DYNAMIC AUDIO-VISUAL NAVIGATION IN CONTINUOUS ENVIRONMENTS WITH MOVING TARGETS

Anonymous authors
Paper under double-blind review

ABSTRACT

In real-world scenarios, sound-emitting entities such as people, pets, and mobile robots may move continuously or emit sound intermittently, and their sounds may be mixed with those from irrelevant sound sources. While recent studies have extended audio-visual navigation from discrete, non-semantic settings to continuous, semantic settings, these advances assume stationary targets. We therefore introduce dynamic audio-visual navigation in continuous environments (DANCE), a new task in which an embodied agent uses audio-visual observations to navigate toward a single sound-emitting target that may be moving or stationary. Navigating toward a moving target is challenging because previously observed spatial cues can quickly become outdated as the target moves. To address this challenge, we propose DyGMap, which couples audio-visual goal estimation with dynamic goal mapping. The former integrates complementary acoustic and visual evidence to estimate the target's semantic category and location, while the latter projects these estimates onto category-specific goal maps and decays outdated or spurious target evidence. Extensive experiments in clean and noisy settings demonstrate that DyGMap improves the success rate by approximately 11.6 percentage points over the strongest evaluated baseline, demonstrating the importance of robust goal reasoning and dynamic goal mapping for efficient navigation.

1 INTRODUCTION

Audio-visual navigation (AVN) requires embodied agents to locate sound-emitting targets in unseen environments using audio-visual observations (Chen et al., 2020). Unlike PointGoal (Chaplot et al., 2020a), ObjectGoal (Batra et al., 2020), and ImageGoal (Krantz et al., 2023) navigation tasks, which specify targets through coordinates, object categories, and reference images, respectively, AVN requires the agent to infer the target's identity and location directly from sensory observations. In this setting, audio provides semantic and spatial cues beyond the agent's current field of view, while vision offers persistent geometric cues and complementary appearance cues when the target is visible. However, integrating these modalities becomes considerably more difficult when the target moves or falls silent, or when its acoustic cues are corrupted by sounds from irrelevant sources.

Early studies consider a single sound source placed at an arbitrary location and restrict the agent to a discrete navigation graph (Gan et al., 2020; Chen et al., 2020; 2021b). Semantic audio-visual navigation (SAVN) (Chen et al., 2021a) grounds limited-duration sounds in visually identifiable objects, while SAVN-CE (Zeng et al., 2026) further extends this semantic formulation to continuous environments with coherent audio-visual observations. Nevertheless, these formulations assume stationary targets. DAV-Nav (Younes et al., 2023) considers moving sound sources but retains the discrete,

Table 1: Comparison of representative audio-visual navigation tasks.

Task	Continuous environment	Intermittent sound	Semantic grounding	Moving target
AVN	✗	✗	✗	✗
SAVN	✗	✓	✓	✗
SAVN-CE	✓	✓	✓	✗
DAV-Nav	✗	✗	✗	✓
DANCE (proposed)	✓	✓	✓	✓

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

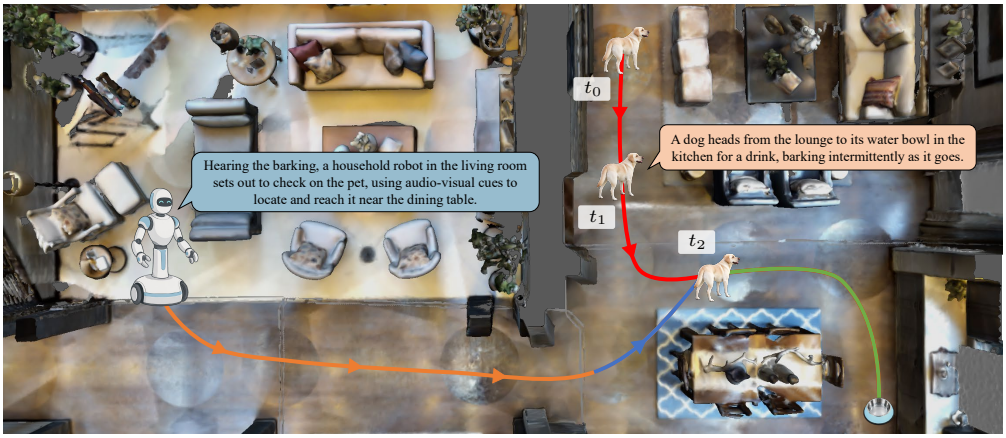


Figure 1: Overview of the DANCE task. The dog's planned and traversed trajectories are shown in green and red, respectively. Orange and blue segments of the robot's trajectory correspond to periods when the dog is barking and when it is silent, respectively.

non-semantic setting and assumes continuous target sound throughout each episode. As summarized in Tab. 1, to our knowledge, no existing formulation jointly considers continuous navigation, intermittent sound emission[1], semantic grounding, and target motion. This leaves an important gap between current benchmarks and realistic navigation toward dynamic acoustic targets.

To address this gap, we introduce ***D**ynamic **A**udio-visual **N**avigation in **C**ontinuous **E**nvironments* (DANCE). As illustrated in Fig. 1, DANCE requires an embodied agent to navigate toward a single semantically grounded acoustic target using audio-visual observations. The target may be either moving or stationary, and its sound emission is independent of its motion state and occurs intermittently for a limited duration. The agent has no episode-specific knowledge of the scene and the target. This setting introduces several challenges beyond stationary-target navigation. As the target moves, previously inferred spatial cues can quickly become outdated, particularly during acoustic silence when audio observations no longer reveal its current location. Meanwhile, existing spatial maps primarily represent static scene structure and lack an explicit mechanism for maintaining time-varying target information (Younes et al., 2023). Dynamic-target navigation therefore requires both robust target estimation from intermittently available audio-visual cues and a spatial representation that preserves scene geometry while continuously adapting to changes in the target state.

Motivated by these challenges, we propose DyGMap, a ***Dy**namic **G**oal **Map**ping* framework that couples audio-visual goal estimation with dynamic goal mapping. DyGMap first employs an audio-visual goal descriptor network to estimate the target's semantic category and position, allowing complementary acoustic and visual evidence to contribute even when they are not simultaneously informative. The estimated goal descriptions are then projected onto category-specific goal maps through a dynamic goal mapping module, which maintains a spatial representation that combines persistent scene geometry with time-varying target information. Explored-region, obstacle, and agent-trajectory maps are retained over time, whereas goal maps are updated using newly observed target evidence and decay when they are no longer refreshed. This mechanism reinforces consistent estimates while gradually suppressing obsolete target locations and transient prediction errors. The dynamic map is further combined with multimodal observations and long-term context to guide navigation, particularly in scenarios involving target motion and acoustic silence.

We adapt, retrain, and evaluate representative AVN methods under the DANCE setting. Experimental results show that DyGMap consistently outperforms the evaluated methods by a substantial margin in both clean and noisy settings. Visualizations of navigation trajectories and dynamic goal maps further illustrate the effectiveness of DyGMap, while ablation studies validate the complementary contributions of audio-visual goal estimation and dynamic goal mapping. Our main contributions are summarized as follows:

[1]In this work, intermittent sound emission means that the target may become temporarily silent during its sound-emitting period; see Fig. 5 for an illustration.

- We introduce DANCE, a new audio-visual navigation task in continuous environments, where an agent uses intermittent acoustic cues and visual observations to navigate toward a single sound-emitting target that may be moving or stationary.
- We propose DyGMap, which combines audio-visual goal estimation with a dynamic map that preserves scene geometry while maintaining time-varying target information.
- We conduct comprehensive experiments in both clean and noisy settings and demonstrate the effectiveness of DyGMap, which improves the success rate by approximately 11.6 percentage points over the strongest baseline in each setting.

2 Related Work

Audio-Visual Navigation. AVN has progressively evolved through three settings: discrete non-semantic (Gan et al., 2020; Chen et al., 2020), discrete semantic (Chen et al., 2021a; Liu et al., 2024; Shi et al., 2025), and continuous semantic (Zeng et al., 2026). Despite this progression, most studies assume that the navigation target is stationary. Only a few studies consider moving sound sources, and these are limited to grid-based, non-semantic environments. Yu et al. (2022b) introduce an adversarial sound source whose position, volume, and sound category vary over time to interfere with navigation, while the navigation target itself remains stationary. Younes et al. (2023) investigate navigation toward a moving sound source, where the target destination is randomly sampled from traversable grid locations. Related studies (Yu et al., 2022a; He et al., 2026) adopt the same moving-target navigation setting. To our knowledge, DANCE is the first AVN setting that jointly considers continuous navigation, intermittent sound emission, semantic grounding, and target motion.

Audio-Visual Sound Event Localization and Detection. Sound event localization and detection (SELD) (Adavanne et al., 2019) provides a unified formulation for detecting sound events and estimating their spatial locations (Shimada et al., 2021; 2022). Audio-only approaches exploit multi-channel acoustic signals (He et al., 2021; Shul & Choi, 2024; Hu et al., 2025), while audio-visual approaches additionally incorporate visual information (Berghi et al., 2024; Hong et al., 2025). Existing audio-visual SELD studies typically evaluate perception using pre-recorded, synchronized audio-visual datasets, such as STARSS23 (Shimada et al., 2023). Embodied navigation introduces additional challenges: observations depend on the agent's actions, and target estimates must be updated online from intermittently available cues, as sounds may cease and targets may be occluded or out of view. Such settings have received limited attention in audio-visual SELD research.

Spatial Mapping. Organizing visual observations into persistent spatial maps is crucial for long-horizon navigation. Chaplot et al. (2020a) construct geometric maps encoding obstacles and explored regions, while subsequent studies incorporate scene semantics (Chaplot et al., 2020b; Georgakis et al., 2022; Yu et al., 2023; Zhang et al., 2023). In AVN, Chen et al. (2021b) maintain an acoustic map that aggregates sound intensity at visited locations over time, providing cues about the source's distance and direction. Younes et al. (2023) encode audio spectrograms in a spatial feature space and fuse them with geometric maps. He et al. (2026) introduce a heatmap conditioned on audio-visual observations to represent target locations. In contrast, our approach maintains time-varying target information alongside persistent scene geometry. Specifically, we project target estimates into category-specific goal maps and apply temporal decay to locations without new estimates, reducing the influence of stale spatial evidence and transient prediction errors.

3 Dynamic Audio-Visual Navigation in Continuous Environments

3.1 Task Description

DANCE requires an agent to navigate through an unmapped environment toward a single sound-emitting target that is either moving or stationary with equal probability. The agent is not provided with the target's category, position, trajectory, or motion state. A moving target follows a predefined trajectory until reaching its destination, with potentially sparse and intermittent sound emission that is independent of its motion state. Following SAVN-CE (Zeng et al., 2026), the agent operates with four low-level actions: *MoveForward 0.25 m*, *TurnLeft 15°*, *TurnRight 15°*, and *Stop*. At each time step t, the agent receives a binaural audio segment $\boldsymbol{y}_t$ and an egocentric RGB-D image $\boldsymbol{I}_t$ of resolution 128×128. It also has access to its pose $\boldsymbol{p}_t = [\Delta x_t, \Delta y_t, \Delta \theta_t]$ relative to its initial

Table 2: Statistics and parameter distributions of the DANCE dataset.

Split	Scenes	Target sounds	Target 3D assets	Episodes
train/val/test	59/11/15	100/14/29	301/43/86	1M/1,000/2,000

Parameter	Target		Distractor			Noise
	Onset (s)	Duration (s)	Onset (s)	Duration (s)	SIR (dB)	SNR (dB)
Distribution	$\mathcal{U}(0,5)$	$\mathcal{N}(20,10^2)$	$\mathcal{U}(0,10)$	$\mathcal{N}(10,5^2)$	$\mathcal{U}(0,10)$	$\mathcal{U}(0,20)$

position and orientation. An episode is successful if the agent executes the *Stop* action within 1 m of the target in at most $t_{\max} = 500$ steps. Two settings are considered: a clean setting that isolates the effects of target motion and intermittent sound emission, and a noisy setting that additionally introduces a static acoustic distractor and background noise to interfere with the target sound.

3.2 DANCE Simulator

Built upon Habitat 2.0 (Szot et al., 2021), SoundSpaces 2.0 (Chen et al., 2022), and SAVN-CE (Zeng et al., 2026), the DANCE simulator renders temporally and spatially continuous binaural audio as both the agent and the target move. Audio signals are sampled at 16 kHz, and each simulation time step spans 0.25 s. At time step t, the binaural room impulse response (RIR) from the target at $\boldsymbol{p}_{t,\text{target}}$ to the agent at $\boldsymbol{p}_{t,\text{agent}}$ with orientation $\theta_{t,\text{agent}}$ is expressed as $\boldsymbol{h}_{t,g} = f\left(\text{scene}, \boldsymbol{p}_{t,\text{target}}, \boldsymbol{p}_{t,\text{agent}}, \theta_{t,\text{agent}}\right)$ and has a length of L_t. The left- and right-channel indices are omitted for simplicity. Let $\boldsymbol{s}_{t,g}$ denote the target source signal segment of length L_s. The received waveform $\boldsymbol{y}_{t,g}$ is obtained by convolving $\boldsymbol{s}_{t,g}$ with $\boldsymbol{h}_{t,g}$ and combining the resulting reverberant signal $\boldsymbol{x}_{t,g}$ via overlap-add across time (Diaz-Guerra et al., 2021; Zeng et al., 2026):

$$\boldsymbol{x}_{t,g}[k] = \sum_{m=0}^{L_s-1} \boldsymbol{s}_{t,g}[m]\, \boldsymbol{h}_{t,g}[k-m], \quad \boldsymbol{y}_{t,g}[n] = \sum_{\tau=0}^{t} \boldsymbol{x}_{\tau,g}[(t-\tau)L_s + n], \tag{1}$$

where $k = 0, \ldots, L_t + L_s - 2$ and $n = 0, \ldots, L_s - 1$. The mixed audio observation is given by

$$\boldsymbol{y}_t = \boldsymbol{y}_{t,g} + 10^{\frac{-\text{SIR}}{20}} \boldsymbol{y}_{t,d} + 10^{\frac{-\text{SNR}}{20}} \boldsymbol{s}_{t,n}, \tag{2}$$

where $\boldsymbol{y}_{t,d}$ denotes the audio observation from the distractor, and $\boldsymbol{s}_{t,n}$ denotes the noise source signal.[2] SIR and SNR denote the nominal signal-to-interference ratio and signal-to-noise ratio, which control the amplitudes of the distractor sound and background noise, respectively.

3.3 DANCE Dataset

We instantiate DANCE in photorealistic Matterport3D scenes (Chang et al., 2017) and use *dog*, *cat*, and *parrot* as the target categories. For each category, we collect corresponding source sounds and reference images from public websites.[3] The reference images are then converted into 3D assets using TRELLIS (Xiang et al., 2025). Each episode is defined by a simulated scene, the agent's initial pose, and a target characterized by its initial orientation, start and end positions, motion trajectory, semantic category, 3D asset, source sound, sound onset time, and sound duration. In the noisy setting, the distractor is specified by its position, semantic category, source sound, sound onset time, sound duration, and SIR, while the noise is specified by its source sound and SNR. We adopt the 21 semantic object categories used in SAVi (Chen et al., 2021a) as distractor categories and use sounds from SAVi and BeDAViN (Shi et al., 2025). Background noise samples are drawn from NOISEX-92 (Varga & Steeneken, 1993). For each episode, the target category is sampled with probabilities 0.4, 0.3, and 0.3 for *dog*, *cat*, and *parrot*, respectively. The key parameters of the DANCE dataset are summarized in Tab. 2. To ensure that the agent can feasibly catch the moving target, the target moves at half the speed of the agent. Additional dataset construction details are provided in Appendix C.

4 Method

We propose DyGMap for efficient navigation toward a potentially moving acoustic target. The architecture shown in Fig. 2 integrates four components for target estimation, map maintenance, obser-

[2] The source signals of the target, distractor, and noise are normalized to the same energy level before mixing.

[3] Sounds are from https://freesound.org and images are from https://unsplash.com.

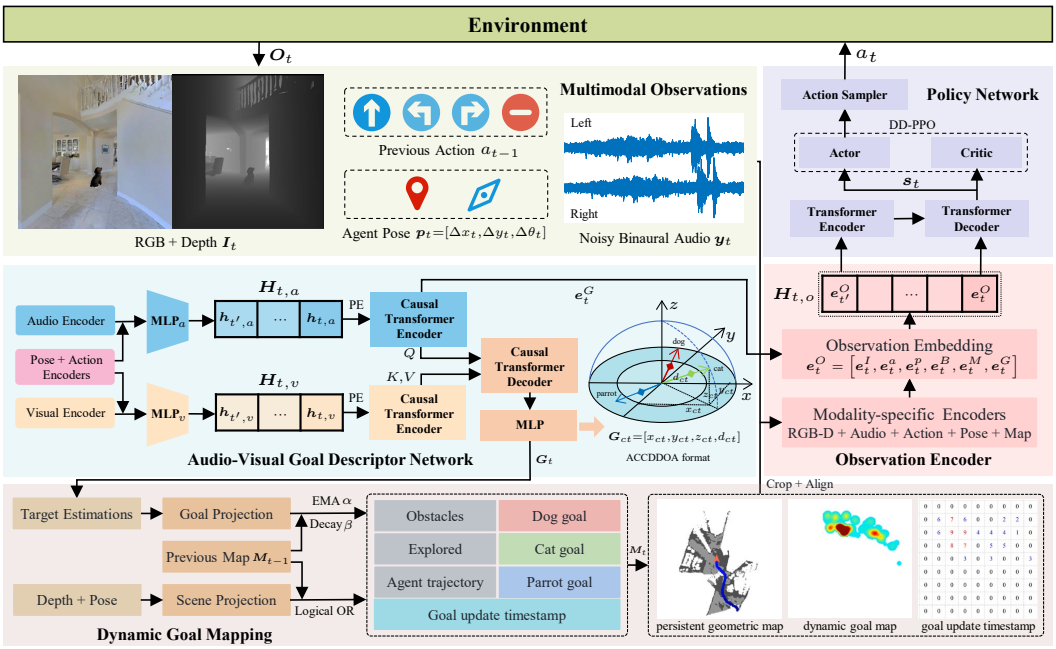


Figure 2: Overview of our proposed navigation framework, DyGMap.

vation encoding, and action prediction. The audio-visual goal descriptor network infers the target's semantic category and location from audio-visual features. These estimates are incorporated into dynamically updated goal maps by the dynamic goal mapping module, which also maintains persistent geometric maps. The observation encoder transforms multimodal observations into embeddings that are accumulated in scene memory throughout the episode. Finally, the policy network draws on the accumulated context in this memory to predict navigation actions.

4.1 Audio-Visual Goal Descriptor Network

Building upon the goal descriptor network (GDN) proposed in MAGNet (Zeng et al., 2026), we leverage visual observations as complementary information to assist goal reasoning. In DANCE, target-related acoustic observations are intermittent and temporally limited, and can be further corrupted by distractor sounds and background noise. Meanwhile, the target may be visually occluded by obstacles due to complex scene geometry. Consequently, during navigation, audio and visual cues may be available simultaneously or individually, or neither may be observable. To handle such situations, we process audio and visual inputs independently and employ cross-modal attention in which audio features query visual features. This design enables the network to exploit complementary cross-modal information for target localization and semantic detection, while episodic memory provides long-term contextual information when current observations are insufficient or unavailable.

The binaural audio signal $\boldsymbol{y}$ is first transformed into complex-valued spectrograms using the short-time Fourier transform (STFT), after which four-channel features are computed to capture semantic and spatial acoustic information (Krause et al., 2024a; Zeng et al., 2026). Details are provided in Appendix D.1. At time step t, the audio features are processed by a convolutional network to obtain the audio embedding $\boldsymbol{e}_t^B$. The previous action a_{t-1} and the agent's pose $\boldsymbol{p}_t$ are encoded into the action embedding $\boldsymbol{e}_t^a$ and pose embedding $\boldsymbol{e}_t^p$ using an embedding layer and a fully connected layer, respectively (Zeng et al., 2026). Meanwhile, RGB and depth observations are independently encoded using ResNet-18 backbones (He et al., 2016), and their features are concatenated to form the visual embedding $\boldsymbol{e}_t^I$. We construct modality-specific representations as follows:

$$\boldsymbol{h}_{t,a} = \mathrm{MLP}_a\left[\boldsymbol{e}_t^B, \boldsymbol{e}_t^p, \boldsymbol{e}_t^a\right], \quad \boldsymbol{h}_{t,v} = \mathrm{MLP}_v\left[\boldsymbol{e}_t^I, \boldsymbol{e}_t^p, \boldsymbol{e}_t^a\right]. \tag{3}$$

The resulting representations are stored in two episodic memories, $\boldsymbol{H}_{t,a}$ and $\boldsymbol{H}_{t,v}$, which retain the most recent N_g embeddings (Zeng et al., 2026). The two memory sequences are first augmented

with positional encodings and independently processed by Transformer encoders, and then fused by a Transformer decoder. The fused representation is passed through an MLP to predict the target's semantic category and position in the activity-coupled Cartesian distance and direction-of-arrival (ACCDDOA) format (Shimada et al., 2021; Krause et al., 2024b; Zeng et al., 2026):

$$\boldsymbol{G}_t = \text{Concat}\left[x_{ct}, y_{ct}, z_{ct}, d_{ct}\right]_{c=0}^{2} = \left[\text{MLP}\left(\text{Decoder}\left(\text{Encoder}_a(\boldsymbol{H}_{t,a}), \text{Encoder}_v(\boldsymbol{H}_{t,v})\right)\right)\right]_t, \quad (4)$$

where $c \in \{0, 1, 2\}$ denotes the category index, $x_{ct}, y_{ct}, z_{ct} \in [-1, 1]$ denote Cartesian components, while d_{ct} denotes the normalized distance. The target direction relative to the agent's current heading is given by $\phi_{ct} = -\text{atan2}(x_{ct}, y_{ct})$, and the activity strength of the predicted category is given by $a_{ct} = \sqrt{x_{ct}^2 + y_{ct}^2 + z_{ct}^2}$. Together, these predictions provide a spatio-semantic description of the target and are further used in Sec. 4.2. Meanwhile, we use the most recent encoded representation from the audio branch, $\boldsymbol{e}_t^G = [\text{Encoder}_a(\boldsymbol{H}_{t,a})]_t$, as the goal embedding for the policy network, which provides a compact contextualized acoustic-memory representation of the target.

4.2 Dynamic Goal Mapping

During each episode, the dynamic goal mapping module integrates scene geometry, the agent trajectory, goal estimates and their update timestamps into a global map $\boldsymbol{M} \in \mathbb{R}^{7 \times L_m \times L_m}$, where L_m denotes the map size. The map encodes obstacles, explored regions, the agent trajectory, three category-specific goal maps, and the most recent update time associated with the goal maps. At time step t, a per-step map $\boldsymbol{m}_t$ is constructed from the depth observation, agent pose, and goal estimates, transformed into the global coordinate frame based on the agent pose, and then aggregated into the global map. The depth observation is geometrically projected onto the map to update the obstacle and explored channels (Chaplot et al., 2020b), while the agent pose is used to update the trajectory channel. The estimated target information is recovered in the global coordinate frame as follows:

$$\begin{aligned} g_{ct,x} &= \Delta x_t + d_{\text{max}} \times d_{ct} \times \cos(\Delta\theta_t + \phi_{ct}), \\ g_{ct,y} &= \Delta y_t - d_{\text{max}} \times d_{ct} \times \sin(\Delta\theta_t + \phi_{ct}), \end{aligned} \quad (5)$$

where d_{max} denotes the distance scaling factor. The resulting coordinates are then discretized into the corresponding goal map as $m_t\left(n, g_{ct,x}, g_{ct,y}\right) = a_{ct}$ if $a_{ct} > \tau$, where $n = c + 3$ denotes the goal map channel index and τ denotes the activity threshold for updating the goal map.

To account for the time-varying goal positions, we introduce a dynamic update mechanism for the goal maps. At each time step t, the global goal map value at location (n, x, y) is updated as follows:

$$M_t(n,x,y) = \begin{cases} m_t(n,x,y), & \text{if } (n,x,y) \text{ receives its first update,} \\ (1-\alpha)M_{t-1}(n,x,y) + \alpha m_t(n,x,y), & \text{if } (n,x,y) \text{ receives a new update,} \\ \beta M_{t-1}(n,x,y), & \text{if } (n,x,y) \text{ was previously updated but not at } t, \end{cases} \quad (6)$$

where α and β denote the smoothing and temporal decay factors, respectively. Newly observed locations are directly initialized, repeated estimates are accumulated using an exponential moving average (EMA), and locations without new estimates gradually decay over time. The goal update timestamp channel is updated as: $M_t\left(6, g_{ct,x}, g_{ct,y}\right) = (t+1)/t_{\text{max}}$ for all valid goal-map updates. This mechanism emphasizes recent target estimates while progressively suppressing stale locations and transient erroneous predictions, which is particularly important when the target moves over time.

4.3 Observation Encoder and Policy Network

The observation encoder combines features extracted by modality-specific encoders into a compact observation embedding. The visual, action, pose, and audio encoders follow the corresponding architectures described in Sec. 4.1. Meanwhile, a local map is cropped from the global map $\boldsymbol{M}_t$ based on the agent's position and aligned with its orientation before being encoded by a convolutional network to obtain the map embedding $\boldsymbol{e}_t^M$. These embeddings are concatenated with the goal embedding $\boldsymbol{e}_t^G$ to form the observation embedding: $\boldsymbol{e}_t^O = \left[\boldsymbol{e}_t^I, \boldsymbol{e}_t^a, \boldsymbol{e}_t^p, \boldsymbol{e}_t^B, \boldsymbol{e}_t^M, \boldsymbol{e}_t^G\right]$. To condition navigation decisions on recent observations, the latest N_o observation embeddings are retained in a scene memory $\boldsymbol{H}_{t,o}$ (Fang et al., 2019; Chen et al., 2021a). The policy network uses a Transformer encoder to capture temporal dependencies within this memory. A Transformer decoder then combines the encoded memory with an MLP projection of the current observation embedding to obtain the latent state: $\boldsymbol{s}_t = \text{Decoder}\left[\text{Encoder}(\boldsymbol{H}_{t,o}), \text{MLP}\left(\boldsymbol{e}_t^O\right)\right]$. An actor head maps $\boldsymbol{s}_t$ to an action distribution, while a critic head estimates the state value. The agent then samples an action a_t from this distribution and executes it to navigate toward the target.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Baselines. We compare our method against representative baselines on the DANCE dataset. Detailed descriptions and implementation settings of these methods are provided in Appendix D. We further introduce four oracle variants of our method by removing the GDN and replacing its predictions with oracle ACCDDOA labels. Oracle ACCDDOA uses the oracle labels as observation inputs only while the target is acoustically active, whereas Oracle ACCDDOA Persist. retains access persistently until the target is found (Zeng et al., 2026). Oracle Goal Map and Oracle Goal Map Persist. additionally use the oracle labels to update the goal map, following the same availability settings as Oracle ACCDDOA and Oracle ACCDDOA Persist., respectively.

Evaluation Metrics. The following metrics are adopted: (1) **SR**: success rate; (2) **SPL**: success weighted by path length (Anderson et al., 2018); (3) **SNA**: success weighted by the number of actions (Chen et al., 2021b); (4) **DTG**: geodesic distance from the agent to the target at episode termination; (5) **SWS**: success when the target is silent (Chen et al., 2021a); (6) **SWM**: success while the target is moving; (7) **DSR**: the fraction of episodes in which the agent stops near the distractor instead of the target (Zeng et al., 2026). Detailed descriptions are provided in Appendix E.1.

Implementation Details. The GDN is trained with persistent ACCDDOA labels to continue updating target estimates after the target stops emitting sound. The policy network is trained with decentralized distributed proximal policy optimization (DD-PPO) (Wijmans et al., 2020) using the standard PPO loss (Schulman et al., 2017). Each learning-based method is trained for up to 240M steps per stage, with early stopping following Zeng et al. (2026). All experiments were completed in approximately two months using 40 A100 GPUs. Additional details are provided in Appendix D.

5.2 EXPERIMENTAL RESULTS

Tabs. 3 and 4 report navigation performance in the clean and noisy settings, respectively, as mean $\pm$ standard deviation over five evaluation runs (see Appendix D). All metrics are presented in percentage terms, with DTG reported in meters. Additional navigation results analyzing the effects of acoustic availability, navigation distance, acoustic interference, motion state and target category are provided in Appendix F.3, while goal reasoning results are reported in Appendix F.5.

Table 3: Navigation performance in the clean setting.

Method	SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑
Random	3.15 $\pm$ 0.00	2.03 $\pm$ 0.00	1.09 $\pm$ 0.00	16.59 $\pm$ 0.00	2.40 $\pm$ 0.00	0.60 $\pm$ 0.00
ObjectGoal	12.45 $\pm$ 0.00	10.28 $\pm$ 0.00	9.83 $\pm$ 0.00	12.04 $\pm$ 0.00	4.20 $\pm$ 0.00	7.15 $\pm$ 0.00
AV-Nav (Chen et al., 2020)	26.68 $\pm$ 0.37	21.32 $\pm$ 0.33	17.81 $\pm$ 0.33	9.61 $\pm$ 0.09	9.44 $\pm$ 0.32	10.81 $\pm$ 0.45
SMT + Audio	35.04 $\pm$ 0.52	29.08 $\pm$ 0.50	25.77 $\pm$ 0.48	8.03 $\pm$ 0.07	13.32 $\pm$ 0.75	13.80 $\pm$ 0.38
SAVi (Chen et al., 2021a)	40.08 $\pm$ 0.82	32.90 $\pm$ 0.58	29.65 $\pm$ 0.48	7.82 $\pm$ 0.11	16.53 $\pm$ 0.90	15.88 $\pm$ 0.52
DAV-Nav (Younes et al., 2023)	43.58 $\pm$ 0.73	34.19 $\pm$ 0.52	29.23 $\pm$ 0.51	8.05 $\pm$ 0.20	**22.71** $\pm$ 0.79	15.64 $\pm$ 0.32
MAGNet (Zeng et al., 2026)	43.43 $\pm$ 0.76	36.22 $\pm$ 0.60	35.18 $\pm$ 0.59	6.59 $\pm$ 0.05	13.79 $\pm$ 0.80	17.99 $\pm$ 0.19
DyGMap (Ours)	**55.20** $\pm$ 0.94	**47.08** $\pm$ 0.77	**44.77** $\pm$ 0.72	**5.19** $\pm$ 0.10	21.94 $\pm$ 0.77	**22.03** $\pm$ 0.61
Oracle ACCDDOA	48.00 $\pm$ 0.00	43.41 $\pm$ 0.00	40.11 $\pm$ 0.00	5.89 $\pm$ 0.00	16.40 $\pm$ 0.00	21.30 $\pm$ 0.00
Oracle Goal Map	63.60 $\pm$ 0.00	55.16 $\pm$ 0.00	52.87 $\pm$ 0.00	4.25 $\pm$ 0.00	28.75 $\pm$ 0.00	24.55 $\pm$ 0.00
Oracle ACCDDOA Persist.	77.70 $\pm$ 0.00	62.99 $\pm$ 0.00	56.35 $\pm$ 0.00	3.06 $\pm$ 0.00	46.45 $\pm$ 0.00	25.65 $\pm$ 0.00
Oracle Goal Map Persist.	79.05 $\pm$ 0.00	68.86 $\pm$ 0.00	58.02 $\pm$ 0.00	3.17 $\pm$ 0.00	46.80 $\pm$ 0.00	26.85 $\pm$ 0.00

Table 4: Navigation performance in the noisy setting.

Method	SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑	DSR↓
AV-Nav (Chen et al., 2020)	23.81 $\pm$ 0.92	16.69 $\pm$ 0.89	13.50 $\pm$ 0.76	11.03 $\pm$ 0.10	13.44 $\pm$ 0.54	7.13 $\pm$ 0.72	1.13 $\pm$ 0.09
SMT + Audio	24.38 $\pm$ 0.82	19.34 $\pm$ 0.48	17.17 $\pm$ 0.43	10.39 $\pm$ 0.08	10.44 $\pm$ 0.52	9.93 $\pm$ 0.45	1.26 $\pm$ 0.16
SAVi (Chen et al., 2021a)	22.47 $\pm$ 0.42	18.44 $\pm$ 0.37	14.69 $\pm$ 0.33	10.67 $\pm$ 0.08	11.12 $\pm$ 0.38	9.01 $\pm$ 0.24	0.95 $\pm$ 0.23
DAV-Nav (Younes et al., 2023)	30.77 $\pm$ 0.27	23.77 $\pm$ 0.24	20.24 $\pm$ 0.23	9.29 $\pm$ 0.16	15.79 $\pm$ 0.53	11.64 $\pm$ 0.32	0.59 $\pm$ 0.20
MAGNet (Zeng et al., 2026)	29.43 $\pm$ 0.73	23.24 $\pm$ 0.45	22.41 $\pm$ 0.48	8.55 $\pm$ 0.10	10.64 $\pm$ 0.31	12.16 $\pm$ 0.14	1.03 $\pm$ 0.16
DyGMap (Ours)	**42.37** $\pm$ 0.85	**32.45** $\pm$ 0.63	**30.97** $\pm$ 0.58	**7.33** $\pm$ 0.11	**18.82** $\pm$ 0.77	**14.60** $\pm$ 0.32	**0.49** $\pm$ 0.07

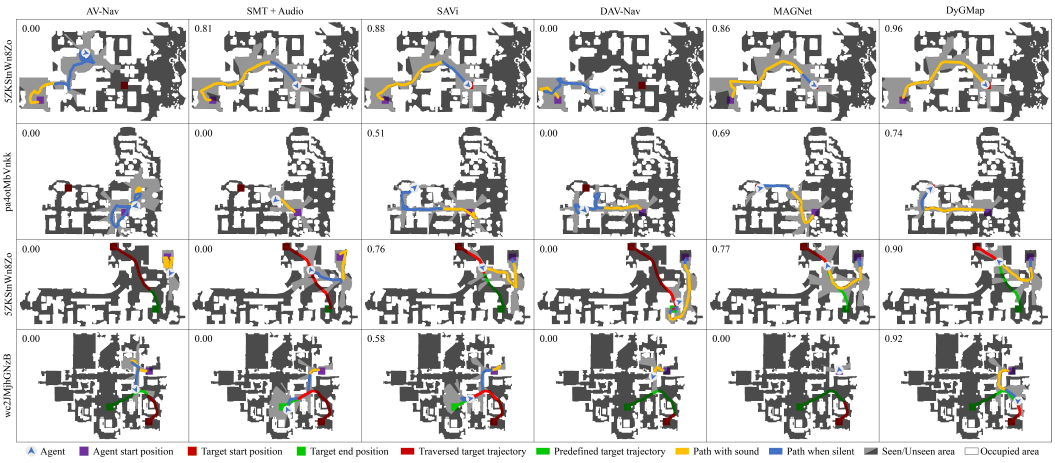


Figure 3: Navigation trajectories of different methods on the DANCE dataset in the clean setting. The number in the corner of each subfigure indicates the corresponding SPL score.

Random achieves only 3.15% SR despite using an oracle *Stop* decision, highlighting the difficulty of locating a target without a learned navigation policy. ObjectGoal improves SR to 12.45% but remains substantially behind methods that incorporate acoustic observations, suggesting that the target category and visual observations alone provide limited guidance for navigation. Among the audio-based baselines, DAV-Nav and MAGNet perform comparably and achieve the strongest overall performance. DAV-Nav is designed for moving-target navigation, whereas MAGNet addresses stationary-target navigation in continuous environments. DyGMap consistently outperforms both methods and achieves substantial improvements across most metrics in the evaluated settings. In particular, its higher SPL and SNA reflect more efficient navigation, while its higher SWM indicates that a larger proportion of episodes end in successful interception while the target is still moving.

The improvement from Oracle ACCDDOA to Oracle Goal Map demonstrates the benefit of explicitly organizing target estimates in a spatial representation. Meanwhile, the substantially stronger performance of the persistent-oracle variants indicates that limited access to up-to-date target information is a major source of performance degradation. Persistent access also narrows the performance gap between the oracle variants, suggesting that differences in representation become less influential when accurate target information is continuously available.

Compared with the clean setting, acoustic distractors and background noise substantially degrade the navigation performance of all audio-based methods by corrupting the semantic and spatial cues associated with the target sound. Despite this interference, DSR remains substantially lower than SR for these methods, indicating that they can generally distinguish the target from the distractor. Although DyGMap still achieves the best overall navigation performance, it also suffers a substantial performance drop, indicating considerable room for improvement under the noisy setting.

5.3 Qualitative Visualization of Navigation and Dynamic Mapping

We visualize representative navigation trajectories of different methods in Fig. 3. The first two and last two rows correspond to episodes with stationary and moving targets, respectively. In the stationary-target examples, DyGMap follows relatively direct routes to the target and can continue navigating after sound emission ceases. In the moving-target examples, DyGMap successfully intercepts the target before it reaches the end of its predefined trajectory. We further visualize the evolution of the constructed maps at representative stages in Fig. 4. During exploration, the geometric map progressively records obstacles and explored regions. Once target evidence becomes available, the target estimates provided by the GDN are projected onto the goal map, which is combined with the geometric map to guide navigation. As the target moves and new estimates arrive, the corresponding goal-map regions are updated with new evidence, while locations that are not refreshed gradually decay. After the target becomes silent, DyGMap continues to approach the target using retained multimodal context and the dynamic goal map, eventually intercepting it successfully.

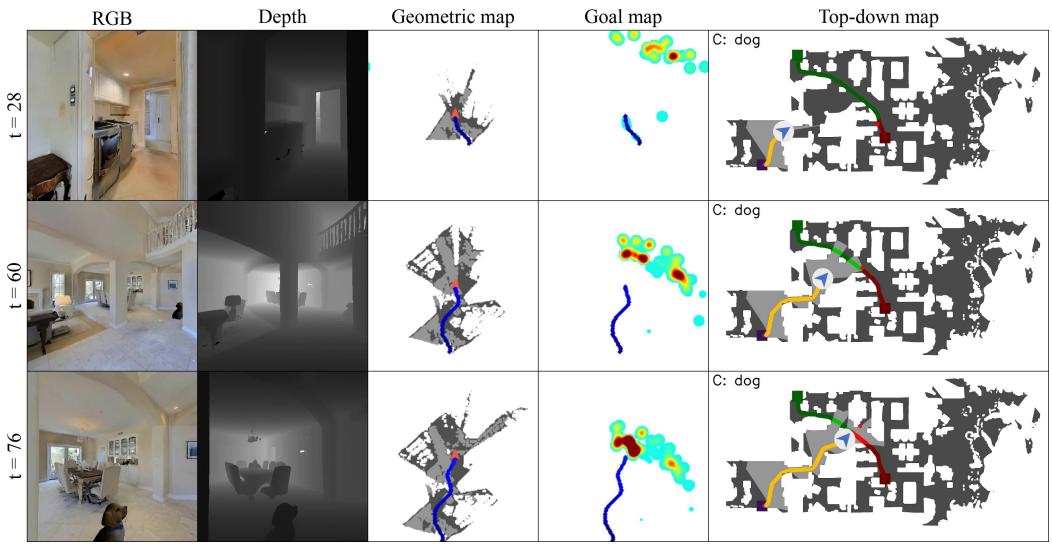


Figure 4: Example maps constructed by DyGMap. The target is a dog, with sound onset and offset at steps 0 and 64, respectively. In the geometric maps, obstacles and explored regions are shown in black and gray, respectively. For visualization, the category-specific goal maps are summed across categories, with brighter regions indicating stronger activity. The upper-left label of each top-down map denotes the estimated target category, and the colored circle marks the estimated target position.

Table 5: Ablation study of the proposed method in the noisy setting.

GDN		Map			SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑	DSR↓
Audio	Vision	Scene	Goal	Dynamic							
✓	✗	✗	✗	✗	31.74 $\pm$ 0.71	27.08 $\pm$ 0.56	23.56 $\pm$ 0.57	8.62 $\pm$ 0.08	11.85 $\pm$ 0.61	12.97 $\pm$ 0.56	0.72 $\pm$ 0.17
✓	✓	✗	✗	✗	33.14 $\pm$ 0.40	28.00 $\pm$ 0.27	25.82 $\pm$ 0.36	8.33 $\pm$ 0.05	11.04 $\pm$ 0.49	13.90 $\pm$ 0.51	0.64 $\pm$ 0.10
✓	✓	✓	✗	✗	38.37 $\pm$ 0.46	29.14 $\pm$ 0.31	27.68 $\pm$ 0.26	7.80 $\pm$ 0.09	17.40 $\pm$ 0.68	13.88 $\pm$ 0.57	**0.43** $\pm$ 0.07
✓	✓	✓	✓	✗	40.06 $\pm$ 0.76	29.11 $\pm$ 0.43	27.87 $\pm$ 0.36	7.40 $\pm$ 0.12	**20.01** $\pm$ 0.63	14.20 $\pm$ 0.59	0.54 $\pm$ 0.12
✓	✗	✓	✓	✓	37.10 $\pm$ 0.62	30.71 $\pm$ 0.61	27.09 $\pm$ 0.57	7.73 $\pm$ 0.07	16.78 $\pm$ 0.40	**14.76** $\pm$ 0.34	0.68 $\pm$ 0.10
✓	✓	✓	✓	✓	**42.37** $\pm$ 0.85	**32.45** $\pm$ 0.63	**30.97** $\pm$ 0.58	**7.33** $\pm$ 0.11	18.82 $\pm$ 0.77	14.60 $\pm$ 0.32	0.49 $\pm$ 0.07

5.4 Ablation Study

We conduct an ablation study in the noisy setting to examine the contributions of visual evidence in the GDN and the proposed mapping components, with results reported in Tab. 5. Incorporating visual evidence into the GDN improves navigation performance under acoustic interference. The goal reasoning results in Appendix F.5 further indicate generally improved goal reasoning performance during sound-emitting periods. Both the geometric map and the goal map provide notable navigation performance gains, while the dynamic update mechanism further improves performance, highlighting the importance of persistent scene representation, explicit goal mapping, and dynamic goal-map updates for navigation. Additional clean-setting ablation results are provided in Appendix F.2.

6 Conclusion

In this paper, we introduce DANCE, a benchmark for audio-visual navigation in continuous environments toward a single semantically grounded acoustic target that may be moving or stationary. We propose DyGMap, which combines audio-visual estimation of the target's semantic category and location with dynamic goal mapping. The framework preserves geometric scene information while updating category-specific goal maps and decaying outdated target evidence, reducing residual traces from obsolete locations and transient prediction errors. Our evaluation of representative navigation methods in both the clean and noisy settings demonstrates the effectiveness and robustness of DyGMap in navigating toward stationary and moving targets. Future work will explore explicit target motion modeling and extend DyGMap to real-world deployment.

AI USE STATEMENT

In this work, we used TRELLIS (Xiang et al., 2025) to generate 3D target assets from reference images for the DANCE dataset. We also used other generative AI tools to discuss the proposed ideas and conceptual frameworks, provide methodological feedback, assist with translation, and interpret results. The research ideas, methodological decisions, and overall study design were developed and determined by the authors. We have not used generative AI tools to implement the core methods of this work, clean or reformat datasets, or support qualitative or thematic data analysis. The use of generative AI tools to formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, or propose or refine hypotheses is not applicable to this work. We also used generative AI tools to generate the robot illustration in Fig. 1, suggest experimental parameters, create or edit visualization code, search and analyze existing literature, and improve the readability of the manuscript. All AI-assisted content was carefully reviewed, verified, and revised by the authors as appropriate. We take full responsibility for the final content of this work, including all text, code, and other artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide comprehensive information to facilitate the reproducibility of our work. The task definition is described in Sec. 3.1. The simulator implementation is described in Sec. 3.2, and additional acoustic-field visualization is provided in Appendix B.2. The DANCE dataset is introduced in Sec. 3.3, with its key parameters summarized in Tab. 2. The detailed episode generation procedure and statistics are provided in Appendices C.2 and C.3, respectively. The model architecture is presented in Sec. 4, while the training procedure, implementation details, hyperparameters, and training settings are described in Sec. 5.1 and Appendix D.1. The evaluation metrics are introduced in Sec. 5.1, with their detailed formulations provided in Appendix E.1. The computational resources, training configurations, and training time are reported in Sec. 5.1 and Appendix D. The configurations and implementation details of the compared methods are documented in Appendix D.2. The training steps for each method in the clean and noisy settings are reported in Appendix D.3. We additionally provide examples of source sounds, rendered binaural audio, generated 3D assets, and DANCE episodes in the supplementary materials. The source code, DANCE dataset, and pretrained models will be publicly released upon publication.

REFERENCES

Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen. Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. *IEEE J. Sel. Topics Signal Process.*, 13(1):34–48, Mar. 2019.

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.

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.

Davide Berghi, Peipei Wu, Jinzheng Zhao, Wenwu Wang, and Philip JB Jackson. Fusion of audio and visual embeddings for sound event localization and detection. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 8816–8820, 2024.

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. In *Proc. Int. Conf. 3D Vis.*, pp. 667–676, 2017.

Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural SLAM. In *Proc. Int. Conf. Learn. Represent.*, pp. 2193–2210, 2020a.

Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Ruslan Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *Proc. Adv. Neural Inform. Process. Syst.*, volume 33, pp. 4247–4258, 2020b.

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 *Proc. Eur. Conf. Comput. Vis.*, pp. 17–36, 2020.

Changan Chen, Ziad Al-Halah, and Kristen Grauman. Semantic audio-visual navigation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 15516–15525, 2021a.

Changan Chen, Sagnik Majumder, Ziad Al-Halah, Ruohan Gao, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Learning to set waypoints for audio-visual navigation. In *Proc. Int. Conf. Learn. Represent.*, pp. 4861–4876, 2021b.

Changan Chen, Carl Schissler, Sanchit Garg, Philip Kobernik, Alexander Clegg, Paul Calamia, Dhruv Batra, Philip Robinson, and Kristen Grauman. SoundSpaces 2.0: A simulation platform for visual-acoustic learning. In *Proc. Adv. Neural Inform. Process. Syst.*, volume 35, pp. 8896–8911, 2022.

David Diaz-Guerra, Antonio Miguel, and Jose R Beltran. gpuRIR: A Python library for room impulse response simulation with GPU acceleration. *Multimed. Tools Appl.*, 80(4):5653–5671, 2021.

Kuan Fang, Alexander Toshev, Li Fei-Fei, and Silvio Savarese. Scene memory transformer for embodied agents in long-horizon tasks. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 538–547, 2019.

Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *Proc. IEEE Int. Conf. Robotics Autom.*, pp. 9701–9707, 2020.

Georgios Georgakis, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, and Kostas Daniilidis. Learning to map for active semantic goal navigation. In *Proc. Int. Conf. Learn. Represent.*, pp. 24559–24577, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 770–778, 2016.

Yuanzheng He, Yuyi Liu, Chenfan Zhang, Dongchen Zhu, and Lei Wang. Learning to chase: Adaptive audio-visual navigation for moving sounds in complex environments. *Pattern Recognit. Lett.*, 206:69–74, 2026.

Yuhang He, Niki Trigoni, and Andrew Markham. SoundDet: Polyphonic moving sound event detection and localization from raw waveform. In *Proc. Int. Conf. Mach. Learn.*, volume 139, pp. 4160–4170, 2021.

Hengyi Hong, Qing Wang, Ruoyu Wei, Mingqi Cai, and Xin Fang. MVANet: Multi-stage video attention network for sound event localization and detection with source distance estimation. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 1–5, 2025.

Jinbo Hu, Yin Cao, Ming Wu, Fang Kang, Feiran Yang, Wenwu Wang, Mark D Plumbley, and Jun Yang. PSELDNets: Pre-trained neural networks on a large-scale synthetic dataset for sound event localization and detection. *IEEE Trans. Audio, Speech, Lang. Process.*, 33:2845–2860, 2025.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *Proc. Int. Conf. Learn. Represent.*, pp. 1–15, 2015.

Jacob Krantz, Theophile Gervet, Karmesh Yadav, Austin Wang, Chris Paxton, Roozbeh Mottaghi, Dhruv Batra, Jitendra Malik, Stefan Lee, and Devendra Singh Chaplot. Navigating to objects specified by images. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 10916–10925, 2023.

Daniel Aleksander Krause, Guillermo García-Barrios, Archontis Politis, and Annamaria Mesaros. Binaural sound source distance estimation and localization for a moving listener. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 32:996–1011, Jan. 2024a.

Daniel Aleksander Krause, Archontis Politis, and Annamaria Mesaros. Sound event detection and localization with distance estimation. In *Proc. Eur. Signal Process. Conf.*, pp. 286–290, 2024b.

Xiulong Liu, Sudipta Paul, Moitreya Chatterjee, and Anoop Cherian. CAVEN: An embodied conversational agent for efficient audio-visual navigation in noisy environments. In *Proc. AAAI Conf. Artif. Intell.*, pp. 3765–3773, 2024.

Archontis Politis, Annamaria Mesaros, Sharath Adavanne, Toni Heittola, and Tuomas Virtanen. Overview and evaluation of sound event localization and detection in DCASE 2019. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 29:684–698, 2021.

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

Zhanbo Shi, Lin Zhang, Linfei Li, and Ying Shen. Towards audio-visual navigation in noisy environments: A large-scale benchmark dataset and an architecture considering multiple sound-sources. In *Proc. AAAI Conf. Artif. Intell.*, pp. 14673–14680, 2025.

Kazuki Shimada, Yuichiro Koyama, Naoya Takahashi, Shusuke Takahashi, and Yuki Mitsufuji. ACCDOA: Activity-coupled Cartesian direction of arrival representation for sound event localization and detection. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 915–919, 2021.

Kazuki Shimada, Yuichiro Koyama, Shusuke Takahashi, Naoya Takahashi, Emiru Tsunoo, and Yuki Mitsufuji. Multi-ACCDOA: Localizing and detecting overlapping sounds from the same class with auxiliary duplicating permutation invariant training. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 316–320, 2022.

Kazuki Shimada, Archontis Politis, Parthasaarathy Sudarsanam, Daniel A. Krause, Kengo Uchida, Sharath Adavanne, Aapo Hakala, Yuichiro Koyama, Naoya Takahashi, Shusuke Takahashi, Tuomas Virtanen, and Yuki Mitsufuji. STARSS23: An audio-visual dataset of spatial recordings of real scenes with spatiotemporal annotations of sound events. In *Proc. Adv. Neural Inform. Process. Syst.*, volume 36, pp. 72931–72957, 2023.

Yusun Shul and Jung-Woo Choi. CST-former: Transformer with channel-spectro-temporal attention for sound event localization and detection. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 8686–8690, 2024.

Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. In *Proc. Adv. Neural Inform. Process. Syst.*, volume 34, pp. 251–266, 2021.

Andrew Varga and Herman JM Steeneken. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech Commun.*, 12(3):247–251, 1993.

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 *Proc. Int. Conf. Learn. Represent.*, pp. 5648–5668, 2020.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3D latents for scalable and versatile 3D generation. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 21469–21480, 2025.

Abdelrahman Younes, Daniel Honerkamp, Tim Welschehold, and Abhinav Valada. Catch me if you hear me: Audio-visual navigation in complex unmapped environments with moving sounds. *IEEE Robot. Autom. Lett.*, 8(2):928–935, Feb. 2023.

Bangguo Yu, Hamidreza Kasaei, and Ming Cao. L3MVN: Leveraging large language models for visual target navigation. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pp. 3554–3560, 2023.

Yinfeng Yu, Lele Cao, Fuchun Sun, Xiaohong Liu, and Liejun Wang. Pay self-attention to audio-visual navigation. In *Proc. Brit. Mach. Vis. Conf.*, pp. 46, 2022a.

Yinfeng Yu, Wenbing Huang, Fuchun Sun, Changan Chen, Yikai Wang, and Xiaohong Liu. Sound adversarial audio-visual navigation. In *Proc. Int. Conf. Learn. Represent.*, pp. 26129–26152, 2022b.

Yichen Zeng, Hebaixu Wang, Meng Liu, Yu Zhou, Chen Gao, Kehan Chen, and Gongping Huang. Semantic audio-visual navigation in continuous environments. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 22369–22379, 2026.

Jiazhao Zhang, Liu Dai, Fanpeng Meng, Qingnan Fan, Xuelin Chen, Kai Xu, and He Wang. 3D-aware object goal navigation via simultaneous exploration and identification. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recog.*, pp. 6672–6682, 2023.

A SUPPLEMENTARY MATERIALS

A brief overview of the supplementary materials is provided below:

1. **Audio examples.** The *audio* folder contains examples of source sounds and the corresponding binaural audio observations, as described in Appendix B.1.
2. **Acoustic-field visualization.** Video 1 in the *videos* folder visualizes the temporal evolution of the acoustic field generated by the DANCE simulator, as described in Appendix B.2.
3. **Dataset examples and 3D assets.** The *dataset* folder contains the reference images and corresponding 3D assets presented in Appendix C.1. It also includes a subset of the DANCE dataset described in Appendix C.3, consisting of 200 episodes from the test split.
4. **Comparison of dynamic and static goal maps.** Videos 2–3 and Videos 4–5 in the *videos* folder compare dynamic and static goal maps in the clean and noisy settings, respectively, illustrating the effect of the dynamic goal-map update mechanism discussed in Appendix F.1.
5. **Navigation recordings.** The *videos* folder contains navigation recordings with binaural audio that illustrate the navigation behavior of our proposed method, DyGMap. In the clean setting, Videos 6–9 correspond to the navigation examples shown in Fig. 3, and Video 10 corresponds to the navigation example shown in Fig. 4. In the noisy setting, Videos 11–16 correspond to navigation examples shown in Fig. 14. Video filenames follow the convention: *videoID_clean/noisy_dynamic/static_sceneID_episodeID_goalCategory_SPL*.mp4.

B DANCE SIMULATOR

B.1 AUDIO EXAMPLES

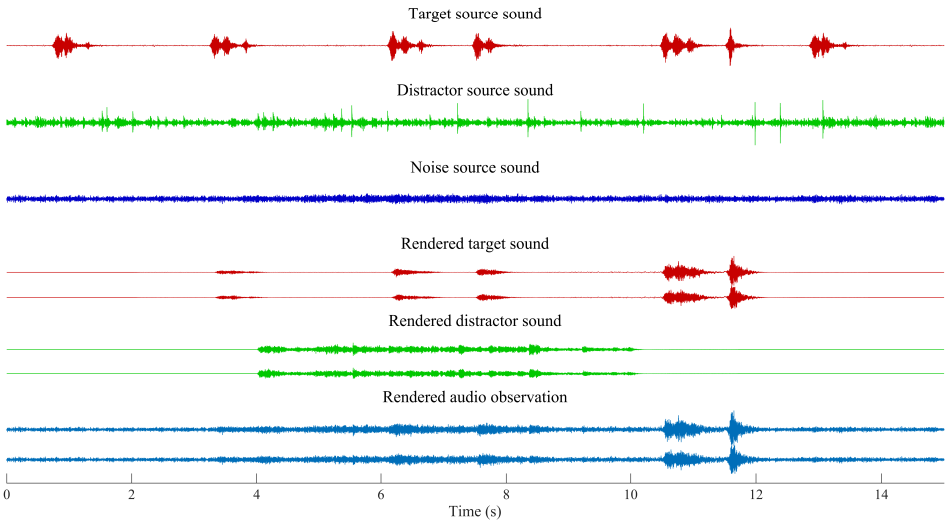


Figure 5: Examples of source sounds and binaural audio rendered by the DANCE simulator.

Fig. 5 presents examples of the source sounds and their corresponding binaural observations rendered by the DANCE simulator. The target sound is active from 2 to 12 seconds, while the distractor sound is active from 4 to 10 seconds. The SIR of the distractor sound is set to 5 dB, and the SNR of the background noise is set to 15 dB.[4] For visualization, the rendered target and distractor waveforms and the mixed binaural observations are amplified by a factor of four. During simulation, the starting indices of the target, distractor, and background-noise source sounds are randomly sampled for each episode to increase acoustic diversity. For this example, all starting indices are fixed at zero

[4]SIR and SNR are used to scale the rendered distractor sound and background noise, respectively. Due to RIR filtering, the resulting signals may not preserve these nominal power ratios.

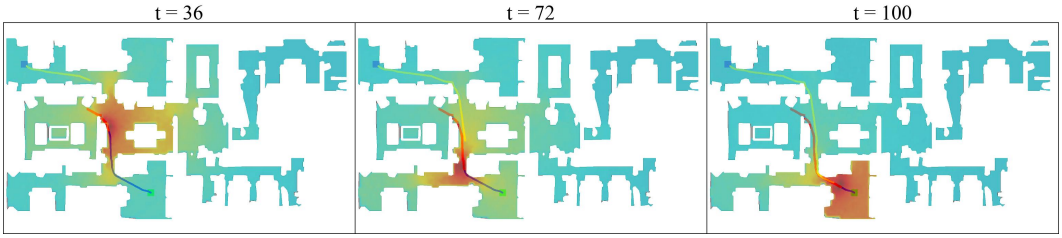


Figure 6: Examples of spatial RIR-energy maps generated by the DANCE simulator.

to temporally align the rendered signals with their corresponding source sounds. The corresponding audio examples are provided in the supplementary material.

B.2 Acoustic Field Visualization

To provide an intuitive visualization of the acoustic dynamics in DANCE, we construct a spatial RIR-energy map as the target and agent move through the scene. We discretize the navigable area of the scene using a grid spacing of 0.2 m and denote the resulting set of navigable positions as $\mathcal{P} = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_N\}$. At each time step t, we place the agent at every position in $\mathcal{P}$ and obtain the corresponding RIRs from the current target position $\boldsymbol{p}_{t,\text{target}}$. Since the agent's orientation is fixed at its initial heading, we omit the orientation argument and express the RIRs at each position $\boldsymbol{p}_i$ as

$$\left[\boldsymbol{h}^l_{t,i}, \boldsymbol{h}^r_{t,i}\right] = f\left(\text{scene}, \boldsymbol{p}_{t,\text{target}}, \boldsymbol{p}_i\right), \tag{7}$$

where $\boldsymbol{h}^l_{t,i}$ and $\boldsymbol{h}^r_{t,i}$ denote the left- and right-channel RIRs, respectively. Both have a length of $L_{t,i}$ and characterize the acoustic transfer functions from the target position $\boldsymbol{p}_{t,\text{target}}$ to position $\boldsymbol{p}_i$. We compute the energy of the RIRs at each navigable position:

$$e_{t,i} = \sum_{k=0}^{L_{t,i}-1} \left(\left|\boldsymbol{h}^l_{t,i}[k]\right|^2 + \left|\boldsymbol{h}^r_{t,i}[k]\right|^2 \right), \tag{8}$$

where $e_{t,i}$ denotes the RIR-energy value at position $\boldsymbol{p}_i$, and the set $\boldsymbol{e}_t = \{e_{t,i}\}_{i=1}^N$ forms the spatial RIR-energy map at time step t. The map visualizes the spatial energy of the acoustic impulse responses, independently of the source waveform.

During visualization, the target moves along its predefined trajectory, while the agent follows the oracle policy toward the moving target. We recompute the RIR-energy distribution map at every time step to visualize how the spatial acoustic field evolves as the target moves. The complete temporal evolution is provided as a video in the supplementary material, and three representative examples are shown in Fig. 6.

C DANCE Dataset Construction

C.1 Source Sounds and Reference Images

Distractor source sounds are drawn from SAVi (Chen et al., 2021a) and BeDAViN (Shi et al., 2025). All 21 distractor categories are represented in each split, with 1,513/269/204 source sounds in the train/val/test splits, respectively. Background noise is drawn from NOISEX-92 (Varga & Steeneken, 1993), which contains 15 noise recordings. Each recording is split temporally at a ratio of 0.7/0.1/0.2 for the train/val/test splits, respectively. DANCE uses dogs, cats, and parrots as target entities, with 143 target source sounds and 430 3D assets in total. Collected reference images are converted into 3D assets using TRELLIS (Xiang et al., 2025). Since each asset is generated from a single reference image, the generated assets may exhibit geometric or appearance artifacts. We therefore manually inspect the generated assets and regenerate those with noticeable artifacts when necessary. Although the assets are not intended to be geometrically exact reconstructions, they provide sufficient visual representations of the target entities for the DANCE task. Example reference images and their corresponding 3D assets are shown in Fig. 7. Target and distractor source recordings, 3D assets, and simulated scenes are disjoint across the training, validation, and test splits. For background noise, non-overlapping temporal segments of each recording are assigned to the three splits.



Figure 7: Example reference images and 3D target assets used in the DANCE dataset.

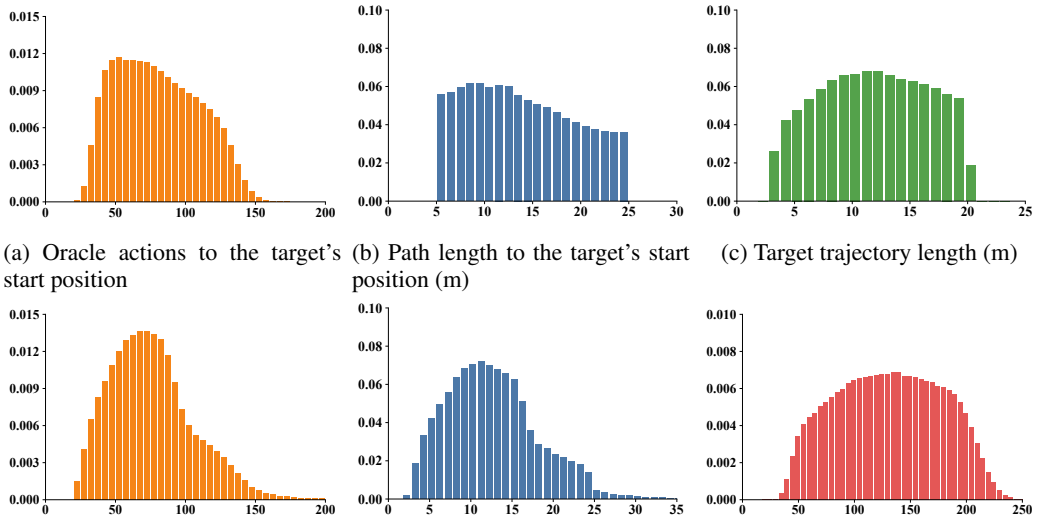
C.2 Episode Generation Procedure

We generate episodes in Matterport3D scenes according to the following procedure:

1. Allocate the number of episodes to each scene in proportion to its navigable area.
2. Randomly sample the start and end positions of the target such that their geodesic distance lies within $[3, 20]$ m. We additionally constrain the ratio of geodesic to Euclidean distance to $[1.1, 2.0]$ to avoid trajectories that are either overly straight or excessively circuitous.
3. Generate the target trajectory between the sampled start and end positions using the agent's action space, including moving forward by 0.25 m and turning left or right by $15°$. We interpolate the target trajectory to double the number of forward-motion waypoints while keeping the turning waypoints unchanged, thereby reducing the target's moving speed to half of the agent's speed.
4. Randomly sample the agent's initial position such that its geodesic distances to both the target start and end positions lie within $[5, 25]$ m. We impose the same path-shape constraint to avoid overly straight or circuitous routes and further require the Euclidean distance between the agent's initial position and every waypoint on the target trajectory to be at least 2 m. The agent's initial heading is uniformly sampled from $[0°, 360°]$.
5. Randomly select an object in the scene belonging to one of the 21 semantic categories used in SAVi and sample a nearby navigable position as the distractor's location. The distractor is required to remain at least 3 m in Euclidean distance from the target trajectory.
6. Generate oracle action sequences and trajectories towards the target under three reference cases: the target located at its start position, at its end position, and moving along its predefined trajectory until the agent and target meet at the theoretical interception point. The theoretical interception point is defined as the point at which the agent and target would meet if the target continued moving along its predefined trajectory while the agent, without knowledge of that trajectory, continuously moved toward the target's current position using oracle actions. The corresponding numbers of actions and path lengths are then computed for evaluation.
7. Sample the acoustic parameters of the target, distractor, and background noise according to the distributions summarized in Tab. 2. The sampled sound durations are then clipped to $[10, 60]$ s for the target and $[5, 30]$ s for the distractor.

C.3 Dataset Statistics

We compute episode-level statistics on the training split and visualize their distributions in Fig. 8. Specifically, we report the number of oracle actions and geodesic distance from the agent's initial position to the target's start position, the target trajectory length, the number of oracle actions and path length to the theoretical interception point when the target moves along its trajectory, and the number of positions along the target trajectory. On the test split, the average numbers of oracle actions for reaching the target's initial position and the theoretical interception point are 83.23 and 81.26, respectively, with corresponding average path lengths of 14.76 m and 13.80 m. The target trajectories have an average length of 12.50 m and contain an average of 136.92 positions.



(a) Oracle actions to the target's start position

(b) Path length to the target's start position (m)

(c) Target trajectory length (m)

(d) Oracle actions to the theoretical interception point

(e) Path length to the theoretical interception point (m)

(f) Number of positions along the target trajectory

Figure 8: Distributions of episode-level statistics on the DANCE training split.

D Implementation and Training Details

All learning-based methods are trained using DD-PPO (Wijmans et al., 2020) with 4 GPUs and 10 parallel workers per GPU. Each policy update uses a 150-step rollout divided into two mini-batches. The reward combines a success reward of +10, a dense progress reward based on the change in geodesic distance to the target, a collision penalty of -0.1, and a time penalty of -0.01 per time step. In our experiments, Random is evaluated without training. ObjectGoal, SMT + Audio, SAVi, and MAGNet follow a two-stage training strategy: the policy is first trained using only the current observation embedding, and then further optimized using a scene memory with a capacity of 150. In contrast, AV-Nav and DAV-Nav use a single-stage training strategy. Each training stage runs for up to 240M environment steps, with early stopping if the validation SPL does not improve for 24M consecutive steps (Zeng et al., 2026).

The checkpoint with the highest validation SPL is selected for evaluation on the test split. Since stochastic acoustic simulation (Chen et al., 2022) may produce slightly different outputs across runs even when the same random seed is used, we evaluate each method five times using the selected checkpoints for learning-based methods and report navigation performance as mean $\pm$ standard deviation. All experiments are conducted on five servers, each equipped with dual AMD EPYC 7742 CPUs and eight NVIDIA A100 GPUs. The DANCE simulator runs at approximately 260 and 200 frames per second (fps) in the clean and noisy settings, respectively. A 240M-step training run takes about 11 days in the clean setting and 14 days in the noisy setting. Overall, the baseline experiments, including training and evaluation, were completed in roughly one month, while developing DyGMap and conducting its training, evaluation, and ablation studies took about one additional month.

D.1 DyGMap

As summarized in Tab. 6, we adopt a three-stage training strategy for DyGMap. The training details of each component are described below.

Table 6: Three-stage training strategy for DyGMap.

Stage	GDN	Map	Observation encoder	Scene memory capacity
Stage 1	Trained	Not used	Trained	$N_o = 1$
Stage 2	Frozen	Used	Trained	$N_o = 1$
Stage 3	Frozen	Used	Frozen	$N_o = 150$

Audio-Visual Goal Descriptor Network. We first transform the binaural audio signals $\boldsymbol{y}$ into complex-valued spectrograms $\boldsymbol{Y}$ using the short-time Fourier transform (STFT) with a window size of 512 samples and a hop size of 160 samples. Then, we compute the mean magnitude spectrogram, interaural phase difference (IPD), and interaural level difference (ILD) as follows (Krause et al., 2024a; Zeng et al., 2026):

$$\begin{aligned} \boldsymbol{Y}_{\text{mean}} &= \left(|\boldsymbol{Y}_l| + |\boldsymbol{Y}_r|\right)/2 + \epsilon, \\ \boldsymbol{Y}_{\text{IPD}} &= \arg\left(\boldsymbol{Y}_l\right) - \arg\left(\boldsymbol{Y}_r\right), \\ \boldsymbol{Y}_{\text{ILD}} &= \left(|\boldsymbol{Y}_l| + \epsilon\right) / \left(|\boldsymbol{Y}_r| + \epsilon\right), \end{aligned} \tag{9}$$

where $\arg(\cdot)$ denotes the phase angle of a complex-valued spectrogram, and subscripts l and r denote the left and right channels, respectively, and $\epsilon = 10^{-8}$ is a small positive constant for numerical stability. The four-channel audio features are constructed as follows:

$$\boldsymbol{B} = [\lg(\boldsymbol{Y}_{\text{mean}}), \sin(\boldsymbol{Y}_{\text{IPD}}), \cos(\boldsymbol{Y}_{\text{IPD}}), \lg(\boldsymbol{Y}_{\text{ILD}})], \tag{10}$$

where $\sin$, $\cos$, and $\lg$ denote the sine, cosine, and base-10 logarithm functions, respectively. Each modality is fused with both the action and pose embeddings using a separate two-layer MLP. The resulting audio and visual representations are independently processed by two-layer Transformer encoders to model their temporal dependencies. A two-layer Transformer decoder then performs cross-modal fusion, followed by a two-layer MLP that predicts the target semantic category and position in the ACCDDOA format.

The GDN is supervised using oracle ACCDDOA labels and trained on complete episodes collected by the navigation policy, following the procedure of MAGNet (Zeng et al., 2026). The oracle ACCDDOA labels used for supervision are persistent until the target is found, requiring the GDN to update target estimates using episodic memory and currently available observations when acoustic observations from the target are unavailable. To preserve temporal continuity, incomplete episodes are retained across iterations and extended with subsequently collected transitions until completion. Episodes shorter than 30 steps are discarded to ensure that the target has emitted sound at least once during the episode. Variable-length episodes are padded to a common length and processed in batches with a batch size of 2, rather than individually with gradient accumulation (Zeng et al., 2026). Padded time steps are masked out during training. Causal masks are applied to the Transformer encoders and the cross-modal Transformer decoder to prevent information leakage from future time steps. The episodic memory capacity is set to $N_g = 128$. The network is optimized using mean squared error (MSE) loss and Adam (Kingma & Ba, 2015) with a learning rate of 1×10^{-3}.

Dynamic Goal Mapping Module. The map construction and update operations are non-learnable. At the beginning of each episode, the global map $\boldsymbol{M}$ is initialized as a zero-filled 1024×1024 grid with a resolution of 0.1 m per cell. The map is anchored at the agent's initial pose, with its initial position at the map center and its initial heading pointing upward (north). The module projects depth observations onto the explored and obstacle maps and the agent's pose onto the trajectory map, and aggregates these per-step maps into the global map using a logical OR operation. The estimated target semantic category and position are mapped onto the category-specific goal maps using Eq. 5 and aggregated into the global map using Eq. 6. We use an activity threshold of $\tau = 0.01$ for goal-map updates and set all goal-map values below 1×10^{-3} to zero. At each time step, a 256×256 local map $\boldsymbol{M}_t^{\text{local}}$ is cropped from the global map, centered at the agent's current position and aligned with its heading. A three-layer convolutional network encodes this local map to obtain the map embedding. In the second training stage, the dynamic goal mapping module is incorporated into the observation pipeline, while the GDN is kept frozen and the policy network is optimized to exploit the resulting map representation. The smoothing factor α and temporal decay factor β are set to 0.5 and 0.95, respectively. The distance scaling factor d_{max} is set to 20 m, and the maximum number of time steps t_{max} in an episode is set to 500.

Policy Network. The policy network consists of a two-layer Transformer followed by separate linear actor and critic heads for predicting the action distribution and state value, respectively. In the second training stage, the scene memory capacity is set to $N_o = 1$, such that the policy network makes decisions based only on the current observation embedding. In the third training stage, the scene memory capacity is increased to $N_o = 150$, allowing the policy network to exploit accumulated temporal context for decision-making. The policy network is optimized using the standard PPO loss (Schulman et al., 2017) with Adam and a learning rate of 2.5×10^{-4}.

D.2 BASELINES

We provide the implementation and training details of the compared methods[5] as follows:

(1) **Random** samples navigation actions according to the action distribution estimated from oracle trajectories in the training split and executes a perfect *Stop* action once near the target. The resulting action probabilities are 66.59% for *MoveForward*, 16.24% for *TurnLeft*, 15.85% for *TurnRight*, and 1.32% for *Stop*. During testing, *Stop* is removed from the sampling distribution and the probabilities of the remaining three actions are renormalized, while a perfect *Stop* is executed automatically when the agent is sufficiently close to the target.

(2) **ObjectGoal** is a visual navigation baseline conditioned on the goal category. We adopt the MAGNet model architecture, provide the oracle target category as a one-hot vector, and train the model on the DANCE dataset.

(3) **AV-Nav** extracts features from visual observations and binaural audio spectrograms and employs a GRU to aggregate temporal information (Chen et al., 2020). We use the official implementation and adapt it to the proposed DANCE setting. Since the original model does not use the agent's relative pose, we retain its original input configuration without introducing additional pose information, unlike other learning-based baselines that inherently incorporate pose observations.

(4) **SMT + Audio** is an audio-based baseline introduced in SAVi (Chen et al., 2021a), built upon the scene memory transformer (SMT) (Fang et al., 2019) to aggregate audio and other sensory observations over time. We construct this baseline following the official SAVi implementation and remove the goal descriptor network.

(5) **SAVi** adopts separate networks to estimate goal semantic category and position, and uses the goal estimates to guide navigation (Chen et al., 2021a). We retrain both the category classifier and position predictor using samples separately from each setting, following the architecture and training procedure of the official implementation. Training samples are collected using a mixed policy, where the agent follows oracle actions with a probability of 0.7 and random actions with a probability of 0.3. For random actions, the probabilities of *MoveForward*, *TurnLeft*, and *TurnRight* are set to 0.6, 0.2, and 0.2, respectively. Each data-collection episode contains 120 steps. If the agent reaches the target before the step limit, it subsequently navigates to its initial position, the target's start position, and the target's end position to continue collecting samples until the episode terminates. We generate up to 1,000 data-collection episodes per scene. Audio observations are clipped to match the simulation step duration (0.25 s, 4,000 samples), resulting in 7.16M, 0.12M, and 0.24M audio segments for the training, validation, and test splits, respectively.

The category classifier and position predictor use only the current audio observation to estimate the target category and position, respectively. The category classifier is trained for 50 epochs using the Adam optimizer, cross-entropy loss, a learning rate of 1×10^{-3}, and a batch size of 1024. The best checkpoint is selected based on validation performance and achieves classification accuracies of 67.32% and 54.29% on the test split in the clean and noisy settings, respectively. We pretrain the position predictor on the same collected dataset using MSE loss and continue updating it during subsequent policy optimization. After pretraining, 23.04% and 12.22% of its predictions fall within 1 m of the target position on the test split in the clean and noisy settings, respectively. During inference, when an audio observation is available at time step t, the estimates are updated as follows:

$$\boldsymbol{c}_t = \lambda \boldsymbol{c}_{t-1} + (1-\lambda)\hat{\boldsymbol{c}}_t, \quad \boldsymbol{l}_t = \lambda \boldsymbol{l}_{t-1} + (1-\lambda)\hat{\boldsymbol{l}}_t, \tag{11}$$

where $\hat{\boldsymbol{c}}_t$ denotes the output of the category classifier, $\hat{\boldsymbol{l}}_t$ denotes the estimated target position obtained by transforming the output of the position predictor using the agent pose, and $\boldsymbol{c}_t$ and $\boldsymbol{l}_t$ are the corresponding accumulated estimates, respectively. Following the official implementation, we set the smoothing factor λ to 0.5. When the audio observation is unavailable, the category estimate remains unchanged, while the target position is updated according to the previous estimate and the agent's pose. Audio availability is determined by checking whether the audio observation is all zeros. This criterion is generally suitable for the clean setting, but becomes less reliable in the noisy setting, where distractor sound may overlap with or occur outside the active period of the target sound, while background noise persists throughout the episode. Consequently, the category

[5]The method proposed by He et al. (2026) is excluded due to the lack of publicly available code and insufficient implementation details for reliable reproduction.

Table 7: Training steps (in millions) for each method in the clean and noisy settings.

Method	Clean		Noisy	
	Steps at each stage	Total	Steps at each stage	Total
ObjectGoal	23.41 + 4.49	27.90	–	–
AV-Nav	122.07	122.07	99.62	99.62
SMT + Audio	67.13 + 109.12	176.25	119.19 + 66.22	185.41
SAVi	83.90 + 136.07	219.97	69.43 + 71.79	141.22
DAV-Nav	174.08	174.08	148.69	148.69
MAGNet	88.92 + 89.92	178.84	103.68 + 9.16	112.84
DyGMap (Ours)	99.22 + 66.36 + 95.04	260.62	96.32 + 70.94 + 55.12	222.38
Oracle ACCDDOA	59.17 + 82.97	142.14	–	–
Oracle Goal Map	81.23 + 75.66	156.89	–	–
Oracle ACCDDOA Persist.	48.09 + 30.20	78.29	–	–
Oracle Goal Map Persist.	52.92 + 64.19	117.11	–	–

classifier and position predictor may continue updating their target estimates even when the audio observation contains only distractor sound or background noise. This behavior may explain the substantial performance drop in the noisy setting reported in Tab. 4 of the main paper.

(6) **DAV-Nav** fuses audio-visual information in a spatial feature space to exploit geometric correlations between local maps and acoustic observations for navigation toward moving sound-emitting targets (Younes et al., 2023). Its geometric map consists of an obstacle channel and an explored-region channel to represent the scene structure. Since the original implementation uses a grid-based approach to construct the geometric map, which is incompatible with our continuous setting, we replace its map construction procedure with the one used in DyGMap. The map resolution is set to 0.1 m. The global and local map sizes are 1200×1200 and 200×200, respectively. We encountered practical difficulties in adapting its original grid-based waypoint selection mechanism to the DANCE setting, and this mechanism is also incompatible with our current training framework. We therefore replace the waypoint-selection head with an action-prediction head that directly predicts the action distribution, consistent with all other learning-based methods evaluated in this work.

(7) **MAGNet** employs a memory-augmented GDN to jointly infer the spatial and semantic properties of a single target in the ACCDDOA format (Shimada et al., 2021; Krause et al., 2024b). It integrates audio observations, self-motion cues, and episodic memory, enabling target estimation even during acoustic silence (Zeng et al., 2026). We adapt the official implementation to the proposed DANCE setting and, as in DyGMap, pad variable-length episodes to a common length during training.

D.3 Training Steps

The training steps recorded at the selected checkpoints for each method in the clean and noisy settings are reported in Tab. 7 for reference. These counts exclude three sources of additional training: the steps required to trigger early stopping after at least 24M steps without improvement in validation SPL, the steps completed before implementation corrections or configuration changes, and the training steps of the existing pretrained models used for initialization.

Adapting existing methods to the new DANCE setting involved correcting implementation errors and revising task-specific configurations during development. Following such changes, training was sometimes restarted from weights learned in earlier runs. For example, ObjectGoal was initially trained for 196.42M steps with the target category provided only during the target's sound-emitting period. We subsequently changed this configuration to provide the ground-truth category throughout the episode and restarted training from previously learned weights.

We also reused pretrained weights across methods and acoustic settings to reduce training time. When a method was built upon another method, its compatible components were initialized using pretrained weights from the base method. Specifically, DAV-Nav was initialized using AV-Nav weights, and SAVi using SMT + Audio weights. The persistent oracle variants were initialized from their corresponding non-persistent variants. Pretraining in the noisy setting was likewise initialized from the corresponding models trained in the clean setting. Consequently, the actual training effort substantially exceeded the reported counts in Tab. 7, which should not be interpreted as complete training budgets or used directly to compare training efficiency across methods.

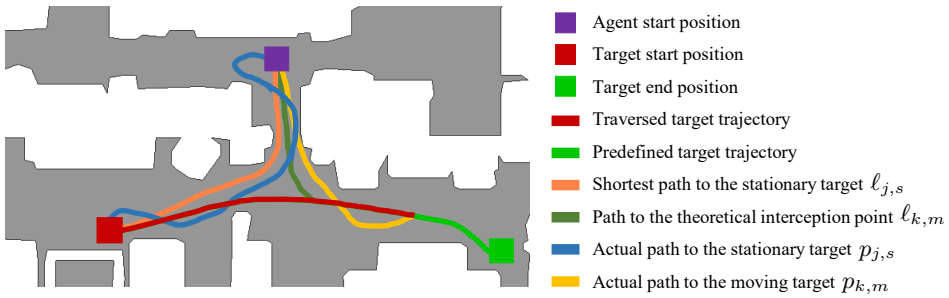


Figure 9: Example paths for stationary and moving targets.

E Evaluation Metrics

E.1 Navigation Metrics

Let N_s and N_m denote the numbers of stationary-target and moving-target episodes in the test split, respectively, with $N = N_s + N_m$ being the total number of episodes. An episode terminates when the agent explicitly executes the *Stop* action within $t_{\max} = 500$ steps; otherwise, it is automatically terminated at the step limit and counted as a failure. For the i-th episode, $d_{i,t}$ denotes the geodesic distance from the agent to the target at termination, and $S_i = \mathbb{I}\big(a^i_{\text{last}} = \textit{Stop}\big) \cdot \mathbb{I}\big(d_{i,t} \le 1\big)$ is the success indicator, where a^i_{last} is the last action executed by the agent in the i-th episode and $\mathbb{I}(\cdot)$ is the indicator function. The navigation metrics used in Sec. 5.1 are defined as follows:

(1) **SR** is the success rate of finding the target:

$$\text{SR} = \frac{1}{N} \sum_{i=1}^{N} S_i. \tag{12}$$

(2) **SPL** is the success weighted by path length (Anderson et al., 2018; Younes et al., 2023):

$$\text{SPL} = \frac{1}{N} \left(\sum_{j=1}^{N_s} \frac{S_{j,s} \cdot \ell_{j,s}}{\max\left(p_{j,s}, \ell_{j,s}\right)} + \sum_{k=1}^{N_m} \frac{S_{k,m} \cdot \ell_{k,m}}{\max\left(p_{k,m}, \ell_{k,m}\right)} \right). \tag{13}$$

As illustrated in Fig. 9, for the j-th stationary-target episode, $\ell_{j,s}$ is the shortest-path length from the agent's start position to the target's start position; for the k-th moving-target episode, $\ell_{k,m}$ is the path length from the agent's start position to the theoretical interception point. The quantities $p_{j,s}$ and $p_{k,m}$ denote the actual path lengths traveled by the agent until episode termination in the corresponding settings. Only successful episodes contribute nonzero terms to SPL, whereas failed episodes contribute zero because their success indicators are zero.

(3) **SNA** is the success weighted by the number of actions (Chen et al., 2021b; Younes et al., 2023):

$$\text{SNA} = \frac{1}{N} \left(\sum_{j=1}^{N_s} \frac{S_{j,s} \cdot \ell^a_{j,s}}{\max\left(p^a_{j,s}, \ell^a_{j,s}\right)} + \sum_{k=1}^{N_m} \frac{S_{k,m} \cdot \ell^a_{k,m}}{\max\left(p^a_{k,m}, \ell^a_{k,m}\right)} \right). \tag{14}$$

Analogous to the path-length definitions in SPL, for the j-th stationary-target episode, $\ell^a_{j,s}$ denotes the minimum number of actions required to reach the target's start position; for the k-th moving-target episode, $\ell^a_{k,m}$ denotes the number of actions required to reach the theoretical interception point. The quantities $p^a_{j,s}$ and $p^a_{k,m}$ denote the actual numbers of actions executed by the agent until episode termination in the corresponding settings.

(4) **DTG** is the mean geodesic distance from the agent to the target at episode termination:

$$\text{DTG} = \frac{1}{N} \sum_{i=1}^{N} d_{i,t}. \tag{15}$$

(5) **SWS** is the fraction of episodes in which the agent successfully reaches the target outside its sound-emitting period (Chen et al., 2021a):

$$\text{SWS} = \frac{1}{N}\sum_{i=1}^{N} S_i \cdot \mathbb{I}\big(\text{target is silent}\big). \tag{16}$$

(6) **SWM** is the fraction of episodes in which the agent successfully reaches the moving target before it arrives at its destination:

$$\text{SWM} = \frac{1}{N}\sum_{i=1}^{N} S_i \cdot \mathbb{I}\big(\text{target is moving}\big). \tag{17}$$

(7) **DSR** is the fraction of episodes in which the agent stops within 1 m of the distractor rather than the target (Zeng et al., 2026):

$$\text{DSR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\big(a^i_{\text{last}} = \textit{Stop}\big) \cdot \mathbb{I}\big(d_{i,d} \le 1\big), \tag{18}$$

where $d_{i,d}$ is the geodesic distance from the agent to the distractor at episode termination in the i-th episode.

Among these metrics, SR, SPL, SNA, and DTG are primary navigation metrics that directly measure overall navigation performance. Higher SR, SPL, and SNA indicate better performance, whereas lower DTG is preferable. In contrast, SWS and SWM characterize the target state at the time of successful navigation and should therefore be interpreted jointly with the primary navigation metrics rather than as strictly monotonic performance measures.

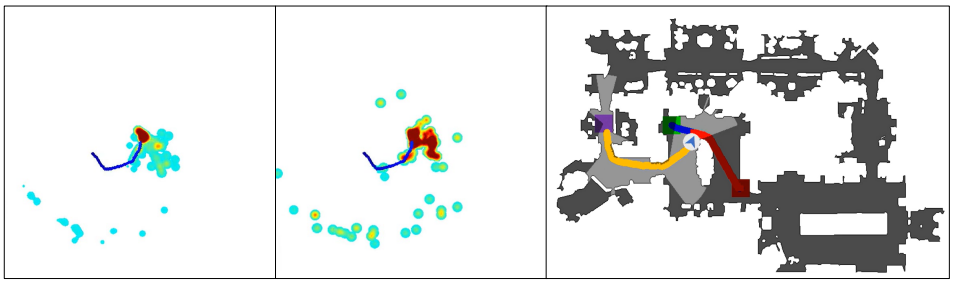
SWS is particularly sensitive to navigation efficiency and the timing of success. For example, an agent following an efficient trajectory may reach the target before it becomes acoustically inactive, in which case the episode contributes to SR but not to SWS. In contrast, another agent taking a longer trajectory or executing redundant actions may reach the same target after it becomes silent, causing the episode to contribute to both SR and SWS. Therefore, a higher SWS does not necessarily indicate better navigation performance and should be interpreted jointly with SPL and SNA. The last two rows of Tab. 8 further illustrate this distinction: DyGMap achieves a lower SWS but a higher SNA than DyGMap w/o vision, despite similar success rates. For a qualitative example, compare the successful trajectories in the first row of Fig. 3.

SWM is introduced in this work to evaluate the agent's ability to reach a target while it is still moving, rather than reaching it only after it approaches its destination and becomes stationary. An episode contributes to SWM only if the agent reaches the target before it reaches the end of its predefined trajectory. Consequently, differences in path length and action efficiency can alter whether a successful episode is counted toward SWM when the interception occurs near the end of the target trajectory. However, since the target is typically in motion over a relatively long temporal interval, moderate differences in navigation efficiency are less likely to change its moving state at the time of interception than its acoustic activity state. Thus, SWM more directly reflects the agent's ability to intercept a moving target.

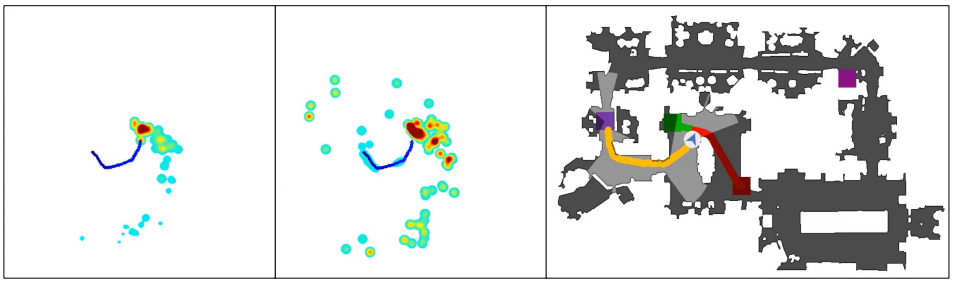
DSR is a diagnostic metric introduced by Zeng et al. (2026) to assess the agent's susceptibility to acoustic distractors. It reflects both whether the agent tends to navigate toward a sounding object and whether it can correctly distinguish the target from the distractor. Notably, an agent that never responds to any sounding object may also achieve a DSR of zero, despite exhibiting poor navigation behavior. Therefore, DSR should be interpreted jointly with the other navigation metrics when diagnosing agent behavior in the presence of acoustic interference. In DANCE, the target and distractor are relatively distinguishable in both visual appearance and acoustic characteristics. Moreover, the nominal SIR is sampled from $[0, 10]$ dB, resulting in a distractor scaling factor no greater than one under the adopted source normalization and mixing procedure. These properties reduce the likelihood of severe distractor confusion and may contribute to generally low DSR values.

E.2 Goal Reasoning Metrics

In the agent coordinate system, the x-, y-, and z-axes point rightward, upward, and backward, respectively, whereas in the SELD coordinate system, they point rightward, forward, and upward.



(a) Clean setting



(b) Noisy setting

Figure 10: Constructed dynamic (left) and static (middle) goal maps, together with the corresponding top-down maps (right).

The azimuth ϕ_{ct} is defined as 0° in the forward direction and increases counter-clockwise within $[-180^\circ, 180^\circ]$. The audio sensor and target are both placed at a height of 1.5 m. Therefore, the relative elevation is always 0°, yielding $z_{ct} = 0$.

We jointly evaluate detection and localization performance using standard SELD metrics (Politis et al., 2021; Krause et al., 2024b): (1) error rate $ER_{\leq 20^\circ}$ and F1-score $F_{\leq 20^\circ}$ with a 20° localization tolerance, and (2) localization error LE_{CD}, localization recall LR_{CD}, and relative distance error RDE for correctly detected events. Following Shimada et al. (2021), target category c is predicted as active at time step t if its predicted activity strength a_{ct} exceeds a threshold of 0.5:

$$\hat{A}_{ct} = \mathbb{I}\left(a_{ct} > 0.5\right), \tag{19}$$

where $\hat{A}_{ct}$ denotes the predicted activity state. Predicting multiple categories as active at the same time step is valid. If no category is predicted as active, the prediction is classified as silence.

F Additional Experimental Results

In this section, we provide additional experimental results for a more comprehensive evaluation of the proposed method. We first compare dynamic and static goal maps in both the clean and noisy settings, followed by additional ablation studies in the clean setting. We then analyze navigation performance with respect to target acoustic availability, navigation distance, acoustic interference, motion state, and target category.

F.1 Comparison of Dynamic and Static Goal Maps

To better understand the importance of dynamic goal mapping for DANCE, we compare the constructed dynamic and static goal maps both in the clean and noisy settings. For a controlled comparison, both maps are constructed using the same oracle action sequence, ensuring that the agent and target follow identical trajectories. The only difference lies in the goal-map update mechanism: the dynamic goal map applies temporal decay to attenuate outdated target estimates, whereas the static goal map does not apply temporal decay to locations without new estimates, i.e., the decay factor β is set to 1. The resulting goal maps are compared in Fig. 10, together with corresponding top-down maps. Videos illustrating the construction process are provided in the supplementary material.

Table 8: Ablation study of DyGMap in the clean setting.

GDN		Map			SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑
Audio	Vision	Scene	Goal	Dynamic						
✓	✗	✗	✗	✗	44.17 ± 0.14	38.75 ± 0.09	34.96 ± 0.18	6.90 ± 0.05	15.34 ± 0.43	17.73 ± 0.37
✓	✓	✗	✗	✗	41.76 ± 0.49	36.07 ± 0.41	34.38 ± 0.50	6.78 ± 0.08	12.95 ± 0.52	18.16 ± 0.49
✓	✓	✓	✗	✗	51.06 ± 0.51	43.54 ± 0.38	41.38 ± 0.42	5.72 ± 0.09	18.83 ± 0.51	20.39 ± 0.57
✓	✓	✓	✓	✗	54.03 ± 0.49	45.66 ± 0.39	43.59 ± 0.32	5.50 ± 0.06	20.95 ± 0.36	21.26 ± 0.38
✓	✗	✓	✓	✓	**55.28** ± 0.46	**47.57** ± 0.35	42.70 ± 0.41	5.31 ± 0.04	**23.75** ± 0.33	20.09 ± 0.50
✓	✓	✓	✓	✓	55.20 ± 0.94	47.08 ± 0.77	**44.77** ± 0.72	**5.19** ± 0.10	21.94 ± 0.77	**22.03** ± 0.61

F.2 Ablation Studies in the Clean Setting

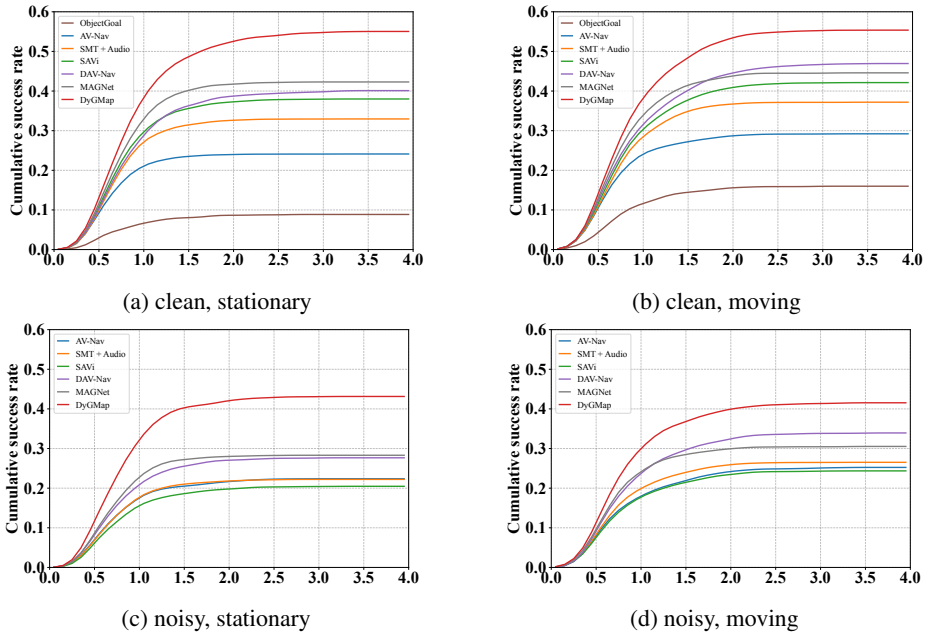
The ablation study in the main paper demonstrates the importance of visual integration in the GDN and dynamic goal mapping for navigation performance in the noisy setting. We further conduct an additional ablation study in the clean setting, with results reported in Tab. 8. For subsequent analyses, we refer to the variants in the fourth and fifth rows as DyGMap w/o dynamic and DyGMap w/o vision, respectively. DyGMap w/o dynamic disables temporal decay by setting $\beta = 1$ and removes the goal update timestamp channel. These components are ablated together because they play complementary roles in accounting for changes in target location over time: temporal decay discounts older target evidence, while the timestamp channel encodes the timing of goal-map updates. DyGMap w/o vision removes visual information from the GDN.

Comparing the results in the clean and noisy settings (Tab. 8 and Tab. 5) shows that incorporating visual evidence into the GDN provides limited benefits in the clean setting but yields more pronounced improvements under acoustic interference. One possible explanation is that, during target sound emission, clean acoustic observations already provide informative semantic and spatial cues, leaving less room for improvement from visual information. Moreover, when the target is occluded or outside the agent's field of view, visual observations may contain little target-specific information, and incorporating such features may introduce noise into target estimation. In the noisy setting, acoustic distractors and background noise reduce the reliability of acoustic cues. Visual evidence, when available, may therefore provide more useful complementary information to support target estimation and navigation. These factors may help explain the condition-dependent benefits of visual integration. Compared with static goal mapping, dynamic goal-map updates improve most metrics in both the clean and noisy settings. This is consistent with the importance of suppressing outdated target information and maintaining an up-to-date goal representation, regardless of the presence of additional acoustic interference.

F.3 Factor-Wise Navigation Analysis

To better understand the DANCE task, we analyze navigation performance with respect to five factors: acoustic availability, navigation distance, acoustic interference, motion state, and target category. We pool the results over five evaluation runs and report the mean navigation performance.

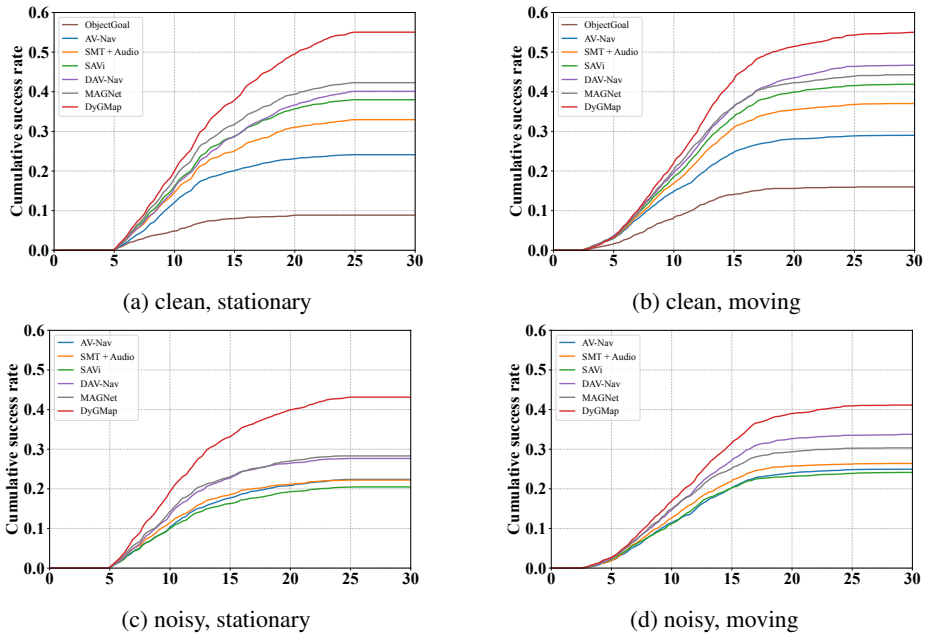
Acoustic Availability. When the target becomes silent, the agent can no longer use direct acoustic cues from the target to update its location estimate and must instead rely on historical information and other available observations. This challenges the agent to maintain useful target estimates rather than merely follow the current sound direction. We characterize the relative availability of target-related acoustic cues using the action ratio, defined as the ratio of the number of steps required to follow the oracle trajectory to the number of steps during which the target is acoustically active (Chen et al., 2021a; Zeng et al., 2026). For ratios greater than 1, a larger value indicates a larger fraction of actions that must be executed without target-related acoustic cues. Fig. 11 presents the cumulative success rates with respect to the action ratio in the clean and noisy settings, separately for stationary and moving targets. Across all four conditions, cumulative success rates increase sharply at low action ratios before leveling off at higher ratios. This indicates that successful episodes are concentrated at smaller ratios, where target-related acoustic cues remain available for a larger portion of the navigation process. Although the baselines exhibit slightly different performance patterns across conditions, DyGMap consistently achieves the highest cumulative success rates across the four conditions, supporting its effectiveness under temporally limited target-related acoustic cues.



(a) clean, stationary (b) clean, moving

(c) noisy, stationary (d) noisy, moving

Figure 11: Cumulative success rates as a function of the action ratio among different methods.

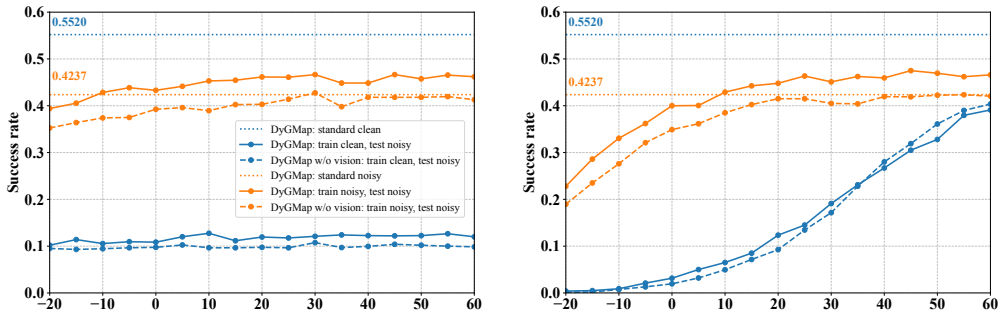


(a) clean, stationary (b) clean, moving

(c) noisy, stationary (d) noisy, moving

Figure 12: Cumulative success rates as a function of geodesic distance among different methods.

Navigation Distance. We use geodesic distance to characterize the spatial difficulty of navigation. Longer geodesic distances generally require more movement and exploration, increasing the need to maintain reliable target information over time. They may also lead to weaker acoustic signals due to propagation attenuation, making distant targets less salient and more susceptible to interference from distractor sounds and background noise. Fig. 12 presents the cumulative success rates with respect to geodesic distance under the four combinations of acoustic setting and target motion state. The curves increase with geodesic distance and gradually plateau at larger distances, indicating that long-distance episodes contribute relatively few additional successes. DyGMap achieves higher cumulative success rates across the evaluated distance range in all four conditions.



(a) SIR (dB) at a fixed SNR of 20 dB

(b) SNR (dB) at a fixed SIR of 10 dB

Figure 13: Effects of distractor SIR and background-noise SNR on the success rates of DyGMap and DyGMap w/o vision trained in the clean and noisy settings.

Acoustic Interference. We conduct additional evaluations to examine how distractor sounds and background noise affect DyGMap and DyGMap w/o vision under different training conditions. Models trained in either the clean or noisy setting are evaluated under controlled acoustic interference. We separately vary the distractor SIR and background-noise SNR from -20 to 60 dB in 5 dB increments, fixing SNR at 20 dB when varying SIR and SIR at 10 dB when varying SNR. Higher SIR and SNR values correspond to weaker distractor sound and background noise, respectively. Fig. 13a shows relatively modest changes in success rate across the tested SIR range, whereas the training conditions have a substantially larger effect. The clean-trained DyGMap achieves a success rate of only about 10% under these noisy evaluations, corresponding to an approximately 80% relative reduction compared with the standard clean setting. In contrast, with SNR fixed at 20 dB, the noisy-trained model remains substantially more robust across the tested SIR range and generally achieves higher success rates than its aggregate performance in the standard noisy setting.

Fig. 13b reveals a stronger dependence on background-noise level. One possible explanation is that background noise affects observed audio signals throughout navigation, whereas the distractor contributes interference only during its sound-emitting period. The success rates of the clean-trained models drop to nearly zero below -10 dB SNR and increase substantially as the SNR rises. The noisy-trained models remain more effective under strong background noise and approach a plateau as the noise weakens. Among the clean-trained models, DyGMap generally outperforms DyGMap w/o vision at lower SNRs, whereas the ordering reverses at higher SNRs, with a crossover at approximately 35 dB. This observation suggests that the benefit of visual integration varies with the reliability of acoustic observations. In contrast, the noisy-trained DyGMap generally outperforms its counterpart without vision across the tested SNR range.

Overall, strong performance in the clean setting does not necessarily translate to robust performance under acoustic interference. Models trained only in the clean setting suffer substantial degradation under noisy conditions, even when the distractor sound and background noise are relatively weak. In contrast, training with acoustic interference substantially improves generalization across the evaluated distractor and background-noise levels. These results highlight the importance of exposing the model to acoustic interference during training and further show that the benefits of visual integration depend on both the training conditions and the level of acoustic interference.

Motion State. The target's motion state imposes different demands on the agent and can influence its navigation behavior. To examine its effect, we conduct new evaluation runs for different methods under stationary-only and moving-only settings, rather than a mixed setting in which the two motion states are sampled with equal probability. According to Eq. 17, SWM is normalized by the total number of episodes N. Since moving-target episodes constitute approximately half of all episodes in the mixed setting, SWM in the moving-only setting is expected to be approximately twice that in the mixed setting, assuming comparable performance on moving-target episodes.

As shown in Tab. 9, all methods perform marginally better in the moving-only setting than in the stationary-only setting. This performance gap may be partly explained by differences in episode difficulty between the stationary-target and moving-target settings, as suggested by the distributions shown in Fig. 8. As reported in Appendix C.3, reaching the theoretical interception point requires

Table 9: Navigation performance with respect to the target's motion state.

Setting	Motion state	Method	SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑
Clean	Stationary	ObjectGoal	9.85	8.32	7.62	13.00	3.35	–
		AV-Nav	25.18	20.14	16.25	10.07	9.98	–
		SMT + Audio	33.15	27.37	23.70	8.54	13.21	–
		SAVi	37.77	31.37	27.38	8.27	15.90	–
		DAV-Nav	39.85	31.62	26.47	8.86	20.09	–
		MAGNet	41.22	35.16	33.51	7.15	12.95	–
		DyGMap w/o dynamic	53.47	46.10	43.10	5.97	22.00	–
		DyGMap (Ours)	**55.57**	**48.01**	**44.93**	**5.60**	**23.33**	–
	Moving	ObjectGoal	15.75	13.33	12.82	11.13	5.20	13.80
		AV-Nav	27.89	22.09	18.95	9.20	10.21	20.09
		SMT + Audio	35.92	29.36	26.60	7.45	13.78	27.39
		SAVi	43.32	35.21	32.35	7.10	18.36	32.33
		DAV-Nav	46.38	36.28	31.82	7.32	**24.62**	30.06
		MAGNet	44.44	36.59	35.93	5.93	14.58	36.17
		DyGMap w/o dynamic	55.28	46.14	45.01	4.97	20.73	43.10
		DyGMap (Ours)	**56.80**	**47.66**	**46.09**	**4.76**	22.47	**44.58**
Noisy	Stationary	AV-Nav	22.18	15.49	12.23	11.47	12.22	–
		SMT + Audio	22.50	18.20	15.62	11.13	9.93	–
		SAVi	20.15	16.41	12.53	11.28	10.51	–
		DAV-Nav	26.87	20.86	17.50	10.21	13.54	–
		MAGNet	27.51	22.26	20.90	9.42	9.16	–
		DyGMap w/o dynamic	39.33	28.61	26.60	7.98	**20.07**	–
		DyGMap (Ours)	**40.90**	**31.66**	**29.54**	**7.95**	17.99	–
	Moving	AV-Nav	25.58	18.23	15.41	10.37	13.55	15.10
		SMT + Audio	27.16	21.52	19.76	9.76	11.94	20.91
		SAVi	25.62	21.10	17.56	10.01	11.25	18.08
		DAV-Nav	34.55	26.68	23.38	8.58	17.57	23.91
		MAGNet	32.08	25.21	24.69	7.77	11.55	25.85
		DyGMap w/o dynamic	40.83	29.56	29.09	**6.82**	**20.27**	28.22
		DyGMap (Ours)	**43.57**	**33.00**	**32.28**	6.86	19.48	**31.09**

an average of 81.26 oracle actions and a path length of 13.80 m, compared with 83.23 actions and 14.76 m for reaching the target's initial position. These lower action counts and shorter paths may make some moving-target episodes easier to complete within the limited target-sound emission window. Target motion may also affect performance by changing the observations available during navigation. For stationary targets, previously acquired location estimates remain spatially valid even when target-related acoustic cues are unavailable. However, if the target is distant, occluded, or poorly localized from its initial observations, the agent may continue navigating toward an inaccurate fixed location without receiving substantially different spatial evidence. In contrast, moving targets require the agent to continually update its location estimate and adapt its navigation trajectory, but target motion can also create additional sensing opportunities. The target may emerge from occlusion, enter the agent's field of view, move closer to the agent, or reach positions from which its sound or appearance provides more informative cues. These additional observation opportunities, together with the target's lower speed relative to the agent, may help explain the stronger performance observed in moving-target episodes. In the moving-only setting, SWM ranges from approximately 59% to 88% of SR across methods, indicating that most successful episodes terminate while the target is still moving. The remaining gap between the two metrics corresponds to successful episodes in which the agent reaches the target after it has arrived at its destination.

Comparing DyGMap with DyGMap w/o dynamic shows that dynamic goal updates improve navigation performance for both stationary and moving targets, with the magnitude of the gains depending on the acoustic setting and target motion state. In the clean setting, the SR gain is larger for stationary targets than for moving targets (2.10 versus 1.52 percentage points), whereas the opposite holds in the noisy setting (1.57 versus 2.74 percentage points). For stationary targets, repeated accurate estimates can refresh the same map region, preserving useful spatial evidence while erroneous locations that no longer receive updates gradually decay. For moving targets, maintaining accurate localization across consecutive time steps can be more challenging. When reliable updates are intermittent, temporal decay may weaken previously accurate location estimates that still provide

Table 10: Navigation performance with respect to the target's semantic category.

Setting	Category	Method	SR↑	SPL↑	SNA↑	DTG↓	SWS↑	SWM↑
Clean	Dog	ObjectGoal	14.00	11.40	10.78	11.76	4.78	7.30
		AV-Nav	25.53	20.28	16.77	9.92	8.76	10.48
		SMT + Audio	34.33	28.30	24.97	8.26	12.37	14.07
		SAVi	40.43	33.07	29.34	8.00	16.79	16.05
		DAV-Nav	43.44	33.86	28.78	7.98	**22.87**	15.98
		MAGNet	43.71	36.30	35.34	6.50	12.97	18.66
		DyGMap (Ours)	**54.78**	**46.84**	**44.54**	**5.24**	21.58	**22.68**
	Cat	ObjectGoal	10.55	8.52	8.18	12.16	4.41	7.09
		AV-Nav	29.04	23.08	19.61	9.23	10.43	12.00
		SMT + Audio	36.31	30.30	26.98	7.75	13.70	15.02
		SAVi	40.63	33.28	30.64	7.52	17.01	17.07
		DAV-Nav	45.80	35.96	30.98	7.76	23.31	17.26
		MAGNet	42.71	35.76	34.55	6.61	13.92	18.14
		DyGMap (Ours)	**55.72**	**47.48**	**45.01**	**5.01**	**23.34**	**23.06**
	Parrot	ObjectGoal	12.29	10.64	10.30	12.33	3.02	6.99
		AV-Nav	25.67	20.83	17.31	9.58	9.34	9.91
		SMT + Audio	34.63	28.86	25.57	8.00	14.37	11.91
		SAVi	38.87	32.17	28.94	7.92	15.54	14.18
		DAV-Nav	41.13	32.59	27.84	8.52	**21.74**	13.16
		MAGNet	43.86	36.65	35.70	6.71	14.93	16.75
		DyGMap (Ours)	**55.24**	**46.99**	**44.85**	**5.33**	20.83	**19.77**
Noisy	Dog	AV-Nav	23.06	16.03	12.89	11.20	12.06	6.94
		SMT + Audio	24.00	18.87	16.76	10.46	10.02	9.67
		SAVi	21.99	17.86	14.21	10.70	10.84	9.43
		DAV-Nav	31.48	24.18	20.53	9.11	16.77	11.41
		MAGNet	28.49	22.11	21.19	8.52	10.67	12.42
		DyGMap (Ours)	**41.96**	**31.51**	**30.01**	**7.26**	**20.12**	**15.45**
	Cat	AV-Nav	24.98	17.47	14.15	10.82	14.52	7.37
		SMT + Audio	23.78	18.88	16.64	10.41	9.92	10.14
		SAVi	22.74	18.88	14.93	10.64	10.93	9.13
		DAV-Nav	29.29	22.49	19.25	9.43	14.30	11.56
		MAGNet	29.92	23.90	23.06	8.57	10.46	11.46
		DyGMap (Ours)	**44.13**	**34.35**	**32.82**	**7.11**	**17.86**	**14.71**
	Parrot	AV-Nav	23.59	16.78	13.69	11.00	14.33	7.15
		SMT + Audio	25.71	20.63	18.46	10.25	11.72	10.09
		SAVi	22.91	18.84	15.16	10.67	11.80	8.20
		DAV-Nav	31.42	24.65	20.95	9.42	16.03	12.10
		MAGNet	30.32	24.23	23.53	8.56	10.81	12.59
		DyGMap (Ours)	**40.91**	**31.64**	**30.28**	**7.71**	**17.92**	**13.12**

useful search cues before a reliable new estimate becomes available. In the clean setting, where historical estimates are relatively reliable, the trade-off between suppressing outdated information and retaining useful spatial context may limit the incremental benefit for moving targets. Under acoustic interference, moving targets pose the combined challenges of localization errors and changes in the true target position: historical estimates may be initially incorrect or become outdated as the target moves. The greater need to suppress both types of unreliable information may explain the larger gains for moving targets in the noisy setting.

Target Category. Different target categories exhibit distinct acoustic and visual characteristics that may affect navigation performance. Variations in the spectral and temporal structure of source sounds can influence target recognition and localization, while differences in appearance and physical size may affect the visual evidence available to the agent. Tab. 10 reports the navigation performance for dog, cat, and parrot targets in both the clean and noisy settings. Performance varies across methods and target categories. ObjectGoal, which uses visual observations without acoustic input, generally achieves its best performance on dog-target episodes. The remaining methods incorporate acoustic observations to support navigation. In the clean setting, AV-Nav, SMT + Audio, SAVi, DAV-Nav, and DyGMap generally achieve slightly better performance on cat-target episodes than on dog- and parrot-target episodes, whereas MAGNet shows comparatively weaker performance on cat-target episodes. In the noisy setting, the relative ordering of target categories becomes less con-

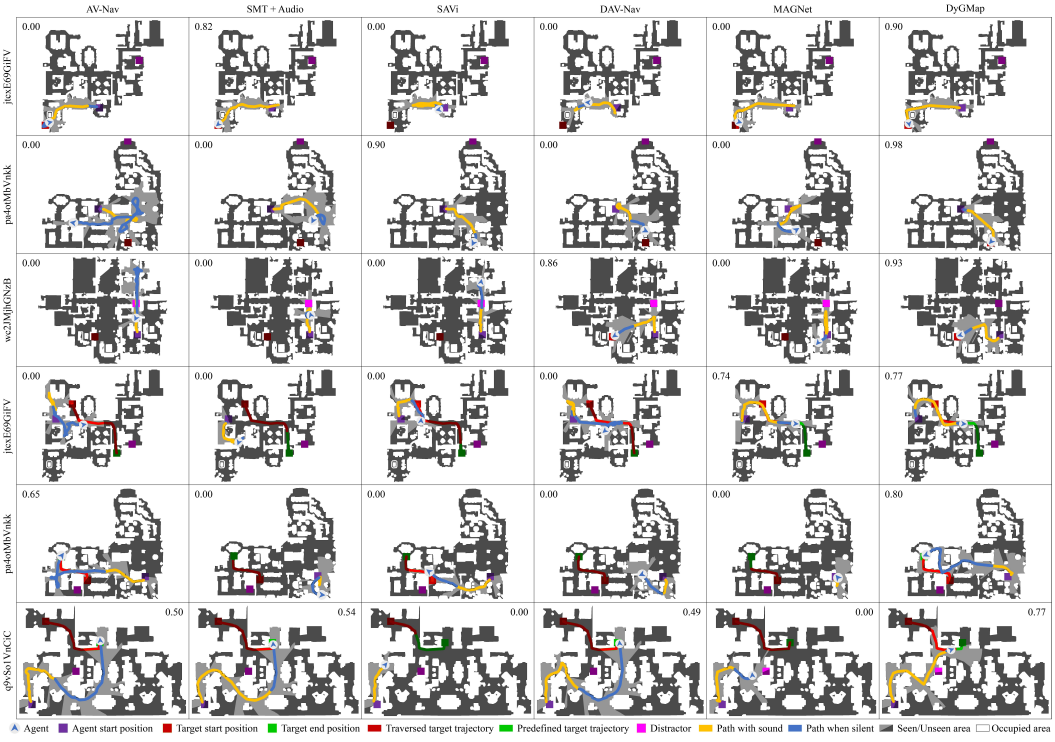


Figure 14: Navigation trajectories in the noisy setting.

sistent across methods. For DyGMap, however, performance remains generally better on cat-target episodes than on dog- and parrot-target episodes, with a more pronounced advantage in the noisy setting than in the clean setting. Although dogs are sampled more frequently than cats and parrots (0.4 versus 0.3 for each), this higher sampling probability does not consistently correspond to better navigation performance. The observed category differences may reflect a combination of acoustic characteristics, visual observability, and the composition of the evaluated episodes.

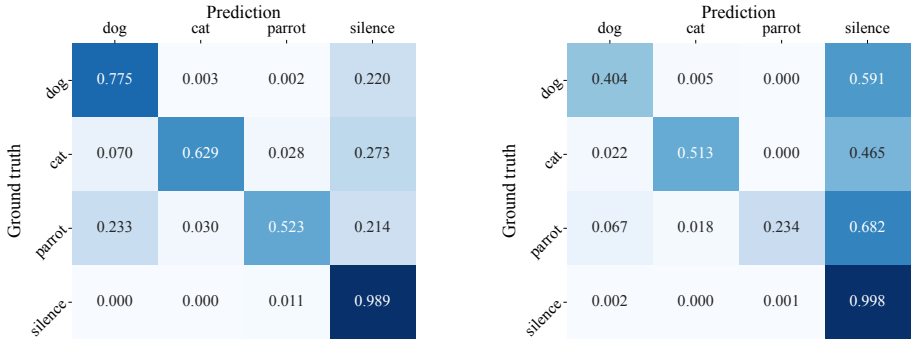
F.4 Additional Navigation Trajectories

We present additional trajectory visualizations to provide further qualitative insights into the navigation behavior of different methods. Fig. 14 shows representative trajectories in the noisy setting with additional acoustic interference. The first three rows correspond to stationary-target episodes, while the last three correspond to moving-target episodes.

Among these examples, the third row of Fig. 14 provides a representative case for analyzing the effect of an acoustic distractor. In most other visualized episodes, the distractor is relatively far from both the agent and the target and therefore has limited influence on the navigation trajectory. In this more challenging case, however, most baseline methods are initially attracted toward the distractor, causing their trajectories to deviate from the target. Although they later turn away from the distractor and attempt to correct their trajectories, the resulting detours consume substantial navigation time and make it difficult to reach the target after its sound becomes unavailable. AV-Nav passes the distractor and continues searching, but later turns back and stops between the distractor and the agent's initial position. SMT + Audio circles around the distractor and terminates after the target has been silent for some time, while SAVi moves past the distractor and continues forward before eventually stopping. MAGNet spends substantial time correcting its trajectory but fails to reach the target after target-related acoustic evidence becomes unavailable. DAV-Nav is also initially attracted by the distractor but successfully corrects its trajectory and reaches the target even after it becomes silent. In contrast, DyGMap shows little deviation toward the distractor and navigates more directly toward the target, exhibiting the most effective behavior among the evaluated methods.

Table 11: Goal reasoning performance in the clean and noisy settings.

Setting	Method	*Active*					*Silent*				
		$ER_{<20°}\downarrow$	$F_{<20°}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	$RDE\downarrow$	$ER_{<20°}\downarrow$	$F_{<20°}\uparrow$	$LE_{CD}\downarrow$	$LR_{CD}\uparrow$	$RDE\downarrow$
Clean	SAVi	0.879	0.138	47.19	0.591	0.287	0.974	0.020	52.28	0.346	0.315
	MAGNet	0.690	0.343	27.43	0.625	0.068	0.819	0.220	35.16	0.444	**0.074**
	DyGMap w/o vision	0.706	0.341	**23.94**	0.580	0.074	**0.787**	**0.244**	**32.30**	0.583	0.086
	DyGMap (Ours)	**0.639**	**0.395**	26.59	**0.665**	**0.055**	0.788	0.224	39.05	**0.722**	0.086
Noisy	SAVi	0.973	0.028	65.95	0.400	0.387	0.992	0.009	76.10	0.353	0.398
	MAGNet	0.856	0.178	42.80	**0.539**	0.101	0.914	0.115	47.58	0.431	0.144
	DyGMap w/o vision	0.852	0.192	38.32	0.476	0.093	**0.885**	**0.146**	**45.53**	**0.445**	0.105
	DyGMap (Ours)	**0.848**	**0.200**	**37.67**	0.442	**0.077**	0.910	0.120	50.15	0.416	**0.103**



(a) Clean setting (b) Noisy setting

Figure 15: Confusion matrices of target-category detection $\hat{A}_{ct}$ in the clean and noisy settings.

F.5 Goal Reasoning Results

We evaluate goal reasoning performance using persistent oracle ACCDDOA labels as the ground truth. Although the target's acoustic activity ends when it stops emitting sound, navigation still requires updating its category and location estimates using other available observations and the observation history. We therefore evaluate both the sound-emitting and silent periods, denoted as *active* and *silent*, respectively. The silent-period evaluation specifically assesses the ability to maintain target estimates without further acoustic evidence from the target. Goal estimates are evaluated along trajectories generated by the corresponding navigation policies. Since longer navigation after sound cessation generally makes target estimation more difficult, we limit the silent-period evaluation to at most 30 steps immediately after the target becomes silent to facilitate a fairer comparison across methods. Under this evaluation protocol, the ratio of evaluated *active* to *silent* time steps is approximately $4:1$. Tab. 11 reports the results in both settings, with all metrics macro-averaged across target categories. LE_{CD} is reported in degrees, and the remaining metrics are dimensionless.

During the active period, DyGMap generally outperforms the other evaluated methods, achieving more accurate target detection and localization. These results suggest that complementary visual evidence improves target estimation when target-related acoustic cues are available. For the silent period, since each method is evaluated along trajectories generated by its own navigation policy, differences in agent behavior and observation history affect the difficulty of maintaining and updating target estimates, making cross-method comparisons less controlled. Nevertheless, MAGNet, DyGMap w/o vision, and DyGMap retain reasonable performance during silence, despite some degradation relative to their active-period results. The benefit of visual integration is less consistent in this phase, with DyGMap w/o vision achieving the best results on most metrics.

We further visualize the goal reasoning results of DyGMap. Fig. 15 presents the confusion matrices of target-category detection $\hat{A}_{ct}$ in the clean and noisy settings. In the ground truth, the silence class corresponds to the period before the target begins emitting sound, whereas target-category labels persist after sound cessation. In the predictions, silence indicates that no target category is detected. This corresponds to a correct prediction before sound onset, a missed detection during sound emission, or a failure to maintain the target estimate after sound cessation. The GDN correctly

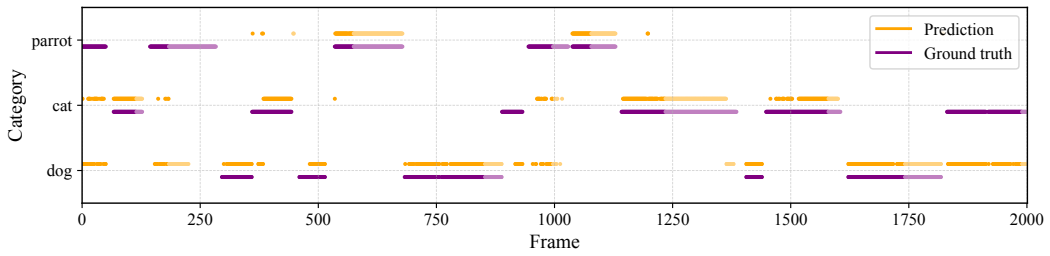


Figure 16: Visualization of target-category detection $\hat{A}_{ct}$ in the clean setting.

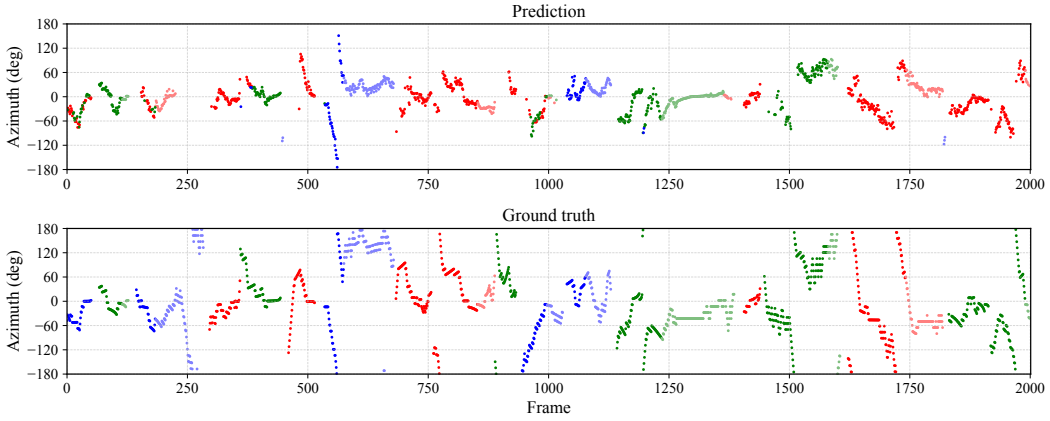


Figure 17: Visualization of azimuth estimation ϕ_{ct} in the clean setting.

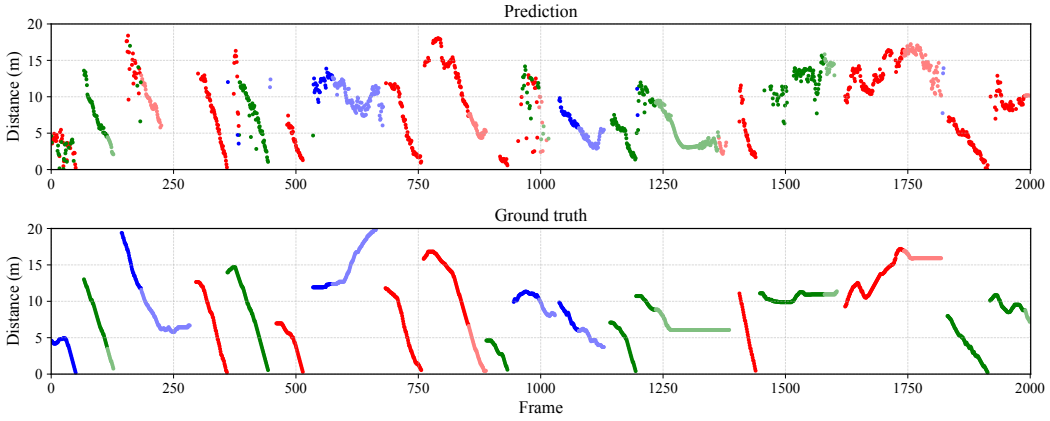


Figure 18: Visualization of distance estimation $d_{\max} \times d_{ct}$ in the clean setting.

classifies nearly all pre-onset silent frames but frequently misclassifies target-category frames as silence, particularly in the noisy setting. In the clean setting, approximately 23% of parrot events are misclassified as dog events, whereas this specific confusion is less evident in the noisy setting. Fig. 16 shows the target category estimation results aggregated across episodes over up to 2,000 time steps (frames), illustrating the GDN's ability to maintain and update target category estimates during both active and silent periods. Fig. 17 and Fig. 18 present the corresponding azimuth estimates ϕ_{ct} and distance estimates $d_{\max} \times d_{ct}$, respectively. Red, green, and blue represent dogs, cats, and parrots, respectively, while lighter shades indicate estimates during the target's silent period. Overall, these visualizations suggest that the GDN can generally estimate target categories and positions during the target's sound-emitting period and maintain target estimates after the target becomes silent.

Table 12: Failure analysis of different methods in the clean and noisy settings.

Setting	Method	Few actions	Short trajectory	Stop before active	Stop after silence	Premature stop	Stop at previous position	Hard to localize	Total failure
Clean	ObjectGoal	1.30	22.15	0.00	5.25	2.70	1.25	81.00	87.55
	AV-Nav	0.14	3.30	0.01	0.57	6.20	3.42	62.42	73.32
	SMT + Audio	0.21	4.10	0.06	1.17	6.24	3.91	53.14	64.96
	SAVi	0.18	4.81	0.00	0.86	5.19	2.66	50.42	59.92
	DAV-Nav	0.31	2.22	0.02	0.64	3.57	2.22	49.73	56.42
	MAGNet	0.19	10.28	0.00	2.35	5.90	3.70	45.02	56.57
	DyGMap w/o dynamic	0.24	2.04	0.06	0.67	5.16	3.58	35.38	45.97
	DyGMap (Ours)	0.09	2.18	0.00	0.56	5.48	3.68	33.46	44.80
Noisy	AV-Nav	0.34	2.94	0.01	0.33	3.92	1.99	68.00	76.19
	SMT + Audio	0.77	7.44	0.04	2.06	4.61	2.80	66.06	75.62
	SAVi	0.70	6.66	0.07	1.18	4.72	2.36	67.66	77.53
	DAV-Nav	0.55	3.69	0.00	1.31	3.61	2.71	61.32	69.23
	MAGNet	0.24	12.82	0.00	2.59	6.59	2.96	57.79	70.57
	DyGMap w/o dynamic	0.10	1.22	0.00	0.51	5.68	4.75	49.04	59.94
	DyGMap (Ours)	0.06	2.42	0.05	0.65	5.26	4.36	47.62	57.63

G Failure Analysis

We define the following diagnostic failure modes for unsuccessful episodes: (1) **Few actions**: The agent stops within 40 actions in an episode where the oracle policy requires more than 40 actions to reach the target. This indicates premature termination with insufficient interaction or exploration. (2) **Short trajectory**: The agent stops after traveling less than 5 m in an episode where the oracle path length exceeds 5 m. This indicates insufficient spatial exploration or movement before termination. (3) **Stop before active**: The agent stops before the target begins emitting sound. This indicates premature termination before target-related acoustic cues become available. (4) **Stop after silence**: The agent stops within 5 steps after the target ceases emitting sound. This indicates difficulty maintaining and exploiting previously acquired target information once target-related acoustic cues become unavailable. (5) **Premature stop**: The agent stops at a geodesic distance in the range $(1.0, 3.0)$ m from the target. This indicates inaccurate target proximity estimation or an incorrect decision about when to execute *Stop*. (6) **Hard to localize**: The agent stops at a geodesic distance greater than 5.0 m from the target. This indicates difficulty in reliably localizing the target or making sufficient progress toward it. (7) **Stop at previous position**: The agent stops at a location less than 1 m in Euclidean distance from a position previously occupied by the moving target and more than 2 m in geodesic distance from the target's position at episode termination. This indicates difficulty distinguishing the current target position from outdated spatial estimates. These modes characterize different failure behaviors and are not mutually exclusive.

Tab. 12 summarizes the failure statistics of different methods in the clean and noisy settings. All values are normalized by the total number of episodes and reported as percentages. The overall failure patterns are similar across the two settings, with hard to localize generally becoming more frequent in the noisy setting. Few actions and stop before active are rare across nearly all methods and settings, suggesting that most failures do not arise from agents terminating after a relatively short period of navigation or before receiving target-related acoustic evidence. Similarly, stop after silence occurs relatively infrequently, indicating that agents generally continue navigation after target-related acoustic cues become unavailable rather than immediately terminating. For short trajectory, most methods exhibit relatively low rates, whereas ObjectGoal and MAGNet exhibit noticeably higher values. Premature stop occurs at a moderate but relatively consistent rate across methods, indicating that agents make inaccurate stopping decisions before entering the success radius. In contrast, stop at previous position captures failures in which the agent stops near a previously occupied target location, suggesting the need to reassess the target's current location before deciding to stop. Among all failure modes, hard to localize is substantially more frequent, suggesting that reliably locating and reaching the target constitutes the dominant challenge. Overall, the failure patterns indicate that maintaining accurate target localization throughout navigation is a major bottleneck, while acoustic interference further exacerbates this difficulty. These results highlight the importance of robust target estimation and tracking for improving navigation performance in DANCE.