This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

RoboTron-Nav: A Unified Framework for Embodied Navigation Integrating Perception, Planning, and Prediction

Yufeng Zhong† Chengjian Feng† Feng Yan Fanfan Liu Liming Zheng Lin Ma‡

Meituan

https://yvfengzhong.github.io/RoboTron-Nav

Abstract

In language-guided visual navigation, agents locate target objects in unseen environments using natural language instructions. For reliable navigation in unfamiliar scenes, agents should possess strong perception, planning, and prediction capabilities. Additionally, when agents revisit previously explored areas during long-term navigation, they may retain irrelevant and redundant historical perceptions, leading to suboptimal results. In this work, we propose ***RoboTron-Nav****, a unified framework that integrates perception, planning, and prediction capabilities through multitask collaborations on navigation and embodied question answering tasks, thereby enhancing navigation performances. Furthermore, RoboTron-Nav employs an adaptive 3D-aware history sampling strategy to effectively and efficiently utilize historical observations. By leveraging large language model, RoboTron-Nav comprehends diverse commands and complex visual scenes, resulting in appropriate navigation actions. RoboTron-Nav achieves an 81.1% success rate in object goal navigation on the* CHORES-$\mathbb{S}$ *benchmark, setting a new state-of-the-art performance.*

1. Introduction

Embodied navigation [1] is a key component of embodied artificial intelligence [12, 18, 19, 31, 38, 39], especially language-guided visual navigation [2, 5, 11, 16, 17, 24–26]. Unlike autonomous driving [13], language-guided visual navigation requires agents to comprehend natural language instructions and autonomously explore unseen visual environments to locate target objects, which presents unique challenges. Specifically, to reliably navigate in unfamiliar visual environments, agents must effectively perceive surrounding scenes, plan strategies to achieve their goals, and predict appropriate navigation actions.

† Equal contribution. ‡ Corresponding author.

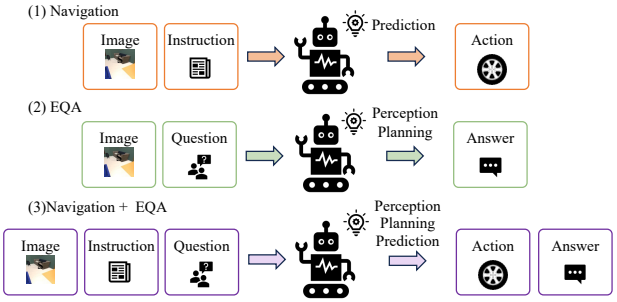


Figure 1. Multitask collaboration. In (1) navigation tasks, only actions are produced, missing the perception and planning found in (2) EQA tasks. (3) Multitask collaboration unifies perception, planning, and prediction for a more comprehensive model.

Recently, significant advancements in language-guided visual navigation have been driven by the robust understanding and generalization capabilities of visual-language models (VLMs) that seamlessly integrate visual and textual information. For example, in Object Goal Navigation (ObjectNav) [2–5, 7, 11, 24–26, 30, 36, 40], numerous approaches [3, 4, 7, 30, 36, 40] have achieved impressive results by using VLMs to align and integrate textual and visual inputs, enabling agents to better understand complex environments and navigate to target objects more effectively. Although current navigation models excel at locating target objects, they often struggle to provide efficient path planning and explain the reasoning behind their path choices, as shown in the *first row* of Fig. 1. This challenge is largely due to traditional navigation datasets focusing primarily on recording coordinate trajectories and neglecting the hierarchical reasoning process employed by humans. For instance, humans typically first identify rooms, then locate furniture, and finally search for specific objects.

Additionally, we have observed a significant amount of redundancy in observations during navigation, as agents frequently revisit the same locations multiple times. As illustrated at the *top* of Fig. 2, the white circle marks areas that the agent revisits multiple times. At (1) $T = 71$ and

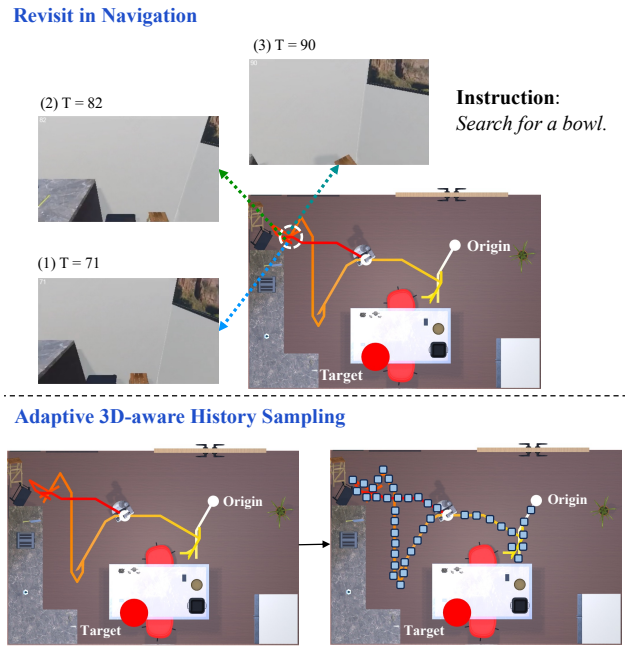


Figure 2. *Top:* During long-term navigation, agents may revisit the same areas multiple times, making previous observations redundant. For instance, at $T = 71$, $T = 82$, and $T = 90$, the scenes observed are very similar, highlighting the redundancy in the agent's perception. *Bottom:* By applying the Adaptive 3D-aware History Sampling strategy, we sample the dense and disorganized trajectory into sparse and organized steps, reducing observational redundancy while preserving visual information at key locations. The trajectory is produced by the SPOC-driven agent [10].

(2) $T = 82$, the agent focuses on the countertop area and two chairs. At (3) $T = 90$, the scene remains largely unchanged, capturing an environment similar to that at (1) and (2) once again. These observations demonstrate that the scenes within the revisit areas are highly similar, indicating the redundancy present in the agent's perception. Therefore, when agents revisit the same area, previous observations may become redundant and should be discarded.

This scenario raises two key questions driving our research: **(1) How can we clarify agents' decision-making processes? (2) How can we enhance long-term navigation efficiency?** To tackle these challenges, we propose RoboTron-Nav, a unified framework for embodied navigation that integrates perception, planning, and prediction. As illustrated in Fig. 1, RoboTron-Nav adopts a Multitask Collaboration strategy, leveraging joint training on both navigation and embodied question answering (EQA) tasks to fully harness perception and planning for improved prediction capabilities. Within this framework, the EQA task seamlessly integrates perception and planning by requiring agents to answer questions based on their historical observations and navigational trajectories. To facilitate this, we construct EQA datasets that *explicitly model the decision-making processes* involved in navigation, empowering agents to engage in deep reasoning and task planning akin to human cognition, and ultimately enhancing navigation interpretability. More details about our novel EQA dataset construction method are provided in Sec. 3.2.

Furthermore, RoboTron-Nav employs an Adaptive 3D-aware History Sampling strategy to effectively and efficiently minimize observation redundancy by controlling the density and diversity of historical sampling in both spatial and semantic domains. Specifically, we consider the RGB frames as valid observations only if they are either spatially non-adjacent to other frames, or spatially adjacent but captured from different viewpoints. In addition, RoboTron-Nav introduces *position-enhanced historical features* that utilize the agent's positions to augment historical semantic features with trajectory information, thereby preventing redundant exploration of the same location. As illustrated at the *bottom* of Fig. 2, we sample the dense and disorganized trajectory to obtain sparse and organized steps, minimizing redundancy while preserving crucial observations at key locations. Finally, by leveraging the large language model (LLM), RoboTron-Nav can understand diverse instructions and complex visual scenes, resulting in effective navigation. Through extensive experimentation, RoboTron-Nav has demonstrated remarkable capabilities, achieving an impressive 81.1% success rate (SR) in ObjectNav on the CHORES-$\mathbb{S}$ benchmark [10]. This performance not only sets a new state-of-the-art but also represents a substantial 9% absolute improvement over previous methods.

To summarize, we make the following contributions:

- We introduce RoboTron-Nav, a unified framework integrating perception, planning, and prediction, enhancing navigation by multitask collaboration on navigation and EQA tasks.
- RoboTron-Nav utilizes an adaptive 3D-aware history sampling strategy, which minimizes observational redundancy by regulating the density and diversity of historical sampling across both spatial and semantic domains.
- RoboTron-Nav achieves an 81.1% SR in ObjectNav on the CHORES-$\mathbb{S}$ benchmark, establishing a new state-of-the-art, with a 9% absolute improvement over previous methods.

2. Related Work

2.1. Object Goal Navigation

The ObjectNav [25] research landscape can be categorized into closed-world and open-world settings. Closed-world ObjectNav [2–5, 7, 11, 20, 24, 26, 30, 36, 40] focuses on locating objects from a fixed set of object categories (e.g., "chair"). Approaches include end-to-end methods [2, 7, 20, 26, 30, 36] that map sensory inputs directly to actions, and modular architectures [3–5, 11, 24, 40] that separate perception, mapping, and control. While effec-

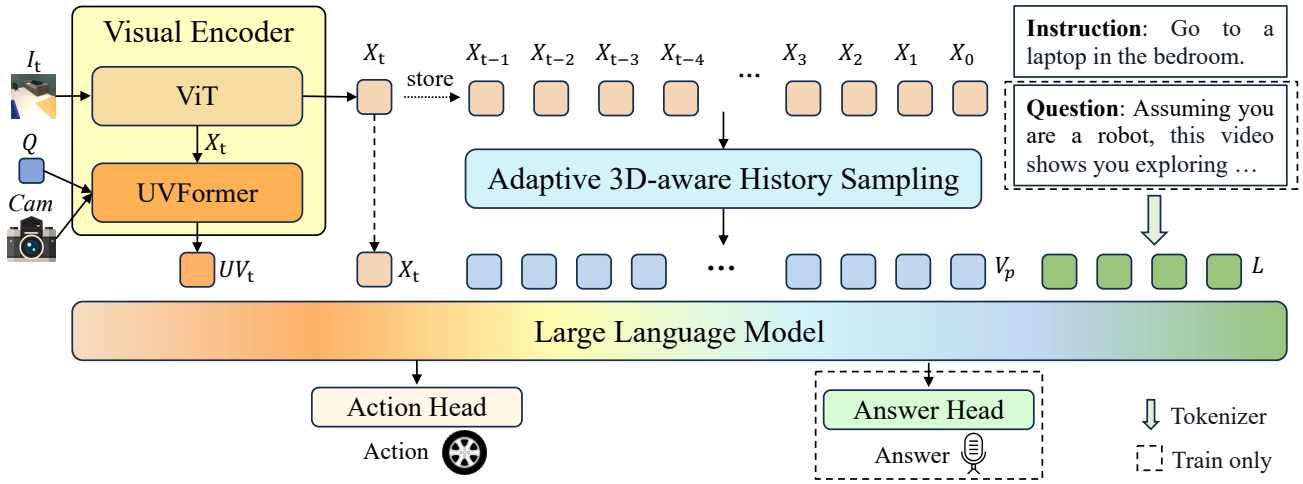


Figure 3. Overview of RoboTron-Nav architecture. The current frame I_t is initially processed through 2D and 3D feature extraction using the visual encoder. Then, historical features are filtered through the Adaptive 3D-aware History Sampling strategy. The visual features obtained, along with the accompanying linguistic instructions, are then fed into the large language model (LLM). Leveraging the Multitask Collaboration strategy, which significantly enhances navigation capabilities through joint training on both navigation and EQA tasks, the LLM produces two outputs via multimodal fusion: executable navigation actions and natural language answers.

tive in controlled settings, these methods are limited by their reliance on predefined object classes. To address this limitation, Open Vocabulary Object Goal Navigation (OVON) [32] allows users to specify any object using natural language (e.g., "find a television"), enabling greater adaptability to real-world scenarios. Although current ObjectNav models are effective at finding target objects, they often struggle to explain their navigation decisions.

2.2. Embodied Question Answering

EQA [6, 13] tasks require agents to explore 3D environments and answer visually grounded questions, integrating visual perception, exploration, and language understanding. For example, agents need to navigate physical spaces to answer questions such as "What color is the car?" The EQA task developed by the MT-EQA dataset [33] introduces scenarios involving multiple objects and locations, thereby increasing its complexity. Furthermore, the MP3D-EQA dataset [29] advances the field by incorporating photorealistic environments with point cloud perception, challenging agents to interact with detailed 3D spaces. Recently, the HM-EQA dataset [23] refines exploration strategies with an "explore until confident" approach, promoting efficient data gathering for accurate decision-making. These aforementioned developments collectively enhance the capabilities of agents in EQA tasks. Based on EQA datasets, some studies [35, 37] advance the field by unifying navigation and EQA tasks into a comprehensive model.

3. Methodology

In this section, we present RoboTron-Nav, a unified framework for embodied navigation that integrates perception, planning, and prediction. We first overview RoboTron-Nav and its core adaptive 3D-aware history sampling strategy (Sec. 3.1). Next, we introduce a method to augment navigation datasets with EQA pairs for explicit decision modeling (Sec. 3.2). Finally, we describe a multitask collaboration strategy (Sec. 3.3) that enhances navigation by jointly training on navigation and EQA tasks.

3.1. Model

As shown in Fig. 3, RoboTron-Nav integrates three components: a visual encoder for current observations, adaptive 3D-aware history sampling for context modeling, and an LLM for action and answer prediction.

3.1.1. Visual Encoder

Our visual encoder integrates 2D and 3D features to represent the current observation frame, with a dedicated emphasis on real-time perceptual cues. It consists of two specialized components: a ViT [21] for 2D feature extraction, and a UVFormer [18] for 3D spatial encoding.

First, we utilize the CLIP-pretrained ViT [21] as the image backbone for 2D feature extraction:

$$X_t = \mathrm{ViT}\left(I_t\right), \tag{1}$$

where $I_t \in \mathbb{R}^{H\times W\times 3}$ denotes the RGB image, and $X_t \in \mathbb{R}^{n_{\text{img}}\times c}$ represents their corresponding 2D features at timestep t. Here, n_{img} indicates the number of image patches while c denotes the feature dimension.

Additionally, inspired by [18], we implement multi-perspective alignment through unified view projection:

$$UV_t = \mathrm{UVFormer}\left(Q, X_t, Cam\right). \tag{2}$$

Here, the UVFormer [18] module processes three inputs: image features X_t, camera parameters Cam, and learnable queries Q that encode spatial positions and semantic features within the robot's 3D workspace. The output $UV_t \in \mathbb{R}^{n_{\text{UV}} \times c}$ aggregates multi-view visual information through a structured 3D feature volume, where n_{UV} denotes the number of 3D visual tokens. See the supplementary material for more details.

3.1.2. Adaptive 3D-aware History Sampling

Utilizing all observed RGB frames can substantially mitigate the issue of catastrophic knowledge forgetting, which is essential for long-term navigation. However, due to GPU memory constraints and high computational costs, storing all image features extracted from these frames is often impractical. Additionally, we have observed a significant amount of redundancy in observations during navigation, as the agent frequently revisits the same locations multiple times. To address this issue, we propose an adaptive 3D-aware history sampling strategy, which selects sparse and organized features based on historical position and viewpoint information, thereby reducing observational redundancy. As shown in Algorithm 1, we sample historical observations by using relative distance to determine spatial adjacency and semantic similarity to identify similar viewpoints. Our adaptive 3D-aware history sampling strategy operates through three sequential phases: initialization, 3D-aware sampling, and adaptive padding.

Initialization. The initialization phase (Lines 1-6) sets up the operational parameters by accepting the input parameters (current step t, window size W, relative position threshold ϵ, and semantic similarity threshold τ) and initializing empty buffers for historical features $\mathbf{V}$ and relative positions $\mathbf{P}$, respectively. The *GetAllObs* function returns all historical observations (i.e., $X_0, \ldots, X_{t-1}$ in Fig. 3), which are stored in $\mathbf{G}$ as a First-in-First-out (FIFO) queue, where $\mathbf{G}[0]$ corresponds to the initial time and $\mathbf{G}[-1]$ to the current time. At time step i, $\mathbf{G}[i]$ includes historical features $\mathbf{G}[i].v$ paired with their corresponding agent absolute positions $\mathbf{G}[i].p = (x, y, z)$.

3D-Aware Sampling. The core 3D-aware sampling phase (Lines 8-23) processes all historical observations in $\mathbf{G}$ in reverse chronological order, from newest to oldest. First, it uses the robot's absolute position p^{ref} at the current step t as the spatial reference to compute relative coordinates p_i^{rel} of historical features $\mathbf{G}[i].v$. Then, using the relative position threshold ϵ and the semantic similarity threshold τ, the method dynamically filters out redundant visual observation. Specifically, for each candidate feature at time step i, we compare it with all previously selected historical features at time step j: if $|p_i^{\text{rel}} - \mathbf{P}[j]| < \epsilon$ (i.e., adjacent in spatial position) and $\cos(\mathbf{G}[i].v, \mathbf{V}[j]) > \tau$ (i.e., captured from similar viewpoints), feature i is excluded from the historical set. By adjusting the value of ϵ and τ, we can control the density and diversity of historical sampling in the spatial and semantic domains, respectively. Concurrently, historical features $\mathbf{G}[i].v$ undergo dimensionality reduction via a $\mathbf{MaxPool}$ operator before being stored to $\mathbf{V}$. This phase terminates when the collected samples reach the target window size W or $\mathbf{G}$ is empty.

Adaptive Padding. The adaptive padding phase (Lines 24-28) ensures fixed-length outputs by replicating the last valid entries whenever the historical features buffer $\mathbf{V}$ does not reach the designated window size W. Through cyclic duplication of $\mathbf{V}[-1]$ and $\mathbf{P}[-1]$, it maintains temporal coherence while fulfilling model input requirements.

Afterwards, we introduce position-enhanced historical features that utilize the agent's positions to augment historical semantic features with trajectory information, preventing redundant exploration of the same location. For obtained historical features $\mathbf{V}$, positional information is criti-

Algorithm 1 Adaptive 3D-aware History Sampling

Input: Current step t, Window size W, Relative Position Threshold ϵ, Semantic Similarity Threshold τ
Output: Historical features $\mathbf{V}$, Relative positions $\mathbf{P}$

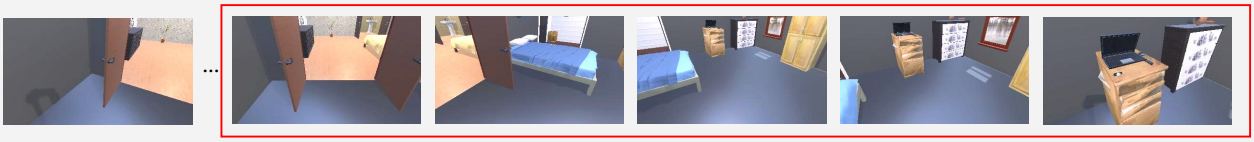
1: $\mathbf{V} \leftarrow \emptyset$, $\mathbf{P} \leftarrow \emptyset$
2: $\mathbf{G} \leftarrow \text{GetAllObs}()$
3: **if** $\mathbf{G} = \emptyset$ **then**
4: # Initialize with dummy data
5: **return** $\mathbf{V}$, $\mathbf{P}$
6: **end if**
7: # Current frame as reference
8: $p^{\text{ref}} \leftarrow \mathbf{G}[-1].p$
9: $k \leftarrow 0$
10: # From current time t to 0
11: **for** $i \leftarrow |\mathbf{G}| - 1$ **to** 0 **do**
12: $p_i^{\text{rel}} \leftarrow \mathbf{G}[i].p - p^{\text{ref}}$
13: **if** $i < |\mathbf{G}| - 1$ **and** $\exists\, j \in [0, k] :\ \|p_i^{\text{rel}} - \mathbf{P}[j]\|_2 < \epsilon$ **and** $\cos(\mathbf{G}[i].v, \mathbf{V}[j]) > \tau$ **then**
14: # Skip redundant frames
15: **continue**
16: **end if**
17: $\mathbf{V} \leftarrow \mathbf{V} \oplus \mathbf{MaxPool}(\mathbf{G}[i].v)$
18: $\mathbf{P} \leftarrow \mathbf{P} \oplus \mathbf{p}_i^{\text{rel}}$
19: $k \leftarrow k + 1$
20: **if** $k = W$ **then**
21: **break**
22: **end if**
23: **end for**
24: **while** $k < W$ **do**
25: # Padding with last valid frame
26: $\mathbf{V} \leftarrow \mathbf{V} \oplus \mathbf{V}[-1]$
27: $\mathbf{P} \leftarrow \mathbf{P} \oplus \mathbf{P}[-1]$
28: $k \leftarrow k + 1$
29: **end while**

Instruction-centric question

Instruction: go to a laptop in the bedroom

Question: Assuming you are a robot, this video shows you exploring a new indoor environment and using visual information to {**Instruction**}. Please **describe the surrounding scene** and your **path planning** to {**Instruction**}, and **analyze the reasons for this path planning based on common sense**. For example, when **searching for a bowl in the kitchen**, you need to first find the kitchen and then search for the bowl inside it; when **searching for a remote control**, you walk towards the living room because it is more likely to be found on the coffee table there.

Spatiotemporally coherent context construction



Structured responses

Answer: Based on the provided images, here is a **description of the surrounding scene** and the path planning to reach the laptop in the bedroom:

1. Initial Scene: The first image shows an open doorway leading into a room with a wooden floor. The room appears to be a bedroom, as there is a bed visible with **a yellow bedspread**.
2. Entering the Bedroom: Moving through the doorway, the second image shows another bed with **a blue bedspread**, indicating that this is a different bedroom or another angle of the same bedroom. **The room has a dark floor and walls**, with some furniture visible.
3. Identifying the Target: The third image shows a dresser with a laptop on top of it. This dresser is located against the wall, and there is another piece of furniture (possibly a chest of drawers) next to it.
4. Approaching the Laptop: The subsequent images show a closer view of the dresser with the laptop. The path involves moving directly towards the dresser, avoiding any obstacles such as the bed or other furniture.
5. Final Approach: The final images show the robot getting closer to the dresser, with the laptop becoming more prominent. The robot navigates to the front of the dresser where the laptop is placed.

Path Planning Analysis:

1. Initial Orientation: The robot starts by identifying the general layout of the room and locating the target object (the laptop).
2. Direct Path: The robot takes a direct path towards the laptop, as there are no significant obstacles in the way. This is efficient and minimizes the distance traveled.
3. Obstacle Avoidance: The robot **avoids the bed and other furniture** by navigating around them, ensuring a clear path to the dresser.
4. Visual Confirmation: As the robot moves closer, it continuously confirms the location of the laptop to ensure it is on the correct path.

Common Sense Reasoning:

1. Bedroom Context: **Laptops are commonly found in bedrooms on desks, dressers, or nightstands**. The robot uses this common knowledge to identify potential locations for the laptop.
2. Efficient Navigation: The robot chooses the most efficient path, avoiding unnecessary detours and focusing on reaching the target quickly.

In summary, the robot uses visual information and common sense reasoning to navigate the bedroom efficiently, identifying and approaching the dresser where the laptop is located. The path planning involves direct movement towards the target while avoiding obstacles and continuously confirming the target's location.

Figure 4. EQA dataset extension framework. The framework comprises three key stages: (1) standardized reasoning steps through instruction-centric question formulation, (2) decision context modeling via spatiotemporally coherent context construction, and (3) structured response generation integrating scene analysis, path planning, and commonsense reasoning. This hierarchical approach enables robots to systematically identify targets and formulate explainable navigation strategies while maintaining human-like decision transparency.

cal for recording the agent's historical trajectory and planning efficient paths for future exploration. Hence, we obtain position-enhanced historical features by using the agents' relative positions $\mathbf{P}$ as positional encodings for the historical features $\mathbf{V}$. This integration enriches spatial semantics with 3D trajectory information, ensuring observation consistency and effectively preventing redundant exploration of previously visited locations. As agents move on the ground, their positions p involve constantly changing x and y coordinates, while the z coordinate, representing height, remains constant. Consequently, we focus on encoding the agent's positions along the x and y axes to effectively capture their 3D trajectory. Inspired by [28], we propose a 2D encoding through axis-separated frequency projections:

$$\omega_k = e^{-2k(\log(10000))/d}, \quad d = c/2, \tag{3}$$

$$PE_x = \left[\sin(x\omega_{\lfloor m/2 \rfloor}), \cos(x\omega_{\lfloor m/2 \rfloor})\right]_{m=0}^{d-1}, \tag{4}$$

$$PE_y = \left[\sin(y\omega_{\lfloor n/2 \rfloor}), \cos(y\omega_{\lfloor n/2 \rfloor})\right]_{n=0}^{d-1}, \tag{5}$$

$$PE(x, y) = PE_x \oplus PE_y, \tag{6}$$

where $\oplus$ operation merges x/y-axis encodings into a unified spatial. We add the positional encoding $PE(x, y)$ to each historical feature in $\mathbf{V}$, and then pass these features through a fully connected layer to obtain the position-enhanced historical features $V_p \in \mathbb{R}^{n_{\text{his}} \times c}$, where n_{his} denotes the number of historical features.

3.1.3. LLM

Finally, we integrate the position-enhanced historical features V_p with the current observations UV_t and X_t from the visual encoder to create visual tokens for the LLM. In this framework, UV_t and X_t capture the agent's ongoing environment, while V_p offers context from past observations, effectively filtering out redundant positions. Similarly, input instructions or questions are transformed into language tokens $L \in \mathbb{R}^{n_{\mathrm{L}} \times c}$ via a tokenizer, where n_{L} denotes the length of the language tokens. These visual and language tokens are combined and fed into the LLM. The LLM utilizes its abilities in multimodal alignment and comprehension to process these tokens, decoding the outputs through specialized heads. For navigation datasets, it employs an action head to generate the executable action, while for EQA datasets, it uses an answer head (i.g., LLM head) to produce the natural language answer to input questions.

3.2. Embodied Question Answering

Current navigation models merely mimic human navigation trajectories, lacking deep thinking and task planning similar to human cognitive processes. For instance, humans typically first identify related rooms, then locate furniture, and finally search for specific objects. This lack of interpretability stems from traditional navigation datasets, which mainly focus on recording coordinate trajectories while neglecting the hierarchical reasoning used by humans. To address this issue, we have developed a new method for extending navigation datasets with EQA pairs that explicitly models the decision-making process involved in navigation.

As shown in Fig. 4, we instruct GPT-4o [14] to generate perception of the surrounding environment, navigation planning, and navigation analysis for each navigation objective. It consists of three stages: instruction-centric question, spatiotemporally coherent context construction, and structured responses.

Instruction-centric question. We construct an instruction-centric question to elicit comprehensive first-person navigation analysis (e.g., "*go to a laptop in the bedroom*") using two synergistic mechanisms: role specification ("*assuming you are a robot*") and exemplar-based prompting (two navigation reasoning and planning examples, see Fig. 4). This ensures structured outputs covering: (1) scene description (e.g., bedroom furniture layout), (2) path planning (e.g., obstacle avoidance), and (3) common sense reasoning (e.g., typical laptop locations).

Spatiotemporally coherent context construction. To ensure alignment with the adaptive 3D-aware history sampling strategy established in Sec. 3.1.2, we systematically select the final W frames from each navigation trajectory. This frame range captures critical target-approaching phases while maintaining temporal consistency with upstream visual processing modules. By adopting this temporal window, we construct spatiotemporally coherent contexts that encode successful navigation patterns and preserve motion dynamics essential for trajectory analysis.

Structured responses. Building on the prepared inputs, we synthesize instruction-aligned multimodal prompts by combining textual questions with curated visual sequences. GPT-4o [14] then generates structured responses in three phases. First, the surrounding scene description identifies architectural features (e.g., "*the room has a dark floor and walls*") and object semantics (e.g., "*a yellow/blue bedspread*"). Second, path planning analysis devises hierarchical navigation strategies (e.g., "*enter bedroom* $\rightarrow$ *circumvent bed* $\rightarrow$ *final approach*"). Third, common sense reasoning incorporates human-environment interaction norms (e.g., "*laptops are commonly found on desks, dressers, or nightstands*"). Each phase operates sequentially, ensuring explicit alignment between perception, action planning, and contextual knowledge.

3.3. Multitask Collaboration

To address the limitations inherent in single-task learning, which only encompasses navigation tasks, we introduce a multitask collaboration strategy that significantly enhances navigation capabilities through joint training on both navigation and EQA tasks. As shown in Fig. 3, during the training phase, this strategy enables the model to learn from both tasks concurrently, with inputs consisting of navigational instructions and EQA questions. Consequently, the model can effectively utilize the perceptual and planning skills derived from EQA pairs to refine and improve its navigation prediction capabilities. By integrating these tasks, the model gains a comprehensive understanding of spatial relationships and decision-making processes, which are critical for accurate navigation.

During inference, the model focuses solely on predicting navigation actions, effectively leveraging the perceptual and planning abilities acquired during training for precise and efficient navigation. Please refer to the supplementary material for details on the training objective.

4. Experiments

In this section, we first outline our experimental settings, then compare our proposed method with state-of-the-art techniques, and finally provide a detailed analysis of our approach, including qualitative results and ablation studies.

4.1. Experimental Settings

Dataset and Metrics. We evaluate our proposed method on the CHORES-$\mathbb{S}$ ObjectNav benchmark [10]. Besides, we extend the navigation dataset with EQA pairs for joint training, as described in Sec. 3.2. We follow the official evaluation split of the CHORES-$\mathbb{S}$ ObjectNav benchmark [10], which contains 200 trajectories in 200 testing

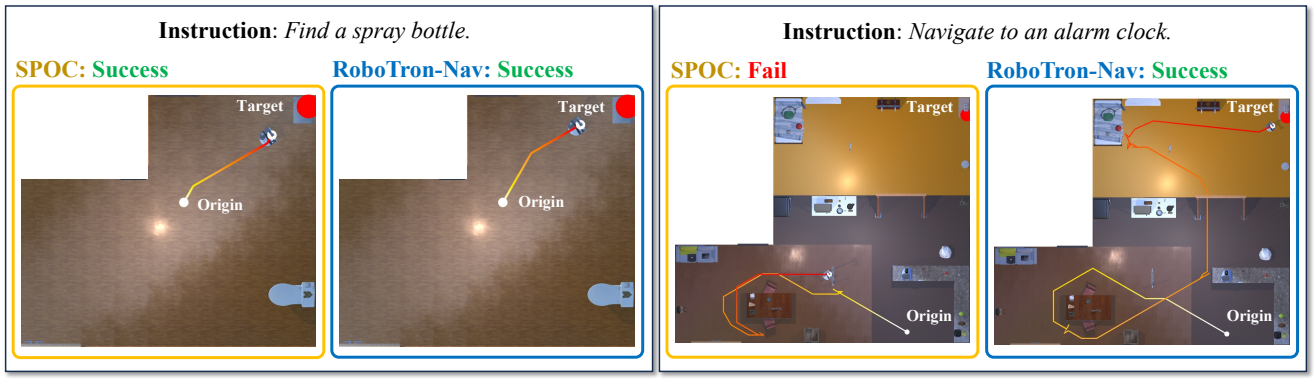


Figure 5. *Left:* Both RoboTron-Nav and SPOC [10] generate the shortest path when the agent is near the target (e.g., in the same room). *Right:* In contrast, SPOC [10] fails due to repeatedly searching along the same paths when distant from the target (e.g., in different rooms), whereas RoboTron-Nav succeeds by effectively avoiding revisiting areas and exploring new ones.

Method	SR ↑	SEL ↑	%Rooms
PIRLNav [22]	20.0	7.0	-
JSRL [27]	21.0	15.6	-
EmbSigLIP [15]	36.5	24.5	42.2
SPOC [10]	57.0	46.2	51.5
SPOC* [10]	60.0	30.5	-
RING [9]	72.1	**53.0**	-
RoboTron-Nav	**81.1**	43.9	67.8

Table 1. Performance comparison of different navigation methods on the CHORES-S ObjectNav benchmark [10].

houses. To comprehensively demonstrate the effectiveness of our method, we evaluate performance using three metrics: success rate (SR), episode-length weighted success (SEL) [8], and percentage of rooms visited (%Rooms) [10]. The additional explanations of the metrics and implementation details are provided in the supplementary materials.

4.2. Performance Comparisons

As shown in Table 1, we compare our method with PIRLNav [22], JSRL [27], EmbSigLIP [10], SPOC [10], SPOC* [10], and RING [9]. PIRLNav [22] and JSRL [27] are first pretrained with imitation learning, and then fine-tuned via reinforcement learning with sparse rewards. EmbSigLIP [10] denotes an EmbCLIP [15] model upgraded with a SigLIP [34] backbone. SPOC [10] imitates shortest paths by modeling optimal routes, while SPOC* [10] is trained on a larger set of expert trajectories. RING [9] introduces a universal indoor navigation policy transferable across diverse robots without additional fine-tuning.

As presented in Table 1, PIRLNav [22] and JSRL [27] achieve relatively low SRs (20.0 and 21.0), which can be attributed to the challenges of fine-tuning with sparse reward signals in reinforcement learning. EmbSigLIP [15] also demonstrates limited navigation performance (SR 36.5, SEL 24.5). SPOC [10] shows improvement, achieving an SR of 57.0 and a SEL of 46.2 by focusing on shortest path

Exp	HO	SP	PEHF	SR ↑	SEL ↑	%Rooms
1	✗	✗	✗	31.3	**16.7**	65.6
2	✓	✓	✗	39.2	14.4	77.7
3	✓	✗	✓	40.3	9.2	81.1
4	✓	✓	✓	**42.5**	16.0	62.3

Table 2. Ablation study on 3D-aware history sampling, where HO, SP, and PEHF stand for historical observations, sampling, and position-enhanced historical features, respectively.

Exp	benchmark	Navigation	EQA	SR ↑	SEL ↑	%Rooms
1	ObjectNavRoom	✓	✗	42.5	16.0	62.3
2	ObjectNavRoom	✓	✓	**47.2**	**21.9**	65.8
3	ObjectNav	✓	✗	72.5	33.5	68.4
4	ObjectNav	✓	✓	**81.1**	**43.9**	67.8

Table 3. Ablation on the multitask collaboration strategy.

learning. SPOC* [10] further increases SR to 60.0 via additional expert trajectory training, though at the cost of longer paths, reducing SEL to 30.5. RING [9] achieves a much higher SR of 72.1 and the best SEL of 53.0, demonstrating strong generalization and efficiency. RoboTron-Nav leverages adaptive 3D-aware history sampling and multitask collaboration to achieve the highest SR of 81.1, effectively utilizing historical data for improved path decision-making and enhancing environmental understanding through EQA tasks. These results demonstrate that RoboTron-Nav excels in long-term navigation, surpassing other methods by integrating planning across both navigation and EQA tasks.

4.3. Qualitative Results

To further illustrate the effectiveness of our unified framework for embodied navigation, some visualization results compared with SPOC [10] are illustrated in Fig. 5. In the left case, both RoboTron-Nav and SPOC [10] are capable of generating the shortest path when the agent is positioned close to the target, such as being in the same room. Both methods demonstrate a strong understanding of the envi-

Exp	W	ϵ	τ	SR ↑	SEL ↑	%Rooms
1	20	0.1	0	32.6	13.2	74.8
2	40	0.1	0	34.8	13.0	74.7
3	60	0.1	0	42.1	15.2	75.0
4	80	0.1	0	38.9	**16.7**	74.3
5	100	0.1	0	35.2	14.3	72.3
6	60	0.05	0	41.5	10.3	79.9
7	60	0.1	0	42.1	15.2	75.0
8	60	0.15	0	41.3	12.6	77.8
9	60	0.2	0	40.5	13.7	72.1
10	60	0.1	0.9	42.3	15.6	64.2
11	60	0.1	0.95	**42.5**	16.0	62.3
12	60	0.1	0.99	41.5	14.6	73.2

Table 4. Analysis of different hyper-parameters for the 3D-aware history sampling strategy.

ronment, allowing them to navigate efficiently without significant difficulties. However, the right case illustrates that when the agent is farther from the target, such as in different rooms, SPOC [10] faces challenges by repeatedly searching the same paths, leading to inefficiencies in navigating complex environments. In contrast, RoboTron-Nav avoids revisiting explored areas, enabling new path exploration and enhanced navigation efficiency, thus excelling in long-term navigation.

4.4. Ablation Studies

To demonstrate the effectiveness of different modules within our unified framework, we conduct ablation studies on the ObjectNavRoom benchmark [10], which has only 1/5 the sample size of ObjectNav. See the supplementary materials for more details about the ObjectNavRoom.

Does the adaptive 3D-aware history sampling help? As shown in Table 2, we conduct an ablation study on three key components in adaptive 3D-aware history sampling: historical observations (HO), sampling (SP), and position-enhanced historical features (PEHF). Exp. 1, which uses no historical observations ($W = 0$), relies solely on current visual input and yields the lowest SR, underscoring the value of historical information. Comparing Exp. 2 and 4, removing PEHF leads to a 2.9 decrease in SR, indicating that positional information helps avoid redundant exploration and improves efficiency. Similarly, between Exp. 3 and 4, omitting sampling in historical observations (i.e., $\epsilon = 0$, $\tau = 1$) causes a 2.2 drop in SR, showing that sampling reduces redundancy and enhances navigation accuracy.

Does the multitask collaboration strategy help? We conduct an ablation study on the multitask collaboration strategy to evaluate the benefits of joint training on both navigation and EQA tasks. As demonstrated in Table 3, Exp. 2, which additionally utilizes the EQA dataset during the training phase, achieves an increase of 4.7 in SR and 5.9 in SEL compared to Exp. 1, which only employs the navigation dataset. This result indicates that integrating the EQA dataset allows agents to leverage perception and planning to improve their prediction capabilities, thereby enhancing navigation abilities. Moreover, we have identified a remarkable improvement: Exp. 4, trained on the ObjectNav benchmark, which contains five times more samples than ObjectNavRoom, exhibits a **8.6** increase in SR compared to Exp. 3. This suggests that increasing the volume of EQA data can significantly enhance navigation performance.

Effect of hyper-parameters. We investigate the performances using different values of the window size W, the relative position threshold ϵ, and the semantic similarity threshold τ for the adaptive 3D-aware history sampling strategy. Here, W controls the length of historical context, while ϵ and τ jointly determine the density and diversity of historical sampling in the spatial and semantic domains, respectively. As shown in Table 4, we conducted comprehensive grid search experiments to analyze specifically how the window size W (20-100), relative position threshold ϵ (0.05-0.2), and semantic similarity threshold τ (0.9-0.99) affect performance. Increasing W initially improves performance by providing richer historical information, with the best results at $W = 60$ (around 6 meters). However, when W exceeds 60, the inclusion of excessive history starts to overwhelm the model, leading to reduced performance. For spatial sampling, increasing ϵ up to 0.1 improves performance by reducing spatial redundancy through lowering the sampling frequency, with optimal results at $\epsilon = 0.1$. However, further increases make sampling overly sparse and reduce performance. For semantic sampling, increasing τ up to 0.95 improves performance by preserving useful contextual information, with optimal results at $\tau = 0.95$. However, further increases introduce excessive redundancy and reduce performance. Therefore, we adopt $W = 60$, $\epsilon = 0.1$, and $\tau = 0.95$ for our adaptive 3D-aware history sampling strategy.

5. Conclusion

In this paper, we present the RoboTron-Nav framework, which significantly advances language-guided visual navigation by effectively integrating perception, planning, and prediction through multitask collaboration strategy. Additionally, RoboTron-Nav adopts an adaptive 3D-aware history sampling strategy to enhance navigation by efficiently utilizing historical data to minimize redundancy. The framework's impressive 81.1% success rate in ObjectNav on the CHORES-$\mathbb{S}$ benchmark underscores its effectiveness. Although RoboTron-Nav has achieved remarkable results in simulated environments, there is still a long way to go to achieve truly human-like embodied navigation. Future work will address these limitations by exploring the integration of larger, more diverse datasets to further enhance RoboTron-Nav's generalization capabilities across various environments.

References

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

[2] Tommaso Campari, Paolo Eccher, Luciano Serafini, and Lamberto Ballan. Exploiting scene-specific features for object goal navigation. In *European Conference on Computer Vision*, pages 406–421. Springer, 2020. 1, 2

[3] Tommaso Campari, Leonardo Lamanna, Paolo Traverso, Luciano Serafini, and Lamberto Ballan. Online learning of reusable abstract models for object goal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14870–14879, 2022. 1, 2

[4] Matthew Chang, Arjun Gupta, and Saurabh Gupta. Semantic visual navigation by watching youtube videos. *Advances in Neural Information Processing Systems*, 33:4283–4294, 2020. 1

[5] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33:4247–4258, 2020. 1, 2

[6] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–10, 2018. 3

[7] Heming Du, Lincheng Li, Zi Huang, and Xin Yu. Object-goal visual navigation via effective exploration of relations among historical navigation states. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2563–2573, 2023. 1, 2

[8] Ainaz Eftekhar, Kuo-Hao Zeng, Jiafei Duan, Ali Farhadi, Ani Kembhavi, and Ranjay Krishna. Selective visual representations improve convergence and generalization for embodied ai. *arXiv preprint arXiv:2311.04193*, 2023. 7

[9] Ainaz Eftekhar, Rose Hendrix, Luca Weihs, Jiafei Duan, Ege Caglar, Jordi Salvador, Alvaro Herrasti, Winson Han, Eli VanderBil, Aniruddha Kembhavi, et al. The one ring: a robotic indoor navigation generalist. *arXiv preprint arXiv:2412.14401*, 2024. 7

[10] Kiana Ehsani, Tanmay Gupta, Rose Hendrix, Jordi Salvador, Luca Weihs, Kuo-Hao Zeng, Kunal Pratap Singh, Yejin Kim, Winson Han, Alvaro Herrasti, et al. Spoc: Imitating shortest paths in simulation enables effective navigation and manipulation in the real world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16238–16250, 2024. 2, 6, 7, 8

[11] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2616–2625, 2017. 1, 2

[12] Yiyang Huang, Yuhui Hao, Bo Yu, Feng Yan, Yuxin Yang, Feng Min, Yinhe Han, Lin Ma, Shaoshan Liu, Qiang Liu, et al. Dadu-corki: Algorithm-architecture co-design for embodied ai-powered robotic manipulation. In *Proceedings of the 52nd Annual International Symposium on Computer Architecture*, pages 327–343, 2025. 1

[13] Zhijian Huang, Chengjian Feng, Feng Yan, Baihui Xiao, Zequn Jie, Yujie Zhong, Xiaodan Liang, and Lin Ma. Drivemm: All-in-one large multimodal model for autonomous driving. *arXiv preprint arXiv:2412.07689*, 2024. 1, 3

[14] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 6

[15] Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: Clip embeddings for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14829–14838, 2022. 7

[16] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 104–120. Springer, 2020. 1

[17] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020. 1

[18] Fanfan Liu, Feng Yan, Liming Zheng, Chengjian Feng, Yiyang Huang, and Lin Ma. Robouniview: Visual-language model with unified view representation for robotic manipulation. *arXiv preprint arXiv:2406.18977*, 2024. 1, 3, 4

[19] Daqin Luo, Chengjian Feng, Yuxuan Nong, and Yiqing Shen. Autom3l: An automated multimodal machine learning framework with large language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 8586–8594, 2024. 1

[20] Arsalan Mousavian, Alexander Toshev, Marek Fišer, Jana Košecká, Ayzaan Wahid, and James Davidson. Visual representations for semantic target driven navigation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8846–8852. IEEE, 2019. 2

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 3

[22] Ram Ramrakhya, Dhruv Batra, Erik Wijmans, and Abhishek Das. Pirlnav: Pretraining with imitation and rl finetuning for objectnav. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17896–17906, 2023. 7

[23] Allen Z Ren, Jaden Clark, Anushri Dixit, Masha Itkina, Anirudha Majumdar, and Dorsa Sadigh. Explore until confident: Efficient exploration for embodied question answering. *arXiv preprint arXiv:2403.15941*, 2024. 3

[24] Sohan Rudra, Saksham Goel, Anirban Santara, Claudio Gentile, Laurent Perron, Fei Xia, Vikas Sindhwani, Carolina Parada, and Gaurav Aggarwal. A contextual bandit approach for learning to plan in environments with probabilistic goal configurations. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5645–5652. IEEE, 2023. 1, 2

[25] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. 2

[26] William B Shen, Danfei Xu, Yuke Zhu, Leonidas J Guibas, Li Fei-Fei, and Silvio Savarese. Situational fusion of visual representation for visual navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2881–2890, 2019. 1, 2

[27] Ikechukwu Uchendu, Ted Xiao, Yao Lu, Banghua Zhu, Mengyuan Yan, Joséphine Simon, Matthew Bennice, Chuyuan Fu, Cong Ma, Jiantao Jiao, et al. Jump-start reinforcement learning. In *International Conference on Machine Learning*, pages 34556–34583. PMLR, 2023. 7

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 5

[29] Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied question answering in photorealistic environments with point cloud perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6659–6668, 2019. 3

[30] Wei Xie, Haobo Jiang, Shuo Gu, and Jin Xie. Implicit obstacle map-driven indoor navigation model for robust obstacle avoidance. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6785–6793, 2023. 1, 2

[31] Feng Yan, Fanfan Liu, Liming Zheng, Yufeng Zhong, Yiyang Huang, Zechao Guan, Chengjian Feng, and Lin Ma. Robomm: All-in-one multimodal large model for robotic manipulation. *arXiv preprint arXiv:2412.07215*, 2024. 1

[32] Naoki Yokoyama, Ram Ramrakhya, Abhishek Das, Dhruv Batra, and Sehoon Ha. Hm3d-ovon: A dataset and benchmark for open-vocabulary object goal navigation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5543–5550. IEEE, 2024. 3

[33] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L Berg, and Dhruv Batra. Multi-target embodied question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6309–6318, 2019. 3

[34] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 7

[35] Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024. 3

[36] Sixian Zhang, Xinhang Song, Weijie Li, Yubing Bai, Xinyao Yu, and Shuqiang Jiang. Layout-based causal inference for object navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10792–10802, 2023. 1, 2

[37] Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13624–13634, 2024. 3

[38] Liming Zheng, Feng Yan, Fanfan Liu, Chengjian Feng, Zhuoliang Kang, and Lin Ma. Robocas: A benchmark for robotic manipulation in complex object arrangement scenarios. *arXiv preprint arXiv:2407.06951*, 2024. 1

[39] Liming Zheng, Feng Yan, Fanfan Liu, Chengjian Feng, Yufeng Zhong, Yiyang Huang, and Lin Ma. Dataplatter: Boosting robotic manipulation generalization with minimal costly data. *arXiv preprint arXiv:2503.19516*, 2025. 1

[40] Minzhao Zhu, Binglei Zhao, and Tao Kong. Navigating to objects in unseen environments by distance prediction. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10571–10578. IEEE, 2022. 1, 2