RAO-Nav: Probing Omni-Language Models for Zero-shot Semantic Audio-Visual Navigation

Qilang Ye[1,2], Meng Liu[3,2] *, Yu Zhou[1,2] *

[1]VCIP & TMCC & DISSec, College of Computer Science &
College of Cryptology and Cyber Science, Nankai University
[2]Zhongguancun Academy
[3]Shandong University
s-yql25@bza.edu.cn, mengliu.sdu@gmail.com, yzhou@nankai.edu.cn

Abstract

We explore whether Omni-Language Models (OLMs) can be directly applied to zero-shot Semantic Audio-Visual Navigation (SAVN). Recent work demonstrates that even state-of-the-art specialized models still struggle to achieve generalist multimodal navigation, despite extensive task-specific training. In this paper, we introduce **RAO-Nav**, short for **R**easoning **A**ll-in-One **O**LM, a deployment pipeline for zero-shot SAVN. By leveraging the rich implicit audio-visual knowledge encoded in OLMs, the embodied agent is enabled to "*hear*", "*see*", "*reason*", and "*act*" in the environment. To further elicit the built-in thinking ability of OLMs, we propose a test-time **L**atent **N**avigation **R**easoning (LNR) module that can be seamlessly integrated into the decoding space. LNR encourages the model to retrieve more target-relevant observations and make effective navigation decisions. Through comprehensive experiments, we show that our framework surpasses existing state-of-the-art baselines on public SAVN benchmarks without using any training data. Moreover, we introduce a new *Global Navigation Instruction* setting to further evaluate the ability of OLMs to serve as embodied navigation agents. Code: https://github.com/rikeilong/OmniAV_Nav

1 Introduction

Unlike most embodied navigation tasks that focus on egocentric vision [1, 2, 3], audio-visual navigation [4, 5] enables the agent to navigate toward a target location with audio signals in complex real or virtual environments. Furthermore, Semantic Audio-Visual Navigation (SAVN) [6, 7] extends this task by requiring an agent to use short-duration sound messages to predict what the target is and where it is. It is more challenging not only because the audio signal can be ambiguous, but also because the target may be blocked by walls or other obstacles and cannot always be seen directly [8, 9]. To deploy in such cases, an embodied agent must go beyond mere visual perception and combine sparse auditory cues with current visual observations to make effective navigational decisions.

Recent progress in Large Language Models (LLMs) [10] has begun to influence embodied robotics [11, 12, 13]. Early efforts [14] mainly use LLMs as high-level planners or language-centric modules to support navigation. However, it mainly focuses on language-driven perception and scene understanding via collaboration among multi-LLMs, which limits its practicality in real-world deployment. More recently, the development of Omni-Language Models (OLMs[1]) [15, 16, 17, 18] has extended the world knowledge to allow models to process not only language, but also audio and

*Corresponding authors.

[1]In this paper, OLMs refer to models capable of processing image, video, and audio inputs.

Preprint.

visual inputs [19, 20, 21]. With carefully crafted prompts, they can be adapted to a wide range of downstream tasks without task-specific training.

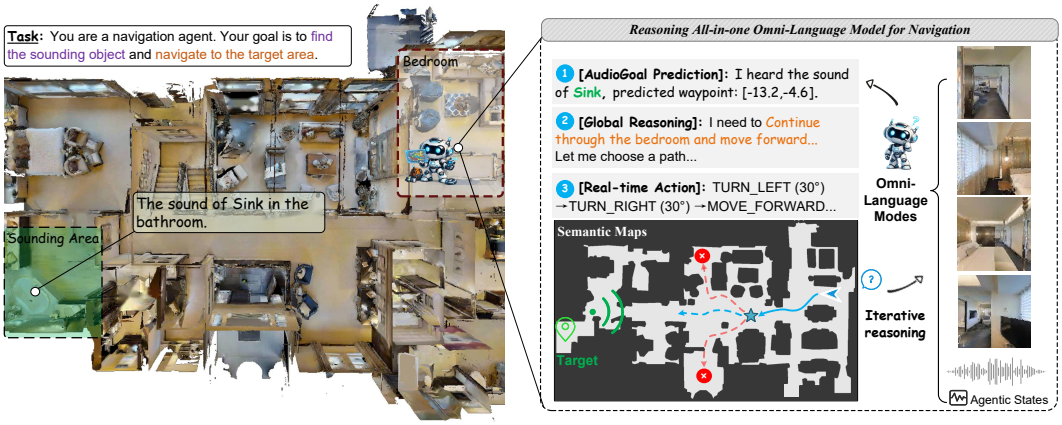

Figure 1: We propose the RAO-Nav pipeline built on a single OLM to achieve zero-shot audio-visual navigation. It can be deployed in both real-world and virtual environments and iteratively performs navigation in three steps: *AudioGoal Prediction*, *Global Reasoning*, and *Real-time Action*.

This naturally raises a question: *can OLMs serve as the main brain for SAVN?* Similar to how humans solve audio-visual navigation by first identifying the sounding object and then moving toward the target, OLMs also require reacting to such tasks with three fundamental aspects: **1) can the model infer what the sounding target may be and where it is**? **2) can it form a rough action plan according to the surrounding scene**? **3) can it act autonomously based on new observations to complete a navigation task**? First, although OLMs can recognize the category of a sound, they do not naturally provide explicit Sound Source Localization (SSL) [22, 23], making it difficult to ground audio understanding into a precise spatial position. Second, Qwen-Omni is not originally designed for discrete embodied action prediction, so directly mapping multimodal observations to output actions remains unstable. Third, a single OLM decision is often limited by short planning horizons, which makes it prone to spinning in place during action execution. Since OLMs have shown strong multimodal understanding ability, how to effectively elicit such reasoning ability for SAVN remains an open challenge.

To address the aforementioned challenges, we propose **RAO-Nav**, a deployment pipeline that uses a single OLM to achieve zero-shot SAVN. RAO-Nav decomposes the navigation into an iterative three-stage process: *AudioGoal Prediction*, *Global Reasoning*, and *Real-time Action*. First, OLMs infer the target category via their built-in audio understanding ability. To locate the target, we design a distinct location module to guide the frozen OLM with dual-frequency spectrograms. Then, the model performs a high-level navigation guidance (e.g., leave the current room and follow the hallway for a long stretch) with the semantic map. At the execution step, the model receives continuous RGB observations and updates the action sequence in the environment. In this way, RAO-Nav allows a single OLM to connect multimodal perception, planning, and action within one unified pipeline. To further improve the reasoning ability of the OLM during navigation, we introduce a test-time **L**atent **N**avigation **R**easoning (LNR) module. LNR's goal is to encourage the model to focus on more target-relevant observations during global reasoning. As a result, the model can make more stable and effective decisions in long-horizon and uncertain environments.

We compare our RAO-Nav with existing specialized models and multi-LLM collaboration frameworks on public SAVN benchmarks. Results show that our method outperforms most existing baselines without using any task-specific training data. Moreover, the single-OLM paradigm offers substantial convenience for real-world deployment, as an embodied agent can be driven to complete navigation tasks using only a server equipped with a single GPU. To further test whether OLMs can serve as practical embodied agents beyond simulation platform settings, we introduce a new Global Navigation Instruction setting based on SAVN. It evaluates the model under more realistic language-guided global perception and navigation scenarios. The results show that, although OLMs demonstrate promising zero-shot ability, deploying them in real-world natural-language-guided audio-visual navigation remains challenging. Our contributions are summarized as follows:

- We propose RAO-Nav for SAVN, exploiting the powerful audio-visual understanding ability of OLMs to navigate effectively.
- Building on this pipeline, we further introduce a test-time LNR module to better elicit the reasoning ability of OLMs during navigation.
- Experiments on public SAVN benchmarks show that our method achieves clear zero-shot improvements over strong baselines without using any task-specific training data.

2 Related Work

Semantic Audio Visual Navigation. SAVN has developed with the emergence of embodied simulators and realistic audio-visual environments (such as Habitat [8, 9] and SoundSpaces [5, 7]). Early research [6] primarily focused on learning end-to-end navigation strategies by jointly modelling visual and auditory observations. For example, SAVi [6] introduces a benchmark for evaluating whether a model can drive navigation to audible semantic targets. These works establish the basic problem setting, where the agent must infer the target from sparse audio cues and navigate under partial visual observations. Recent research has further advanced this field in several directions, including modular transfer learning [24], knowledge-enhanced scene priors [25], and more realistic scenarios incorporating moving sound sources in complex, unmapped environments [26]. Such extensions push SAVN beyond static indoor settings and make the task closer to real-world deployment. In addition to standard SAVN, AVLEN [27] extends the task to audio-visual-language embodied navigation, exploring oracle-language to support navigation. These efforts show that SAVN is gradually moving from task-specific policy learning toward richer multimodal reasoning and interaction.

SAVN with MLLMs. MLLMs [15, 28, 29, 30, 31] refer to models that understand both video and audio inputs and output text instructions. Recent studies [14, 32] investigate whether language-centric multimodal agents can perform navigation through prompt engineering. RILA [14] introduces a reflective and imaginative language agent for zero-shot SAVN, showing that LLM-based planning can leverage multimodal perception to guide exploration. This line of work suggests that large models may provide a new alternative to heavily trained navigation policies. NavBench [33] probes the navigation capability of MLLMs under zero-shot settings, and the results show that although open-source MLLMs exhibit promising spatial and action reasoning ability, they still struggle with robust long-horizon decision-making. Moreover, OctoNav [34] explores free-form embodied navigation and argues that it is difficult for MLLMs to achieve generalist embodied navigation. These findings indicate that strong multimodal understanding alone does not directly translate into reliable embodied control. Therefore, we believe that applying MLLMs designed for general-purpose scenarios to SAVN remains a challenge that has not yet been fully explored.

3 RAO-Nav Pipeline

In this section, we study whether a single OLM can solve the SAVN task [6] in a zero-shot manner. As shown in Fig. 2, the agent is required to identify a sounding target and navigate toward it in complex and unseen environments by jointly using auditory and visual observations. Instead of relying on task-specific training trajectories, we aim to directly leverage the intrinsic multimodal perception and reasoning ability of OLMs for navigation. To this end, we propose **RAO-Nav**, a unified deployment pipeline that enables the model to perform *AudioGoal Prediction*, *Global Reasoning*, and *Real-time Action* in a human-like step-by-step manner.

3.1 Problem Definition

Consider an agent in a previously unseen 3D indoor environment, where navigation is performed over a finite action space $\mathcal{A} = \{\texttt{stop}, \texttt{move_forward}, \texttt{turn_left}, \texttt{turn_right}\}$. The goal of SAVN is to locate and reach a sounding target object in the scene. To better adapt the formulation to OLMs, we do not introduce redundant components used in previous work [27], such as the immediate reward, the probability of observation, or the transition probability. Instead, we formulate the sequential decision process characterized by the tuple $(\mathcal{H}, \mathcal{S}, \mathcal{A}, \mathcal{O})$. At each time step t, $\mathcal{O}_t$ denotes the current ego-centric RGB image, $a_t \in \mathcal{A}$ denotes the audio signal, and $s_t \in \mathcal{S}$ denotes agent states such as angle and distance to the predicted goal. Similar to [6], we keep a history of past observation $\mathcal{H}_t = \{\mathcal{O}_1, \ldots, \mathcal{O}_t\}$ as the memory for reasoning. Based on these inputs, OLMs

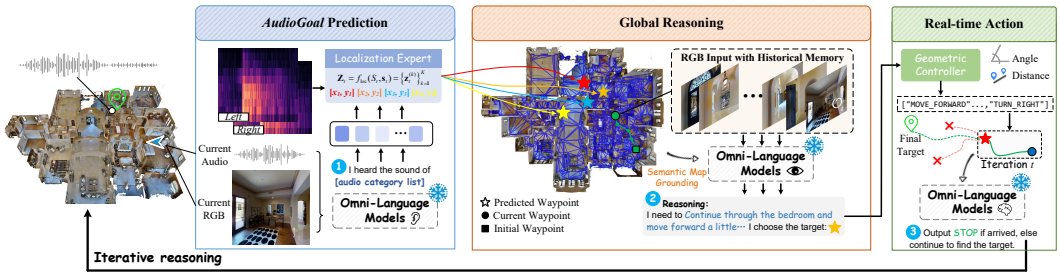

Figure 2: **Overview of the RAO-Nav Pipeline for zero-shot SAVN.** Given a scene and a queried episode, the agent receives the current RGB observation and the incoming audio signal (or `None` if no new audio is observed). ❶ The OLM first performs audio classification and predicts the target waypoint at time step t with the Localization Expert. Conditioned on multiple target waypoints and historical RGB observations, ❷ the OLM performs language-driven navigation reasoning to choose a target. The selected target is executed by a low-level action compiler, ❸ while the OLM continuously determines whether the agent should stop based on the current surroundings. Notably, all OLMs in the pipeline share the same model.

as agents continuously update the current *AudioGoal* to maximize the probability of reaching the sounding target and stopping at the correct location.

3.2 *AudioGoal* Prediction

Fully open-set audio classification for OLMs remains challenging, as reverberation and semantically similar acoustic events are common in both virtual and real-world environments. Since an audio signal may only be present momentarily, we propose an iterative perception–localization strategy built upon OLMs, enabling them to reason about the target object in a more generalizable way for SAVN. Specifically, it is divided into two steps:

Perception. Let $\mathcal{A}_t$ denote the audio observation received at time step t, and let $\mathbf{u}_t$ denote the action executed at this step. We use $\mathcal{A}_{1:i} = \{\mathcal{A}_1, \ldots, \mathcal{A}_i\}$ and $\mathbf{u}_{1:i} = \{\mathbf{u}_1, \ldots, \mathbf{u}_i\}$ to represent the audio observations and action history up to step i, respectively. Given the accumulated audio observations and action history, the OLM produces a $\mathcal{G}$-dimensional vector of audio-category scores:

$$\mathbf{s}_i = G_{\text{obj}}\left(\mathcal{A}_{1:i}, \mathbf{u}_{1:i}\right) \in \mathbb{R}^{|\mathcal{G}|} \tag{1}$$

where $\mathcal{G}$ is the fixed set of the defined audio categories in the SAVi benchmark, and $s_{i,g}$ denotes the score assigned to category $g \in \mathcal{G}$ at step i. These scores are generated directly by the frozen OLM without task-specific training or fine-tuning.

Since an audio signal may only be available for a short period, we model audio prediction as a progressively updated belief. After each executed action, the OLM updates the category scores using the newly accumulated audio observations. We then construct the candidate target set as

$$\hat{\mathcal{G}}_t = \left\{ g \in \mathcal{G} \;\middle|\; \sum_{i=1}^{t} \max\left(0,\ s_{i,g} - \tau\right) \geq \gamma \right\}, \tag{2}$$

where the summation ranges from the first step to the current step ($i = 1, \ldots, t$). In all experiments, we set $\tau = 0.12$ and $\gamma = 0.5$. Since the uniform score is $1/21 \approx 0.048$, τ filters out weak per-step predictions, whereas γ retains categories with sufficiently strong accumulated evidence. The resulting candidate set $\hat{\mathcal{G}}_t$ is then used by the localization module.

Localization. General-purpose OLMs can recognize the semantic category of a sound but do not reliably estimate its spatial location. To provide explicit spatial grounding, we use a plug-in Localization Expert following the localization design of [6]. The expert is used without additional task-specific training and predicts the relative source position from binaural audio features and the OLM-generated category information.

Let $\mathcal{A}_{t,L}$ and $\mathcal{A}_{t,R}$ denote the left and right audio channels at time step t. We first convert each channel into a time-frequency representation and apply logarithmic compression:

$$S_{t,c} = \log\left(1 + F(\mathcal{A}_{t,c})\right), \qquad c \in \{L, R\}, \tag{3}$$

where $F(\cdot)$ denotes the time-frequency transformation. We then stack the two spectrograms $S_t = \mathrm{Stack}(S_{t,L}, S_{t,R})$. This two-channel representation preserves both spectral information and inter-channel differences.

To condition localization on the OLM's semantic prediction, we use the continuous $\mathcal{G}$-dimensional category-score vector $\mathbf{s}_t$ as the localization input. Then, the Localization Expert predicts K possible source positions:

$$\mathbf{Z}_t = f_{\mathrm{loc}}(S_t, \mathbf{s}_t) = \left\{\mathbf{z}_t^{(k)}\right\}_{k=1}^{K}, \qquad \mathbf{z}_t^{(k)} = \left[u_t^{(k)}, v_t^{(k)}\right]. \tag{4}$$

Each predicted position is then transformed into the agent coordinate system:$\hat{\mathbf{c}}_t^{(k)} = \begin{bmatrix} -v_t^{(k)} \\ u_t^{(k)} \end{bmatrix}$, where $\hat{\mathbf{c}}_t^{(k)}$ is the relative position of the k-th possible sound source. In this way, the expert only provides geometric estimation conditioned on the predicted target for navigation, allowing our RAO-Nav pipeline to retain the audio understanding ability of OLMs.

3.3 Global Reasoning

Although OLMs hold powerful transferability, they are still fundamentally language-driven models and are not well-suited for directly predicting discrete actions. To make the planning more robust while keeping the decision space tractable for OLMs, we follow [34] and introduce a QA-driven reasoning paradigm with semantic map grounding. Specifically, given the candidate waypoint set $\mathbf{Z}_t$ at iteration t and the current agent position $\mathbf{w}_t^{\mathrm{pos}}$, the OLM needs to reason jointly over the predicted audio object and the visual observations to determine which path is most consistent with the target semantics, instead of reasoning over the entire discrete action space. We present simplified versions of the task-specific prompts below.

/* **TASK DESCRIPTION** */ \nGiven the predicted target: {...}, the semantic map <img>, and the recent visual observations, please infer how to reach the target location and select the target you consider most suitable.
/* **CANDIDATE TARGETS** */ \nGiven the following targets:\n A. Red\n B. Green\n C. Orange\n D. Yellow \n...
/* **OBESERVATIONS** */ Recent actions: {action_list[-5:]}. Recent RGB inputs: {<img><img><img><img><img><img><img>}.

This explicit multi-waypoints formulation makes better use of the zero-shot decision-making ability of OLMs, while encouraging the agent to explore sufficiently diverse routes for completing SAVN.

3.4 Real-time Action

To convert the selected path into executable low-level actions, we introduce a *Geometric Controller* that compiles the trajectory into a deterministic action sequence. Instead of predicting one or a few actions from local observations at each step, we execute long-horizon actions with the reasoning of OLMs. Given the selected path $P_t^{(z_t)} = \{\mathbf{w}_i^{\mathrm{pos}}\}_{i=0}^{N_t}$, where $\mathbf{w}_0^{\mathrm{pos}}$ is the current position and $\mathbf{w}_{N_t}^{\mathrm{pos}}$ is the predicted goal, we decompose each $\mathbf{w}_0^{\mathrm{pos}} \rightarrow \mathbf{w}_{N_t}^{\mathrm{pos}}$ into a *turn-then-forward* procedure. Specifically, we compute the turning angle θ_i between the current heading and the segment direction, and the travel distance d_i on the xz plane, which can be denoted as:

$$n_i^{\mathrm{turn}} = \left\lfloor \frac{|\theta_i|}{\Delta\theta} \right\rfloor, \quad n_i^{\mathrm{forward}} = \left\lfloor \frac{d_i}{\Delta s} \right\rfloor, \tag{5}$$

where $\Delta\theta$ and Δs denote the fixed rotation angle and forward step size.

After executing each compiled action sequence, the agent moves to a new position and receives updated RGB and audio observations. Furthermore, to stabilize localization across iterations, we apply temporal smoothing to every predicted point goal:

$$\tilde{\mathbf{z}}_t = (1 - \omega)\mathbf{z}_t + \omega\,\Pi(\tilde{\mathbf{z}}_{t-1}), \qquad \omega \in [0, 1], \tag{6}$$

where $\Pi(\cdot)$ projects the previous estimate into the current coordinate frame. When no new audio is observed, we retain the previous estimate instead of forcing a new prediction. During execution, the

OLM continuously monitors the environment and decides whether to terminate with `STOP`. Otherwise, the system re-runs *AudioGoal* prediction and generates a new candidate path, forming a closed-loop predict–plan–decide–execute pipeline.

4 Test-time Latent Navigation Reasoning Module

To reduce the amount of historical RGB observations used in subsequent iterations, we introduce a simple yet effective test-time LNR module that enables the OLM to selectively focus on target-relevant observations without additional training. Specifically, the agent accumulates a sequence of historical RGB embeddings $\mathcal{V} = \texttt{get_input_embeddings}(\mathcal{O}_1, \ldots, \mathcal{O}_t)$. Then, we initialize L latent tokens on the text side, each represented by a special placeholder token `<|latent|>`. These tokens are embedded using the same `get_input_embeddings()` function and concatenated with other multimodal tokens. Formally, these embeddings are denoted as $\mathbf{H} = [h_1, h_2, \ldots, h_L]$ and share the same embedding space as other tokens. Instead of updating model parameters, we define an intrinsic reward based on the model's conditional likelihood:

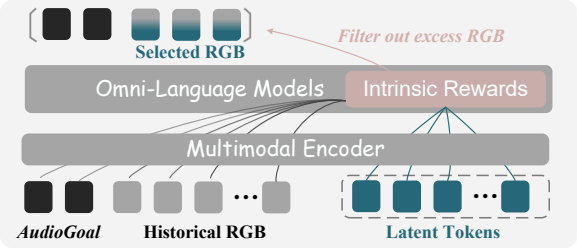

Figure 3: Overview of the proposed LNR module.

$$R(\mathbf{H}^{(t)}) = \log p_\theta(\hat{\mathcal{G}}_t^k \mid \mathcal{V}, \mathbf{H}^{(t)}), \tag{7}$$

where $\hat{\mathcal{G}}_t^k$ denotes the predicted audio goal (represented as text tokens). A higher likelihood indicates that the latent tokens better summarize goal-relevant information from the observations. Then, we iteratively update the latent tokens to maximize this reward by:

$$\mathbf{H}^{(t+1)} = \mathbf{H}^{(t)} + \eta \nabla_{\mathbf{H}} R(\mathbf{H}^{(t)}). \tag{8}$$

where the gradient $\nabla_{\mathbf{H}}$ is computed via backpropagation through the frozen OLM, and η is a step size. After optimization, the latent tokens are aggregated into a query $q = \text{MeanPool}(\mathbf{H})$. Finally, we compute the similarity between the query and each frame embedding to select the top-K RGB embeddings to form a compact subset for the next navigation reasoning step.

5 Experiments

5.1 Experiment Setup

Simulator and Datasets. We evaluate our method in the SoundSpaces [5] simulator built on Habitat [8]. The 3D environments are constructed from the Matterport3D (MP3D) dataset [35], which provides realistic indoor layouts and object labels. Unlike prior SAVN works [6, 27], we focus on unheard sounds and unseen scenes at test time, where the agent must navigate toward sounding target objects without access to task-specific training trajectories. This setting is suitable for evaluating whether a single OLM can generalize to navigation tasks in a zero-shot manner.

Baselines. We evaluate four different types of baselines, including:

- **Specialized Models** are task-specific navigation methods trained with environment trajectories, including AudioGoal [5], SAVi [6], AV-WAN [36], AVLEN [27], and K-SAVEN [9]. These methods are usually optimized with task-specific supervision and rely on large amounts of environment interaction data.
- **Multi-MLLMs Collaborative Frameworks** perform zero-shot navigation through the cooperation of multiple language-based modules, e.g., RILA [14]. In this paradigm, different modules are designed for separate roles, such as audio-visual perception and global planning. While this design can achieve zero-shot SAVN, it increases system complexity and makes real-world deployment less convenient.
- **Pure Vision-Language Models (VLMs) with Audio Descriptions.** This setup is designed to assess whether VLMs possess visual navigation capabilities after a given fixed audio target. We feed audio information as textual descriptions rather than directly reasoning over raw audio-visual inputs. The evaluation model includes Qwen-VL Series (7B/8B/30B) [37, 38], and LLaVA-NeXT [39].

Table 1: Comparison of four types of navigation frameworks on the Matterport3D test dataset in unheard-sound settings. Oracle† denotes a privileged-input diagnostic setting for Pure VLMs. The model is given the ground-truth audio category, the semantic map, and the ground-truth target location marked on the map. It directly predicts low-level actions.

	Method	SR (%)↑	SPL (%)↑	SNA (%)↑	DTG (m)↓	SWS (%)↑
	Supervised					
Specialized Models	AudioGoal	16.5	15.5	10.4	12.8	5.6
	AV-WAN	17.2	13.2	12.7	11.0	6.9
	SAVi	24.8	17.2	13.2	9.9	14.7
	AVLEN	26.2	17.6	14.2	9.2	15.8
	K-SAVEN	34.4	**23.4**	**21.7**	**6.6**	14.3
	Zero-Shot					
Multi-MLLMs	RILA	35.4	11.8	8.7	11.4	20.4
Pure VLMs	LLaVA-NeXT + Oracle†	4.6	2.9	2.4	19.6	2.1
	Qwen2.5-VL-7B + Oracle†	8.6	4.8	3.7	28.7	4.3
	Qwen3-VL-8B + Oracle†	7.9	2.5	4.2	21.3	3.8
	Qwen3-VL-30B + Oracle†	14.7	5.2	7.6	17.6	3.5
w/ RAO-Nav	Baichuan-Omni-7B	37.6	16.7	9.4	15.2	21.2
	Qwen2.5-Omni-7B	36.4	18.4	10.2	11.7	21.4
w/ RAO-Nav + LNR	Baichuan-Omni-7B	38.4	17.5	9.6	13.7	**22.4**
	Qwen2.5-Omni-7B	**39.2**	20.7	12.4	10.9	21.9

- **OLMs w/ RAO-Nav Pipeline** use one unified model to jointly process audio, vision, and interaction history for navigation. We evaluate models including Qwen2.5-Omni-7B [17], and Baichuan-Omni-1.5-7B [16].

Through this comparison, we examine how RAO-Nav differs from heavily trained specialized models as well as from different zero-shot paradigms.

Metrics. We follow standard SAVN evaluation protocols and report Success Rate (SR), Success weighted by Path Length (SPL), Success Rate weighted by Number of Actions (SNA), Success When Silent (SWS), and Distance To Goal (DTG). Moreover, we further evaluate the performance of OLMs in the global navigation instruction setting in Section 5.3.

Implementation Details. Consistent with the standard SAVN setting, the agent is provided with RGB observations at a resolution of 256×256. It also receives audio signals in `wav` format, which are further converted into two-channel spectrograms of size 65×26 for goal prediction. All large models in our framework are deployed on only one NVIDIA A100 GPU (80GB). We evaluate RAO-Nav on 1,000 test episodes, and limit the maximum number of execution steps to 200 for each episode.

5.2 Experimental Results

Main Experiment. We compare different navigation frameworks on the Matterport3D dataset under the unheard-sound setting in Table 1. For supervised methods, K-SAVEN achieves the best SPL and SNA by leveraging oracle guidance, indicating that task-specific training helps the agent follow near-optimal trajectories. However, such approaches rely heavily on curated data and generalize poorly to unseen environments. In the zero-shot setting, RILA achieves a higher SR than all supervised baselines, but suffers from low SPL and SNA, suggesting inefficient and unstable navigation. Moreover, directly applying VLMs performs significantly worse, with SR below 15%, indicating that mapping visual observations to actions without structured reasoning is insufficient.

In contrast, our RAO-Nav pipeline enables a single OLM to perform SAVN effectively. Without LNR, RAO-Nav already achieves strong performance (e.g., 37.6% SR with Baichuan-Omni-7B). With LNR, Qwen2.5-Omni-7B further improves to the best SR of 39.2%, surpassing all supervised baselines, while maintaining competitive SPL and strong stopping performance SWS. This shows that LNR helps filter redundant RGB observations and improves decision stability. However, zero-shot methods still lag behind supervised models in SPL, SWS, and DTG, indicating less efficient trajectories and occasional redundant actions.

Table 2: Experimental results on SAVN in our Global Navigation Instruction settings. Closed and Open indicate whether candidate audio categories are provided, and Map indicates whether a semantic map is provided.

Methods	*Audio Classification*		*Visual Navigation*		All avg.
	Closed	Open	w/o Map	w/ Map	
OLMs with an average accuracy below 40% ↓					
MIO-Instruct-7B *(open-source)*	47.56	6.40	18.95	17.74	22.84
PandaGPT-7B *(open-source)*	45.73	9.88	25.81	28.63	27.04
Video-SALMONN-o1-7B *(open-source)*	53.05	12.21	29.03	30.65	31.25
Video-LLaMA2-7B *(open-source)*	40.85	5.23	23.79	27.02	24.28
OLMs with an average accuracy above 40% ↑					
Baichuan-Omni-7B *(open-source)*	59.76	23.84	35.89	45.56	40.99
Qwen2.5-Omni-7B *(open-source)*	65.24	21.51	37.10	43.15	42.67
ChatGPT-4o *(closed-source)*	75.61	36.05	41.94	53.23	48.80
Gemini2.0-Flash *(closed-source)*	71.34	31.40	50.81	52.02	51.68

Experiment on Global Navigation Instruction. To assess whether current OLMs are suitable for our proposed RAO-Nav deployment, we evaluate their performance under a Global Navigation Instruction setting within the SAVN task. We collect 164 and 172 samples for the closed-set and open-set in the audio classification task, respectively, and 248 samples for the visual navigation task. This setting highlights language-driven navigation, where the agent must rely on high-level instructions rather than step-by-step supervision. We first examine the audio perception capability of OLMs, which is essential for grounding the target sound source. As shown in Table 2, we evaluate models under both closed-set and open-set conditions. In the closed setting, candidate audio categories are provided, while in the open setting, models must infer the target category without prior constraints. The results show that although several OLMs achieve reasonable performance in the closed setting, there is a clear performance drop in the open setting, indicating that robust audio understanding and generalization remain challenging for most models.

Furthermore, we investigate how well OLMs align visual observations with language instructions. Each input consists of multiple candidate navigation options, each paired with a corresponding RGB observation. The model is required to select the option that best matches the target. This setup evaluates whether the model can jointly reason over visual content and language descriptions. The results indicate that most models struggle to consistently identify the correct option, especially in complex scenes with subtle visual differences. Even strong models tend to be distracted by irrelevant visual cues or fail to correctly associate language instructions with the corresponding observations. This suggests that cross-modal alignment between vision and language instruction remains a key bottleneck for OLMs in navigation tasks.

5.3 Ablation Studies

Ablation on Localization Expert. We compare three localization strategies to explore the impact of target position estimation. *Random* refers to sampling target locations randomly on the map, while *OLM-only Localization* lets the model infer the target position directly from the predicted audio category and the semantic map. As shown in Table 3(a), the result shows that current general-purpose OLMs are weak at sound-source localization, which motivates the use of the Localization Expert. We study whether audio-category information is useful for localization. Following the two SAVi variants [6], we compare a spectrogram-only localizer with a localizer that receives the OLM-predicted audio-category information. The result shows that removing the category input reduces the SR from 39.2% to 12.6%. We believe that reliable localization benefits from both acoustic and semantic information.

Ablation on Input Modalities. We further investigate the effect of input modalities in Table 3(b). Interestingly, using only RGB observations achieves the best performance, while incorporating depth information results in a drop in SR and SPL and an increase in DTG. We argue that depth images do not provide additional useful cues for the OLM. Instead, it introduces extra input complexity and increases the reasoning burden of the model. Since RAO-Nav relies on language-driven reasoning over multimodal inputs, adding depth may distract the model from focusing on more informative

Table 3: **Ablations.** We conduct detailed ablations based on Qwen2.5-Omni-7B in SAVN.

(a) Results of different localization strategies. All settings use the same OLM for global reasoning and action decisions. $\hat{\mathcal{G}}_t$ denotes the OLM reasoning output.

Method	SR	SPL	DTG
Random	3.4	1.9	21.7
OLM-only Localization	10.7	5.8	16.4
Localization Expert (w/o $\hat{\mathcal{G}}_t$)	12.6	7.4	14.6
Localization Expert (w/ $\hat{\mathcal{G}}_t$)	**39.2**	**20.7**	**10.9**

(b) Results of different input modalities.

Modality	SR	SPL	DTG
only RGB	**39.2**	**20.7**	**10.9**
w/ Depth	28.6	12.4	18.4

(c) Results of different LNR variants.

Method	SR	SPL	DTG
Random-K	34.6	18.3	13.7
K-NN	37.8	19.3	12.4
LNR	**39.2**	**20.7**	**10.9**

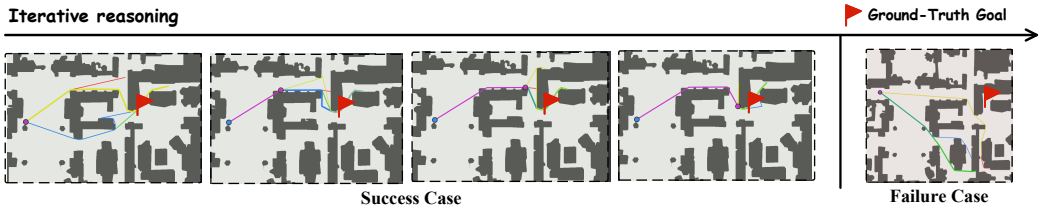

Figure 4: Navigation trajectories in different iterations with RAO-Nav.

semantic and visual signals. Therefore, we adopt a simplified design that only uses RGB inputs, which is more effective for zero-shot navigation.

Ablation on LNR. We compare LNR with two history-selection baselines: randomly selecting K observations (*Random-K*) and selecting the K nearest observations in the embedding space (*K-NN*). Compared with K-NN, LNR improves SR and SPL by 1.4% and reduces DTG by 1.5 m, showing that target-guided latent optimization is more effective than simple observation selection.

5.4 Visualization of Navigation Trajectories

To better understand the behavior of RAO-Nav, we visualize navigation trajectories under the iterative reasoning process, including both success and failure cases. As shown in Fig. 4, in successful cases, the agent is able to progressively refine its decisions by continuously incorporating new visual observations. Although the initial trajectories may deviate from the ground-truth goal due to coarse localization, the model gradually corrects its direction through iterative reasoning and ultimately reaches the sounding target. In contrast, most failure cases are primarily caused by inaccurate sound source localization. When the predicted target position is incorrect, the subsequent reasoning process is guided toward a wrong region, leading the agent to follow suboptimal or misleading paths. As a result, even though the model continues to update its reasoning, it fails to recover and reach the true target. It suggests that while OLMs have strong potential for navigation through iterative reasoning, their performance heavily depends on the accuracy of target localization, highlighting the importance of reliable spatial grounding in SAVN.

6 Conclusion

In this paper, we investigate whether OLMs can serve as a unified solution for zero-shot SAVN. We propose RAO-Nav, a simple yet effective pipeline that enables a single OLM to jointly perform audio understanding, global reasoning, and action execution in an iterative manner. By decomposing navigation into *AudioGoal* Prediction, Global Reasoning, and Real-time Action, our framework bridges multimodal perception and decision-making without any task-specific training. Furthermore, we introduce a test-time LNR module to filter redundant historical RGB observations and encourage the model to focus on target-relevant information during iterative reasoning. Experimental results on SAVN benchmarks demonstrate that RAO-Nav achieves strong zero-shot performance and surpasses existing supervised baselines.

7 Acknowledgments

This research was funded by the Zhongguancun Academy (Grant No. 20240306), the National Natural Science Foundation of China (Grant NO 62376266 and 62406318), CAAI-Tencent Rhino-Bird Open Research Fund, and CAAI-Ant Group Research Fund.

References

[1] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik, "Cognitive mapping and planning for visual navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2616–2625.

[2] Y. Wu, Y. Wu, A. Tamar, S. Russell, G. Gkioxari, and Y. Tian, "Bayesian relational memory for semantic visual navigation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2769–2779.

[3] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3357–3364.

[4] C. Gan, Y. Zhang, J. Wu, B. Gong, and J. B. Tenenbaum, "Look, listen, and act: Towards audio-visual embodied navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9701–9707.

[5] C. Chen, U. Jain, C. Schissler, S. V. Amengual Gari, Z. Al-Halah, V. K. Ithapu, P. Robinson, and K. Grauman, "Soundspaces: Audio-visual navigation in 3d environments," in *Computer Vision – ECCV 2020*, ser. Lecture Notes in Computer Science, 2020.

[6] C. Chen, Z. Al-Halah, and K. Grauman, "Semantic audio-visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15 516–15 525.

[7] C. Chen, C. Schissler, S. Garg, P. Kobernik, A. Clegg, P. Calamia, D. Batra, P. Robinson, and K. Grauman, "Soundspaces 2.0: A simulation platform for visual-acoustic learning," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 8896–8911.

[8] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[9] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. S. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra, "Habitat 2.0: Training home assistants to rearrange their habitat," in *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[10] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, "Llama 2: Open foundation and fine-tuned chat models," *CoRR*, vol. abs/2307.09288, 2023.

[11] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[12] P. Chen, X. Sun, H. Zhi, R. Zeng, T. H. Li, G. Liu, M. Tan, and C. Gan, "a^2 nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models," *arXiv preprint arXiv:2308.07997*, 2023.

[13] M. Deitke, W. Han, A. Herrasti, A. Kembhavi, E. Kolve, R. Mottaghi, J. Salvador, D. Schwenk, E. VanderBilt, M. Wallingford *et al.*, "Robothor: An open simulation-to-real embodied ai platform," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3164–3174.

[14] Z. Yang, J. Liu, P. Chen, A. Cherian, T. K. Marks, J. Le Roux, and C. Gan, "Rila: Reflective and imaginative language agent for zero-shot semantic audio-visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 16 251–16 261.

[15] Q. Ye, Z. Yu, R. Shao, X. Xie, P. Torr, and X. Cao, "CAT: enhancing multimodal large language model to answer questions in dynamic audio-visual scenarios," in *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part X*, vol. 15068, 2024, pp. 146–164.

[16] Y. Li, H. Sun, M. Lin, and et al., "Baichuan-omni technical report," *CoRR*, vol. abs/2410.08565, 2024.

[17] J. Xu, Z. Guo, J. He, and et al., "Qwen2.5-omni technical report," *CoRR*, 2025.

[18] Z. Cheng, S. Leng, H. Zhang, Y. Xin, X. Li, G. Chen, Y. Zhu, W. Zhang, Z. Luo, D. Zhao, and L. Bing, "Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms," *CoRR*, vol. abs/2406.07476, 2024.

[19] OpenAI, "GPT-4 technical report," *CoRR*, vol. abs/2303.08774, 2023.

[20] J. Zhan, J. Dai, J. Ye, Y. Zhou, D. Zhang, Z. Liu, X. Zhang, R. Yuan, G. Zhang, L. Li, H. Yan, J. Fu, T. Gui, T. Sun, Y. Jiang, and X. Qiu, "Anygpt: Unified multimodal LLM with discrete sequence modeling," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, L. Ku, A. Martins, and V. Srikumar, Eds. Association for Computational Linguistics, 2024, pp. 9637–9662.

[21] J. Han, K. Gong, Y. Zhang, J. Wang, K. Zhang, D. Lin, Y. Qiao, P. Gao, and X. Yue, "Onellm: One framework to align all modalities with language," *CoRR*, vol. abs/2312.03700, 2023.

[22] S. Adavanne, A. Politis, and T. Virtanen, "Direction of arrival estimation for multiple sound sources using convolutional recurrent neural network," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 1462–1466.

[23] P.-A. Grumiaux, S. Kitić, L. Girin, and A. Guérin, "A survey of sound source localization with deep learning methods," *The Journal of the Acoustical Society of America*, vol. 152, no. 1, pp. 107–151, 2022.

[24] Z. Al-Halah, S. K. Ramakrishnan, and K. Grauman, "Zero experience required: Plug & play modular transfer learning for semantic visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[25] G. Tatiya, J. Francis, L. Bondi, I. Navarro, E. Nyberg, J. Sinapov, and J. Oh, "Knowledge-driven scene priors for semantic audio-visual embodied navigation," *arXiv preprint arXiv:2212.11345*, 2022.

[26] A. Younes, D. Honerkamp, T. Welschehold, and A. Valada, "Catch me if you hear me: Audio-visual navigation in complex unmapped environments with moving sounds," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 928–935, 2023.

[27] S. Paul, A. K. Roy-Chowdhury, and A. Cherian, "Avlen: Audio-visual-language embodied navigation in 3d environments," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 6236–6249.

[28] Q. Ye, W. Zeng, M. Liu, J. Zhang, Y. Hu, Z. Yu, and Y. Zhou, "When eyes and ears disagree: Can mllms discern audio-visual confusion?" in *AAAI Conference on Artificial Intelligence*, 2026.

[29] Q. Ye, Z. Yu, R. Shao, Y. Cui, X. Kang, X. Liu, P. Torr, and X. Cao, "Cat+: Investigating and enhancing audio-visual understanding in large language models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[30] Q. Ye and Z. Yu, "Pose-promote: Progressive visual perception for activities of daily living," *IEEE Signal Processing Letters*, 2024.

[31] Q. Ye, Y. Zhou, L. He, J. Zhang, X. Guo, J. Zhang, M. Tan, W. Xie, Y. Sun, T. Tan *et al.*, "Sugar: Learning skeleton representation with visual-motion knowledge for action recognition," in *AAAI Conference on Artificial Intelligence*, 2026.

[32] K. Zhou, K. Zheng, C. Pryor, Y. Shen, H. Jin, L. Getoor, and X. E. Wang, "Esc: Exploration with soft commonsense constraints for zero-shot object navigation," in *International Conference on Machine Learning*. PMLR, 2023, pp. 42 829–42 842.

[33] Y. Qiao, H. Hong, W. Lyu, D. An, S. Zhang, Y. Xie, X. Wang, and Q. Wu, "Navbench: Probing multimodal large language models for embodied navigation," *arXiv preprint arXiv:2506.01031*, 2025.

[34] C. Gao, L. Jin, X. Peng, J. Zhang, Y. Deng, A. Li, H. Wang, and S. Liu, "Octonav: Towards generalist embodied navigation," *arXiv preprint arXiv:2506.09839*, 2025.

[35] A. X. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *arXiv preprint arXiv:1709.06158*, 2017.

[36] C. Chen, S. Majumder, Z. Al-Halah, R. Gao, S. K. Ramakrishnan, and K. Grauman, "Learning to set waypoints for audio-visual navigation," in *International Conference on Learning Representations (ICLR)*, 2021.

[37] P. Wang, S. Bai, S. Tan, and et al., "Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution," *CoRR*, vol. abs/2409.12191, 2024.

[38] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu, "Qwen3-vl technical report," *arXiv preprint arXiv:2511.21631*, 2025.

[39] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Llava: Large language and vision assistant," *arXiv preprint arXiv:2304.08485*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly summarize the main contributions and are consistent with the scope and results presented in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the main limitations of the current approach and clarifies the scope in which the conclusions should be interpreted.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theoretical results are presented with their assumptions, and complete proofs are provided in the paper or supplementary material.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides sufficient methodological and experimental details for reproducing the main empirical findings.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code and data will be released upon acceptance.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The training setup, data splits, hyperparameters, and optimization details are reported clearly enough to understand the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports appropriate measures of experimental variability or statistical significance for the key results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper specifies the compute resources used, including the main hardware configuration and relevant execution details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research is conducted in accordance with the NeurIPS Code of Ethics and does not involve procedures that violate these guidelines.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses both beneficial applications and possible risks or negative societal impacts of the proposed work.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The paper describes safeguards and release considerations to reduce the risk of misuse of the data, models, or other assets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Existing datasets, models, and codebases are properly credited, and their licenses or terms of use are explicitly acknowledged.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We only used LLMs for minor language polishing and improving the clarity of some text descriptions in the manuscript.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.