Semantic Audio-Visual Navigation

Changan Chen[1,2] Ziad Al-Halah[1] Kristen Grauman[1,2]
[1]UT Austin [2]Facebook AI Research

Abstract

Recent work on audio-visual navigation assumes a constantly-sounding target and restricts the role of audio to signaling the target's spatial placement. We introduce semantic audio-visual navigation*, where objects in the environment make sounds consistent with their semantic meanings (e.g., toilet flushing, door creaking) and acoustic events are sporadic or short in duration. We propose a transformer-based model to tackle this new semantic AudioGoal task, incorporating an inferred goal descriptor that captures both spatial and semantic properties of the target. Our model's persistent multimodal memory enables it to reach the goal even long after the acoustic event stops. In support of the new task, we also expand the SoundSpaces audio simulation platform to provide semantically grounded object sounds for an array of objects in Matterport3D. Our method strongly outperforms existing audio-visual navigation methods by learning to associate semantic, acoustic, and visual cues.*

1. Introduction

An autonomous agent interacts with its environment in a continuous loop of action and perception. The agent needs to reason intelligently about all the senses available to it (sight, hearing, proprioception, touch) to select the proper sequence of actions in order to achieve its task. For example, a service robot of the future may need to locate and fetch an object for a user, go empty the dishwasher when it stops running, or travel to the front hall upon hearing a guest begin speaking there.

Towards such applications, recent progress in visual navigation builds agents that use egocentric vision to travel to a designated point in an unfamiliar environment [22, 34, 38, 10], search for a specified object [40, 9, 33, 8], or explore and map a new space [13, 10, 15, 10]. Limited new work further explores expanding the sensory suite of the navigating agent to include hearing as well. In particular, methods tackling the AudioGoal challenge use sound to get key directional and distance information about a sounding target to which the agent must navigate (e.g., a ringing phone) [11, 19, 12].

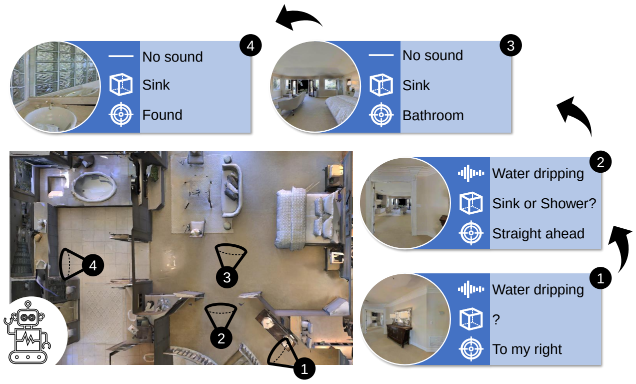


Figure 1: Proposed task: Semantic audio-visual navigation. The agent must navigate to a sounding object in the 3D environment. Since the sound may stop while the agent searches for the object, the agent is incentivized to learn the association between how objects look and sound, and to build contextual models for where different semantic sounds are more likely to occur (e.g., water in the bathroom).

While exciting first steps, existing audio-visual (AV) navigation work has two key limitations. First, prior work assumes the target object constantly makes a steady repeating sound (e.g., alarm chirping, phone ringing). While important, this corresponds to a narrow set of targets; in real-world scenarios, an object may emit a sound only briefly or start and stop dynamically. Second, in current models explored in realistic 3D environment simulators, the sound emitting target has neither a visual embodiment nor any semantic context. Rather, target sound sources are placed arbitrarily in the environment and without relation to the semantics of the scene and objects. As a result, the role of audio is limited to providing a beacon of sound announcing where the object is.

In light of these limitations, we introduce a novel task: *semantic audio-visual navigation*. In this task, the agent must navigate to an object situated contextually in an environment that only makes sound for a certain period of time. Semantic audio-visual navigation widens the set of real-world scenarios to include acoustic events of short temporal duration that are semantically grounded in the environment. It offers new learning challenges. The agent must learn not only how to associate sounds with visual objects, but also how to leverage the semantic priors of objects (along with any acoustic cues) to reason about where the object is likely located in the scene. For example, hearing the dishwasher stop running and issue its end of cycle chime should suggest both what visual object to search for as well as the likely paths for finding it, i.e., towards the kitchen rather than the bedroom. Notably, in the proposed task, the agent is not given any external information about the goal (such as a displacement vector or name of the object to search for). Hence the agent must learn to leverage sporadic acoustic cues that may stop at any time as it searches for the source, inferring what visual object likely emitted the sound even after it is silent. See Figure 1.

To tackle semantic AudioGoal, we introduce a deep reinforcement learning model that learns the association between how objects look and how they sound. Before seeing the target object, the model learns to hypothesize the goal properties (e.g., location and object category) from the received acoustic cues. Coupled with a transformer, it learns to attend to the previous visual and acoustic observations in its memory—conditioned on the predicted goal descriptor—to navigate to the audio source. Furthermore, to support this line of research, we instrument audio-visual simulations for real scanned environments such that semantically relevant sounds are attached to semantically relevant objects.

We evaluate our model on 85 large-scale real-world environments with a variety of semantic objects and their sounds. Our approach outperforms state-of-the-art models in audio-visual navigation with up to 8.9% improvement in SPL. Furthermore, our model is robust in handling short acoustic signals emitted by the goal with varying temporal duration, and compared to the competitors, it more often reaches the goal after the acoustic observations end. In addition, our model maintains good performance in the presence of environment noise (distractor sounds) compared to baseline models. Finally, we demonstrate the potential for embodied agents to learn about how objects look and sound through interactions with the 3D environment.

2. Related work

Visual navigation. To navigate autonomously, traditionally a robot builds a map with 3D reconstruction algorithms (i.e., SLAM) and then plans a path using the map [17]. Recent work instead learns navigation policies directly from egocentric observations [22, 33, 28]. A popular task is PointGoal navigation, where the goal position is given to the agent by its relative displacement vector from the agent's position [22, 28, 34, 38]. Alternatively, in the ObjectGoal setting, the agent is given a semantic object label rather than the goal location, and is tasked with navigating to the nearest instance of that category (e.g., go to a table) [40, 4, 9]. In contrast to both PointGoal and ObjectGoal, in the proposed setting the agent is not given specific goal information. Instead, it needs to react to an acoustic event to determine what kind of object is sounding and navigate to it. Furthermore, unlike ObjectGoal, the agent needs to navigate to the specific object instance that emitted the sound rather than any instance of that category. Our task represents real-world scenarios where dynamic objects draw the attention of an agent and call it to action (e.g., the sound of a heavy object falling somewhere upstairs).

Audio-visual navigation Recent work leverages audio in embodied navigation for the AudioGoal navigation task [11, 19]. In that setup, the agent navigates to a sound-emitting goal using both visual and acoustic observations [11, 19, 12]. As discussed above, prior methods assume the goal is sounding continuously through the episode and that it does not have a visual embodiment. While suitable for certain events like a fire alarm, many acoustic events are short and infrequent (e.g., glass breaking, door slamming, a person calling for help). We consider a generalized setting where the audio signal is only available for a limited period of time and the agent must find it using both initial acoustic cues and the goal semantics. In addition, we augment the SoundSpaces audio simulations [11] for Matterport3D [6] to portray semantically relevant object-level sounds, an advance over the simulations used in prior work [11, 19, 12] that inserted a small set of sounds randomly in the environments without any visual embodiment.

Memory models for 3D environments While it is common to use an implicit memory representation in navigation to aggregate observations, e.g., a recurrent neural network [27, 34, 11, 26, 3, 29], other methods leverage explicit map-based memories to record occupancy [22, 13, 32, 10, 12] or object locations [9, 5]. To capture long-term dependencies while avoiding designer bias in the memory encoding, another promising direction is to use a transformer architecture [37] to record observations and poses [15]. We build in this direction and introduce a scene memory transformer that, unlike prior work, 1) is multimodal and 2) leverages an explicit learned *goal descriptor* to attend to the memory. Our memory model learns audio-visual associations between the goal and the observations from the scene, a crucial functionality as we demonstrate in experiments.

Audio-visual learning in video Work in passive (non-embodied) video analysis also explores the link between object appearance and sound. This includes audio source separation methods that disentangle sounds based on object appearance [14, 30, 1, 39, 20, 21] as well as self-supervised video representation learning methods [25, 31, 16, 18]. Our approach also learns how to associate objects with their sounds. However, in contrast to previous video approaches, our approach learns in the context of an agent's interaction with a 3D environment. Namely, our agent learns to associate an object category inferred from audio with its visual representation and contextual scene cues at the same time it learns to navigate efficiently.

3. Semantic Audio-Visual Navigation

We introduce the novel task of semantic audio-visual navigation. In this task, the agent is required to navigate in a complex, unmapped environment to find a semantic sounding object—semantic AudioGoal for short. Different from AudioGoal [11, 19], the goal sound *need not be periodic*, has *variable duration*, and is associated with a meaningful *semantic object* (e.g., the door creaking is associated with the apartment's door). This setting represents common real world events and, as discussed above, poses new challenges for embodied learning. Relying on audio perception solely to produce step-by-step actions is not sufficient, since the audio event is relatively short. Instead, the agent needs to reason about the category of the sounding object and use both visual and audio perception to predict its location.

3D environments and simulator. Consistent with the active body of computer vision work on embodied AI done in simulation, and to facilitate the reproducibility of our work, we rely on a visually and acoustically realistic simulation platform to model an agent moving in complex 3D environments. We use SoundSpaces [11], which enables realistic audio rendering of arbitrary sounds for the real-world environment scans in Replica [36] and Matterport3D [6]. We use the Matterport environments due to their greater scale and complexity. SoundSpaces is Habitat-compatible [34] and allows rendering arbitrary sounds at any pair of source and receiver (agent) locations on a uniform grid of nodes spaced by 1 m. Next we explain how we extend this audio data to provide semantically meaningful sounds.

Semantic sounds data collection. We use the 21 object categories defined in the ObjectGoal navigation challenge [4] for Matterport3D environments: chair, table, picture, cabinet, cushion, sofa, bed, chest of drawers, plant, sink, toilet, stool, towel, tv monitor, shower, bathtub, counter, fireplace, gym equipment, seating, and clothes. All of these categories have objects that are visually present in Matterport environments. By rendering object-specific sounds at the locations of the Matterport objects, we obtain semantically meaningful and contextual sounds. For example, the water flush sound will be associated with the toilet in the bathroom, and the crackling fire sound with the fireplace in the living room or the bedroom. We filter out object instances that are not reachable by the navigability graph of the simulation environments, namely where the distance to the closest node is greater than 1 m. The number of remaining object instances for train/val/test is 303/46/80 on average for each object category.

We consider two types of sound events: object-emitted and object-related. Object-emitted sounds are generated by the object, e.g., a toilet flushing, whereas object-related sounds are caused by peoples' interactions with the object, e.g., food being chopped on the counter. To provide a variety of sounds, we download long copyright-free audio clips that include sounds of various actions being performed on the 21 object categories. We split the original clips (average length 81s) evenly into train/val/test clips. These splits allow the characteristics of the unheard sounds (i.e., waveforms not heard during training) to be related to those in the training set, while still preserving natural variations.[1] The duration of the acoustic phase in each episode is randomly sampled from a Gaussian of mean 15s and deviation 9s, clipped for a minimum 5s and maximum 500s. If the sampled duration is longer than the length of the audio clip, we replay the clip. See the Supp. video for examples.

Action space and sensors. The agent's action space is *MoveForward*, *TurnLeft*, *TurnRight*, and *Stop*. The last three actions are always valid, while *MoveForward* only takes effect when the node in front of the agent is reachable from that position (no collision). The sensory inputs are egocentric binaural sound (two-channel audio waveforms), RGB, depth, and the agent's current pose.

Episode specification and success criterion. An episode of semantic AudioGoal is defined by 1) the scene, 2) the agent start location and rotation, 3) the goal location, 4) the goal (object) category and 5) the duration of the audio event. In each episode in a given scene, we choose a random object category and a random instance of that category as the goal. The agent's start pose is also randomly positioned in the scene. In semantic AudioGoal, the agent has to stop near the particular sounding object instance, not simply any instance of the class. This is a stricter success criterion than ObjectGoal [4], which judges an episode as successful if the agent stops near any instance of that category. We define a set of viewpoints around each object within 1 m of the object's boundary; issuing the *Stop* action at any of these viewpoints is considered a successful termination of the episode.

4. Approach

We propose a novel model for the semantic audio-visual navigation task. Our approach uses a persistent multimodal memory along with a transformer model, which, unlike RNN-based architectures (e.g., [11]) or reactive ones (e.g., [19]), allows it to directly attend to observations with

[1] Note that even the same waveform will sound different when rendered in a new environment; the sound received by the agent is a function of not only that waveform but also the environment geometry and the agent's position relative to the source.

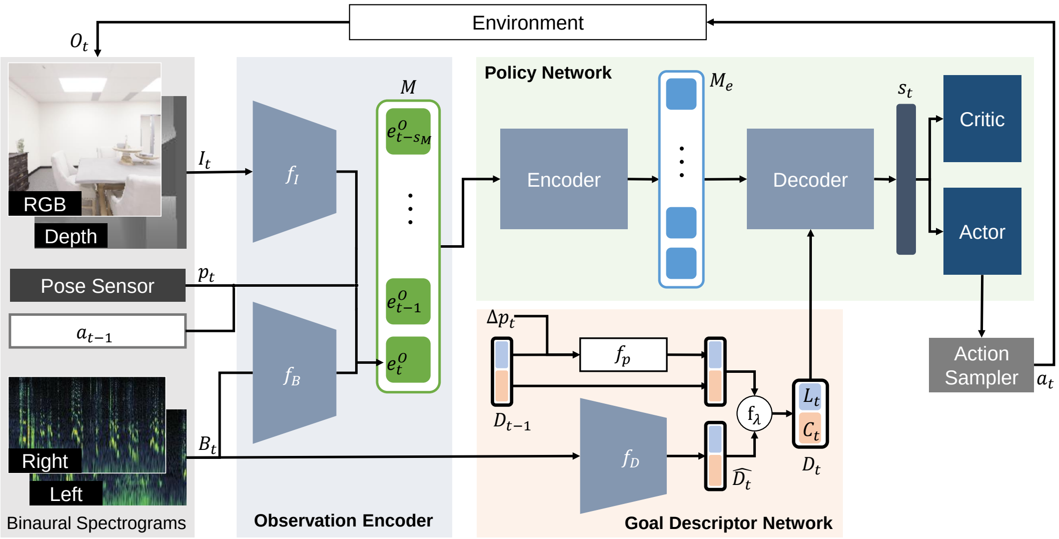


Figure 2: In our model, the agent first encodes input observations and stores their features in memory M. Then it uses the acoustic cues to dynamically infer and update a *goal descriptor* D_t of the target object, which contains both location L_t and object category C_t information about the goal. By conditioning the agent's scene memory on the goal descriptor, the learned state representation s_t preserves information most relevant to the goal. Our transformer-based policy network attends to the encoded observations in M with self-attention to reason about the 3D environment seen so far, and it attends to M_e with D_t to capture possible associations between the hypothesized goal and the visual and acoustic observations to predict the state s_t. Then, s_t is fed to an actor-critic network, which predicts the next action a_t. The agent receives its reward from the environment based on how close to the goal it moves and whether it succeeds in reaching it.

various temporal distances from the current step to locate the goal efficiently. Furthermore, our model learns to capture the goal information from acoustic events in an explicit descriptor and uses it to attend to its memory, thus enabling the agent to discover any spatial and semantic cues that may help it reach the target faster.

Our approach has three main components (Figure 2): 1) an Observation Encoder that maps the egocentric visual and acoustic observations received by the agent at each step to an embedding space; 2) a Goal Descriptor Network that produces a goal descriptor based on the encoded observations; and 3) a Policy Network that given the encoded observations and the predicted goal descriptor, extracts a descriptor-conditioned state representation and outputs the action distribution. Next, we describe each module. We defer CNN architecture details to Sec. 4.4.

4.1. Observation Encoder

At each time step t, the agent receives an observation $O_t = (I_t, B_t, p_t, a_{t-1})$, where I is the egocentric visual observation consisting of an RGB and depth image; B is the received binaural audio waveform represented as a two-channel spectrogram; p is the agent pose defined by its location and orientation (x, y, θ) with respect to its starting pose p_0 in the current episode; and a_{t-1} is the action taken at the previous time step.

Our model encodes each visual and audio observation with a CNN, $e^I_t = f_I(I_t)$ and $e^B_t = f_B(B_t)$. Then, the observation O_t encoding is $e^O_t = [e^I_t, e^B_t, p_t, a_{t-1}]$ where [...] is a concatenation operation. The model stores the encoding of the observations up to time t in memory $M = \{e^O_i : i = \max\{0, t - s_M\}, \ldots, t\}$ (see Figure 2 second column), where s_M is the memory size.

4.2. Goal Descriptor Network

As described in Sec. 3, the agent does not receive direct information about the goal; rather, it needs to rely solely on its observations to set its own target. Audio carries rich cues about the target—not only its relative direction and distance from the agent, but also the type of object that may have produced the acoustic event. Hence, we leverage the acoustic signal to automatically predict the goal properties, namely its location (spatial) and object category (semantics). Both properties are crucial for successful navigation. The estimated goal location gives the agent an idea of where to find the goal. However, since the acoustic event may be short-lived, and the estimate may be wrong, the agent cannot solely rely on this initial estimate. Our model thus aims to also leverage the goal semantics in terms of both the object's likely appearance and the scene's visual context.

The goal descriptor network is a CNN f_D such that $\hat{D}_t = f_D(B_t)$, where $\hat{D}_t$ is the step-wise estimate of the descriptor and consists of two parts: the current estimate of the goal location $\hat{L}_t = (\Delta x, \Delta y)$ relative to the agent's current pose p_t, and its predicted object label $\hat{C}_t$. To reduce the impact of noise from a single prediction, the agent aggregates the current estimate with the previous goal descriptor $D_t = f_\lambda(\hat{D}_t, D_{t-1}, \Delta p_t) = (1-\lambda)\hat{D}_t + \lambda f_p(D_{t-1}, \Delta p_t)$, where $f_p(\cdot)$ transforms the previous goal location $\hat{L}_{t-1}$ based on the last pose change Δp_t (the goal label is unaffected by this transformation), and λ is the weighting factor, which is set to 0.5 based on validation. When sound stops (i.e., the sound intensity becomes zero), the agent maintains its latest estimate D_t by simply transforming the previous descriptor based on the pose change Δp_t to obtain the current descriptor $D_t = f_p(D_{t-1}, \Delta p_t)$.

4.3. Policy Network

Our reinforcement learning policy network is based on a transformer architecture, which has been shown to be effective in addressing the long-term dependencies that RNNs fail to capture [37, 15]. Using the memory M collected so far in the episode, the transformer proceeds by encoding these observation embeddings with a self-attention mechanism to capture any possible relations among the inputs, yielding the encoded memory $M_e = \text{Encoder}(M)$. Then, using the predicted goal descriptor D_t, a decoder network attends to all cells in the encoded memory M_e to calculate the state representation $s_t = \text{Decoder}(M_e, D_t)$. An actor-critic network uses s_t to predict the action distribution and value of the state. The actor and the critic are each modelled by single linear layer neural network. Finally, an action sampler samples the next action a_t from this action distribution, determining the agent's next motion in the 3D scene.

4.4. Training

To train the goal descriptor network, we generate pairs of ground truth locations and categories from the simulator for the array of training sounds, and train the prediction network in a supervised fashion. For the category prediction portion, we find off-policy training gives good accuracy; hence we pre-train the classifier on 3.5M collected spectrogram-category pairs at a variety of positions in the training environments and freeze it during policy training. In contrast, location prediction is learned better on-policy. We train the L_t predictor on-policy, which brings it the benefit of matching the training data distribution with policy behavior, leading to higher accuracy (see Supp.). We use the same experience collected for policy training to train the location predictor and update them at the same frequency. We use the mean squared error loss for the location predictor and the cross entropy loss for the goal object label predictor.

For policy training, we follow a two-stage training paradigm (as shown to be effective for transformer-based models [15]) using decentralized distributed proximal policy optimization (DD-PPO) [38]. In the first stage, we set the memory size $s_M = 1$ (the most recent observation) to train the observation encoder without attention. Then, in the second stage, we freeze the observation encoder and train the rest of the model with the full memory size. In both stages, the loss consists of a value network loss to reduce the error of state-value prediction, a policy network loss to produce better action distributions, and an entropy loss to encourage exploration. We refer readers to PPO [35] for more details. To train the policy, we reward the agent with $+10$ if it reaches the goal successfully and issue an intermediate reward of $+1$ for reducing the geodesic distance to the goal, and an equivalent penalty for increasing it. We also issue a time penalty of -0.01 per time step to encourage efficiency.

To avoid sampling easy episodes (e.g., short or straight-line paths), we require the geodesic distance from the starting pose to the goal to be greater than 4 m and the ratio of Euclidean distance to geodesic distance to be greater than 1.1. We collect 0.5M/500/1000 episodes for train/val/test splits for all 85 Matterport3D SoundSpaces environments.

We train our model with Adam [24] with a learning rate of 2.5×10^{-4} for the policy network and a learning rate of 1×10^{-3} for the descriptor network. We roll out policies for 150 steps, update them with collected experiences for two epochs, and repeat this procedure until convergence. We train all methods, both ours and the baselines, for 300M steps for them to fully converge.

At every time step, the agent receives a binaural audio clip of 1s, which is represented as two 65×26 spectrograms. The audio is computed by convolving the appropriate impulse response from SoundSpaces with the target waveform, thereby generating the sound the agent would hear in that environment at that position relative to the source. The RGB and depth images are center cropped to 64×64. Both the observation encoder CNNs f_B and f_I and the descriptor networks f_D use a simplified ResNet-18 [23] that is trained from scratch. For the transformer model, we use one encoder layer and one decoder layer, which employ multi-attention with 8 heads. The hidden state size is 256 and the memory size s_M is set to be 150, matching the frequency of policy updates.

5. Experiments

Baselines. We compare our model to the following baselines and existing work:

1. **Random**: A random baseline that uniformly samples one of three actions and executes *Stop* automatically when it reaches the goal (perfect stopping).
2. **ObjectGoal RL**: An end-to-end RL policy with a GRU

	Heard Sounds					*Unheard Sounds*				
	Success ↑	SPL ↑	SNA ↑	DTG ↓	SWS ↑	Success ↑	SPL ↑	SNA ↑	DTG ↓	SWS ↑
Random	1.4	3.5	1.2	17.0	1.4	1.4	3.5	1.2	17.0	1.4
ObjectGoal RL	0.9	0.5	0.4	16.7	0.7	0.9	0.5	0.4	16.7	0.7
Chen et al. [11]	21.6	15.1	12.1	11.2	10.7	18.0	13.4	12.9	12.9	6.9
Gan et al. [19]	29.3	23.7	23.0	11.3	14.4	15.9	12.3	11.6	12.7	8.0
AV-WaN [12]	20.9	16.8	16.2	10.3	8.3	17.2	13.2	12.7	11.0	6.9
SMT [15] + Audio	22.0	16.8	16.0	12.4	8.7	16.7	11.9	10.0	12.1	8.5
SAVi (**Ours**)	**33.9**	**24.0**	**18.3**	**8.8**	**21.5**	**24.8**	**17.2**	**13.2**	**9.9**	**14.7**

Table 1: Navigation performance on the SoundSpaces Matterport3D dataset [11]. Our SAVi model has higher success rates and follows a shorter trajectory (higher SPL) to the goal compared to the state-of-the-art. Equipped with its explicit goal descriptor and having learned semantically grounded object sounds from training environments, our model is able to reach the goal more efficiently—even after it stops sounding—at a significantly higher rate than the closest competitor (SWS).

encoder and RGB-D inputs (no audio). It is given the one-hot encoding of the true category label as an additional input to search for the goal object instance. This baseline is widely used in ObjectGoal tasks [22, 9, 29, 7]. We train this method for 800M steps. Details in Supp.

3. **Chen et al. [11]**: An end-to-end RL policy developed for AV navigation that encodes past memory with a GRU RNN and is trained to reach the goal using audio and visual observations.

4. **Gan et al. [19]**: a map-based approach developed for AV navigation that predicts the goal location from binaural spectrograms alone and then navigates with an analytical path planner on an occupancy map that it progressively builds by projecting depth images. It uses a separate audio classifier to stop (see Supp. for details).

5. **AV-WaN [12]**: A SoTA hierarchical RL model for AV navigation that records acoustic observations on the ground plane, predicts waypoints, and uses a path planner to move towards these waypoints using a sequence of navigation actions.

6. **SMT [15] + Audio**: We adapt the scene memory transformer (SMT) model [15] to our task by also encoding the audio observation in its memory. Unlike our model, it does not explicitly predict the goal description and relies only on the cues available in memory to reach the goal. The last observation embedding is used as decoder input to decode M_e and predict the state representation.

All models use the same reward function and inputs. For all methods, there is no actuation noise since audio rendering is only available at grid points (see [11] for details).

Metrics. We evaluate the following navigation metrics: 1) success rate: the fraction of successful episodes; 2) success weighted by inverse path length (SPL): the standard metric [2] that weighs successes by their adherence to the shortest path; 3) success weighted by inverse number of actions (SNA) [12]: this penalizes collisions and in-place rotations by counting number of actions instead of path lengths; 4) average distance to goal (DTG): the agent's distance to the goal when episodes are finished; 5) success when silent (SWS): the fraction of successful episodes when the agent reaches the goal after the end of the acoustic event.

Navigation results. Following the evaluation protocol defined by [11] we evaluate all models in two settings: 1) *heard sounds*—train and test on the same sound 2) *unheard sounds*—train and test on disjoint sounds. In both cases, the test environments are always unseen, hence both require generalization. All results are averaged over 1,000 test episodes.

Table 1 shows the results. We refer to our model as **SAVi**: **S**emantic **A**udio-**V**isual Nav**i**gation model. Our approach outperforms all other models by a large margin on all metrics—with 8.9%, 0.3%, 7.2%, 7.2% absolute gains in SPL on *heard sounds* and 3.8%, 4.9%, 4%, 5.3% SPL gains on *unheard sounds* compared to Chen et al. [11], Gan et al. [19], AV-WaN [12], and SMT [15], respectively. This shows our model leverages audio-visual cues intelligently and navigates to goals more efficiently. Although Gan et al. [19] also leverage external supervision for training the location predictor, which leads to good performance on *heard sounds*, this is not enough to solve the task in the more challenging *unheard sounds* setting. Our method has the advantage of allowing the agent to fully leverage the semantic and spatial cues from audio along with its visual perception to locate the sounding objects. According to previously reported results, AV-WaN represents the state-of-the-art for AudioGoal audio-visual navigation. Our SAVi model's gains over AV-WaN show both 1) the distinct new challenges offered by the semantic AudioGoal task, and 2) our model's design effectively handles them.[2]

[2] While AV-WaN [12] reports large performance improvements over

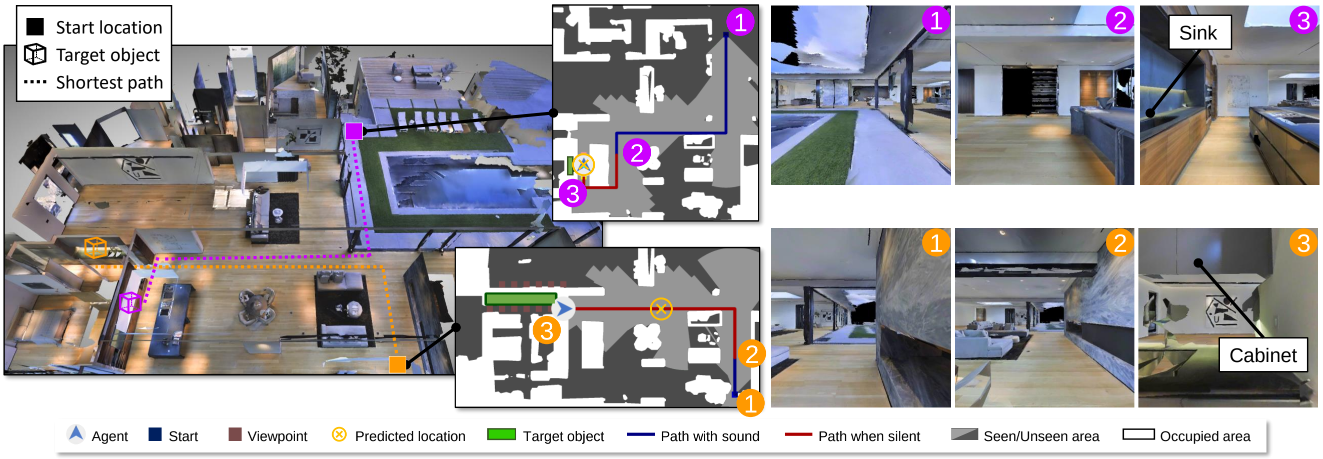


Figure 3: Navigation trajectories for our SAVi model from the test data. In the first episode (top/magenta) the agent hears a water dripping sound and in the second episode (bottom/orange) a sound of opening and closing a door. For each episode, we show three egocentric visual views (right) sampled from the agent's trajectory at the start location ①, when the sound stops ②, and at the end location ③. We see that for the top episode, the acoustic event lasts for two thirds of the trajectory and when the sound stops the agent has an accurate estimate of the object location that helps it find the sounding object (the sink). The second episode (bottom) has a much shorter acoustic event. The agent's estimate of the object location is inaccurate when the sound stops but still helps the agent as a general directional cue. The agent leverages this spatial cue and the semantic cue from its estimate of the object category, a cabinet, to attend to its multimodal memory and find the object in the kitchen and end the episode successfully.

In addition, our model improves the success-when-silent (SWS) metric by a large margin compared to the closest competitor. This emphasizes the advantage of our goal descriptor module. The explicit and persistent descriptor for the goal in our model helps to maintain the agent's focus on the target even after it stops emitting a sound. Although the SMT+Audio [15] model also has access to a large memory pool and can leverage implicit goal information from old observations, lacking our goal descriptor and the accompanying goal-driven attention, it underperforms our model by a sizeable margin.

As expected, Random does poorly on this task due to the challenging complex environments. Although ObjectGoal RL has the goal's ground truth category label as input, it fails in most cases. This shows that knowing the category label by itself is insufficient to succeed in this task; the agent needs to locate the specific instance of that category, which is difficult without the acoustic cues.

Example navigation trajectories. Figure 3 shows test navigation episodes for our SAVi model. The agent uses its acoustic-visual perception and memory along with the spatial and semantic cues from the acoustic event, whether from a long event (water dripping sound) or a short one (opening and closing a door sound), to successfully find the target objects (the sink and the cabinet). Please see the Supp. video for navigation examples.

Distractor sounds. In our tests so far, there is a single acoustic event per episode, whether comprised of a heard or unheard sound (Table 1). Next, we generalize the setting further to include *unheard distractor sounds*—sounds happening simultaneously with the target object. This corresponds to real-world scenarios, for example, where the door slams shut while the AC is humming. For this setting to be well-defined, the agent must know which sound is its target; hence, we input the one-hot encoding of the target object to all models and concatenate it with their state features. For our model, in addition to replacing C_t with this one-hot encoding, we also use it as input to the location prediction network along with B_t. This allows the location prediction network to learn to identify which of the sounds mixed in the input needs to be localized. We use the 102 periodic sounds from SoundSpaces [11] as the set of possible distractor sounds, which are disjoint from the target object sounds curated for this work. We divide these 102 sounds into non-overlapping 73/11/18 splits for train, validation, and test, and hence the distractor sound at test time is unheard. In each episode, we randomly position one distractor sound in the environment at a location different from the goal.

Chen et al. [11] on the standard AudioGoal task, we do not observe similar margins between the two models here. We attribute this to temporal gaps in the memory caused by AV-WaN's waypoint formulation—which are not damaging for constantly sounding targets, but do cause problems for semantic AudioGoal (see Supp. for details).

	Success ↑	SPL ↑	SNA ↑	DTG ↓	SWS ↑
Chen et al. [11]	4.0	2.4	2.0	14.7	2.3
AV-WaN [12]	3.0	2.0	1.8	14.0	1.6
SMT [15]+Audio	4.2	2.9	2.1	14.9	2.8
SAVi (**Ours**)	**11.8**	**7.4**	**5.0**	**13.1**	**8.4**

Table 2: Navigation performance on *unheard sounds* in the presence of unheard distractor sounds.

	Success ↑	SPL ↑	SNA ↑	DTG ↓	SWS ↑
C_t-only	20.5	13.5	11.6	9.8	11.0
L_t-only	23.9	16.2	**13.5**	**9.3**	13.8
w/o aggregation	21.9	14.3	11.1	9.7	13.4
Full model	**24.8**	**17.2**	13.2	9.9	**14.7**

Table 3: Ablation experiment results.

Table 2 shows the results. While the performance of the baselines suffers from the distracting environment noise, our agent is still able to reach a success rate of 11.8% and SPL of 7.4%, which is 7.6% and 4.5% higher than the best-performing baseline. This shows the proposed inferred goal descriptor helps the agent attend to important observations to capture semantic and spatial cues, making our model more robust to the environment noise. That said, the absolute performance declines for all methods in this hard setting. We plan to investigate ways to explicitly separate the "clutter" sounds in future work.

Analyzing the goal descriptor. Next we ablate the two main components in the goal descriptor, location and category, to study their relative impact for the *unheard sounds* experiment in Table 1. Table 3 shows that ablating any component results in a performance drop. L_t has a comparatively larger impact on our model's performance.

Next we analyze the *successful* episodes in the context of L_t and C_t. For 56% of them, our model ends the episode by stopping at its own estimate of the goal location in its descriptor, suggesting that the agent has successfully used its directional sound prediction to guide its movements. On the other hand, for the other 44%, the agent stops at a (correct) location *different* than L_t, suggesting that the agent has relied more on the visual context cues leading to the anticipated object C_t. In fact, if we inject a random category label instead of C_t at the start of the episode, success rates and SPL drop up to 8%. This shows the learned associations between spatial and semantic cues are important for success; breaking these associations with random category labels forces the agent to attend to contradictory cues about the goal in its memory, increasing the chance of failure.

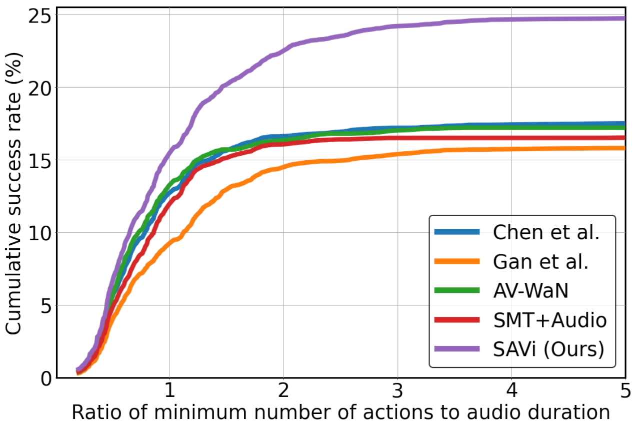


Figure 4: Cumulative success rate vs. silence percentage.

Goal descriptor accuracy and aggregation. The goal descriptor network has two main modules: 1) $f_D(\cdot)$, which produces the current descriptor estimate and 2) an aggregation function $f_\lambda(\cdot)$, which aggregates the current estimate with the previous goal descriptor. Next we evaluate goal prediction accuracy with and without aggregation, as well as how aggregation impacts the navigation performance.

The average location prediction error is 8.1 m and the average category prediction accuracy is 64.5% with aggregation, and 8.4 m, 53.6% without aggregation. Aggregation is important because the source sound is divided into 1s clips for each step, and the characteristics of the sound in some seconds are harder to identify, e.g., the silent moment between pulling and pushing a chest of drawers. Essentially, aggregation stabilizes the goal descriptor prediction. See Supp. for the distribution of prediction accuracy over distance to goal. Navigation performance is affected as well: success rate and SPL drop about 3 points without aggregation ("w/o aggregation" ablation in Table 3).

Robustness to silence duration. Figure 4 analyzes specifically how the models perform after the goal sound stops. We plot the cumulative success rate vs. silence ratio for all models, where the latter is the ratio of the minimum number of actions required to reach the goal to the duration of audio. A point (x, y) on this plot means the fraction of successful episodes with ratios up to x among all episodes is y. When this ratio is greater than 1, no agent can reach the goal before the audio stops. The greater this ratio is, the longer the fraction of silence, and hence the harder the episode. Indeed, we see for all models the success rate accumulates more slowly as the ratio becomes bigger. However, while the success rates of Chen et al. [11], Gan et al. [19], AV-WaN [12], and SMT [15] increase only marginally for ratios greater than 1, our model shows a noticeable increase after the ratios surpass 1 and even 2. This indicates our model is able to cope with long silence to reach goals, thanks to the guidance of our predicted goal descriptor and its attention on the memory.

6. Conclusions

We introduce the task of semantic audio-visual navigation in complex 3D environments. To support this task, we expand an existing audio simulation platform to provide semantically grounded object sounds. We introduce a transformer-based model that learns to predict a goal descriptor capturing both spatial and semantic properties of the target. By encoding the observations conditioned on this goal descriptor, our model learns to associate acoustic events with visual observations. We show that our approach outperforms existing state-of-the-art models. We provide an in-depth analysis of the impact of the goal descriptor and its components, and show that our model is more robust to long silence duration and acoustic distractors. In future work, we are interested in generalizing policies learned in these high quality simulators to test in the real world.

References

[1] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. *Interspeech*, 2018. 2

[2] 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. 6

[3] 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 *CVPR*, 2018. 2

[4] Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. Objectnav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020. 2, 3

[5] Vincent Cartillier, Zhile Ren, Neha Jain, Stefan Lee, Irfan Essa, and Dhruv Batra. Semantic mapnet: Building allocentric semanticmaps and representations from egocentric views. *arXiv preprint arXiv:2010.01191*, 2020. 2

[6] 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. *International Conference on 3D Vision (3DV)*, 2017. MatterPort3D dataset license available at: http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf. 2, 3

[7] Matthew Chang, Arjun Gupta, and Saurabh Gupta. Semantic visual navigation by watching youtube videos. In *NeurIPS*, 2020. 6

[8] Devendra Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *CVPR*, 2020. 1

[9] Devendra Singh Chaplot, Dhiraj Gandhi, Abhinav Gupta, and Ruslan Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *NeurIPS*, 2020. 1, 2, 6

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

[11] 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, 2, 3, 6, 7, 8, 11

[12] Changan Chen, Sagnik Majumder, Ziad Al-Halah, Ruohan Gao, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Learning to set waypoints for audio-visual navigation. *arXiv preprint arXiv:2008.09622*, 2020. 1, 2, 6, 8, 11

[13] Tao Chen, Saurabh Gupta, and Abhinav Gupta. Learning exploration policies for navigation. In *ICLR*, 2019. 1, 2

[14] 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. *SIGGRAPH*, 2018. 2

[15] Kuan Fang, Alexander Toshev, Li Fei-Fei, and Silvio Savarese. Scene memory transformer for embodied agents in long-horizon tasks. In *CVPR*, 2019. 1, 2, 5, 6, 7, 8

[16] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017. 2

[17] Jorge Fuentes-Pacheco, José Ruíz Ascencio, and Juan M. Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43:55–81, 2012. 2

[18] Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J. Guibas. Geometry guided convolutional neural networks for self-supervised video representation learning. In *CVPR*, June 2018. 2

[19] 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, 2, 3, 6, 8, 11

[20] Ruohan Gao and Kristen Grauman. 2.5D visual sound. In *CVPR*, 2019. 2

[21] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *ICCV*, 2019. 2

[22] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In *CVPR*, 2017. 1, 2, 6

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2015. 5

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[25] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018. 2

[26] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, 2020. 2

[27] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, et al. Learning to navigate in complex environments. In *ICLR*, 2017. 2

[28] Dmytro Mishkin, Alexey Dosovitskiy, and Vladlen Koltun. Benchmarking classic and learned navigation in complex 3d environments. *arXiv preprint arXiv:1901.10915*, 2019. 2

[29] Arsalan Mousavian, Alexander Toshev, Marek Fiser, Jana Kosecka, and James Davidson. Visual representations for semantic target driven navigation. *arXiv preprint arXiv:1805.06066*, 2018. 2, 6

[30] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018. 2

[31] Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018. 2

[32] Santhosh K. Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Occupancy anticipation for efficient exploration and navigation. In *ECCV*, 2020. 2

[33] Nikolay Savinov, Alexey Dosovitskiy, and Vladlen Koltun. Semi-parametric topological memory for navigation. In *ICLR*, 2018. 1, 2

[34] 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. 1, 2, 3

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

[36] 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. 3

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 5

[38] 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. 1, 2, 5

[39] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, 2018. 2

[40] Yuke Zhu, Daniel Gordon, Eric Kolve, Dieter Fox, Li Fei-Fei, Abhinav Gupta, Roozbeh Mottaghi, and Ali Farhadi. Visual Semantic Planning using Deep Successor Representations. In *ICCV*, 2017. 1, 2

7. Supplementary Material

In this supplementary material, we provide additional details about:

1. Video (with audio) for qualitative assessment of our agent's performance. Please listen with headphones to hear the spatial sound properly.
2. Baselines' implementation details and analysis (Sec. 5)
3. Distribution of prediction accuracy over distance to goal (Sec. 5)
4. On-policy location predictor training (Sec. 4.4)

7.1. Qualitative Video

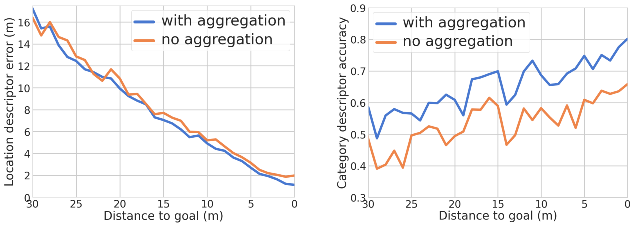
The supplementary video demonstrates the audio simulation platform that we use and shows the comparison between our proposed model and the baselines as well as qualitative analysis for failure cases. Please listen with headphones to hear the binaural audio correctly.

7.2. Baselines' Implementation Details and Analysis

ObjectGoal RL. We implement this baseline by first feeding the RGB-D observations into a CNN (similar CNN to $f_I(\cdot)$ in our model) and concatenating the visual features with a one-hot encoding of the target label. A one-layer GRU memory takes the concatenated feature as input and outputs a state vector of size 512. Similar to our work, this state representation is used by an actor-critic network to predict the action distribution and value of the current state.

Although this baseline has the goal's ground truth label, it fails quite often in reaching the goal. Even if we supply this baseline with a perfect stopping action (i.e., the stopping action is automatically issued when the agent is at the goal location), the success rate in this case is only 1.5%, still much lower than other baselines and our method. This shows knowing the category alone is insufficient to locate the particular object instance and to succeed in this task. The model needs to leverage both visual and acoustic cues to find the goal. This experiment also draws attention to the difference between the proposed semantic AudioGoal and the existing task of ObjectGoal.

Gan et al. [19] We compare to the model from Gan et al. [19] , which trains a goal location predictor in an offline fashion and uses a geometric planner for planning a path to the predicted goal location (shown here for space). We use the same amount of training data for our category predictor to train the goal predictor from [19]. The original model from [19] assumes a continuous periodic acoustic event and it cannot handle sporadic or short acoustic events like those considered in this work. To improve the existing model to perform in this task, we augment its goal location predictor with our update operation f_λ (Sec. 4.2) for transforming the predicted location when the audio goal becomes silent.



(a) (b)

Figure 5: Error analysis of the location predictions and the category predictions in the goal descriptor as a function of the agent's geodesic distance to goal.

AV-WaN [12]. While AV-WaN [12] reports large performance improvements over Chen et al. [11] on the standard AudioGoal task (see [12] for details), we do not observe similar margins between the two models here. Both models, AV-WaN [12] and Chen et al. [11], use RNNs to encode the state representation; however, AV-WaN accumulates the observations at the waypoint prediction level while Chen et al. does so at each step. We speculate that this behavior creates large temporal gaps in the memory for AV-WaN, which makes it harder for the model to adapt to the more challenging task of semantic AudioGoal; the sound may stop at any moment, and the AV-WaN model may not be able to capture the last important acoustic cues in between waypoints. Our model outperforms both since it can keep track of a large set of observations and leverage this information at each step while navigating.

7.3. Distribution of Goal Descriptor Accuracy

Figure 5 shows how the location descriptor error and the category descriptor accuracy change as the agent gets closer to the goal with and without temporal aggregation. The location error is measured as the Euclidean distance between the predicted and the ground truth goal location. The category accuracy is measured by whether the correct goal is predicted or not. We can see that the error of both predictions get lower as the agent gets closer to the goal location and the temporal aggregation leads to higher performance.

7.4. On-policy Location Predictor Training

As noted in the main paper, we find training the location predictor on-policy and online leads to higher accuracy compared to using a pretrained model. If we use an off-policy model in our approach (i.e., similar to the location predictor trained for Gan et al.), this version underperforms our model by 4.8% success rate and 4.7% SPL on the *unheard sound* test split.