rnd: uniformly random choice function.

for *episode in episodes* **do**
    includeSecondSound = rnd([True, False])
    if *includeSecondSound* **then**
        secondAudio = rnd(listOfSounds)
    end
    includeDistractor = rnd([True, False])
    for *step in steps* **do**
        audio = computeAudio(targetAudio, agentPosition,targetPosition)
        if *includeSecondSound* **then**
            audio += computeAudio(secondAudio, agentPosition,targetPosition)
        end
        distractorStep = rnd([True, False])
        if *includeDistractor **and** distractorStep* **then**
            distractorAudio = rnd(listOfSounds)
            distractorPosition = rnd(listOfNodes)
            audio += computeAudio(distractorAudio agentPosition,distractorPosition)
        end
        spectrogram = computeSpectrogram(audio)
        augment = rnd([True, False])
        if *augment* **then**
            aug = rnd([timeMasking, frequencyMasking, both])
            applySpecAugment(spectrogram, aug)
        end
    end
end

---

position and not with regard to the shortest reachable intersection with the target's trajectory. As a result, in this setting the dense reward no longer directly points to the optimal policy, reducing the value of the supervisory signal. Finally, a small time penalty of -0.01 for each step incentivises the agent to find short paths.

4.3. Complex Audio Scenarios

Current approaches [9, 10, 21] mostly focus on relatively simple scenarios with a single sound-emitting source in a clean audio environment, while the explorations of sound disturbances remains limited, as discussed in Sec. 2. Inspired by challenges of real world scenarios, we design complex audio-scenarios in which the agent is confronted with second sound-emitting sounds at the goal location, noisy audio sensors, and distractors sounds at different locations. To provide
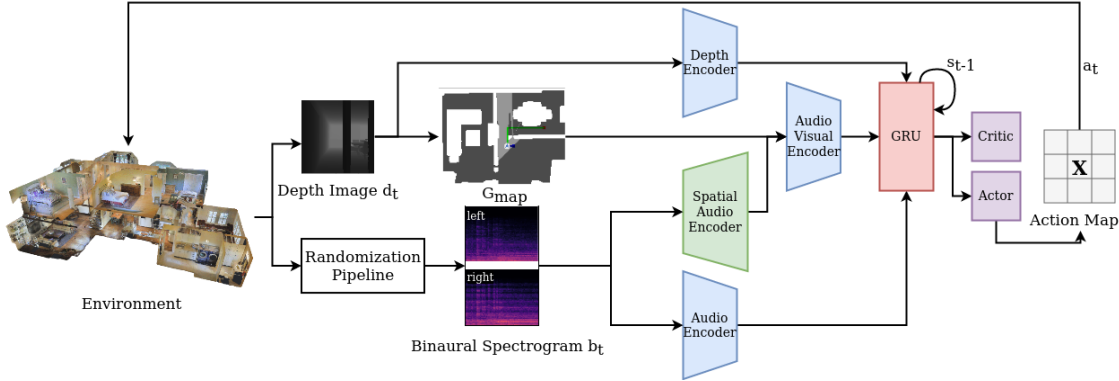
Figure 2. Our proposed architecture. The depth image is projected into an allocentric geometric map G_{map}. A novel channel consisting of a Spatial Audio Encoder and Audio-Visual Encoder fuses the spatial information inherent in geometric map and audio signal. A GRU then combines this channel with separate depth and audio encodings. A PPO agent then produces close-by waypoints that are executed by a Djikstra planner.

the agent with a more realistic, highly diverse training experience to ensure the agent focuses on the directional and spatial information inherent in the audio signal to improve generalization to unheard and noisy environments at test time.

We design highly randomized audio scenarios with perturbations and augmentations on both episode and step level. The full randomization pipeline is shown in Algo. 1 and consists of three elements:

- Distractor Sounds: With a fixed probability, each episode includes a distractor. If the episode includes a distractor, we randomize whether the distractor is audible each step, and if it is audible, a distractor sound is chosen uniformly random from all training sounds, excluding the current target sound. Then we further randomize its location in the environment.

- Second Sound: For each episode, with a certain probability we overlay an additional audio signal coming from the same position over the target sound.

- Spectogram Augmentations: In each step, a random augmentation is applied to the spectogram. We draw on [32] to construct a set of augmentations consisting of (none, time masking, frequency masking, both).

These scenarios on one hand increase the difficulty of the task, requiring the agent to reason about the elements in the audio signal. At the same time it largely increases the diversity of training experience, which has shown to be very beneficial, particularly in scenarios with limited data, such as the comparatively small audio dataset of 102 sounds used here. All augmentations are purely based on the training sound dataset, avoiding any leaks from the validation or test sounds.

4.4. Spatial Audio-Visual Fusion

Existing work focuses on direct end-to-end reinforcement learning from sound and visual inputs to actions. AV-Nav [9] individually encodes RGB-D and audio, while AV-WaN [10] further structures the inputs into a geometric and acoustic map before encoding them individually. Both then concate-

nate the individual features and let a standard GRU cell combine them. [21] estimate a local occupancy map and use the audio signal to estimate the relative goal location, then combine these inputs by providing them to a planner to produce actions. But none of these approaches provide a clear structure to learn to combine these modalities.

Binaurally perceived spectrograms from the sound source contain a large amount of information about the space and room geometry, due to how the sound propagates through the rooms and reflects off of walls and obstacles. Previous work has shown that this information can reveal room geometries [35]. We hypothesize that learning to extract and focus on this information and to learn to combine it with the spatial information from geometric maps is an appropriate architectural prior for audio navigation tasks. Furthermore, we hypothesize that a structure that succeeds to focus on this part of the audio information is more likely to generalize to unheard sounds and to succeed in noisy and distracting audio environments.

As AV-WaN we construct and continuously update an allocentric geometric map G_{map} from the depth inputs d_t. The map has two channels, one for explored/unexplored and one for occupied/free space. We then propose an early fusion of learned audio features and a joint audio-spatial encoder of the geometric map and encoded audio features based on convolutional layers. We introduce a Spatial Audio Encoder module that maps the binaural spectrogram into a spatial feature space. An Audio-Visual Encoder then convolves the channel-wise concatenation of the geometric map and encoded audio features, while also reducing it in dimensionality. Similar to AV-Nav and AV-WaN we then use an RNN as memory component and train the agent end-to-end with Proximal Policy Optimization (PPO) [40]. The overall architecture is shown in Fig. 2. Exact definitions of all modules are provided in the supplementary material.

Action Parametrization: While AV-Nav [9] reasons directly in the raw action space of the simulator, AV-WaN [10]

Model	Replica						MP3D					
	Multiple Heard			Unheard			Multiple Heard			Unheard		
	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA
AV-Nav [9]	54.1	73.9	30.3	34.1	51.1	16.7	53.7	69.8	30.2	28.3	38.2	14.8
AV-WaN [10]	64.5	**91.2**	49.1	27.3	42.5	20.4	55.4	81.4	44.0	39.6	56.6	31.1
Ours	**71.9**	85.9	**53.7**	**48.6**	**63.6**	**35.4**	**66.2**	**86.7**	**48.5**	**46.3**	**60.6**	**33.8**
AV-Nav + complex	**63.3**	86.1	32.2	44.3	63.4	21.3	50.7	70.3	29.7	32.6	48.8	17.2
AV-WaN + complex	62.9	**86.4**	**48.2**	47.8	73.7	36.2	59.6	**86.7**	45.5	47.4	69.4	35.6
Ours + complex	51.2	75.6	35.2	**54.0**	**78.0**	**36.7**	**66.3**	86.4	**47.7**	**55.0**	**73.0**	**38.6**
Ours + complex + dynamic	46.9	60.8	34.8	37.4	52.7	27.5	**66.4**	84.1	**48.9**	48.6	65.3	35.6

Table 1. Results on the **static** Audio Goal task **without** complex scenarios. The heard experiments represent experiments trained on multiple sounds and evaluated on the same sounds but in unseen environments. The unheard sounds experiments represent experiments trained on multiple sounds and evaluated on multiple unheard sounds in unseen environments.

demonstrated further improvements from learning to select waypoints on a higher level of abstraction. The agent chooses from a 9×9 action map centered on the agent's current position. A simple Dijkstra planner then navigates to this waypoint. While acting in such a lifted Markov Decision Process can be beneficial, far away waypoints correspond to a lower control frequency, ignoring up to ten observations while the planner is executing the required actions to reach the selected waypoint. While the loss of information from these observations might be negligible in scenarios with a clean, static sound source, it becomes much more important to continuously integrate the audio observations over time in the noisy and dynamic audio scenarios that we present in this paper. Correspondingly we find it beneficial to decrease the size of the action map to 3×3, providing an efficient middle ground between the benefits of learning waypoints and decreasing the number of unprocessed observations to a maximum of four, using the same planner as [10].

5. Experimental Evaluation

In this section, we present results from experimental evaluations of both existing approaches and our proposed architecture on the static and dynamic AudioGoal tasks. Furthermore, we study the impact of training and evaluating in complex audio scenarios.

5.1. Task Setups

We tackle the tasks of static and dynamic AudioGoal navigation. For each task, we train all agents in two scenarios: the clean audio setup used in [9, 10] and the complex audio scenarios with noisy audio, distractors and second sound-emitting source that we introduced in Sec. 4.3. All agents are trained on multiple sounds and evaluated in two settings: on heard sounds in unseen environments and on unheard sounds in unseen environments. We use the same train/val/test splits protocol used by [9, 11], where Replica splits into 9/4/5 scenes and Matterport3D (MP3D) splits into 59/10/12 scenes. The 102 different sounds are split into 73/11/18. The same split is applied to any other audio signals,

such as distractors. This signifies that evaluations on unheard sounds are also required to handle unheard disturbances.

Metrics: We use the following metrics to evaluate the navigation performance:

- *Success rate (SR)*: The share of successful episodes of all test episodes. An episode is considered a success if the agent executes the *stop* action at the goal location.
- *Success weighted by path length (SPL) [1]*: This metric calculates the ratio of the length of the shortest path to the goal to the length of the executed path for the successful episodes.
- *Success weighted by number of actions (SNA) [10]*: The ratio of the number of actions needed to follow the shortest path to the actual number of actions the agents took to reach the same goal. In contrast to the SPL this metric takes the number of orientation changes into account.
- *Dynamic success weighted by path length (DSPL)*: The primary metric for the novel task of dynamic AudioGoal Navigation. It is calculated as the ratio of the length of the path to the earliest reachable intersection and the length of the actual executed path for the successful episodes, see Sec. 4.1.
- *Dynamic success weighted by number of actions (DSNA)*: Equivalently to the DSPL, we calculate an adjusted version of the SNA with respect to the same definition of the earliest reachable intersection.

Baselines: We compare our approach to two current state-of-the-art methods: AV-Nav [9] and AV-WaN [10]. *AV-Nav [9]* is an end-to-end reinforcement learning agent that directly encodes audio and visual observations to select actions using audio-visual observations. We use the authors' code to train and evaluate the approach. *AV-WaN [10]* is the current state-of-the-art for static AudioGoal task, which predicts intermediate waypoints to the goal depending on audio observation, geometric, and acoustic maps. We also use the code provided by the authors. Hyperparameters for all trained models are listed in the supplementary material.

Model	Replica						MP3D					
	Multiple Heard			Unheard			Multiple Heard			Unheard		
	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA
AV-Nav + complex	55.3	**81.1**	27.5	43.8	65.1	21.5	46.7	69.4	25.6	36.9	56.1	19.4
AV-WaN + complex	**54.0**	80.0	**40.9**	43.0	66.5	32.3	57.6	**84.6**	45.6	49.0	72.3	37.3
Ours + complex	47.9	73.7	33.1	**47.8**	**73.4**	**33.3**	**64.1**	86.4	46.8	**55.5**	**76.0**	**40.4**

Table 2. Results on the **static** Audio Goal task **with** complex scenarios. The heard experiments represent experiments trained on multiple sounds and evaluated on the same sounds but in unseen environments. The unheard sounds experiments represent experiments trained on multiple sounds and evaluated on multiple unheard sounds in unseen environments.

Model	Replica						MP3D					
	Multiple Heard			Unheard			Multiple Heard			Unheard		
	DSPL	SR	DSNA	DSPL	SR	DSNA	DSPL	SR	DSNA	DSPL	SR	DSNA
AV-Nav [9]	44.0	69.5	14.6	22.3	35.0	7.5	39.9	70.1	16.4	21.8	38.7	8.8
AV-WaN [10]	56.6	88.8	33.3	**23.9**	**38.7**	**13.9**	57.9	85.8	34.9	26.2	38.4	16.0
Ours	**62.4**	**93.1**	**33.6**	23.1	36.0	12.4	**63.9**	**95.0**	**36.7**	**29.9**	**46.3**	**17.6**
AV-Nav + complex	43.1	67.0	14.6	23.5	38.4	7.5	37.7	67.3	15.5	20.4	35.5	8.4
AV-WaN + complex	**47.3**	**75.0**	**28.3**	29.4	51.6	**17.6**	55.4	86.7	33.8	36.2	61.5	22.1
Ours + complex	36.2	57.7	20.7	**29.9**	**68.0**	17.0	**61.6**	**91.9**	**36.2**	**45.8**	**74.0**	**26.0**

Table 3. Results on the **dynamic** Audio Goal task **without** complex scenarios. The heard experiments represent experiments trained on multiple sounds and evaluated on the same sounds but in unseen environments. The unheard sounds experiments represent experiments trained on multiple sounds and evaluated on multiple unheard sounds in unseen environments.

5.2. Static AudioGoal Task

We first evaluate the static AudioGoal task, in which the agents are trained on multiple heard sounds and evaluated in unseen apartments with either the heard or unheard sounds. The upper half of Tab. 1 shows the results for the original setup without complex audio scenarios. Our proposed novel channel for a learned spatial fusion achieves similar or even better performance on the heard sounds and significantly improves on unheard sounds, increasing the success rate from 51.1 to 63.6 percent on Replica and 56.6 to 60.6 percent on MP3D, with similar improvements for SPL and SNA. The bottom part of the table shows the effects of training on the complex audio scenarios described in Sec. 4.3 and evaluation on the standard benchmark. Training in these scenarios provides extensive improvements for the generalization to unheard sounds across all models, with improvements of over 30 percentage points for certain models. Again, our model achieves the highest performance across all metrics. Combined, our new architecture and the complex scenarios increase the performance from an SPL of 27.3 to 54.0 on Replica and 39.6 to 55.0 on MP3D, strongly outperforming previous state-of-the art results on this benchmark. A further decomposition of the effects of the individual elements of the complex scenarios can be found in the supplemental material.

We observe a deterioration in performance of our architecture when trained on the complex scenarios and evaluated only on the heard sounds in Replica. This can be attributed to the small dataset size or the complex scenarios remove certain channels that the model could previously exploit to overfit on heard sounds. We then further evaluate the impact of including dynamic target sounds within the training. We let the sound source move in a random selection of half of the episodes and remain static in the other half to test the models ability to learn both tasks simultaneously. While we observe a negative impact of this training setup on Replica, performance on MP3D remains stable. The differences between moving and static sounds can be more considerable on the much smaller apartments in Replica. We proceed to evaluate the performance on the complex audio scenarios. Results are shown in Tab. 2. We find that these scenarios indeed signify a more challenging task, with lower performance across all models. Though notably, the drop is much smaller on the large MP3D dataset. The performance on MP3D unheard sounds is even up to par with the non-complex scenarios. Again we find that our architecture consistently achieves the best results in generalization to unheard sounds.

5.3. Dynamic AudioGoal Task

We then evaluate the models on the moving sound task. All models are trained on multiple heard sounds with half of the episodes containing a moving and the other half containing a static sound source. The models are then evaluated on the dynamic sounds for both heard and unheard sounds. Results are shown in Tab. 3. Overall, we find that all models are able to solve a significant share of the tasks, with success rates of 69.5-95 percent on heard sounds. This is roughly similar to the performance on static sound. But we find a much larger gap in the performance on unheard sounds, both

Model	Replica						MP3D					
	Multiple Heard			Unheard			Multiple Heard			Unheard		
	DSPL	SR	DSNA	DSPL	SR	DSNA	DSPL	SR	DSNA	DSPL	SR	DSNA
AV-Nav + complex	32.9	60.1	10.5	25.5	45.2	8.1	35.6	62.6	13.7	25.1	44.3	9.7
AV-WaN + complex	**44.0**	**66.1**	**25.6**	**34.2**	52.1	**19.9**	53.7	84.6	32.9	42.9	70.9	26.2
Ours + complex	32.6	55.3	18.8	27.5	**52.6**	16.1	**59.9**	**92.8**	**33.3**	**52.1**	**84.1**	**30.4**

Table 4. Results on the **dynamic** Audio Goal task **with** complex scenarios. The heard experiments represent experiments trained on multiple sounds and evaluated on the same sounds but in unseen environments. The unheard sounds experiments represent experiments trained on multiple sounds and evaluated on multiple unheard sounds in unseen environments.

Replica   Matterport3D

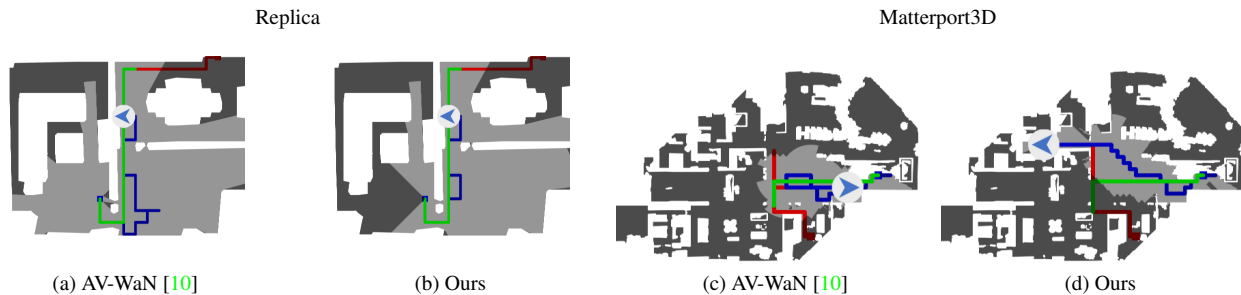(a) AV-WaN [10]   (b) Ours   (c) AV-WaN [10]   (d) Ours

Figure 3. Example episodes of the heard dynamic audio-visual navigation task on the replica (left) and Matterport3D (right) dataset. For each, the AV-WaN agent is shown on the left and our architecture on the right, both trained without complex scenarios. The paths of the agent and sound source are shown in blue and red, respectively. Green shows the path to the earliest reachable intersection as defined for the DSPL metric.

in terms of success rates and optimality of the paths (SPL vs DSPL). In terms of overall performance, we find that our architecture again clearly performs best on all clean audio setups, except for slight advantages for AV-WaN on Replica unheard sounds. Training on the complex scenarios again proves very beneficial for generalization to unheard sounds for all models except AV-Nav. Our trained architecture with these complex scenarios achieves large improvements from a success rate of 38.7 for the best baseline on clean scenarios to 68.0 on replica and from 38.7 to 74.0 on MP3D. Tab. 4 evaluates the same models on the complex audio scenarios. On replica, we find the best performance changes between AV-WaN and ours, depending on the metric, while on MP3D our approach consistently performs the best.

Fig. 3 depicts example episodes for the AV-WaN and our agent on heard sounds, illustrating some of the challenges of this task. Optimal behaviour can significantly differ from simply moving directly towards the initial position of the sound. Furthermore, acting suboptimally early on can have a large impact on the required path later on and agent's may have to quickly change direction if the target moves past them.

6. Conclusion

We introduce the novel dynamic audio-visual navigation benchmark together with a novel metric that quantifies the gap to optimal behavior. We demonstrate that this new task poses further challenges over existing benchmarks. We introduce complex audio scenarios based on audio-specific aug-

mentations, perturbations, and randomizations and demonstrate that this provides substantial benefits in generalization to unheard sounds. Lastly, we introduce an architecture with an inductive bias to allow the agent to spatially fuse the geometric information inherent in the audio and visual observations. Combined with training on the new audio scenarios, this results in large overall improvements for the generalization to unheard sounds and the best performance on the dynamic audio-visual navigation task on MP3D.

Limitations and Future Work While we are able to further close the gap in the generalization to unheard sounds, there remains a significant gap to optimal performance as measured by the SPL. This gap is even larger for the novel task of dynamic audio-visual navigation, indicating the need for further research. The current audio dataset only consists of 102 different sounds. While randomizations and perturbations can significantly increase the variety in this data, introducing larger audio datasets is a promising avenue to scale to more complex and robust models.

So far, audio-visual navigation tasks have remained constrained to simulation. While we introduce more complex audio scenarios, these still do not capture all the challenges of the real world. A particular restriction in contrast to the continuous real world is the discrete navigation grid of the simulator. Binaural room impulse responses are only available for these predefined locations, prohibiting the exploration of continuous audio streams, actuation noise or continuous actions in simulation. Sim-to-real transfer of these tasks will be a very important next step.

Acknowledgements

References

[1] Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018. 4, 6

[2] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3674–3683, 2018. 2

[3] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Eur. Conf. Comput. Vis.*, 2018. 2

[4] F. Belkhouche, B. Belkhouche, and P. Rastgoufard. Parallel navigation for reaching a moving goal by a mobile robot. *Robotica*, 25(1):63–74, 2007. 2

[5] Federico Boniardi, Abhinav Valada, Rohit Mohan, Tim Caselitz, and Wolfram Burgard. Robot localization in floor plans using a room layout edge extraction network. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 5291–5297, 2019. 1

[6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *Int. Conf. on 3D Vision*, pages 667–676, 2017. Matterport3D dataset licence available at: http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf. 2, 3, 11

[7] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. In *Int. Conf. Learn. Represent.*, 2019. 2

[8] Changan Chen, Ziad Al-Halah, and Kristen Grauman. Semantic audio-visual navigation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 15516–15525, 2021. 1, 2

[9] 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 *Eur. Conf. Comput. Vis.*, pages 17–36, 2020. Sound dataset licence available at: https://github.com/facebookresearch/sound-spaces/blob/main/LICENSE. 1, 2, 3, 4, 5, 6, 7, 11, 12

[10] Changan Chen, Sagnik Majumder, Ziad Al-Halah, Ruohan Gao, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Learning to set waypoints for audio-visual navigation. In *Int. Conf. Learn. Represent.*, 2020. 1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13

[11] Changan Chen, Wei Sun, David Harwath, and Kristen Grauman. Learning audio-visual dereverberation. *arXiv preprint arXiv:2106.07732*, 2021. 1, 6

[12] Kevin Chen, Junshen K Chen, Jo Chuang, Marynel Vázquez, and Silvio Savarese. Topological planning with transformers for vision-and-language navigation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11276–11286, 2021. 2

[13] Kevin Chen, Juan Pablo de Vicente, Gabriel Sepulveda, Fei Xia, Alvaro Soto, Marynel Vázquez, and Silvio Savarese. A behavioral approach to visual navigation with graph localization networks. *arXiv preprint arXiv:1903.00445*, 2019. 2

[14] Yu Fan Chen, Miao Liu, Michael Everett, and Jonathan P How. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning. In *IEEE Int. Conf. on Robotics and Automation*, pages 285–292, 2017. 2

[15] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–10, 2018. 2

[16] Heming Du, Xin Yu, and Liang Zheng. Vtnet: Visual transformer network for object goal navigation. In *Int. Conf. Learn. Represent.*, 2020. 2

[17] Christine Evers, Alastair H. Moore, Patrick A. Naylor, Jonathan Sheaffer, and Boaz Rafaely. Bearing-only acoustic tracking of moving speakers for robot audition. In *IEEE Int. Conf. on Digital Signal Processing*, pages 1206–1210, 2015. 2

[18] Scott Forer, Santosh Balajee Banisetty, Logan Yliniemi, Monica Nicolescu, and David Feil-Seifer. Socially-aware navigation using non-linear multi-objective optimization. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1–9, 2018. 2

[19] Madeleine Fortin, Patrice Voss, Catherine Lord, Maryse Lassonde, Jens Pruessner, Dave Saint-Amour, Constant Rainville, and Franco Lepore. Wayfinding in the blind: larger hippocampal volume and supranormal spatial navigation. *Brain*, 131(11):2995–3005, 2008. 1

[20] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. In *Adv. Neural Inform. Process. Syst.*, 2018. 2

[21] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *IEEE Int. Conf. on Robotics and Automation*, pages 9701–9707, 2020. 1, 2, 4, 5

[22] Matti Grohn, Tapio Lokki, and Tapio Takala. Static and dynamic sound source localization in a virtual room. *J. of the Audio Engineering Society*, 2002. 2

[23] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2616–2625, 2017. 1, 2

[24] John Hershey and Javier Movellan. Audio vision: Using audio-visual synchrony to locate sounds. In S. Solla, T. Leen, and K. Müller, editors, *Adv. Neural Inform. Process. Syst.*, volume 12, 2000. 2

[25] Juana Valeria Hurtado, Laura Londoño, and Abhinav Valada. From learning to relearning: A framework for diminishing bias in social robot navigation. *Frontiers in Robotics and AI*, 8:69, 2021. 2

[26] Juana Valeria Hurtado, Rohit Mohan, Wolfram Burgard, and Abhinav Valada. Mopt: Multi-object panoptic tracking. *arXiv preprint arXiv:2004.08189*, 2020. 2

[27] Stephen James, Andrew J Davison, and Edward Johns. Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task. In *Conf. on Robot Learning*, pages 334–343, 2017. 3

[28] Dinesh Jayaraman and Kristen Grauman. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1238–1247, 2018. 2

[29] E. Kidron, Y.Y. Schechner, and M. Elad. Pixels that sound. In *IEEE Conf. Comput. Vis. Pattern Recog.*, volume 1, pages 88–95, 2005. 2

[30] Sagnik Majumder, Ziad Al-Halah, and Kristen Grauman. Move2hear: Active audio-visual source separation. *arXiv preprint arXiv:2105.07142*, 2021. 1, 2

[31] Kazuhiro Nakadai and Keisuke Nakamura. Sound source localization and separation. *Wiley Encyclopedia of Electrical and Electronics Engineering*, pages 1–18, 1999. 2

[32] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019. 3, 5

[33] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Int. Conf. on Machine Learning*, pages 2778–2787, 2017. 2

[34] Alban Portello, Patrick Danès, and Sylvain Argentieri. Active binaural localization of intermittent moving sources in the presence of false measurements. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3294–3299, 2012. 2

[35] Senthil Purushwalkam, Sebastia Vicenc Amengual Gari, Vamsi Krishna Ithapu, Carl Schissler, Philip Robinson, Abhinav Gupta, and Kristen Grauman. Audio-visual floorplan reconstruction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1183–1192, 2021. 1, 5

[36] Caleb Rascon and Ivan Meza. Localization of sound sources in robotics: A review. *Robotics and Autonomous Systems*, 96:184–210, 2017. 2

[37] Fereshteh Sadeghi and Sergey Levine. CAD2RL: real single-image flight without a single real image. In Nancy M. Amato, Siddhartha S. Srinivasa, Nora Ayanian, and Scott Kuindersma, editors, *Robotics: Science and Systems*, 2017. 3

[38] 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 *Int. Conf. Comput. Vis.*, 2019. HabitatLab licence available at: https://github.com/facebookresearch/habitat-lab/blob/v0.1.6/LICENSE. 2, 3

[39] Paul Scerri, Prasanna Velagapudi, Balajee Kannan, Abhinav Valada, Christopher Tomaszewski, John Dolan, Adrian Scerri, Kumar Shaurya Shankar, Luis Bill, and George Kantor. Real-world testing of a multi-robot team. In *Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 1213–1214, 2012. 2

[40] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5, 12

[41] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019. 3

[42] 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. Replica dataset licence available at: https://github.com/facebookresearch/Replica-Dataset/blob/main/LICENSE. 2, 3, 11

[43] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Eur. Conf. Comput. Vis.*, pages 247–263, 2018. 2

[44] Abhinav Valada and Wolfram Burgard. Deep spatiotemporal models for robust proprioceptive terrain classification. *The Int. J. of Robotics Research*, 36(13-14):1521–1539, 2017. 1

[45] J.-M. Valin, F. Michaud, B. Hadjou, and J. Rouat. Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach. In *IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 1033–1038, 2004. 2

[46] Francisco Rivera Valverde, Juana Valeria Hurtado, and Abhinav Valada. There is more than meets the eye: Self-supervised multi-object detection and tracking with sound by distilling multimodal knowledge. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11612–11621, 2021. 1

[47] Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied question answering in photorealistic environments with point cloud perception. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6659–6668, 2019. 2

[48] 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 *Int. Conf. Learn. Represent.*, 2019. 1, 2

[49] Wei Yang, Xiaolong Wang, Ali Farhadi, Abhinav Gupta, and Roozbeh Mottaghi. Visual semantic navigation using scene priors. *arXiv preprint arXiv:1810.06543*, 2018. 2

[50] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L Berg, and Dhruv Batra. Multi-target embodied question answering. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6309–6318, 2019. 2

[51] Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Towards distraction-robust active visual tracking. *arXiv preprint arXiv:2106.10110*, 2021. 2, 3

[52] Qingbao Zhu, Jun Hu, and Larry Henschen. A new moving target interception algorithm for mobile robots based on subgoal forecasting and an improved scout ant algorithm. *Applied Soft Computing*, 13(1):539–549, 2013. 2

Catch Me If You Hear Me: Audio-Visual Navigation in Complex Unmapped Environments with Moving Sounds

Supplementary Material

Abdelrahman Younes*   Daniel Honerkamp*   Tim Welschehold   Abhinav Valada

University of Freiburg

{younesa, honerkamp, twelsche, valada}@cs.uni-freiburg.de

In this supplementary material, we provide additional details on the publicly released code to run the experiments for our model as well as the baselines, and a video demonstrating the qualitative behavior of the agents. We describe the model architecture in detail and list the hyperparameters for all models. We also analyze the impact of individual components of the complex audio scenarios in an ablation study.

1. Qualitative Results

Fig. 1 depicts further qualitative examples for the AV-WaN [10] agent and our model on the dynamic audio-visual navigation benchmark. Each column shows the same episode for each agent. While both agents initially move in the right direction, the AV-WaN agent ultimately has to move through a much larger part of the apartment, chasing the sound source. In contrast our approach captures the target in its path. Novel challenges of this task include the need to adapt directions to the moving sound and potentially much higher costs for missing the sound source, which then may move further away from the agent.

2. Extended Architecture Details

In this section, we provide additional details on the individual modules of our proposed architecture.

2.1. Depth Encoder

We adopt the depth encoder introduced by AV-Nav [9] baseline. The encoder has three convolutional layers with kernel sizes of (8, 8), (4, 4) and (3, 3), and with strides (4, 4), (2, 2), and (2,2). Each convolution layer is followed by an ReLU activation function. A fully connected layer at the end of the encoder then embeds the features to size of 512 followed by an ReLU activation function.

2.2. Audio Encoder

Based on the architecture of [10], we encode the audio signals with three convolutional layers with kernel sizes of (8, 8), (4, 4) and (3, 3) and strides of (4, 4), (2, 2) and (1, 1) for Replica [42], and kernel sizes of (5, 5), (3, 3), and (3, 3)

and strides of (2, 2), (2, 2), and (1, 1) for Matterport3D [6] datasets. Each convolution layer is followed by an ReLU activation function. A fully connected layer at the the end of the encoder then embeds the features to size of 512 followed by an ReLU activation function.

2.3. Spatial Audio Encoder

This encoder receives as an input the binaural spectrogram b_t of size (2, 65, 69) for Replica and (2, 65, 26) for the Matterport3D datasets. The Spatial Audio Encoder upscales the input to the same dimensionality as the geometrical map, which is (2, 200, 200). Due to the different spectrogram dimensionality of the two datasets, we implement two different encoders. Replica's encoder consists of two transposed convolution layers with kernel sizes of (8,8) and (1,13), and with strides of (3,3) and (1,1), respectively. We employ an ReLU activation function after each convolution layer. For Matterport3D, the encoder consists of two transposed convolution layers followed by one convolution layer. The kernel sizes are (5,2), (4,2) and (1,5) while the strides sizes are (3,4), (1,2) and (1,1), respectively. We also employ an ReLU activation function after each convolution layer.

2.4. Audio-Visual Encoder

We concatenate the output of the Spatial Audio Encoder with the current geometrical map G_{map}. Subsequently, we pass the resulting features through three convolutional layers with ReLU activations, followed by a fully connected layer to embed the features into an embedding of size 512. Finally, we employ an ReLU activation function and feed the output to a GRU. The kernel sizes are (8, 8), (4, 4), and (3, 3) while the strides are (4, 4), (2, 2), and (1, 1).

2.5. Actor-Critic

The GRU is implemented as a single bidirectional GRU cell with hidden size 512 and one recurrent layer. The actor and critic heads take as input the current state of the GRU and estimate the action distribution $\pi(a_t|s_t)$ and value of the state $V(s_t)$, respectively. Both are implemented as a linear layer. The action distribution is a categorical distribution
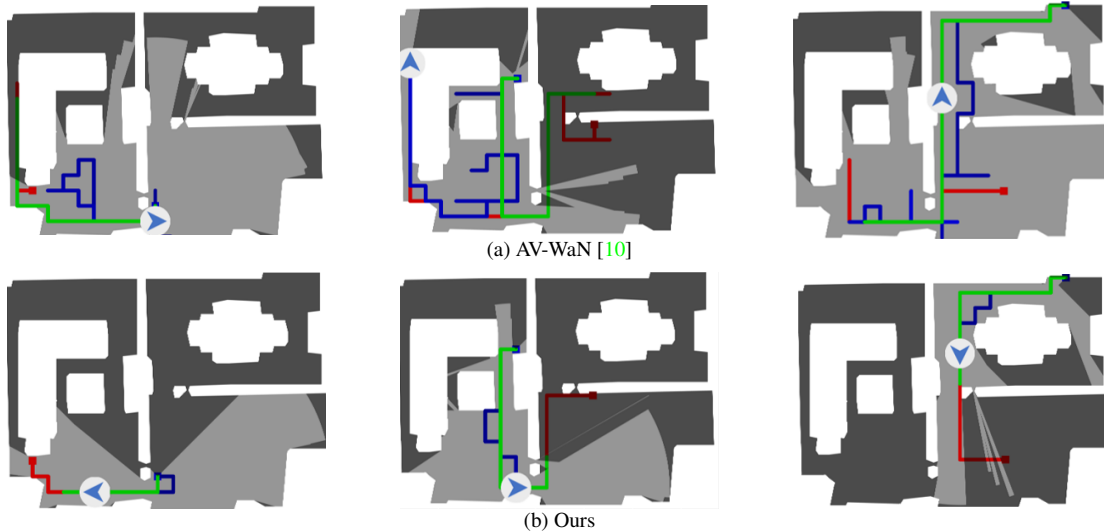
(a) AV-WaN [10]

(b) Ours

Figure 1. Qualitative comparison of the dynamic audio-visual navigation task on the Replica dataset for heard sounds without complex scenarios. Each column shows the same episode for the AV-WaN agent on top and ours in the bottom. The paths of the agent and sound source are shown in blue and red, respectively. Start poses are marked with rectangles. Green shows the path to the earliest reachable intersection as defined for the DSPL metric.

| Model | Replica | | | | | | MP3D | | | | | |
| | Multiple Heard | | | Unheard | | | Multiple Heard | | | Unheard | | |
	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA
Ours	**71.9**	**85.9**	**53.7**	48.6	63.6	35.4	66.2	**86.7**	48.5	46.3	60.6	33.8
Ours + second audio	70.2	85.3	51.6	**56.0**	74.6	**40.4**	62.6	81.8	47.4	51.2	65.0	38.2
Ours + specaugment	56.5	78.4	40.2	53.9	78.7	38.1	59.2	79.7	43.3	39.6	58.3	28.6
Ours + second audio + specaugment	59.2	82.4	41.8	54.5	**80.0**	38.3	62.6	83.2	**48.7**	53.0	**73.3**	**40.7**
Ours + second audio + distractor	61.5	82.2	44.1	48.7	72.1	34.6	64.7	83.3	47.8	52.0	70.1	37.8
Ours + complex	51.2	75.6	35.2	54.0	78.0	36.7	**66.3**	86.4	47.7	**55.0**	73.0	38.6

Table 1. Decomposition of complex scenarios: Evaluation on the **static** Audio Goal task **without** complex scenarios.

with probabilities based on the softmax of the actor's output. Similar to [9, 10], we also add an entropy maximization term for exploration to the Proximal Policy Optimization (PPO) [40] objective.

3. Complexity of Audio Scenarios

In this section, we further analyze the importance of the individual components of the audio scenarios. Tab. 1 shows the performance on the standard static AudioGoal task for models trained with subsets of these components. Focusing on unheard sounds, we find that training with the second audio improves performance on both datasets, while the spectrogram augmentations improve generalization on Replica, but not on Matterport3D. On the other hand, combining both further improves on Matterport3D but not on Replica. Lastly, the addition of distractor sounds further improves generalization on MP3D, but does not have a large impact on Replica. These differences in the impact of the individual elements across datasets show the importance to combine different randomizations and perturbation.

On the heard sounds, the perturbations do not influence performance on MP3D substantially. On Replica, the decomposition shows that the decrease was not due to any single component, but rather any kind of disturbance reduces the performance. This might further indicate that the agent learns to use or overly rely on some features of the audio signal that are not as robust or general to navigate these much smaller environments. On the other hand, the impact of the perturbations might be larger as the distances to the goals are generally shorter and thereby the number of audio observations the agent receives to filter out such disturbances are also fewer.

We then repeat the same evaluation with complex scenarios at test time, shown in Tab. 2. We find that training on all components is required to achieve robustness to these scenarios. No subset of the components is able to induce the same robustness to the full set of perturbations. In particular, the performance of the model trained on clean audio scenarios drops significantly across all settings and metrics, often more than 50%. This demonstrates the importance

Model	Replica						MP3D					
	Multiple Heard			Unheard			Multiple Heard			Unheard		
	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA	SPL	SR	SNA
Ours	24.5	31.7	18.0	20.0	26.9	15.0	40.0	51.3	28.7	32.1	43.1	23.4
Ours + second audio	31.0	41.7	22.6	23.6	33.1	16.8	46.9	62.1	35.8	43.1	56.5	32.8
Ours + specaugment	47.1	67.7	33.3	45.7	67.8	32.3	50.7	70.9	36.5	40.4	58.0	28.7
Ours + second audio + specaugment	**49.3**	70.4	**34.9**	46.1	68.8	32.6	55.6	79.0	43.3	53.4	72.9	**40.7**
Ours + second audio + distractor	38.5	56.1	27.6	32.9	47.9	23.4	49.1	64.9	36.8	41.5	55.3	30.6
Ours + complex	47.9	**73.7**	33.1	**47.8**	**73.4**	**33.3**	**64.1**	**86.4**	**46.8**	**55.5**	**76.0**	40.4

Table 2. Decomposition of complex scenarios: Evaluation on the **static** Audio Goal task **with** complex scenarios.

Hyperparameter	Value
clip param	0.1
ppo epoch	4
num mini batch	1
value loss coef	0.5
entropy coef (AV-Nav)	0.02 (0.2)
lr	2.5e−4
eps	1e−5
max grad norm	0.5
optimizer	Adam
steps number	150
gru hidden size	512
use gae	True
gamma	0.99
tau	0.95
linear clip decay	True
linear lr decay	True
exponential lr decay	False
exp decay lambda	5.0
reward window size	50
number of processes (AV-Nav Matterport3d)	5 (10)
number of updates (AV-Nav)	10,000 (40,000)
time mask param (Replica)	32
frequency mask param (Replica)	12
time mask param (Matterport3d)	12
frequency mask param (Matterport3d)	12

Table 3. Hyperparameters used for training our model and the baselines. Differences across models are shown in parentheses.

of training on these scenarios to achieve robust behavior in noisy and complex audio environments.

4. Hyperparameters

To ensure a fair comparison to the baselines we use the same hyperparameters as reported by [10] for all models, including our proposed architecture. The hyperparameters are listed in Tab. 3. Each episode has a time limit of 500 steps after which it will be stopped and counted as a failure.