ONEDAY: TOWARDS FULL-DAY PERSONALIZED EMBODIED AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied agents powered by multimodal large language models (MLLMs) have shown strong open-vocabulary understanding and long-horizon planning. Yet, existing benchmarks largely focus on single-episode instruction following, where goals are fully specified and horizons are short. In contrast, practical household assistance demands continuous, day-long support that respects user preferences, satisfies temporal constraints, and adapts to evolving context. To close this gap, we introduce **OneDay-Bench**, a benchmark for full-day personalized assistance, consisting of 100 diverse daily plans paired with structured user profiles and human-annotated preference memories. Our benchmark provides comprehensive metrics spanning personalization alignment, task completion, and time management, and validates action executability via grounded interaction in simulation. Building on this benchmark, we present **OneDay-Agent**, an MLLM-driven embodied assistant that executes underspecified natural-language daily plans in dynamic home environments. The agent performs schedule-aware goal synthesis by jointly reasoning over the plan, current observations, and retrieved user preferences, producing temporally ordered and executable actions. Together, we aim to establish a new testbed and baseline for genuinely personalized, day-long embodied assistance. All materials will be open-sourced.

1 INTRODUCTION

Recent advances in multimodal large language models (MLLMs) have accelerated embodied agents, enabling open-vocabulary understanding and long-horizon decision-making for home operations and navigation. Modern systems couple MLLM reasoning with skill libraries, closed-loop feedback, or code generation to execute extended tasks in interactive environments (Ahn et al., 2022; Huang et al., 2022; Liang et al., 2022; Song et al., 2023; Wang et al., 2023). In parallel, large-scale simulators and benchmarks have expanded task diversity and realism, sharpening our understanding of what remains difficult in long-horizon embodied execution (Srivastava et al., 2021; Li et al., 2024a; 2021).

Despite this progress, most agents are still designed and evaluated in single-task settings where goals are fully specified, the horizon is short, and user preferences are ignored. In contrast, a practical home assistant must operate continuously across a day: coordinating multiple interconnected activities, respecting personal preferences, and adhering to time constraints – capabilities that are not yet captured by existing benchmarks or agent architectures.

To bridge this gap, we propose **OneDay**, a full-day personalized embodied assistant with two key components: OneDay-Bench, a benchmark where agents must interpret underspecified daily plans with structured user profiles and perform personalized household chores across multiple time points; and OneDay-Agent, an MLLM-based system that synthesizes daily plans, scene observations, and retrieved user preferences to generate chronologically ordered, executable actions for continuous full-day support in dynamic home environments.

As illustrated in Fig. 1, our benchmark is constructed by composing real-world family scenarios into coherent daily plans. Each plan is formed by sampling 2–4 segments within the same scenario and pairing it with (i) a structured user profile encoding object-level preferences and behavioral habits, and (ii) manually labeled preference memories capturing precise object semantics and group associ-

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

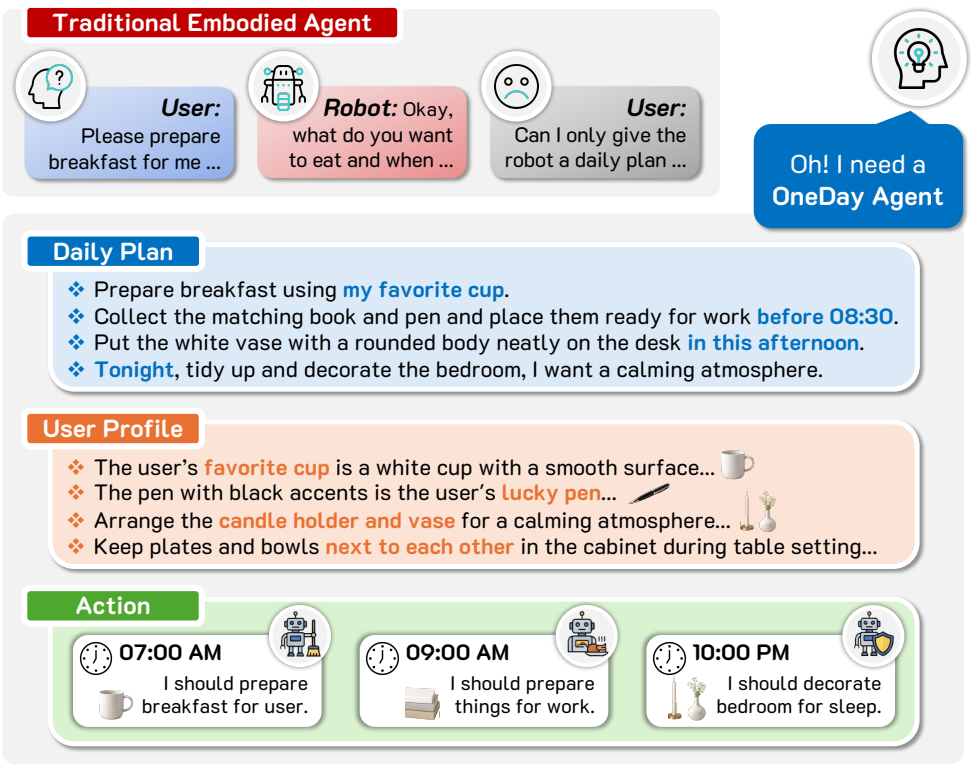


Figure 1: Overview of the OneDay framework. Traditional embodied agents require complete user instructions and detailed task specifications, and often lack full-day execution capabilities. In contrast, our approach requires only an underspecified daily plan expressed in free-form natural language and a user profile to generate a personalized, chronologically ordered sequence of activities throughout the day.

ations. To ensure object reliability, we filter resources that appear fewer than twice or in fewer than 10% of scene-level segments, resulting in 100 distinct daily instances. Each instance contains an underspecified daily plan, a structured user profile, and real-world actions. OneDay-Bench evaluates agents along five complementary dimensions – personalization score, task completion, time management, schedule consistency, and action execution – and aggregates them into a comprehensive final score with an explicit penalty for repetitive behaviors.

Building on this benchmark, we propose OneDay-Agent, a closed-loop assistant driven by an MLLM. At each time step t, the agent consumes four inputs: an underspecified [1]*Daily Plan* P, a structured [2]*User Profile* U, the [3]*Current Scene Observations* O_t, and the [4]*Accumulated Agent Memory* M_t that records completed activities. It performs perceptual schedule goal synthesis to identify the activity required at time t, and then generates a chronologically ordered sequence of low-level skill calls (e.g., navigation, pick-up, place-down) to execute the activity in simulation. After execution, the agent updates its memory via $M_{t+1} \leftarrow M_t \cup \{a_{\text{act}}\}$, which serves as both a global progress tracker for coherent multi-activity planning and a deduplication mechanism to prevent re-executing completed activities over the day.

To assess the current state of full-day personalized embodied assistance, we conduct extensive experiments using a diverse set of MLLMs, covering state-of-the-art closed-source and open-source models, and evaluating them across all five dimensions. The results reveal substantial gaps: models that are strong at language-level preference understanding can be inconsistent in scheduling; models with better time awareness may fail to map preferences to the correct object instances; and even the strongest models often struggle to translate high-level plans into complete, physically executable action sequences in simulated environments. These findings indicate that existing MLLMs remain far from being reliable and personalized OneDay-Agent. Our main contributions are:

- **OneDay-Bench:** A comprehensive benchmark for evaluating embodied agents on full-day personalized assistance. It contains 100 daily instances with structured user profiles and manually labeled preference memories, and evaluates performance across five dimensions: [1]personalization fit, [2]task completion, [3]time management, [4]schedule consistency, and [5]action executability.

- **OneDay-Agent:** A preference-conditional, MLLM-driven embodied assistant that integrates structured preference memories with schedule-aware goal synthesis and persistent agent memory, enabling coherent, personalized, reliable and non-redundant execution in a dynamic home environment.

- Extensive evaluations of state-of-the-art MLLMs on our benchmark, with failure mode analysis and ablations that expose key bottlenecks and motivate future full-day personalized embodied assistants.

2 Related Works

Embodied Agents & Long-Horizon Benchmarks. MLLMs have emerged as effective high-level planners for embodied tasks, commonly paired with skill libraries, code generation, or closed-loop feedback. Representative directions include grounding language through affordance/value models (Ahn et al., 2022), leveraging natural-language self-feedback (Huang et al., 2022), compiling instructions into policy code (Liang et al., 2022), and few-shot grounded planning for household activities (Song et al., 2023). Open-ended agents further study continual skill acquisition and reuse (Wang et al., 2023). Realistic evaluation of household assistance relies on interactive simulators and diverse task suites (Puig et al., 2023; Savva et al., 2019; Deitke et al., 2020; Kolve et al., 2017; Deitke et al., 2022). BEHAVIOR introduced predicate-logic specifications for everyday activities (Srivastava et al., 2021) on iGibson (Li et al., 2021), and BEHAVIOR-1K scales this to 1,000 activities with broader scene coverage (Li et al., 2024a). Other benchmarks emphasize planning and coordination in household environments (Chang et al., 2024). However, agents are still predominantly evaluated on single-episode objectives, and existing benchmarks rarely score user preference satisfaction or schedule adherence over continuous, multi-activity daily plans. **OneDay** targets this gap by unifying preference grounding, temporal scheduling, and physical execution in a full-day evaluation framework.

Personalization & Preference-Guided Planning. Personalized assistance has been explored via learning user-specific organization rules or navigation targets (Kwon et al., 2025a). TidyBot learns tidying preferences from a small number of interactions and generalizes them to new objects and scenes (Wu et al., 2023), while earlier work models tidying preferences through structured latent representations (Kapelyukh & Johns, 2021). Interactive personalized navigation studies how agents identify user-specific objects through dialogue and exploration (Dai et al., 2023). Preference-based Planning formalizes learning and using personalized preferences for plan generation across diverse preference types (Xu et al., 2025). Nevertheless, prior personalization settings typically assume isolated tasks or short horizons, and do not jointly handle temporal constraints, dynamic scene updates, and persistent resolution of underspecified natural-language references against a user profile over an entire day.

Memory for Long-Term & Embodied Agents. Long-term memory is central to MLLM agents, including experience streams with reflection for human-like simulation (Park et al., 2023), tiered memory management (Packer et al., 2023), and long-term user modeling for consistent interactions (Zhong et al., 2023). In embodied scenarios, non-parametric memory can support retrieval and grounded generation (Xie et al., 2024), and hybrid long/short-term memory has been shown to improve planning efficiency in household tasks (Wang et al., 2024). Recent personalization evaluations further indicate that embodied agents still struggle to reliably use episodic memories for routines and multi-memory composition (Kwon et al., 2025b). However, existing memory mechanisms are not designed to deduplicate completed activities, enforce temporal ordering, and maintain preference alignment across an entire day of embodied interaction.

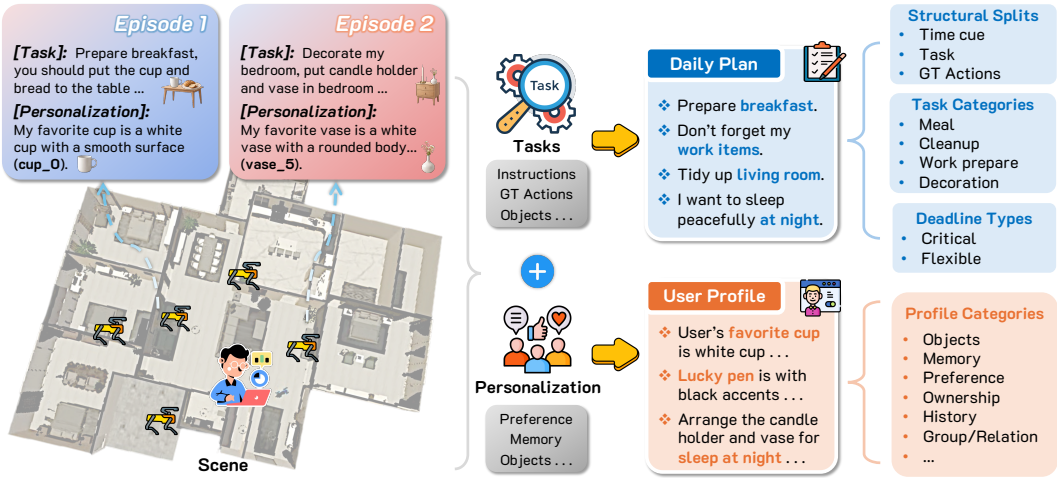


Figure 2: Overview of OneDay-Bench data generation pipeline. Individual scene episodes, each containing task descriptions and personalization information, are sampled and combined into a coherent underspecified daily plan. Object-specific memories are transformed into fuzzy natural language references (e.g., "my favorite cup"), while the corresponding precise object semantics and behavioral patterns are retained in the structured user profile for agent retrieval during evaluation.

3 METHODOLOGY

3.1 OVERVIEW

To study full-day personalized embodied assistance, we introduce **OneDay-Bench** for evaluation and **OneDay-Agent** as a preference-conditioned assistant that can carry out an entire day of tasks. At each time step t, the agent receives an input tuple $\mathcal{I}_t = \{P, \mathcal{U}, \mathcal{O}_t, \mathcal{M}_t\}$, where P is an underspecified natural-language daily plan, $\mathcal{U}$ is a structured user profile encoding object-level preferences and behavioral habits, $\mathcal{O}_t$ denotes current scene observations (e.g., visible objects and their states), and $\mathcal{M}_t$ is the accumulated memory of completed subtasks. The agent outputs $\mathcal{A}_t = \{G_t, S_t\}$, where $G_t = \{g_1, g_2, \ldots, g_k\}$ is the temporally ordered set of subtasks inferred for the current checkpoint, and S_t is a sequence of executable low-level skill calls (e.g., navigate, pick, place) that realizes these subtasks in simulation. This process is summarized as:

$$\mathcal{A}_t = f_\theta\left(P,\ \mathcal{U},\ \mathcal{O}_t,\ \mathcal{M}_t\right), \tag{1}$$

where f_θ denotes the MLLM-based **OneDay-Agent**. This model runs in a **closed loop across the day**: after executing $\mathcal{A}_t$, it updates $\mathcal{M}_t \rightarrow \mathcal{M}_{t+1}$ to reflect completed activities and any preference/schedule changes, enabling consistent and adaptive behavior from morning to night.

3.2 ONEDAY-BENCH: TOWARDS FULL-DAY PERSONALIZED INTELLIGENCE

Data Generation. As shown in Fig. 2, our benchmark composes real-life family episodes into coherent one-day plans. Each plan is created by sampling 2–4 episodes from the same scene, where each episode corresponds to a distinct activity grounded in real object interactions. For every selected scene, we extract **three types of structured information**: **(1)** target objects with titles, asset identifiers, and semantic memories (e.g., ownership, preferences, history); **(2)** source/target furniture derived from the scene's `name_to_receptacle` mapping and `evaluation_propositions`; and **(3)** executable ground-truth actions obtained from the initial state and container layout (e.g., PICK_AND_PLACE, NAVIGATE). We then match these event fragments to activity templates (e.g., preparing breakfast, getting ready for work, evening decorating). Templates come from eight predefined types, each specifying a task category (tight deadline vs. flexible time) and a non-overlapping time window.

To generate underspecified natural-language daily plans, we convert object memories into ambiguous referents via rule-based transformations. For instance, a **"preference"** object becomes *"my*



Figure 3: Overview of OneDay-Agent. At each time checkpoint, the agent takes an underspecified daily plan, a structured user profile, current scene observations, and agent memory as input, and generates a chronologically ordered action sequence for the current activity. The agent memory is updated after each execution to ensure non-redundant full-day scheduling, and the outputs are evaluated across four complementary metrics.

favorite cup", and a **"history"** object becomes *"the cup I use when traveling"*; while deliberately omitting asset identifiers so that the agent must resolve them through the user profile. The resulting plan is a paragraph describing 2–4 activities with time cues (e.g., *"morning"*, *"before 8:30"*, *"tonight"*). The paired user profile records the exact object semantics, behavioral patterns, and group associations needed to disambiguate these referents. To ensure reliability at evaluation, we filter objects that appear fewer than 2 times or in fewer than 10% of scene-level episodes, ensuring required objects are present when needed. This pipeline yields 100 daily instances, each containing an underspecified daily plan, a structured user profile (preference memory), and ground-truth actions.

Evaluation Metrics & Protocols. We evaluate five complementary dimensions: personalization, task completion, time management, grounded task success, and repetitive task penalties. Scores are aggregated within each day and then across the 100 daily instances. The final score combines the four performance scores with a repetition penalty; full definitions, equations, aggregation rules, and time-constraint handling are provided in Appendix A. The experimental configuration and composite score are also specified in Section 4.

3.3 ONEDAY-AGENT

Pipeline. Based on the formal definition in Sec. 3.1, our agent framework operates as a closed-loop, MLLM-based assistant, processing inputs and generating structured action plans at each time checkpoint. As shown in Fig. 3, at each time checkpoint t, the agent receives four inputs: an underspecified daily plan P, a structured user profile $\mathcal{U}$, current scene observations $\mathcal{O}_t$ (visible objects and their locations), and agent memories $\mathcal{M}_t$ recording previously completed activities. The agent first performs schedule-based goal synthesis: it integrates the current time, the daily plan, and retrieved user preferences to determine the activities to be performed at this checkpoint. Then, it

generates an output tuple:

$$\mathcal{A}_t = (a_{\text{act}}, \mathbf{s}_t), \tag{2}$$

where a_{act} is the name of the selected activity (NONE if no operation is required), and $\mathbf{s}_t = \langle o_1, o_2, \ldots, o_k \rangle$ is a chronological sequence containing at most 8 basic skill calls taken from {`Navigate`, `PickAndPlace`, `Pick`, `Place`, `Open`, `Close`}. After execution, the agent updates the memory as: $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t \cup \{a_{\text{act}}\}$, and the updated memory is passed to the next checkpoint to prevent the repeated execution of completed activities.

Prompt Design. The agent is driven by structured system prompts that enforce a strict output format for reliable, comprehensive evaluation. Specifically, the prompts instruct the agent to infer the following information: (1) the current time; (2) the actual activity name at the current checkpoint; (3) the history of completed activities from $\mathcal{M}_t$; and (4) the user profile and a rough daily schedule. The output follows a fixed four-field pattern:

$$\underbrace{\texttt{Thought:}}_{\text{reasoning}} \quad \underbrace{\texttt{Activity:}}_{\text{decision}}$$
$$\underbrace{\texttt{Tool:} \quad o_1, \ldots, o_k}_{\text{action sequence}} \quad \underbrace{\texttt{Done[]}}_{\text{termination}} . \tag{3}$$

The `Activity` field must be taken from the predefined activity name in the daily plan; the output `NONE` indicates that the agent determines that no action needs to be performed at this checkpoint. This structured format decouples the agent's linguistic reasoning from downstream evaluation, allowing the evaluator to independently parse and score each field.

Memory Construction. The OneDay-Agent maintains a structured agent memory $\mathcal{M}_t$ to accumulate the task execution history for the entire day. At each checkpoint, $\mathcal{M}_t$ is serialized into a natural language summary listing all completed activities and their execution timestamps, for example: "I completed 'Prepare Breakfast' at 07:00 and 'Prepare Work' at 08:00." This memory serves two purposes: (1) it provides the agent with a global view of the day's progress to support coherent multi-activity planning; (2) it acts as a deduplication protection mechanism—system prompts explicitly instruct the agent never to repeat any activity already existing in $\mathcal{M}_t$, thereby directly suppressing repetitive task behavior and avoiding a decrease in the final score due to the N_{repeat} value. The user profile $\mathcal{U}$ is treated as a separate persistent memory, which is fully retrieved at each checkpoint, thus ensuring that object-level preferences and behavior patterns are always accessible throughout the day without performance degradation in long-term contexts.

4 Experiments

4.1 Experimental Settings

Implementation Details. We evaluate a diverse set of state-of-the-art large models spanning closed-source, open-source, and embodied-specific categories. Closed-source models include GPT-5 (OpenAI, 2025b), Claude-Sonnet-4.5 (Anthropic, 2025), Qwen-Max (Alibaba Cloud, 2026), and Gemini-2.5-Pro (Comanici et al., 2025). Open-source models cover a broad range of scales (7B–309B), including DeepSeek-R1 (DeepSeek-AI et al., 2026), the Qwen series (Qwen et al., 2025; Yang et al., 2025), MiMo-V2-Flash (Xiao et al., 2026), etc.. We also include an embodied-specific model, RoboBrain2.0 (Team et al., 2025), to test whether domain-specific training benefits personalized household assistance. All models use identical system instructions and are provided with the daily plan, user profile, current scene observations, and agent memory at each checkpoint, without any model-specific fine-tuning. Due to space limits, additional implementation details are deferred to the **Appendix**.

Evaluation Protocols. We report the five metrics defined in Appendix A. Personalization score, task completion score, and time management score are computed from the predicted action sequences, while task success rate is validated through grounded interaction in the Habitat 3.0 simulator: predicted actions are executed in simulation, and success is determined by whether the resulting object states satisfy the evaluation propositions. The repetitive task penalty count (N_{repeat}) accumulates when an agent re-selects activities already completed earlier in the day. The final comprehensive score is computed as follows:

$$\mathcal{F} = 100 \times \left(0.2\,\bar{P} + 0.1\,\bar{C} + 0.2\,\bar{T} + 0.5\,\bar{S}\right) - \lambda \times N_{\text{repeat}}, \tag{4}$$

Table 1: Benchmarking results of all evaluated models on OneDay-Bench. Results are reported using five component metrics: Personalization Score, Task Completion, Time Management, Task Success Rate, and Repetition Penalty Count, together with the resulting Final Score (columns: Pers., Compl., Time, Success, Repeat, and Final). Boldface and underlining indicate the best and second-best results, respectively.

Model	Model Size	Pers. ↑	Compl. ↑	Time ↑	Success ↑	Repeat ↓	Final ↑
Closed-Source Models							
GPT-5	-	52.65	46.69	71.75	26.04	3.00	42.48
GPT-5.2	-	52.90	50.60	67.27	24.30	**0.00**	41.24
GPT-5-mini	-	53.03	46.95	60.37	24.06	**0.00**	39.41
Claude-Sonnet-4.5	-	**88.79**	63.18	**89.40**	29.69	122.00	53.14
Qwen-Max	-	68.32	67.31	80.02	<u>44.73</u>	73.00	<u>56.57</u>
Gemini-2.5-Pro	-	11.07	10.11	34.40	5.51	**0.00**	12.86
Open-Source Models							
DeepSeek-R1	671B	62.73	56.28	77.99	29.68	<u>9.00</u>	48.34
Qwen2.5	7B	74.58	56.47	80.64	22.00	245.00	40.34
Qwen3	8B	84.90	<u>76.60</u>	85.91	42.98	241.00	56.08
Qwen3	30B	<u>87.37</u>	**77.76**	<u>88.72</u>	44.49	393.00	53.45
Qwen2.5	72B	76.35	70.59	82.12	38.15	263.00	49.94
MiMo-V2-Flash	309B	81.68	76.29	89.17	38.15	77.00	**58.56**
InternVL3.5	8B	83.22	75.90	83.10	**45.51**	470.00	49.51
LLaVA-NeXT	8B	60.91	62.19	75.29	29.76	375.00	37.09
Embodied-Specific Models							
RoboBrain2.0	7B	46.76	48.67	69.03	20.14	314.00	28.67
MiMo-Embodied	7B	61.08	56.13	70.92	28.25	50.00	44.64
RynnBrain	8B	77.76	75.79	81.80	45.04	349.00	51.54

where the weights reflect the relative importance of personalization and task success rate in actual family support. We set the repetition penalty coefficient $\lambda = 0.03$ to balance the impact of repetitive behaviors against other performance dimensions, and N_{repeat} is the total number of repetitive activity selections during the entire 100-day assessment period.

4.2 Benchmarking Results & Analyses

Tab. 1 and Fig. 5 (Appendix B) summarize results for benchmarked models in **OneDay-Bench**. **Personalization & Time Management.** Many models exhibit strong language-level preference understanding and time awareness. Claude-Sonnet-4.5 (Anthropic, 2025) achieves the best personalization (88.79) and time management (89.40), followed by the Qwen3 series and MiMo-V2-Flash (Xiao et al., 2026) (81.68). Among embodied-specific models, RynnBrain (Dang et al., 2026) attains competitive personalization (77.76) and time management (81.80), suggesting that domain training can help resolve ambiguous user references. In contrast, Gemini-2.5-Pro (Comanici et al., 2025) performs poorly on both dimensions (11.07 and 34.40), indicating unstable preference grounding. These results suggest that while modern MLLMs can effectively parse and understand user preferences from natural language profiles, successfully grounding these preferences to specific objects in complex scenes remains a key differentiator between high-performing and low-performing models.

Task Completion & Action Execution. Performance drops sharply when evaluated via Habitat 3.0 simulator (Puig et al., 2023) execution. All models achieve task success rates below 45%, and even the strongest performers: Qwen-Max (Alibaba Cloud, 2026) (44.73) and InternVL3.5 (Wang et al., 2025) (45.51), fail on most activities in simulation. Embodied-specific models do not show a consistent advantage: RoboBrain2.0 (Team et al., 2025) reaches only 20.14, indicating that domain-specific training alone does not bridge the gap between high-level preference understanding and low-level physical execution. **Repetition Penalty & Schedule Memory.** A salient failure pattern is excessive repetition among many open-source models. InternVL3.5 (Wang et al., 2025) accumulates 470 repetitions and Qwen3-30B (Yang et al., 2025) accumulates 393, severely reducing their final scores. Closed-source models are generally more disciplined, with the GPT series showing near-zero repetitions. DeepSeek-R1 (DeepSeek-AI et al., 2026) is an exception among open-source models, with only 9 repetitions, plausibly benefiting from its reasoning-oriented design. Among embodied-

Table 2: Ablation study on user profile completeness. The results demonstrate the impact of progressively richer profile information on personalization and task performance. Pers., Time, Success, and Final denote Personalization Score, Time Management, Task Success Rate, and Final Score, respectively.

Model	Size	Profile Setting	Pers. ↑	Time ↑	Success ↑	Final ↑
Claude-Sonnet-4.5	-	*w/o* Profile	31.24	72.35	10.83	24.17
		Ownership Only	54.37	81.62	17.42	35.63
		Preference Only	72.81	86.18	23.15	44.29
		Full Profile	**88.79**	**89.40**	**29.69**	**53.14**
GPT-5	-	*w/o* Profile	28.91	56.42	11.26	20.35
		Ownership Only	39.48	64.83	18.94	31.72
		Preference Only	46.27	68.51	22.37	37.16
		Full Profile	**52.65**	**71.75**	**26.04**	**42.48**
Qwen3-8B	8B	*w/o* Profile	33.47	68.54	9.61	21.38
		Ownership Only	51.82	77.83	18.74	34.72
		Preference Only	67.39	82.76	30.21	46.85
		Full Profile	**84.90**	**85.91**	**42.98**	**56.08**
MiMo-V2-Flash	309B	*w/o* Profile	36.52	74.28	12.37	25.61
		Ownership Only	53.18	82.45	21.84	38.94
		Preference Only	66.73	86.31	31.46	48.37
		Full Profile	**81.68**	**89.17**	**38.15**	**58.56**
MiMo-Embodied	7B	*w/o* Profile	27.84	54.76	8.13	18.92
		Ownership Only	38.61	62.48	14.52	28.37
		Preference Only	49.37	67.35	21.08	36.74
		Full Profile	**61.08**	**70.92**	**28.25**	**44.64**

specific models, MiMo-Embodied (Hao et al., 2025) (50) is notably more consistent than RoboBrain2.0 (Team et al., 2025) (314) and RynnBrain (Dang et al., 2026) (349), but still lags behind closed-source models. This disparity highlights that maintaining temporal reasoning and avoiding redundant task execution over extended horizons requires more than scale or embodied training—it demands mechanisms for tracking completed activities and reasoning about daily progress.

Overall, no single model achieves satisfactory performance across all five metrics simultaneously. The universally low task success rates and widespread repetition penalties confirm that **OneDay-Bench** poses a genuine challenge, and that all-day personalized embodied assistance remains unresolved by state-of-the-art models.

4.3 Ablation Study & Analyses

Effect of User Profiles. Tab. 2 studies the impact of different user-profile components. Removing the profile (i.e., the *w/o Profile* entries) causes large drops across models: personalization falls below 37%, time management decreases by 15–17 points, and final scores drop by more than 25 points, showing that the daily plan alone is insufficient to infer personalized objects and temporal routines. Adding ownership information yields moderate gains, while preference descriptions provide larger improvements: task success rates are typically 8–13 points higher than ownership-only, indicating that semantic cues (e.g., appearance and emotional association) are more informative than ownership labels alone. MiMo-Embodied (Hao et al., 2025) shows smaller absolute gains at each profile level, suggesting that embodied-specific training does not automatically improve the ability to exploit structured profiles. Although the full profile performs best, success rates remain below 45%, underscoring the difficulty of converting preferences into correct physical execution.

Effect of Agent Memory. We ablate the agent memory module on five representative models. As shown in Fig. 4, disabling memory substantially increases repetition penalties and consistently reduces final scores. Without memory, repetitions increase by 171–298 depending on the model (e.g., Qwen3-8B (Yang et al., 2025): 241→412, +171; GPT-5 (OpenAI, 2025b): 3→301, +298), indicating a strong tendency to re-select completed activities without persistent history. Correspondingly, final scores drop by 12.8–23.9 points (e.g., Claude-Sonnet-4.5 (Anthropic, 2025): 53.14→29.21, −23.9; DeepSeek-R1 (DeepSeek-AI et al., 2026): 48.34→25.67, −22.6; GPT-5 (OpenAI, 2025b): 42.48→21.54, −21.0). These results validate agent memory as a critical deduplication mechanism:

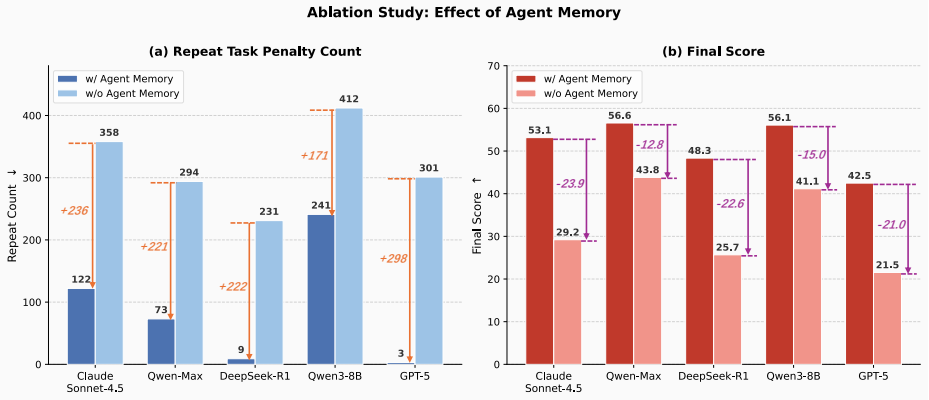


Figure 4: Ablation study on the effect of agent memory. Removing agent memory significantly increases repetition penalty count (a) and consistently degrades final scores across all evaluated models (b), confirming that agent memory is essential for maintaining non-redundant and coherent full-day scheduling.

explicitly recording completed activities prevents repetition penalties and supports coherent scheduling across a day. The large degradation on GPT-5 and Claude-Sonnet-4.5 further suggests that even models with strong baseline scheduling awareness rely heavily on memory to maintain global consistency over long horizons. Notably, the consistency of this pattern across both closed-source and open-source models indicates that memory-based deduplication is a universal requirement for reliable full-day assistance, independent of model architecture or scale.

Failure Case Analysis. We analyze 5,250 failure instances and group them into four categories (Fig. 6, Appendix B). *Incomplete actions* dominate (46.6%): the agent produces partial sequences that omit required operations (e.g., in failure case 4, DeepSeek-R1 (DeepSeek-AI et al., 2026) misses PICK_AND_PLACE:CLOCK_0). *Time management* (19.1%) and *personalization errors* (19.0%) occur at comparable rates: in failure case 1, Qwen2.5-7B (Qwen et al., 2025) incorrectly outputs no task despite an approaching deadline; in failure case 3, Claude (Anthropic, 2025) selects a white desk lamp instead of the user-specified gray laptop. *Repetitive task penalties* (10.6%) arise when the agent fails to condition on execution history; for example, in failure case 2, GPT-5 (OpenAI, 2025b) reselects a task already recorded in agent memory. Overall, the primary bottleneck is not high-level language comprehension, but faithfully translating plans into complete, time-consistent, and preference-aligned executable actions. The dominance of incomplete actions suggests that future work should prioritize improving the completeness and executability of action sequences, potentially through better grounding in physical simulation.

5 Conclusion

In this paper, we presented **OneDay-Bench** and **OneDay-Agent** to move beyond single-task instruction following toward full-day personalized household assistance. **OneDay-Bench** offers 100 daily-plan instances with structured user profiles and human-annotated preference memories, and evaluates agents across personalization alignment, task completion, time management, action executability, and task deduplication. Building on this benchmark, **OneDay-Agent** integrates preference memory with schedule-aware goal synthesis to convert underspecified daily plans into temporally consistent, preference-aligned, executable skill sequences in dynamic home environments. Extensive experiments across state-of-the-art MLLMs reveal that while models often capture preferences and time cues at the language level, grounded success in simulation remains below 45% and repeated-activity errors persist. Our ablation studies further demonstrate that structured user profiles and persistent agent memory are essential for accurate preference grounding, coherent scheduling, and non-redundant execution over extended horizons. Meanwhile, the failure analysis identifies incomplete action sequences as the dominant bottleneck, highlighting the need for tighter integration between high-level reasoning and low-level embodied control. These findings confirm that full-day personalized assistance remains an open and challenging problem, and we hope our benchmark and baseline agent will facilitate future research.

AI USE STATEMENT

Large language models and multimodal large language models were used to construct the agents and assist in generating task descriptions and daily plans. The authors take full responsibility for all content, experiments, and conclusions presented in this paper.

REFERENCES

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, et al. Do as I can, not as I say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Alibaba Cloud. Alibaba Cloud Model Studio: DashScope API reference (calling Qwen models). Alibaba Cloud documentation, February 2026. Last updated: Feb 24, 2026; accessed: 2026-02-28.

Anthropic. Introducing Claude Sonnet 4.5. Anthropic news/announcement, September 2025. Accessed: 2026-02-28.

Matthew Chang, Gunjan Chhablani, Alexander Clegg, et al. PARTNR: A benchmark for planning and reasoning in embodied multi-agent tasks. *arXiv preprint arXiv:2411.00081*, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025.

Yinpei Dai, Run Peng, Sikai Li, and Joyce Chai. Think, act, and ask: Open-world interactive personalized robot navigation. *arXiv preprint arXiv:2310.07968*, 2023.

Ronghao Dang, Jiayan Guo, Bohan Hou, Sicong Leng, Kehan Li, Xin Li, Jiangpin Liu, Yunxuan Mao, Zhikai Wang, Yuqian Yuan, et al. Rynnbrain: Open embodied foundation models. *arXiv preprint arXiv:2602.14979*, 2026.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, et al. Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2026. arXiv preprint, version v2 (revised Jan 4, 2026).

Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew 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*, pp. 3164–3174, 2020.

Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022.

Xiaoshuai Hao, Lei Zhou, Zhijian Huang, Zhiwen Hou, Yingbo Tang, Lingfeng Zhang, Guang Li, Zheng Lu, Shuhuai Ren, Xianhui Meng, et al. Mimo-embodied: X-embodied foundation model technical report. *arXiv preprint arXiv:2511.16518*, 2025.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022.

Ivan Kapelyukh and Edward Johns. My house, my rules: Learning tidying preferences with graph neural networks. *arXiv preprint arXiv:2111.03112*, 2021.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017.

Taeyoon Kwon, Dongwook Choi, Hyojun Kim, Sunghwan Kim, Seungjun Moon, Beong-woo Kwak, Kuan-Hao Huang, and Jinyoung Yeo. Embodied agents meet personalization: Investigating challenges and solutions through the lens of memory utilization. *arXiv preprint arXiv:2505.16348*, 2025a.

Taeyoon Kwon, Dongwook Choi, Sunghwan Kim, Hyojun Kim, Seungjun Moon, Beong-woo Kwak, Kuan-Hao Huang, and Jinyoung Yeo. Embodied agents meet personalization: Exploring memory utilization for personalized assistance. *arXiv preprint arXiv:2505.16348*, 2025b.

Chengshu Li, Fei Xia, Roberto Martin-Martin, Michael Lingelbach, Sanjana Srivastava, Bokui Shen, Kent Vainio, Cem Gokmen, Gokul Dharan, Tanish Jain, Andrey Kurenkov, C. Karen Liu, Hyowon Gweon, Jiajun Wu, Fei-Fei Li, and Silvio Savarese. iGibson 2.0: Object-centric simulation for robot learning of everyday household tasks. *arXiv preprint arXiv:2108.03272*, 2021.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martin-Martin, Chen Wang, Gabrael Levine, et al. BEHAVIOR-1K: A human-centered, embodied AI benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*, 2024a.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. LLaVA-NeXT-Interleave: Tackling multi-image, video, and 3D in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024b.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. *arXiv preprint arXiv:2209.07753*, 2022.

OpenAI. Introducing GPT-5.2. OpenAI release post, December 2025a. Accessed: 2026-02-28.

OpenAI. Introducing GPT-5. OpenAI release post, August 2025b. Accessed: 2026-02-28.

OpenAI. GPT-5 mini model (OpenAI API documentation). OpenAI developer documentation, August 2025c. Model snapshots include gpt-5-mini-2025-08-07; accessed: 2026-02-28.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.

Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*, 2023.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report, 2025. arXiv preprint, version v2 (revised Jan 3, 2025).

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*, pp. 9339–9347, 2019.

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M. Sadler, Wei-Lun Chao, and Yu Su. LLM-Planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2998–3009, 2023.

Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martin-Martin, Fei Xia, Kent Vainio, Zheng Lian, Cem Gokmen, Shyamal Buch, C. Karen Liu, Silvio Savarese, Hyowon Gweon, Jiajun Wu, and Fei-Fei Li. BEHAVIOR: Benchmark for everyday household activities in virtual, interactive, and ecological environments. *arXiv preprint arXiv:2108.03332*, 2021.

BAAI RoboBrain Team, Mingyu Cao, Huajie Tan, Yuheng Ji, Xiansheng Chen, Minglan Lin, Zhiyu Li, Zhou Cao, Pengwei Wang, Enshen Zhou, et al. Robobrain 2.0 technical report. *arXiv preprint arXiv:2507.02029*, 2025.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Zixuan Wang, Bo Yu, Junzhe Zhao, Wenhao Sun, Sai Hou, Shuai Liang, Xing Hu, Yinhe Han, and Yiming Gan. KARMA: Augmenting embodied AI agents with long-and-short term memory systems. *arXiv preprint arXiv:2409.14908*, 2024.

Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. TidyBot: Personalized robot assistance with large language models. *arXiv preprint arXiv:2305.05658*, 2023.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

Quanting Xie, So Yeon Min, Pengliang Ji, Yue Yang, Tianyi Zhang, Kedi Xu, Aarav Bajaj, Ruslan Salakhutdinov, Matthew Johnson-Roberson, and Yonatan Bisk. Embodied-RAG: General non-parametric embodied memory for retrieval and generation. *arXiv preprint arXiv:2409.18313*, 2024.

Manjie Xu, Xinyi Yang, Wei Liang, Chi Zhang, and Yixin Zhu. Learning to plan with personalized preferences. *arXiv preprint arXiv:2502.00858*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report, 2025.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. *arXiv preprint arXiv:2305.10250*, 2023.

APPENDIX

This technical appendix provides additional implementation details and experimental results of OneDay-Bench and OneDay-Agent that are omitted from the main body of this paper due to the page limit. Specifically, this appendix is organized as follows:

- Appendix A contains the complete evaluation metric definitions and protocols.
- Appendix B contains the radar comparison, memory ablation, and failure analysis figures.
- Sec. D provides additional details on the **Implementation Details**, including prompt construction, model configurations, and Habitat 3.0 grounded evaluation setup.
- Sec. E presents additional details on the **Data Generation Pipeline**, including activity template matching, object and memory extraction, underspecified plan generation, and user profile construction.
- Sec. F showcases extended **Visualization Results**, with full-day execution trajectories rendered in the Habitat 3.0 simulator.

A DETAILED EVALUATION METRICS AND PROTOCOLS

Evaluation Metrics & Protocols. We evaluate the agent's performance across *five complementary dimensions*, summing them into a final composite score.

- [1]*Personalization Score* measures the degree to which the agent recognizes and uses the correct user-specific objects. For each activity, this is calculated as the proportion of the required personalized objects selected correctly:

$$P_{\text{act}} = \min\left(1,\ \frac{|\text{correct objects used}|}{|\text{must use objects}|}\right) \times 100\%. \tag{5}$$

- [2]*Task Completion Score* measures the percentage of necessary actions successfully performed in an activity, that is:

$$C_{\text{act}} = \frac{|\text{completed actions}|}{|\text{required actions}|} \times 100\%. \tag{6}$$

  For activities with strict operation sequence requirements, this score is averaged with the sequence fidelity item C_{seq} : $C_{\text{act}} = \frac{C_{\text{act}}+C_{\text{seq}}}{2}$. For deadline-critical activities that fail to pass the time check, both P_{act} and C_{act} are penalized by 0.3 times.

- [3]*Time Management Score* is used to assess progress adherence. For activities with tight deadlines, the score is binary: $T_{\text{act}} = 1$ if completed on time; otherwise, 0. For activities with flexible time windows, partial credit is awarded based on whether the last action occurred within the specified time window:

$$T_{\text{act}} = \begin{cases} 1.0 & \text{last action} \le t_{\text{end}} \\ 0.3 & \text{last action} > t_{\text{end}} \\ 0.5 & \text{no actions recorded} \end{cases} \times 100\%. \tag{7}$$

- [4]*Task Success Rate* is a binary judgment for each activity, requiring correct use of personalized objects and timely completion. For time-limited tasks, it also requires that all required objects appear and the timestamp of the last operation does not exceed the deadline:

$$S_{\text{act}} = \mathbb{1}\left[\begin{smallmatrix}\text{all required actions completed} \\ \wedge\ \text{deadline satisfied}\end{smallmatrix}\right] \times 100\%. \tag{8}$$

- [5]*Repetitive Task Penalty Count* tracks the number of times an agent repeatedly selects a completed activity within the same day. A penalty event is recorded each time the agent's resolved activity decision matches a previously executed activity in the current timeline:

$$N_{\text{repeat}} = \sum_{\text{day}} \sum_{\text{timeline}} \mathbb{1}[\text{activity already completed}]. \tag{9}$$

  The scores for all four consecutive activities within a day are averaged to obtain daily scores $\bar{P}$, $\bar{C}$, $\bar{T}$, and $\bar{S}$.

These four scores are then averaged over 100 days. The *Final Score* combines these metrics with a repetitive task penalty:

$$\mathcal{F} = 100 \times \left(0.2\,\bar{P} + 0.1\,\bar{C} + 0.2\,\bar{T} + 0.5\,\bar{S}\right) - \lambda \times N_{\text{repeat}}, \tag{10}$$

where the weights λ reflect the relative importance of personalization and task success rate in actual family support, and N_{repeat} is the total number of repetitive activity selections during the entire 100-day assessment period.

B Additional Analysis Figures

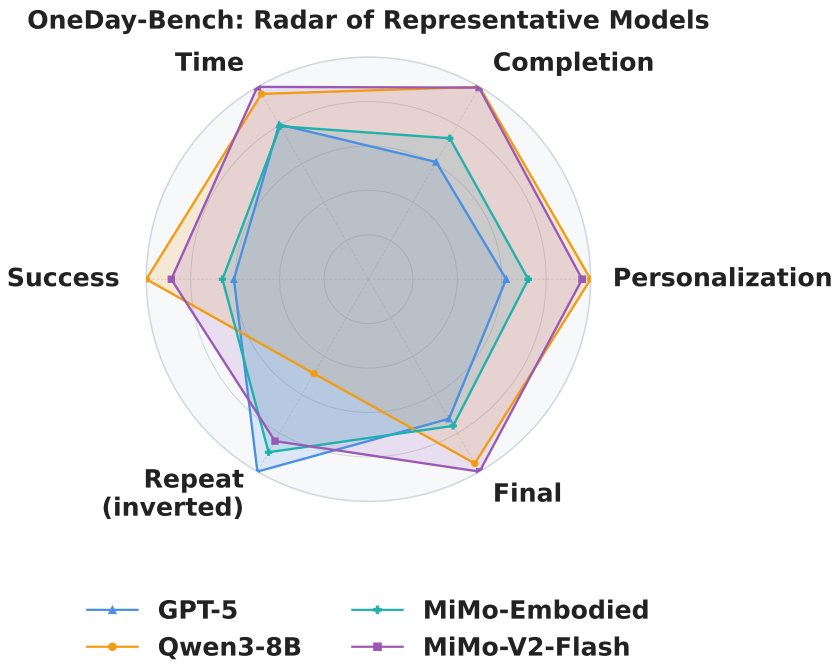


Figure 5: Radar comparison of representative models on OneDay-Bench.

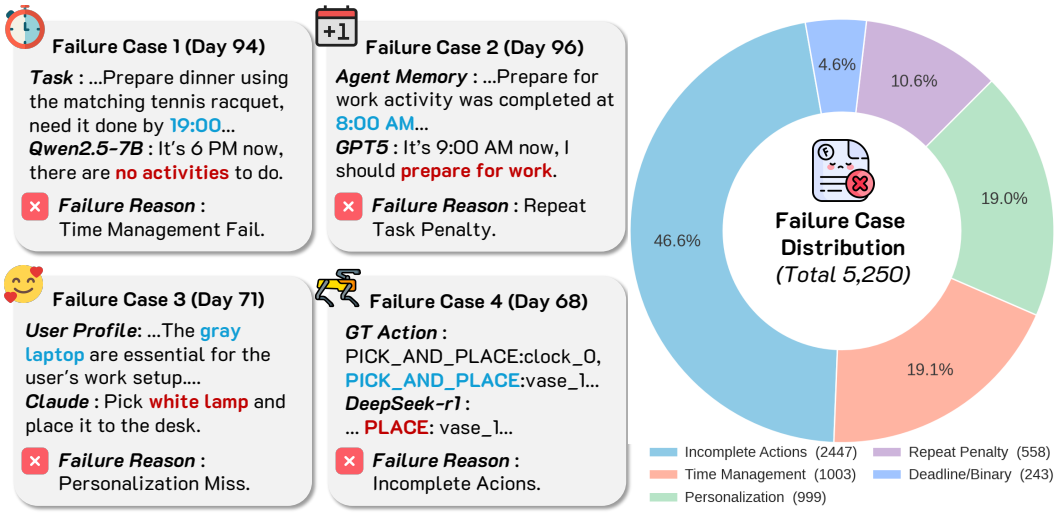


Figure 6: Failure case analysis on OneDay-Bench. Four representative failure types are illustrated: time management failure, repetition penalty, personalization miss, and incomplete action sequence.

C Limitations and Future Work

This work focuses on characterizing the fundamental capabilities required for embodied home assistants. To maintain high experimental fidelity and safety, the current benchmarks leverage high-quality simulations, which serve as a critical proxy for complex physical interactions. Future iterations will focus on cross-domain transfer to diverse real-world settings. We also envision that future work will incorporate more sophisticated lifelong learning mechanisms and nuanced human-in-the-loop feedback to further refine the personalization and reliability of the assistant in diverse user scenarios.

D Implementation Details

All models are evaluated under a unified prompting interface without model-specific fine-tuning. At each checkpoint, the model receives the underspecified daily plan, the structured user profile, the current time and timeline context, and the memory of completed activities. It then predicts an activity decision and a short action sequence. We use a maximum output length of 800 tokens per call, and do not pass an explicit temperature parameter to avoid compatibility issues across providers.

Prompt Construction. The system prompt establishes the agent's role and output contract, while the user prompt is dynamically assembled at every probe time t from six fields: (1) the current timestamp; (2) the ground-truth activity name and its time slot for that checkpoint; (3) the must-use object list from the success criteria; (4) the enumerated list of all valid activity names for the day; (5) the seen-activity history with first-occurrence timestamps, which serves as the primary deduplication signal; and (6) the daily plan text together with the top-8 user-memory hints drawn from the structured user profile. The full system prompt used in all experiments is shown below.

System Prompt (used for all models)

```
You are a household task planning assistant.  You only need to output
text; no real action execution is required.
You will receive:
- Current time
- The activity currently to be executed (the activity name from ground
truth)
- History of completed activities (very important)
- User profile and today's underspecified plan
Your goals:
1) Give the most reasonable activity decision for the current moment.
2) Output a concise text action plan (for offline scoring).
3) If an activity already appears in the completed activity history,
do NOT execute it again.
You MUST strictly follow the output format below:
Thought:  <brief reasoning>
Activity:  <activity name, must come from the input activity list;
output NONE if no action>
Tool:  Navigate[<target>]
Tool:  PickAndPlace[<object>, <destination>]
Done[]
Rules:
- Output at most 8 Tool lines.
- If Activity=NONE, you may output Done[] directly.
- Allowed action names:  Navigate / PickAndPlace / Pick / Place / Open
/ Close.
```

The user prompt template filled at each probe time t is:

User Prompt Template (instantiated per probe)

```
Current time: {probe_time}
Current activity in timeline: {activity_name}
Current activity slot: {time_slot}
Current activity must-use objects: {must_use_objects}
All valid activities:
{list of activity names, one per line}
Activities already done at least once (DO NOT repeat them):
{seen activity name + first-seen timestamp, one per line; "(none)" if
empty}
Daily plan:
{task_input.daily_plan}
User memory hints:
{top-8 memory entries as [type] object: content}
Question:
It is now {probe_time}. What do you plan to do?
Please strictly follow the required output format.
```

Structured Output Space Model outputs follow a fixed four-field format: `Thought`, `Activity`, `Tool`, and `Done[]`. The `Activity` field must be one of the predefined activity names or `NONE`. The action vocabulary is restricted to `Navigate`, `PickAndPlace`, `Pick`, `Place`, `Open`, and `Close`. For stable parsing, each response is limited to at most 8 tool calls.

Models and Hyperparameters. We evaluate **17 models** spanning three categories: closed-source, open-source, and embodied-specific. Closed-source models include GPT-5 (OpenAI, 2025b), GPT-5.2 (OpenAI, 2025a), GPT-5-mini (OpenAI, 2025c), Claude-Sonnet-4.5 (Anthropic, 2025), Qwen-Max (Alibaba Cloud, 2026), and Gemini-2.5-Pro (Comanici et al., 2025). Open-source models cover a wide parameter range (7B–671B): DeepSeek-R1 (DeepSeek-AI et al., 2026) (671B), Qwen2.5 (Qwen et al., 2025) (7B and 72B), Qwen3 (Yang et al., 2025) (8B and 30B), MiMo-V2-Flash (Xiao et al., 2026) (309B), InternVL3.5 (Wang et al., 2025) (8B), and LLaVA-NeXT (Li et al., 2024b) (8B). Embodied-specific models include RoboBrain2.0 (Team et al., 2025) (7B), MiMo-Embodied (Hao et al., 2025) (7B), and RynnBrain (Dang et al., 2026) (8B). All models share an identical inference configuration: maximum output length is set to 800 tokens per checkpoint call, and no explicit temperature is passed to the API to avoid provider-specific compatibility issues. The repetition penalty coefficient is fixed at $\lambda = 0.03$ for all evaluations. Per-model access details and API endpoints are summarized in Table 3.

Habitat 3.0 Evaluation. Task success rate is validated through grounded interaction in the Habitat 3.0 simulator (Puig et al., 2023). Each scene is a realistic multi-room household environment encompassing diverse room types such as kitchens, living rooms, bedrooms, offices, dining rooms, and hallways. At evaluation time, the agent's predicted text action sequence is first parsed by a rule-based extractor and then converted into natural-language instructions executable by the Habitat `EvaluationRunner`. The conversion follows a fixed action vocabulary of six primitives: `Navigate`, `Pick`, `Place`, `Open`, `Close`, and `Done[]`. Each primitive is mapped deterministically to a simulator directive—for example, `Pick[obj]` becomes *"Pick up the `obj`."* and `Place[obj, furn]` becomes *"Place the `obj` on the `furn`."* Object name resolution follows a two-stage lookup: an exact match against the scene's `sim_handle_to_name` table is attempted first, falling back to room- and furniture-alias redirection if the exact handle is not found. Task success is then determined by executing the converted instruction sequence in simulation and checking whether the resulting object states satisfy the ground-truth `evaluation_propositions` defined for each activity. A ground-truth (GT) execution mode—which replays `required_actions` directly from `ground_truth.json`—is additionally provided as an oracle upper bound for the grounded evaluation pipeline.

E Data Generation Pipeline.

The 100 daily instances in OneDay-Bench are generated automatically from real MEMENTO episodes (Kwon et al., 2025a) using a deterministic pipeline. Episodes are first grouped by `scene_id`, and for each day one scene is selected in round-robin order across the 12 HSSD scenes. Within the selected scene, 2–4 episodes are sampled without replacement to form the day's activity

Table 3: Model configurations used in all OneDay-Bench experiments. All models use an identical prompt interface with `max_tokens=800` and no explicit temperature. "–" in the Size column denotes closed-source models whose parameter counts are not publicly disclosed. API mode: **O** = OpenAI-compatible endpoint, **V** = local vLLM service.

Model	Size	Max Tokens	Temperature	API Mode
Closed-Source Models				
GPT-5 (OpenAI, 2025b)	–	800	default	O
GPT-5.2 (OpenAI, 2025a)	–	800	default	O
GPT-5-mini (OpenAI, 2025c)	–	800	default	O
Claude-Sonnet-4.5 (Anthropic, 2025)	–	800	default	O
Qwen-Max (Alibaba Cloud, 2026)	–	800	default	O
Gemini-2.5-Pro (Comanici et al., 2025)	–	800	default	O
Open-Source Models				
DeepSeek-R1 (DeepSeek-AI et al., 2026)	671B	800	default	O
Qwen2.5 (Qwen et al., 2025)	7B	800	default	V
Qwen2.5 (Qwen et al., 2025)	72B	800	default	V
Qwen3 (Yang et al., 2025)	8B	800	default	V
Qwen3 (Yang et al., 2025)	30B	800	default	V
MiMo-V2-Flash (Xiao et al., 2026)	309B	800	default	O
InternVL3.5 (Wang et al., 2025)	8B	800	default	V
LLaVA-NeXT (Li et al., 2024b)	8B	800	default	V
Embodied Specific Models				
RoboBrain2.0 (Team et al., 2025)	7B	800	default	V
MiMo-Embodied (Hao et al., 2025)	7B	800	default	V
RynnBrain (Dang et al., 2026)	8B	800	default	V

Shared: repetition penalty $\lambda = 0.03$, max tool calls $= 8$, memory hints ≤ 8.

pool, with the constraint that at least one activity must be *deadline-critical* (i.e., must be completed before a hard cutoff time) and the remainder may be *flexible-window* tasks.

Activity Template Matching. Each sampled episode is matched to one of **8 predefined activity templates** covering common household scenarios: *Prepare Breakfast*, *Organize Items for Leaving*, *Evening Decoration Arrangement*, *Organize Desk*, *Kitchen Cleanup*, *Living Room Tidy*, *Prepare Dinner*, and *Bedroom Tidy*. Template matching is performed by computing the overlap between the episode's target object categories and each template's `preferred_categories` list; the template with the highest overlap score is selected, with ties broken randomly. Each template specifies a set of candidate time windows (e.g., `06:30–07:30` for breakfast) and a probe count range of 2–5 checkpoints; probe times are distributed uniformly across the assigned window. Non-overlapping time windows are enforced globally across all activities within a day; if all predefined options conflict, a random 1–2 hour window between 06:00 and 22:00 is sampled.

Object and Memory Extraction. For each episode, target objects are extracted from the episode's `metadata.target_objects` field, with each object assigned a canonical `wg_name` (`category_idx`). Object handles and receptacle mappings are recovered from the scene's `name_to_receptacle` table. Each object carries a memory annotation of one of five types—`history`, `preference`, `ownership`, `group`, or `routine`—drawn from human-annotated `object_semantics` and `user_pattern` files. Objects with `history` or `preference` memory types are designated as *must-use* objects (up to 2 per activity) and form the basis for the personalization score.

Underspecified Plan Generation. The daily plan is assembled by concatenating one natural-language fragment per activity, chosen randomly from the template's `daily_plan_fragments`

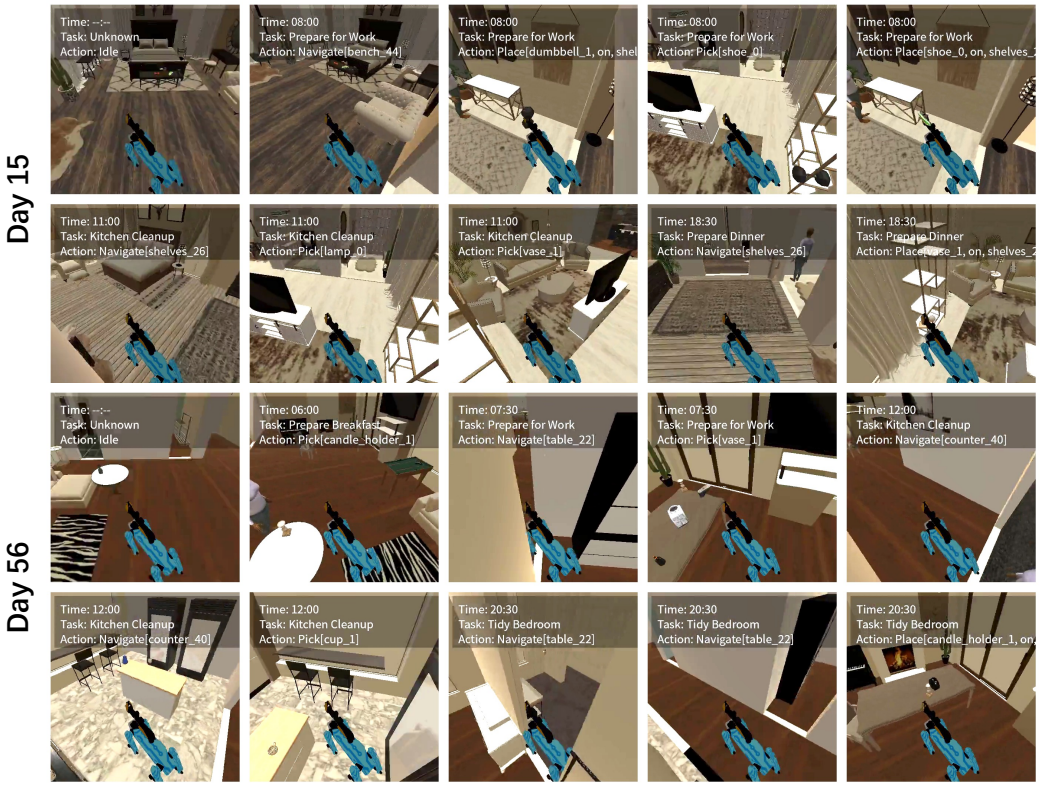


Figure 7: **Qualitative visualization of two full-day execution trajectories in the Habitat 3.0 simulator.** On Day 15 and Day 56, the agent successfully associates fuzzy references in the underspecified daily plan with specific object instances and executes multi-step skill call sequences across multiple time checkpoints, demonstrating coherent schedule-aware goal synthesis and persistent agent memory from morning to night.

pool. Object references are deliberately obfuscated: a `preference` object becomes "my favorite {category}", a `history` object with travel context becomes "the {category} from my travels", and a gift object becomes "the {category} (gift from friend)". Explicit asset identifiers are always omitted, requiring the agent to resolve the correct object instance from the structured user profile at inference time.

User Profile Construction. Each instance is paired with a structured user profile containing four fields: `ownership`, `preference`, `history`, and `group` object lists, together with `user_patterns` encoding per-template action routines. A flat `memories` list is additionally derived from all objects carrying non-empty memory annotations and is provided as the top-8 memory hints in the agent prompt. Work schedules are sampled from 5 predefined variants (work start times ranging from 08:00 to 10:00) and dietary preferences from 4 variants, introducing realistic inter-day variability in the profile context.

Qualitative Assessments. Fig. 7 shows two representative full-day trajectories rendered in Habitat 3.0. On Day 15, the agent completes multiple time-sequential activities: getting ready for work (08:00), cleaning the kitchen (11:00), and preparing dinner (18:30)—via coherent navigation and object-manipulation sequences across different home settings. It remains idle when no task is scheduled and transitions smoothly at the correct checkpoints, illustrating schedule-aware goal synthesis supported by persistent memory. Day 56 further stresses the agent with a denser schedule, including preparing breakfast (06:00), getting ready for work (07:30), cleaning the kitchen (12:00), and tidying the bedroom (20:30). Across both days, the agent resolves fuzzy references to specific object instances (e.g., `candle_holder_1`, `vase_1`, `cup_1`) and executes multi-step actions within

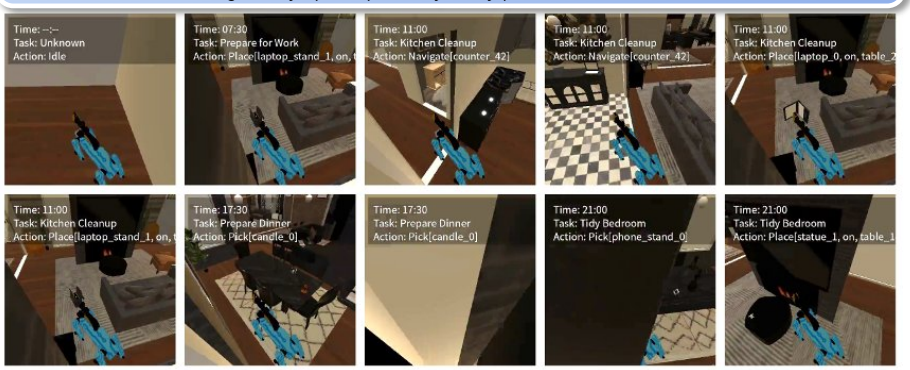
realistic layouts, demonstrating that **OneDay-Agent** can exhibit personalized and time-consistent behavior from morning to night.

F Extended Qualitative Visualization.

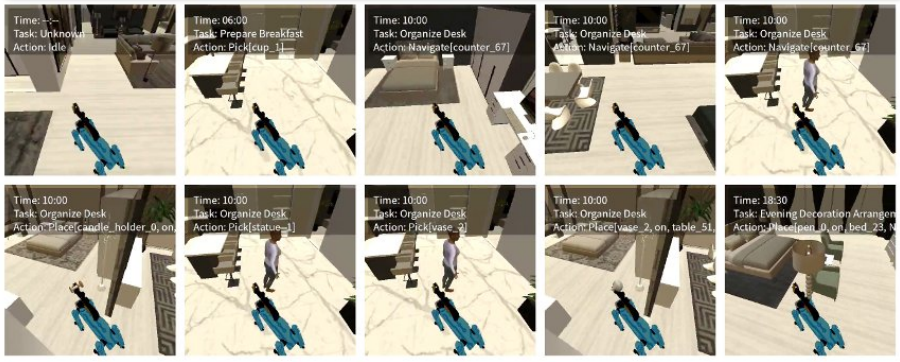
Beyond the two full-day trajectories in Fig. 7, we provide extended qualitative visualizations across **12 additional daily instances** rendered in the Habitat 3.0 simulator and shown in Figs. 8–11. Each panel displays the underspecified daily plan at the top, followed by representative simulator frames annotated with the current time checkpoint, active task name, and low-level action label.

The visualizations collectively highlight three recurring capabilities of OneDay-Agent. *(1) Fine-grained appearance-based grounding.* The agent successfully resolves highly specific object descriptions to concrete simulator instances across a wide variety of categories—including "the black laptop with a sleek design", "the white candle shaped like a gift box with a black bow", "the gray clock with a rounded top and black numerals", "the black kettle with a silver handle", and "the mint green candle with white butterfly designs"—demonstrating that the structured user profile provides sufficient discriminative signal even for visually similar object instances. *(2) Relational and social memory resolution.* Several daily plans include socially grounded references such as "the statue my best friend gave me", "the gift clock from my friend", "the gift vase from my friend", and "my favorite stuffed toy". The agent correctly maps these relational references to the corresponding asset identifiers via the `history` and `ownership` memory fields in the user profile, and executes the appropriate `Pick`-and-`Place` sequences at the correct furniture destinations. *(3) Dense multi-activity scheduling.* Several instances contain 3–4 activities spanning the full day from 06:00 to 21:00, including deadline-critical tasks (e.g., work preparation before 08:30 or 09:30) interleaved with flexible-window tasks (*e.g.*, evening decoration arrangement, living room tidying, bedroom organization). Across all examples, the agent correctly sequences activities in chronological order, remains idle between checkpoints, and avoids re-executing completed activities, confirming that the agent memory module provides effective deduplication across dense schedules.

Daily Plan: Prepare breakfast using the cream-colored jug with a curved handle. and the black cup with a smooth surface. Collect the matching plate and the matching bowl and place them ready for work before 08:30. Put the white vase with a rounded body and narrow neck. neatly on the desk. Tonight, tidy up and place my lucky pen on the bedroom bed.



Daily Plan: Get breakfast ready with the gray clock with a rounded top and black numerals.. Before bed, put the matching phone stand on the nightstand.



Daily Plan: Set up breakfast table with the lamp with a pineapple-shaped neon outline and a black base. and the gold picture frame with a dark central image. Organize my favorite jug in the bedroom.

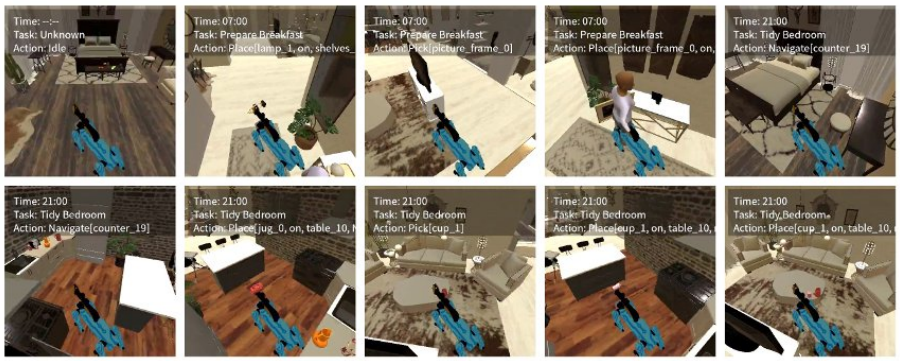


Figure 8: **More Visualization Results of OneDay Agent.**

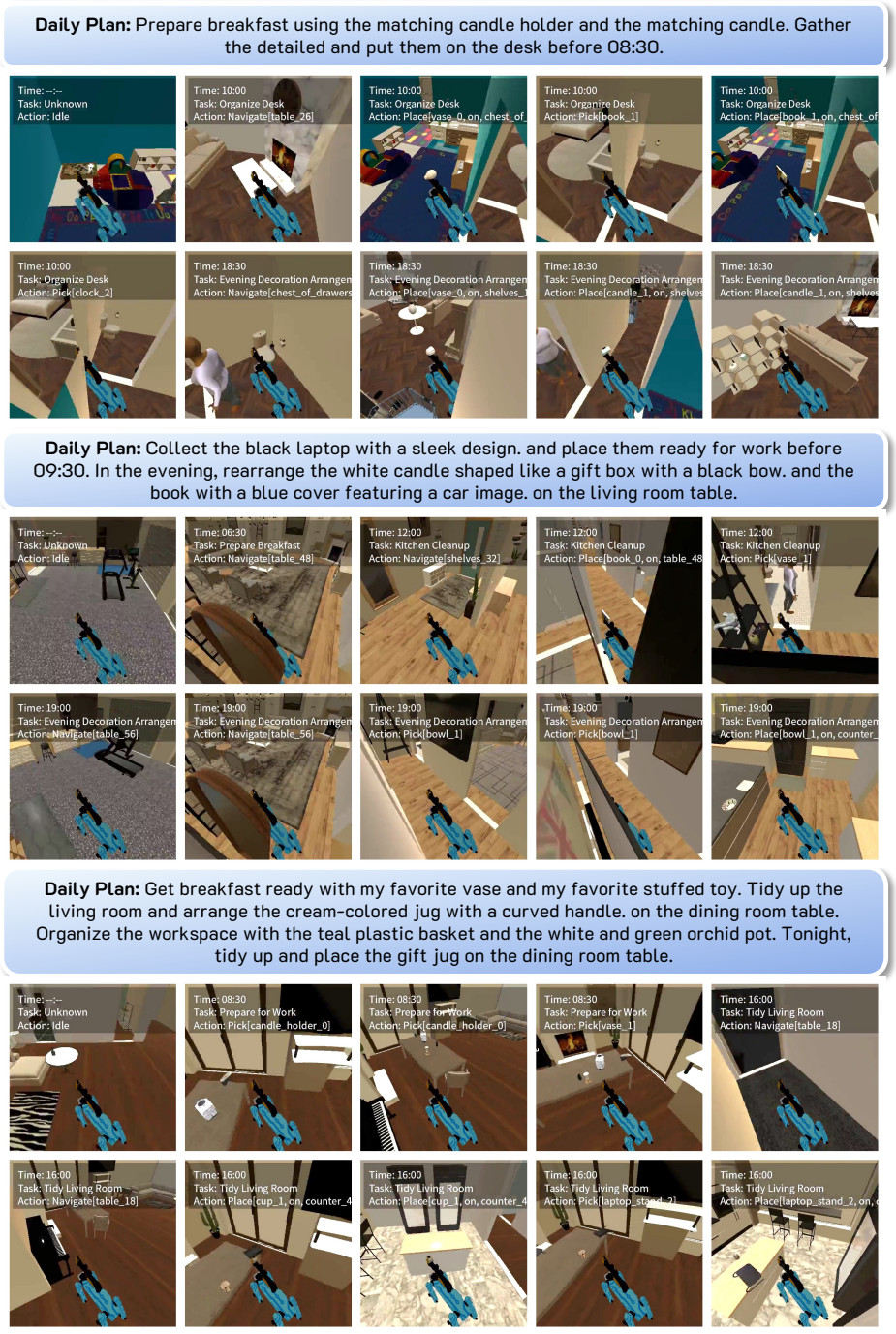


Figure 9: **More Visualization Results of OneDay Agent.**



Figure 10: **More Visualization Results of OneDay Agent.**

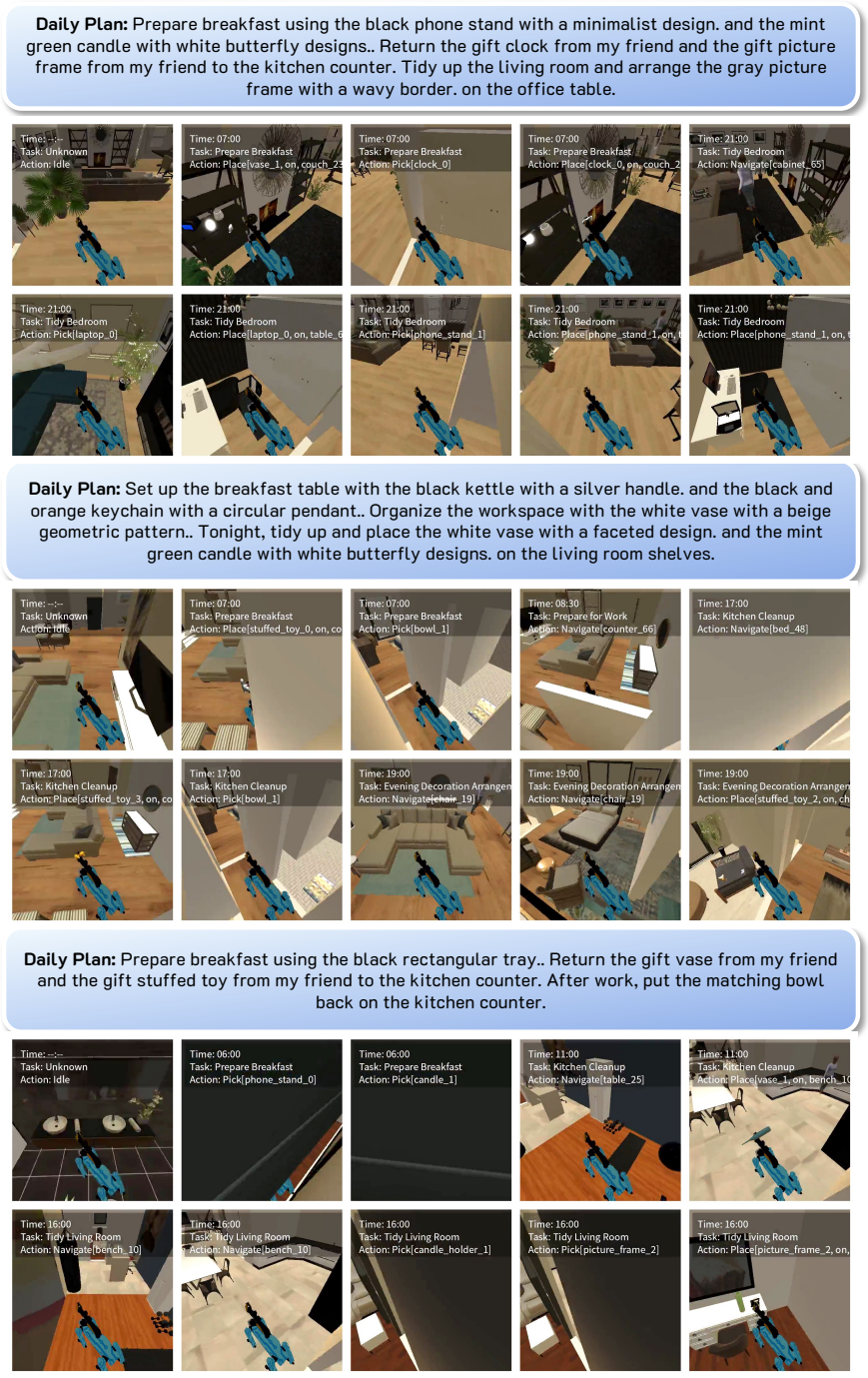


Figure 11: **More Visualization Results of OneDay Agent.**