Human-centered In-building Embodied Delivery Benchmark

Anonymous ACL submission

Abstract

Recently, the concept of embodied intelligence has been widely accepted and popularized, leading people to naturally consider the potential for commercialization in this field. In this work, we propose a specific real-world scenario simulation — human-centered in-building embodied delivery. Furthermore, for this scenario, we have developed a brand-new virtual environment system from scratch, constructing a multi-level connected building space modeled after a polar research station. This environment also includes autonomous human characters and robots with grasping and mobility capabilities, as well as a large number of interactive items. Based on this environment, we have built a delivery dataset containing 13k language instructions to guide robots in providing services. We simulate human behavior through human characters and sample their various needs in daily life. Finally, we proposed a method centered around a large multimodal model to serve as the baseline system for this dataset. Compared to past embodied data work, our work focuses on an immersive virtual environment centered around human-robot interaction for industrial-grade scenarios. We believe this will bring new perspectives and exploration angles to the embodied community. Our code, dataset, and benchmark are publicly available.

1 Introduction

With the rapid development of embodied robotic technology, people are gradually becoming aware of its tremendous potential in various fields. Concurrently, there has been a surge of discussions and explorations within the community regarding embodied skill scenarios, such as navigation (Wang et al., 2024; Hao et al., 2020), manipulation (Li et al., 2023c; Jin et al., 2024), and instruction following (Brohan et al., 2022b; Yenamandra et al., 2023), leading to the proposal of a series of models. Although the skill scenarios are diverse, people are concerned that the current skill scenarios are designed to be overly simplistic for realism-aligned application scenarios (Fu et al., 2024). And there exists a noticeable gap between skill scenarios and real-world application scenarios. Specifically, it is widely believed that existing skill scenarios may be inadequate in fully reflecting the potential issues encountered in actual environments and do not accurately capture users' more precise interaction needs with embodied robots (Bousmalis et al., 2023). Therefore, we argue that this inconsistency with real-world commercial scenarios has hindered the emergence of novel topics within the embodied AI community in recent years. Therefore, we suggest that exploring scenarios closer to real-world applications can help further the development of the embodied AI community (Zador et al., 2023; Yang et al., 2023a).

In this work, we focus on simulating and data construction for a highly anticipated express delivery service scenario called human-centered in-building delivery. In today's society, precise and efficient delivery services are crucial to the success of many top companies. However, unlike large-scale transshipment centers and external express delivery services that rely on public transportation, the last step of the delivery stage faces significant challenges. For instance, private spaces like companies in buildings or high-security residential areas often prohibit external delivery services due to various security and management considerations. Moreover, people typically move around inside buildings to meet their needs and purposes. This delivery process can impose tangible pressure on customers. Therefore, precise item delivery to specified persons in private spaces represents a significant opportunity for robotic services. In order to explore this scenario, our contributions can be divided into the following parts:

Scenario & Task Definition. The real-world indoor service scenario is characterized by the need

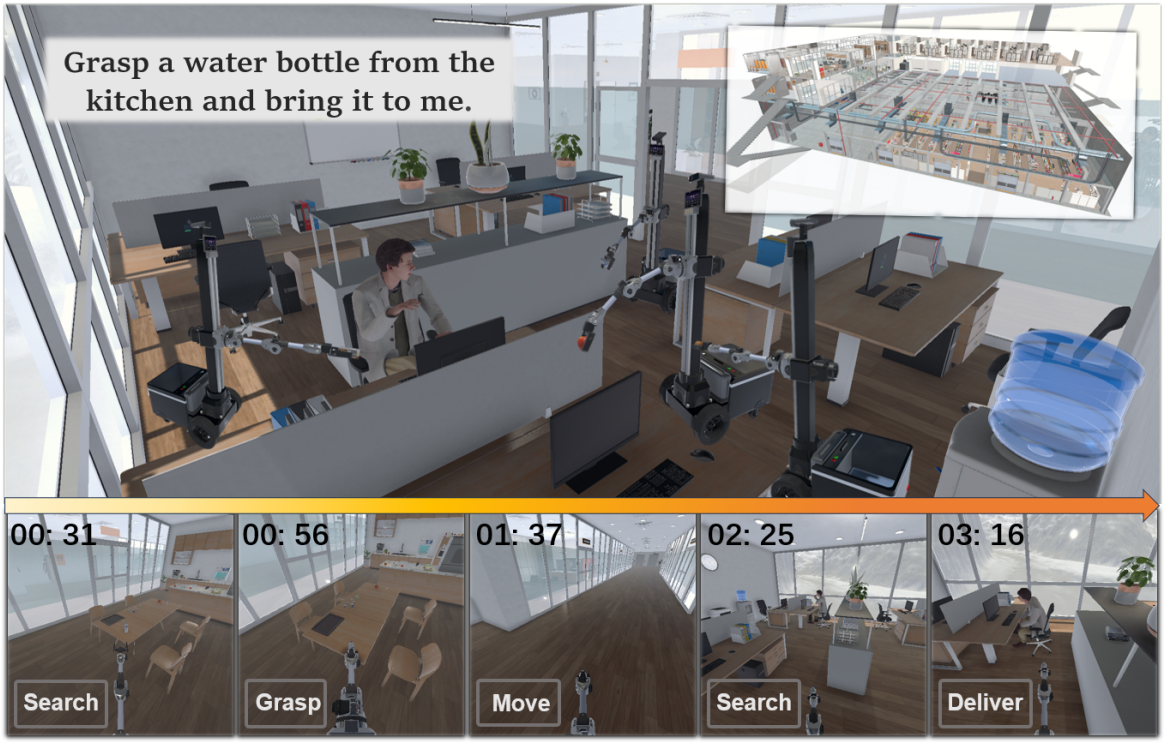

Figure 1: Human-centered in-building embodied delivery describes a task that originates from a real commercial delivery scenario. It mainly refers to the precise delivery service for users in private spaces where external delivery services cannot be used, achieved through embodied robots. This task typically requires the robot to locate the target item based on the user's needs (e.g., *grasp a water bottle from the kitchen and bring it to me.*) across multiple rooms within the three-story building (a polar research station building, *see the thumbnail in the top right corner*) and ultimately deliver it to the designated location/person. The robot needs to consider the user's context (behavior or schedule), as the user will be moving around the building according to their own goals during the delivery.

to account for numerous complex and intervening factors. We analyze the scenario and pinpoint the critical elements of the delivery service, as shown in Figure 1. Followed by formulating task objectives and definitions, which includes establishing the task's premises, context, framework, and scope (Section 3).

Simulation Environment. Grounded in the task setting and business requirements, we have constructed from scratch a novel simulation environment modelled after a real-world polar research station (referred to as the **Polar Research Station Environment**, **PRS**). This environment comprises a three-story building interconnected by stairs and a functional elevator. It integrates common human societal scenarios into a community-like pattern, such as bedrooms, gyms, offices, laboratories, medical rooms, wards, living rooms, leisure spaces, etc. This design aims to cover as a wide range of everyday scenarios within the building as possible. Additionally, to simulate daily activities for delivery services, the environment includes over a dozen virtual human characters engaging in activities according to their individual intentions. Furthermore, we provide a range of interactive objects to support the tasks. Lastly, we have designed a robotic simulation with grasping and moving capabilities to serve human character agents (Section 4).

Dataset. In constructing delivery service data, we initially utilize the large language model (LLM) to generate reasonable daily activities and varied demands for virtual characters (Non-Playable Character, NPC) based on their profiles. Consequently, the robot is required to locate and deliver the appropriate objects to meet the human characters' demands and accomplish task objectives. We continually generate diverse data by modifying character needs, daily routines, and target objects. Furthermore, we incorporate a manual review and refinement stage to ensure the balance of the task data (Section 5).

Baseline. We propose an LMM-based approach as the baseline method, employing a modular architecture encompassing language instruction analysis, multimodal target search, and robotic action execution (Section 6).

Therefore, we will gradually introduce these contents. Due to our substantial workload, additional content will be included in the Appendix.

2 Background

In recent years, the concept of embodied AI has been widely recognized and popularized (Das et al., 2018; Gan et al., 2021; Brohan et al., 2022a, 2023). People have been actively exploring the capabilities of this new form of intelligent entity (Wu et al., 2024a), such as embodied instruction following (Pashevich et al., 2021; Li et al., 2023d; Padmakumar et al., 2022), visual navigation (Hao et al., 2020), and manipulation grasping (Wu et al., 2024b). Furthermore, with the introduction of large models, significant progress has also been made in the field (Mousavian et al., 2019; Murali et al., 2020). However, while researchers, investors, and engineers generally believe that existing skill-driven scenarios may demonstrate the potential of embodied robots (Jiang et al., 2022; Gao et al., 2022), their performance in comprehensive business scenarios remains uncertain, leading to widespread concern.

To mitigate this issue, we believe that introducing simulations of commercial scenarios might be a potential solution. The main difference from current skill-learning-oriented scenarios (Padalkar et al., 2023; Mandlekar et al., 2023) is that commercial scenarios typically prioritize meeting human needs. This not only requires the integration of multiple skills (Wu et al., 2023) to achieve service objectives but also entails incorporating elements such as human-robot interaction (Long et al., 2023), scenario diversity (Deitke et al., 2020), and human behavior portrayal. The benefits of doing so are twofold: firstly, it can make robot training more closely resemble real commercial scenarios, and secondly, it can introduce new, more specific topics to the community, further promoting the community's evolution towards commercialization.

Furthermore, we systematically examined a large number of existing virtual environment systems (such as AI2thor (Kolve et al., 2017), Habitat (Puig et al., 2023), BEHAVIOR-1k (Li et al., 2023a), etc.(Makoviychuk et al., 2021; Handa et al., 2023; James et al., 2020)), which generally struggle to simultaneously support the depiction of commercial scenarios requiring interconnected multi-level architectural spaces, diverse and multi-functional social spaces (such as laboratories, medical rooms), customizable interactive human character and behavior, a plethora of interactable items, and continuously changing motion states supported by physics engines (Todorov et al., 2012; Haviland and Corke, 2023b,a). Thus, we constructed the aforementioned simulation environment from scratch, which is inspired by the polar research stations from the real world. Thus, we choose the human-centered in-building delivery service as an initial exploration into simulating embodied commercial scenarios.

3 Scenario Analysis & Task Definition

In the first place, we need to analyze and abstract the authentic scenario in order to generate actionable tasks and a quantifiable benchmark (Li et al., 2023a). In the context of precise in-building delivery services, referring to general robot tasks (Gao et al., 2022; Li et al., 2024), we have identified several potential key factors:

(1) Robots operate within a relatively fixed building space. (2) The residents within the building are the recipients of the service, and they typically move throughout the building based on personal needs and objectives. Robots can access relevant information about the recipients to better locate and identify them. (3) The transportation service may cover a substantial area, involving different floors and rooms. (4) Robots typically need to understand human instructions in order to search for and retrieve the correct target items, and deliver them to the designated recipients.

Based on the aforementioned scenario requirements, we provide the following task definition and settings, as shown in Table 1.

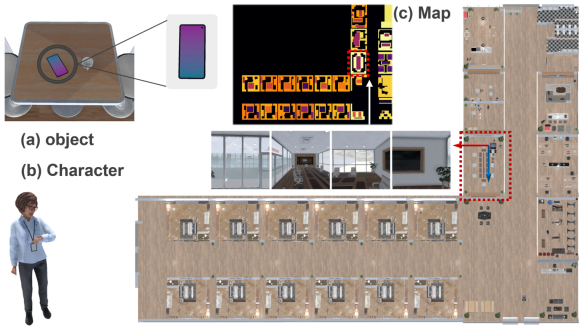

Figure 2: The available information in task. Context information guides robots to have a specific task goal.

The distinction in design between skill scenarios and commercial scenarios lies in their objectives. Skill scenarios tend to focus on exploring the efficiency of a particular skill under given conditions. On the other hand, real application scenarios exploration primarily revolves around identifying which conditions and information are most effective for achieving the ultimate goal in that scenario. There-

Task Setting	Content
Purpose	Deliver the requested item to the vicinity of the designated character.
Delivery Items	Items in the environment that can be grabbed and moved. (see item example in Figure 2)
Customers	Ten virtual human characters with different daily activities inside the building. They will move within the building for their own purposes.
Spatial Scope	The reachable areas within different rooms of a three-story building.
Time Setting	Real-world time, but simulation can be accelerated.
Customer Description	Self introduction and personal image photos, such as "*I'm John, a supervisor who's often busy with meetings and office work. ... my office is Room 2 on second floor. ... with a middle-aged man in a white shirt ... that's me.*" (see the personal image in Figure 2).
Scenario Map	2D projected obstacle map of scenario, and pre-sampled panoramic photos at various locations on the map (see panoramic image of sampling position in 2D obstacle map in Figure 2, which is built in advance or in real-time).
Robot Positioning	We adopt relative localization rules for robot positioning, where its initial position is always set to (0, 0, 0).
Robot Actions	Movement, joint control, and manipulation.
Robot Skills	Local navigation by coordinate, 6-DOF visual grasping, and pose adjustment.
Sensors	Two RGB-D cameras (head and arm), tactile sensors.
Customer Instruction	Describe the goal, Identify the target object, describe its location, and confirm the target person of the delivery. For example, "*Fetch the blue-packaged **water bottle** from the wooden **dining table** in the kitchen and deliver it to **Imani**, the woman in the blue shirt with black glasses, in the kitchen room*".
Success Criteria	Place the target object within a 3 meter range of the target person.
Constraints	Completion within 8 minutes without any dangerous collisions and unavailability of environmental metadata.

Table 1: Task settings and configuration of human-centered in-building embodied delivery task.

fore, for this delivery task, we strive to provide as comprehensive and diverse sensor information as possible to assist the robot in completing the task. Additionally, we continuously optimize the scenario and task design based on feedback from dataset users by adding more information channels.

4 Simulation Environment

Virtual environments typically need to meet the task requirements. Clearly, to depict corresponding physical indoor scenarios, existing environments are still constrained by factors such as the richness of the scene, the complexity of space, character portrayal, continuous environmental state systems, long-term operation, and the setup of items and robots. Therefore, we construct a brand-new virtual environment to support the tasks, as shown in Figure 3. Next, we will introduce the main features of the simulation environment.

Social Scenarios & Space with Height. As we mentioned, we need diverse common social scenarios. In existing work, we often see common indoor spaces such as kitchens, bedrooms, and living rooms, but less common are places like supermarkets, medical rooms, and studios. However, activities in different places vary greatly, and the premise of depicting diverse human activities in commercial scenarios is to include these settings. Moreover, it is the people constantly moving within these spaces that give them unique semantics. Additionally, we notice that existing work often confines scenes to a "flat" plane, with rare descriptions of "space with height", greatly limiting the spatial utilization of virtual environments in depicting complex scenes. Our virtual environment takes these factors into consideration.

Human Character. As mentioned earlier, the behavior of humans in commercial scenarios needs

Figure 3: PRS environment includes three-story buildings, items, human characters, and robots.

emphasis. The activities of the robot actually revolve around human activities. Therefore, in our virtual environment, we support an LLM-driven human character system (Non-Playable Character, NPC agents design) that controls goals, actions, and interactions. In this task, we mainly adopt various forms of daily activities (working, resting, simple socializing, etc.) to depict the actions of characters. Since the delivery task is closely related to the positions of characters, we primarily drive the movement of characters within the building based on schedule information.

Continuous Environment State. Our virtual environment is primarily driven by a physics engine at its core, containing items with physical properties, so almost all movements are continuous (with exceptions for specific object state changes controlled by scripts and interfaces). Even when we use robot control interfaces similar to the ALFRED style (Shridhar et al., 2020) (AI2-THOR(Kolve et al., 2017)), such as "*pick_obj()*", the movements it executes require real-time implementation through continuous body control.

Robot Configuration. We use a robot with grasping and movement capabilities. It is equipped with visual perception (RGB-D) and simple tactile perception based on rigid body collision. At the core, we have prepared various control methods for it. Users can control the robot either through an ALFRED-style interface (typically invoked by high-level action and LMMs with object segmentation) or through a ROS-like interface.

5 Dataset

We elaborate on the data collection process, encompassing the generation of language instructions, item placement, and scene construction. Additionally, we present the data annotation methodology and results. Tasks, environments, and agents all adhere to the PRS environment settings. The task scenes are located within a three-story building in the PRS environment. Target objects encompass all interactive items, while functional equipment follows physical engines and basic logic. Language instructions originate from a task generator, refined and reviewed manually through an LLM to ensure accuracy and diversity (more details see Appendix C). Although NPCs can move and act independently in the environment (Li et al., 2023b), and numerous interactive items and devices are present, we can configure and access all their states from the ground up, such as naming identifiers, spatial coordinates, and physical attributes. Based on the comprehensive environmental data obtained, we can generate tasks in real-time using preset scalable templates (Liu et al., 2019) and polish task instructions with a large language model. Moreover, the environment's data interface can easily acquire relevant task information to evaluate the computation methods for generated task results. (1) We constructed an environment-related PRS corpus, collecting verbs, nouns, and adverbs corresponding to actions, items, and locations. This corpus includes manually designed scalable templates that map verbs, nouns, and adverbs into sentences related to the current environment using reason-

Benchmark	PRS(Ours)	ALFRED (Shridhar et al., 2020)	EQA (Das et al., 2018)	VirtualHome (Puig et al., 2018)	BEHAVIOR-1K (Li et al., 2024)	Habitat (Puig et al., 2023)	iGibson (Shen et al., 2021)
Directive	✓	✓	✓	-	✓	✓	-
Continuous State	✓	-	-	-	✓	✓	✓
Articulated Joints	✓	-	-	-	✓	-	-
Mobile Characters	✓	-	-	✓	-	✓	✓
Autonomous NPC	✓	-	-	-	-	-	-
Elevator	✓	-	-	-	-	-	-
Long-term	✓	-	-	-	-	-	-
Human-centered	✓	-	-	-	-	-	-
Multi-floor	✓	-	-	-	✓	-	-

Table 2: Comparison between PRS delivery tasks and existing popular embodied simulation dataset.

	Test Set	Validation Set
Task	918	5730
Instructions	1836	11460
Annotation	GPT-4	GPT-4
Check	Manual	GPT-4
Ground Truth	-	✓
Scenes	19	24
Object Categories	42	43

Table 3: The validation and test sets of the dataset encompass distinct NPC behaviors, task contexts, and linguistic instructions. We validate against potential common scenarios as a benchmark for solutions without emphasizing training and fine-tuning. However, the validation set contains appropriately annotated information, which can also be used as a training set.

able grammatical rules (Xu et al., 2022). (2) We continuously generate task statements in the running simulated environment, refine and polish them using an LLM, and finally screen them to obtain 13296 language instructions, as shown in Table 3.

Unlike household tasks in Table 2, we specialize in the indoor environment of buildings and pay particular attention to how robots can deliver items across different floors and rooms in the building based on the needs of service recipients.

6 Baseline Method

The baseline method comprises multiple modules (Min et al., 2021), including the language, vision, and action modules, as shown in Figure 4, for tasks such as language parsing, navigation search, scene understanding, object recognition, segmentation, action, localization, and object manipulation.

6.1 Language Module

The language module utilizes a large language model (LLM) to process instructions ins and character introductions $intro$, outputting executable sequences based on pre-defined prompts (Wei et al., 2022), which specify the extraction of target information and visual feature from the task context, $LLM(ins, intro, prompt) = res$. The prompt defines the output format with the fixed symbols and includes result examples to facilitate the alignment of LLM output (Zamfirescu-Pereira et al., 2023). For instance, using the regular expression to decode relevant information, $RE(res) =< obj, recep_{obj}, room_{obj}, npc, room_{npc} >$. The executable sequences generated by LLM break down the task into subtasks, including corresponding information, such as target object search $[obj, ecep_{obj}, room_{obj}]$ (e.g., "*white cup, dinner counter, kitchen*"), grasping $[obj]$, delivery to $[room_{npc}]$ (e.g., "*office*"), and person search $[npc]$ (e.g., "*a man with grey coat*"). Subsequently, the robot sequentially performs these subtasks to accomplish the overall task accurately.

6.2 Vision Module

With information (*"a white cup"*) from the language module, the robot localize the specified object. The intricate spatial layout within indoor environments results in diverse positional arrangements of interactive objects, posing challenges for visual models in object detection. Furthermore, identifying diminutive, occluded, or container-enclosed objects presents a formidable obstacle (Zhu et al., 2021). To enhance search and recognition efficiency, we incorporate large receptacle information (*"dinner counter"*) derived from task instructions, or *"the*

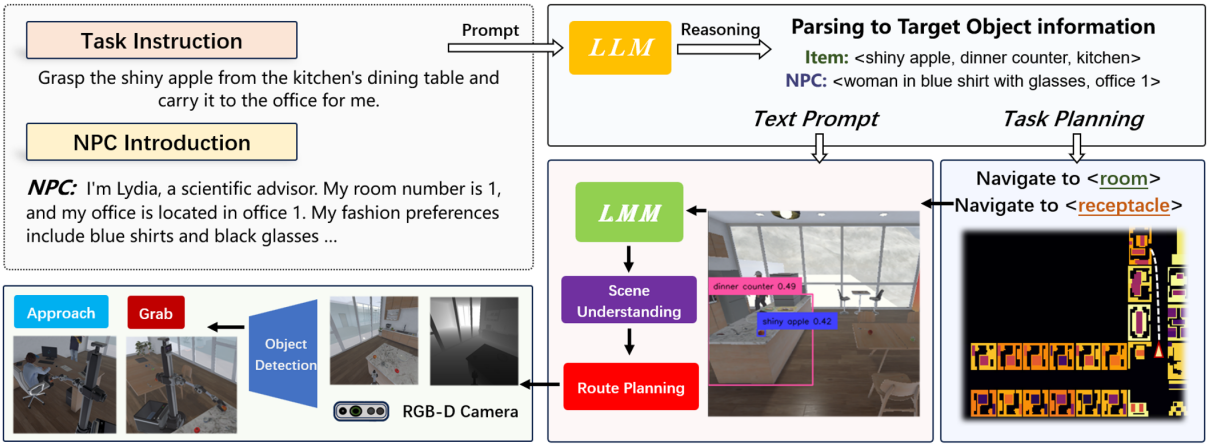

Figure 4: Modular method for the robot delivery task with LLM and LMM. LLMs and LMMs can achieve robotic perception and environmental interaction by invoking ROS-like APIs of PRS simulator.

apple on the dining table", *"the water cup on the desk."* Nonetheless, conducting direct object recognition within a room for item retrieval may yield considerable errors (Inoue and Ohashi, 2022). Suppose the confidence threshold is set excessively high. In that case, the visual model may struggle to identify the target object due to the item's unknown specific location and the potential distance of the robot from it. This can result in a limited number of pixels occupied by the object within the field of view, making it challenging to distinguish its visual features. Conversely, setting the confidence threshold too low can readily lead to erroneous identifications, such as mistakenly identifying objects with similar features as the intended target. Therefore, upon searching the target object in a room, the robot performs scene understanding using a large multimodal model (LMM) (Lu et al., 2023). Utilizing the current image captured by the robot's camera and the textual description of the target object as input, coupled with a constraining prompt such as *"Are there a white cup or large dining counter in the picture? Please answer 'yes' if so, and 'no' if not."* The robot rotates its head camera to observe the environment and employs the output of the LMM to ascertain the target object's presence within its field of view. Once the detection of the target object is detected by the LMM, the robot proceeds to either approach the object or utilize a visual model (Kirillov et al., 2023) to recognize and segment its mask. Leveraging the robust text-image alignment capabilities of LMM allows the robot to make reasonable decisions while mitigating ineffective visual processing outcomes. The visual module capitalizes on the LMM's capacity to align comprehensive visual data with object descriptions. Conversely, the small-scale visual model focuses on local information to accurately segment the mask of the target object, thereby facilitating an efficient and precise search process.

6.3 Action Module

Robots rely on environmental information and task context to execute navigation, approach, and grasping actions. Following the spatial directives provided in the output of the language module, the robot navigates to the designated area, such as "the kitchen." By aligning the target room information extracted from the instructions with the scene on the semantic map, the robot uses the A-star algorithm to plan an approximate route from its current location to the target room and navigates accordingly (Murray and Cakmak, 2022). With the segmented mask of the target object through the search strategy and recognition process, in conjunction with depth data captured by the depth camera, the robot calculates the approximate distance between itself and the target object. Should the distance exceed the desired range (1m-3m), the robot devises a local route based on map data to maneuver close to the target object. It adjusts its position and orientation by aligning its camera and body with the target object (item or NPC). The robot accurately identifies the item and acquires its 2D mask for object grasping, serving as input parameters for the PRS grasping API to complete the target grasping action (Corke and Haviland, 2021).

Method	Task SR	Parsing	Manipulation	Human Search	Time Spent
Rule-Based + GD	3.4	25.9	14.7	21.1	3.15
GLM-4V	18.7	68.3	45.7	78.6	4.68
GLM-4V + GD	23.7	65.2	53.4	83.2	4.27
GPT-4V+ GD	28.5	69.7	57.1	85.2	5.13
GPT-4o + GD	**32.2**	**71.7**	**59.2**	**88.7**	4.59

Table 4: Results on test set. **Method** includes various models for the baseline method, and GD is Grounding DINO(Liu et al., 2023) object detection model. **Task SR** is the success rate (SR) of the complete robot delivery task, **Parsing** is the SR of language instruction parsing to correct target information, **Manipulation** represents object grasping SR, **Human Search** is human character search SR, and **Time Spent** is time used (minute) on successful task execution. The LMM API used in the experiments is provided by the 2024 version.

7 Experiments

7.1 Evaluation Metrics

Two conditions determine the success of a delivery task: 1) successfully grasping the target object, which requires providing an accurate mask of the target object within an appropriate range, and 2) locating the target character, which is considered successful if the robot is within a Cartesian distance of 3m from the character. The ultimate goal is to deliver the object to the vicinity of the target character. Therefore, the delivery task can be decomposed into two sub-tasks, and the experiment will evaluate the efficiency and success rate (SR) of completing these sub-tasks (Shridhar et al., 2020).

7.2 Experimental Setup

Task success is fulfilling all the prescribed success conditions in the instructions. During task execution, the execution results are checked after all agent robots complete their actions. Each agent is allowed only one attempt and cannot repeat the execution. Any incorrect use of interface parameters, collisions with obstacles, interactions with the wrong target, or dangerous movements will result in task failure. The task execution time is limited to 8 minutes (Li et al., 2024; Shridhar et al., 2020) (balance the limit and tractability for the task). Access to underlying environmental data is not permitted, but all robot interfaces provided by the PRS simulator are available. The PRS simulator offers interfaces for robot control and sensing. For the experiment, LLM employs GPT-4 and GLM-4 (Zeng et al., 2022) (given the similarity in instruction processing capabilities between GLM-4 and GLM-4V, they are not separately listed in the table), LMM utilizes GPT-4V(ison) (Yang et al., 2023b) and GLM-4V (Wang et al., 2023), and the visual detection model uses Grounding DINO (Liu et al., 2023), all in a zero-shot setting. The performance of each sub-module will be tested separately without other modules' results.

7.3 Result

With a test set of 918 tasks, the efficiency of each module was calculated in the experiment, including language parsing, object search, and virtual human character recognition. Experimental results in Table 4 were compared, and the GPT-4o-based method achieved a task success rate of 32.2%.

8 Conclusions

This work integrates previous work on skill-learning scenarios and explores a specific in-building scenario with human-robot interaction at its core. Specifically, we have constructed a brand-new immersive environment system for human-centred in-building delivery services, including multi-level spatial buildings, diverse functional rooms, multi-role behavior systems, robots, and item systems, as well as a delivery service dataset and a baseline system. We believe that a significant and promising direction for the future is to integrate existing skills to simulate specific, determined, commercial, high-fidelity simulation scenarios, ultimately aiming to drive the development of community technology toward commercialization and realism-aligned robot service.

9 Limitations

Despite our efforts to address key issues in robotic task design, several limitations are present in our work. The simulation environment, though comprehensive, is confined to a three-story building, which restricts the scope and generalizability of

our findings to all real-world scenarios. While we incorporated a variety of scene categories, the number of environments remains limited, and the multi-level space architecture we developed lacks sufficient diversity. Additionally, the autonomous NPCs used for human-robot interaction have constrained capabilities due to performance limitations. Furthermore, the behavior of NPCs has been simplified to reduce computational costs, which may impact the realism of the simulated environment. Additionally, the language prompts are restricted to English, which limits the diversity of the dataset, and the study focuses on a single complex simulated environment, in contrast to the unpredictability and diversity found in real-world settings. Despite manual verification, the content generated by the large language models (LLM) and large multimodal models (LMM) may still exhibit biases and imbalances.

10 Ethical Considerations

Ethical concerns are central to our work, especially in data collection and robot deployment. While we have undertaken manual reviews to minimize biases, the dataset may still contain unintentional biases that may not reflect the full range of user needs, particularly those outside mainstream indoor scenarios. Additionally, although our study adheres to existing laws and regulations governing robot usage, deploying mobile robots in indoor spaces may raise privacy concerns for users and others. The human-robot interaction aspect of our robot delivery task could also introduce potential safety hazards and risks of object damage in real-world applications. Although ethical considerations are not explicitly discussed in the paper, we emphasize the importance of ensuring no privacy infringements or data breaches occur in the simulation and dataset. Researchers utilizing these resources are required to agree to basic terms, taking responsibility for any outcomes. We encourage open-sourcing of the code to promote transparency and foster community collaboration.

References

Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Manon Devin, Alex X Lee, Maria Bauza Villalonga, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, and 1 others. 2023. Robocat: A self-improving generalist agent for robotic manipulation. *Transactions on Machine Learning Research*.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, and 1 others. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, and 1 others. 2022a. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*.

Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, and 1 others. 2022b. Do as i can, not as i say: Grounding language in robotic affordances. In *the 6th Annual Conference on Robot Learning*.

Peter Corke and Jesse Haviland. 2021. Not your grandmother's toolbox–the robotics toolbox reinvented for python. In *the 2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11357–11363. IEEE.

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

Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, and 1 others. 2020. Robothor: An open simulation-to-real embodied ai platform. In *the IEEE/CVF conference on computer vision and pattern recognition*, pages 3164–3174.

Zipeng Fu, Tony Z Zhao, and Chelsea Finn. 2024. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*.

C Gan, J Schwartz, S Alter, M Schrimpf, J Traer, J De Freitas, J Kubilius, A Bhandwaldar, N Haber, M Sano, and 1 others. 2021. Threedworld: A platform for interactive multi-modal physical simulation. *Advances in Neural Information Processing Systems (NeurIPS)*.

Xiaofeng Gao, Qiaozi Gao, Ran Gong, Kaixiang Lin, Govind Thattai, and Gaurav S Sukhatme. 2022. Dialfred: Dialogue-enabled agents for embodied instruction following. *IEEE Robotics and Automation Letters*, 7(4):10049–10056.

Ankur Handa, Arthur Allshire, Viktor Makoviychuk, Aleksei Petrenko, Ritvik Singh, Jingzhou Liu, Denys Makoviichuk, Karl Van Wyk, Alexander Zhurkevich, Balakumar Sundaralingam, and 1 others. 2023. Dextreme: Transfer of agile in-hand manipulation from simulation to reality. In *the 2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5977–5984. IEEE.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *the IEEE/CVF conference on computer vision and pattern recognition*, pages 13137–13146.

Jesse Haviland and Peter Corke. 2023a. Manipulator differential kinematics: Part 2: Acceleration and advanced applications. *IEEE Robotics & Automation Magazine*.

Jesse Haviland and Peter Corke. 2023b. Manipulator differential kinematics: Part i: Kinematics, velocity, and applications. *IEEE Robotics & Automation Magazine*.

Yuki Inoue and Hiroki Ohashi. 2022. Prompter: Utilizing large language model prompting for a data efficient embodied instruction following. *arXiv preprint arXiv:2211.03267*.

Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. 2020. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. 2022. Vima: General robot manipulation with multimodal prompts. In *the NeurIPS 2022 Foundation Models for Decision Making Workshop*.

Yixiang Jin, Dingzhe Li, A Yong, Jun Shi, Peng Hao, Fuchun Sun, Jianwei Zhang, and Bin Fang. 2024. Robotgpt: Robot manipulation learning from chatgpt. *IEEE Robotics and Automation Letters*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, and 1 others. 2023. Segment anything. In *the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, and 1 others. 2017. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, and 1 others. 2024. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, and 1 others. 2023a. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *the Conference on Robot Learning*, pages 80–93. PMLR.

Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C Karen Liu. 2023b. Controllable human-object interaction synthesis. *arXiv preprint arXiv:2312.03913*.

Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. 2023c. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation. *arXiv preprint arXiv:2312.16217*.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, and 1 others. 2023d. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and 1 others. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.

Yang Liu, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. 2019. Experience-based causality learning for intelligent agents. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(4):1–22.

Yuxing Long, Xiaoqi Li, Wenzhe Cai, and Hao Dong. 2023. Discuss before moving: Visual language navigation via multi-expert discussions. *arXiv preprint arXiv:2309.11382*.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. 2023. Unified-io 2: Scaling autoregressive multimodal models with vision, language, audio, and action. *arXiv preprint arXiv:2312.17172*.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and 1 others. 2021. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. 2023. Mimicgen: A data generation system for scalable robot learning using human demonstrations. *arXiv preprint arXiv:2310.17596*.

So Yeon Min, Devendra Singh Chaplot, Pradeep Kumar Ravikumar, Yonatan Bisk, and Ruslan Salakhutdinov. 2021. Film: Following instructions in language with modular methods. In *the International Conference on Learning Representations*.

Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 2019. 6-dof graspnet: Variational grasp generation

for object manipulation. In *the IEEE/CVF international conference on computer vision*, pages 2901–2910.

Adithyavairavan Murali, Arsalan Mousavian, Clemens Eppner, Chris Paxton, and Dieter Fox. 2020. 6-dof grasping for target-driven object manipulation in clutter. In *the 2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6232–6238. IEEE.

Michael Murray and Maya Cakmak. 2022. Following natural language instructions for household tasks with landmark guided search and reinforced pose adjustment. *IEEE Robotics and Automation Letters*, 7(3):6870–6877.

Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, and 1 others. 2023. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*.

Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. 2022. Teach: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2017–2025.

Alexander Pashevich, Cordelia Schmid, and Chen Sun. 2021. Episodic transformer for vision-and-language navigation. In *the IEEE/CVF International Conference on Computer Vision*, pages 15942–15952.

Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. 2018. Virtualhome: Simulating household activities via programs. In *the IEEE conference on computer vision and pattern recognition*, pages 8494–8502.

Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, and 1 others. 2023. Habitat 3.0: A co-habitat for humans, avatars, and robots. In *The Twelfth International Conference on Learning Representations*.

Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne Tchapmi, and 1 others. 2021. igibson 1.0: a simulation environment for interactive tasks in large realistic scenes. In *the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7520–7527. IEEE.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749.

Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *the 2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE.

Hongcheng Wang, Andy Guan Hong Chen, Xiaoqi Li, Mingdong Wu, and Hao Dong. 2024. Find what you want: Learning demand-conditioned object attribute space for demand-driven navigation. *Advances in Neural Information Processing Systems*, 36.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, and 1 others. 2023. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Ruihai Wu, Haoran Lu, Yiyan Wang, Yubo Wang, and Hao Dong. 2024a. Unigarmentmanip: A unified framework for category-level garment manipulation via dense visual correspondence. *arXiv preprint arXiv:2405.06903*.

Ruihai Wu, Chuanruo Ning, and Hao Dong. 2023. Learning foresightful dense visual affordance for deformable object manipulation. In *the IEEE/CVF International Conference on Computer Vision*, pages 10947–10956.

Tianhao Wu, Mingdong Wu, Jiyao Zhang, Yunchong Gan, and Hao Dong. 2024b. Learning score-based grasping primitive for human-assisting dexterous grasping. *Advances in Neural Information Processing Systems*, 36.

Zhuoqun Xu, Liubo Ouyang, and Yang Liu. 2022. Task-driven and experience-based question answering corpus for in-home robot application in the house3d virtual environment. In *the Thirteenth Language Resources and Evaluation Conference*, pages 6232–6239.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. 2023a. Learning interactive real-world simulators. In *the NeurIPS 2023 Workshop on Generalization in Planning*.

Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023b. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9(1):1.

Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin S Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander Clegg, John M Turner, and 1 others. 2023. Homerobot: Open-vocabulary mobile manipulation. In

Conference on Robot Learning, pages 1975–2011. PMLR.

Anthony Zador, Sean Escola, Blake Richards, Bence Ölveczky, Yoshua Bengio, Kwabena Boahen, Matthew Botvinick, Dmitri Chklovskii, Anne Churchland, Claudia Clopath, and 1 others. 2023. Catalyzing next-generation artificial intelligence through neuroai. *Nature communications*, 14(1):1597.

JD Zamfirescu-Pereira, Richmond Y Wong, Bjoern Hartmann, and Qian Yang. 2023. Why johnny can't prompt: how non-ai experts try (and fail) to design llm prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–21.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, and 1 others. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. 2021. Soon: Scenario oriented object navigation with graph-based exploration. In *the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12699.

A Benchmark

To investigate preliminary commercial scenarios of robotic applications, we introduce the human-centered in-building embodied delivery task, aiming to deliver specified items to the vicinity of the target human character. We developed the PRS simulation environment and collected a dataset related to delivery tasks. Consequently, the delivery task benchmark encompasses a simulator, environment API, dataset, evaluation metrics, and baseline methods, as follows:

```
.
└── prs-delivery
    ├── simulator
    │   ├── Linux
    │   ├── Windows
    │   └── macOS
    ├── dataset
    │   ├── validation set
    │   └── test set
    ├── environment information
    │   ├── semantic map
    │   ├── NPC information
    │   ├── Robot.urdf
    │   └── environment setting
    ├── baseline Method
    │   └── LLM
    │   ├── LMM
    │   ├── object detection model
    │   └── main
    ├── evaluation
    └── API document
```

B Simulator

In light of our conceptualization of robotic service, and in order to better simulate comprehensive application scenarios, we have developed the Polar Research Station (PRS), a three-story building containing different rooms, providing (1) a PhysX-supported physical environment, (2) autonomous characters for performing human behaviors, (3) robots with perception sensors and interaction abilities, (4) interactive objects and devices with continuous state changes, and (5) available API for LLMs and LMMs. Figure 6 shows rich scenes that are close to the real world. The PRS rendering engine utilizes Unity and offers a diverse range of Python APIs. The resource is user-friendly (Python-only) and can even run without a GPU.

C Task Dataset

The dataset is represented as a JSON file in Listing 1, and task initialization, execution, and evaluation are accomplished using the Python API. The task involves a variety of objects with different styles, as depicted in the Figure 7. NPCs in the environment engage in continuous simulated life activities, generating various needs over time, such as eating, drinking, working, and resting. At these moments, NPCs potentially require certain items to fulfill their demands (e.g., food, drinks, mobile phones). Thus, we simulate robot delivery services by collecting these needs. By querying environmental data, we automatically gather a large number of delivery tasks. We refine the language content using the LLM and conduct manual checks and corrections, as shown in Figure 8. Specifically, we introduce LMM to perform textual annotation (visual feature description) of image data to decrease manual work and increase diversity. Figure 10 indicates ten distinct NPCs as the service targets, each with their own profile and preferences. Figure 11 illustrates the spatial distribution of scenes within the task set, demonstrating long-range visibility across spaces.

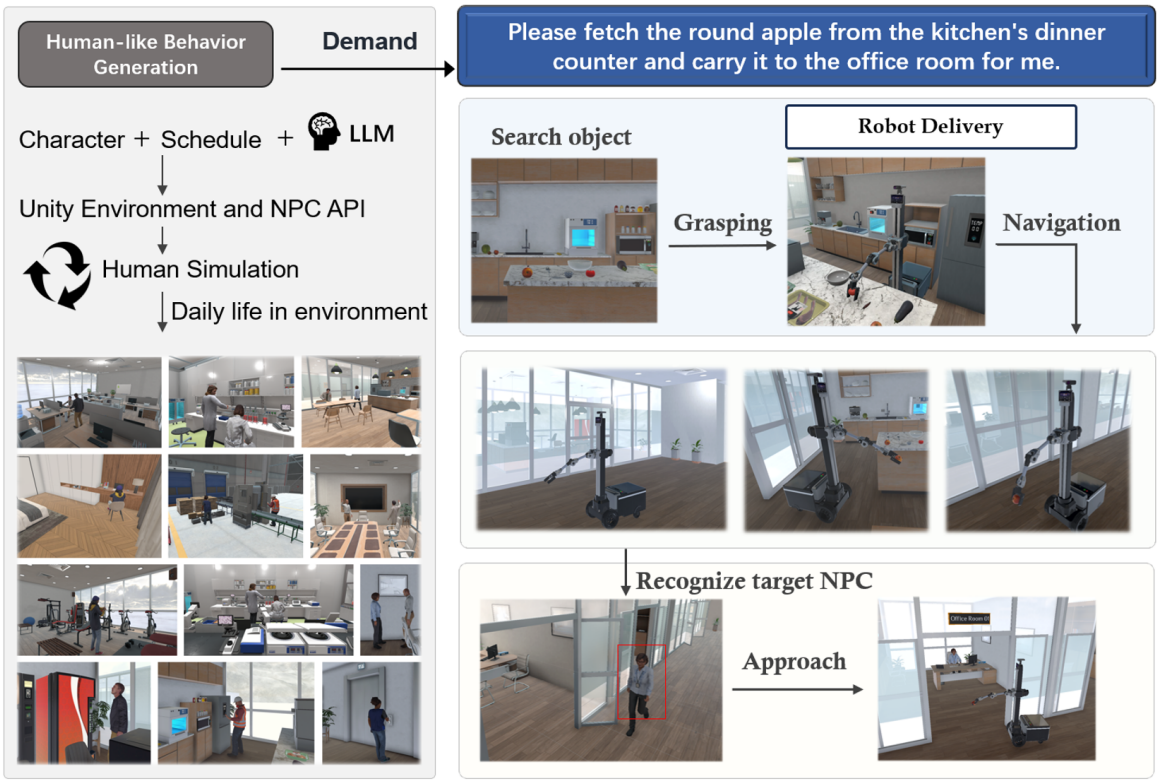

Figure 5: A data generation instance. We generate human activities, target objects, robot positions, task instructions, and a complete process of robot execution based on the settings combined with large language models.

Figure 6: Two-dimensional floor plan of the Polar Research Station building.

Listing 1: JSON file format for a human-centered in-building embodied delivery task, including task parameters and descriptions.

```
{
        "task_id": "1
            _2025_02_11T12_45_49_10_1_1"
        "npc_name": "Imani",
        "npc_id": 1,
        "time": "2025-02-11T12:45:49",
        "npc_action": "sit",
        "npc_position": {
            "x": -16.02390480041504,
            "y": 0.0,
            "z": -8.445791244506836},
        "target_object_name": "
            WaterBottle_Blue_1",
        "target_object_type": "
            WaterBottleBlue",
        "target_object_pos": {
            "x": -16.878999710083008,
            "y": 0.7600002288818359,
            "z": -5.263000011444092},
        "directive": [
            "Grasp the blue water bottle
                 from the wooden dining
                table in the kitchen and
                 bring it to me in the
                kitchen room.",
            "Fetch the blue-packaged
                water bottle from the
                wooden dining table in
                the kitchen and deliver
                it to Imani, the woman
                in the blue shirt with
                black glasses, in the
                kitchen room."],
        "npc_description": "I'm Imani, a
             scientific advisor at a
            polar research station. My
            room number is 1, and my
            office is located in office
            1. I often lead a regular
            life. My fashion preferences
             include blue shirts and
            black glasses."
}
```

Figure 7: Interactive objects for grasping and delivery with physical properties and authentic textures.

C.1 Data Augmentation

The delivery tasks encompass 10 NPCs, 23 rooms, and 47 types of items. We utilize various NPC information (e.g., names, occupations, habits) and actions, and alter the positions of NPCs, items, and robots to enrich the benchmark distribution. For each robotic delivery task, natural language instructions with relevant context are provided to simulate the robot's instruction following.

C.2 Dataset Split

Unlike past supervised learning settings, we propose that embodied tasks in simulated scenarios need not be based on the independent identically distributed (IID) assumption. Consequently, we modify the setting from the traditional "train-develop-test" to "free mode-develop-test," omitting an explicit training set (with ground truth information included in the validation set for training or fine-tuning purposes). In the free mode, researchers can freely collect data without restrictions to develop and debug solutions, such as visual recognition, scene understanding, and search strategies. We argue that this setting is more advantageous for large multimodal models (zero-shot) and closer to real-world scenarios, where it is impossible to pre-acquire all user scenarios but rather to handle various potential scenarios with general solutions.

C.3 Accessibility

We have made the PRS simulator and robot delivery dataset available and accessible to all. The simulator is provided in Linux (Ubuntu), macOS, and Windows, with continuous updates and maintenance. We explicitly offer a usable API and usage examples. Additionally, we have opened an online result evaluation for the validation and test sets by Eval AI.

C.4 Responsibility

We are responsible for the content of the simulator and dataset, ensuring no infringement or privacy breaches. Researchers must agree to our basic terms before usage, which include taking responsibility for outcomes resulting from utilizing these resources for development, deployment, and research. We encourage researchers to open-source their code to facilitate community efforts.

D Delivery Process

Figure 9 shows that the task can be decomposed into several subtasks, each with explicit goals and termination conditions. The robot delivers items amidst dynamic environmental changes and NPC behaviors. Additionally, the environment features numerous interactive objects, resulting in unpredictable circumstances throughout the task, as illustrated in Figure 12.

D.1 Robotic Skill

In the delivery task, some executions are simplified for industrial and standard processes. Robot manipulation and navigation remain significant challenges, with different AI models addressing various robot types. We focus on comprehensive simulation of scenarios and performance evaluation without delving into the details of robot skill learning. Consequently, based on robotics standards, we provide a high-level API (ROS-like, e.g., *prs.agent.goto_target_goal((-2.25, 0.1, -7.25), radius=1.7), prs.agent.object_interaction(input_matrix=segment*

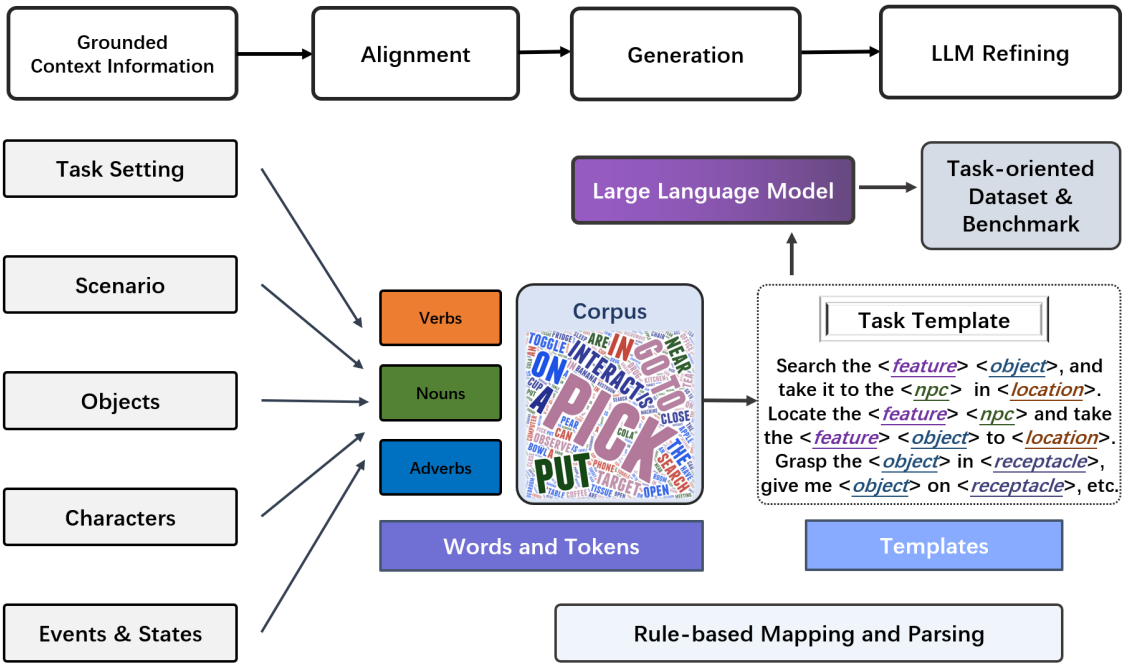

Figure 8: Automatic generation pipeline of the task instruction with corpus, templates, and LLM.

matrix, manipulation=1, camera_type=0)) for navigation and grasping. Specifically, we offer a rough obstacle map and semantic and observation image sampling (facilitating scene comprehension and room differentiation), obviating the need for robot SLAM in large spaces during each task execution. For the robot to successfully grasp the target object, a correct segment mask must be provided within a 1.2m range, with the PRS environment already offering built-in coordinate transformation, inverse kinematics (IK) calculations, and joint control. Thus, the task solutions utilize ROS-like APIs, which abstract the specific robot model and align more closely with general algorithms. The ROS-like API setup allows for robot morphology and structure modifications at a low cost, enhancing sim2real performance.

D.2 Baseline Reproducibility

We employ a zero-shot setting (LLM, LMM, zero-shot object detection model) instead of model fine-tuning in the baseline method. We have released the baseline and dataset document. This setup holds significant advantages in reproduction and secondary development. Besides replacing models, researchers can explore better prompts, target searching, navigation strategies, semantic alignment, context processing, etc., to enhance the robot's efficiency.

E Contributions

Threads	Frames Per Second (FPS)		
Nums	RTX3090	RTX4050	RTX4090
1	93.1	62.7	123.5
2	62.4	45.6	96.4
3	43.9	28.4	67.1
4	30.5	16.7	45.2
5	21.4	-	33.7
6	-	-	26.3

Table 5: Performance of the PRS environment under different thread counts and GPU. The PRS environment supports parallel simulation and achieves high frame rates across multiple hardware configurations. Based on a robust physics engine (PhysX), it enables robots to collect data for vision models or attempt manipulations for reinforcement learning (RL). We have considered agent training (RL or multimodal data collection) within the PRS environment.

Our work introduces a novel simulation environment that advances robotic learning and interaction in complex, long-term, and human-centered settings. Unlike previous benchmarks, we emphasize realistic environmental contexts, time-sensitive interactions, and human-robot collaboration. Key contributions include:

E.1 Robotic Manipulation Framework

Grasping is fundamental to robotic manipulation, placing, moving, opening, and closing. In error analysis, most failures arise from unsuccessful grasps due to action failure or target misidentification. High-level commands primarily cause manipulation errors, as low-level inverse kinematics and joint dynamics seldom lead to significant failures.

E.2 Enhanced Simulation Environment

Unlike Behavior-1K, our simulation integrates multiple rooms into multi-layered buildings and incorporates a time dimension, enabling continuous learning over long-term simulations. Compared to Habitat 3.0, we introduce LLM-driven NPCs with needs-based autonomous behaviors, supporting human-robot collaboration and realistic service-oriented tasks.

E.3 Physics-Driven Robotics Benchmark

Built on a robust physics engine, the PRS environment supports robotic data collection for vision models and reinforcement learning (RL). We provide 40+ object categories for manipulation verification and fine-tuning, improving the generalization of robotic grasping models. Additional 3D resources further enhance robotic interaction capabilities.

E.4 Realistic Task Design

In contrast to Omnigibson (Behavior-1K), we define robotic delivery tasks with human simulation, environmental context, complex spatial semantics, and long-term sequences rather than isolating navigation or manipulation as standalone tasks. Our benchmark integrates autonomous NPCs and multi-room, multi-story environments to reflect real-world task demands, supporting long-term simulations and practical robotic service modeling.

E.5 Scalability and Generalization

With a modular architecture and efficient rendering pipeline, we continuously expand scenarios and architectural structures to enhance model training and generalization testing. Given that no simulation fully replicates the real world, we assess zero-shot robotic performance in the PRS delivery task to evaluate LMM-based robotic systems' ability to provide practical services. We provide standardized APIs, including ROS-like interfaces and LMM integration, facilitating real-world deployment.

E.6 Agent Training and RL Support

The PRS platform enables RL and multimodal data collection, offering APIs for sensor signal retrieval (RGB-D, tactile), environmental data access, and motor control (discrete motion, joint angles, forces). Interactive elements such as articulated objects, NPCs, elevators, and virtual devices enrich training environments. An automatic task generator enhances usability, allowing large-scale task generation for model validation.

E.7 Performance and Efficiency

The PRS environment maintains a physics frame rate of 60 fps, meets RL requirements, and achieves rendering rates exceeding standard RGB-D cameras. Motor-level and joint control APIs support precise robotic learning and benchmarking.

E.8 Zero-Shot Benchmarking for LMMs

Our baseline employs zero-shot evaluation without fine-tuning, testing the general understanding capabilities of Large Multimodal Models (LMMs) in robotic tasks. LLM-generated language instructions resemble real-world dialogues and undergo manual review to remove biases and ambiguity. Customizable instruction generation enhances task scenario diversity.

E.9 Occlusion Handling in 3D Environments

Instead of relying on point cloud completion, our approach allows robots to change viewpoints dynamically to locate objects, leveraging multiple perspectives for robust target identification.

Connected, Multi-Layered Spatial Contexts Unlike prior works such as ProcTHOR, our environment features semantically rich multi-layer buildings rather than isolated rooms, incorporating autonomous NPCs with daily routines. This fosters realistic robotic interactions, supporting human-like task engagement and long-term goal execution. Error Analysis in Robotic Delivery By evaluating sub-task performance in baseline models, we identified key failure sources:

- 40%: Manipulation failures (grasping errors)
- 29%: Language inference errors
- 18%: Object localization failures
- 13%: Inability to locate target NPCs

Since delivery success depends on completing all sub-tasks, improving each module enhances overall

performance. Our findings highlight grasping and object search as significant challenges in robotic service applications.

E.10 LMMs in Robotics

LMMs provide extensive world knowledge and reasoning abilities, supporting instruction parsing, behavior planning, perception, and decision-making. Unlike traditional robotic learning, LMMs eliminate the need for frequent retraining of visual and language models. In our PRS delivery benchmark, a zero-shot LMM-based system achieved a 32.2% task success rate, demonstrating LMMs' potential while underscoring remaining challenges in object search and manipulation.

E.11 Distinctions

Our work advances robotic simulation by integrating realistic human-robot interactions, scalable task modeling, and robust environmental dynamics, bridging the gap between academic research and real-world deployment.

E.11.1 Autonomous NPCs

Virtual human characters that engage in independent activities based on environmental context rather than serving as mere obstacles.

E.11.2 Long-Term Simulation

The environment supports the continuous execution of multiple tasks within the same spatiotemporal setting rather than resetting after each task instance.

E.11.3 Human-Centered Tasks

Robotic execution is guided by contextual information about specific NPCs rather than purely linguistic instructions.

F Limitations

Although we have considered data augmentation and variations in style, we only constructed a three-story building and thus cannot cover all scenarios. Our dataset content has been manually verified, but the generated content of LLM and LMM may still exhibit bias and imbalance. To reduce computational expense, we simplified NPC behaviors. We simulated a robot application scenario, but the real world is far more complex and unpredictable. LLM prompts include different tone and content requirements to synthesize diverse and universal data, albeit limited to English content.

G Future Work

In-building delivery is a realistic commercial scenario, differing from the popular factory assembly line scenario in that it involves more consideration of human-robot interaction. Therefore, in our future work, we will introduce (1) richer user interaction behaviors, such as users being able to send real-time location hints to the robot, and (2) longer-term user behavioral data, enabling the robot to summarize user behavior patterns for more precise service autonomously. (3) More diverse scenarios, items, and tasks. Our business scenario design, virtual environment setup, and dataset collection will iterate and continuously improve alongside research efforts in the community, commercial developments, the robotics industry, and user research.

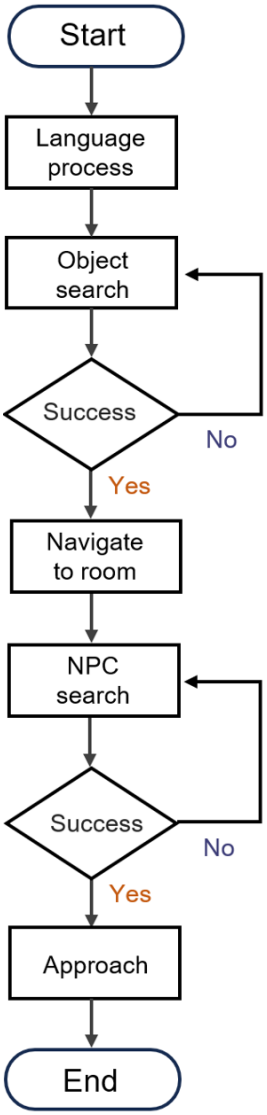

Figure 9: A flowchart visually representing the sequential steps and decision points involved in the human-centered in-building embodied delivery task.

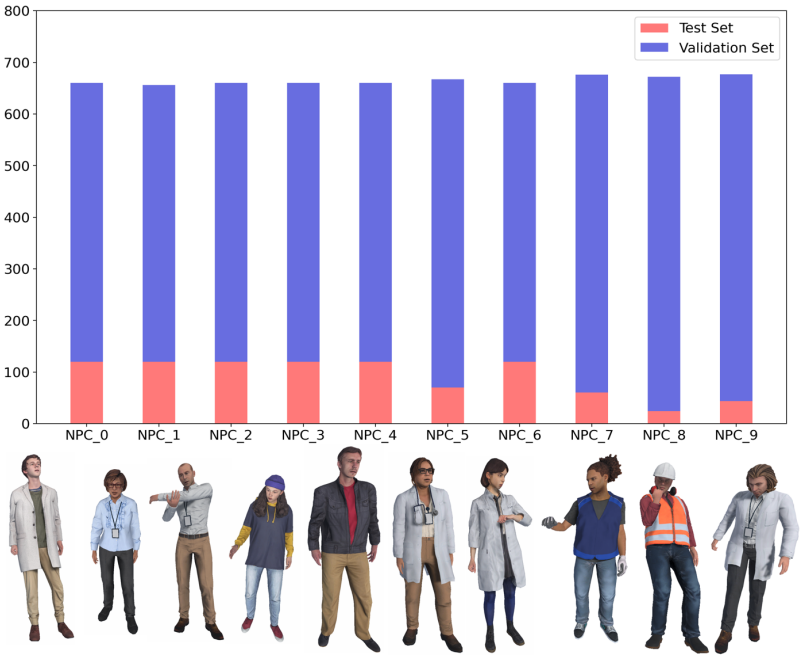

Figure 10: The frequency of NPC appearances with different jobs and habits in the PRS dataset.

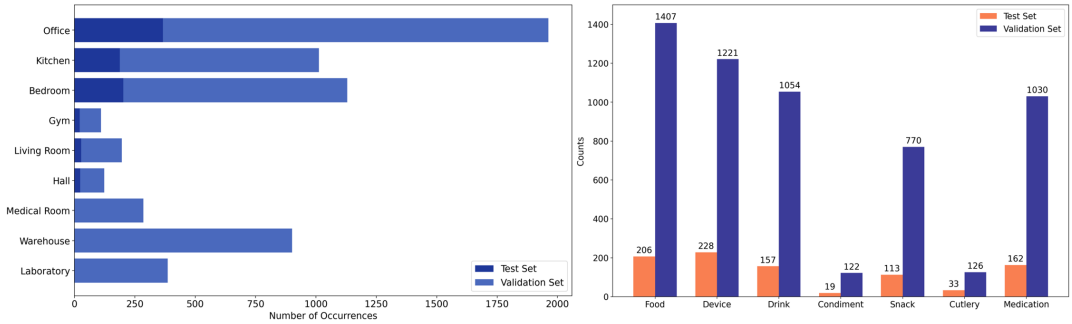

Figure 11: Statistics of different scenes and interactive object categories in the dataset.

Figure 12: Demonstrations showcasing examples of the delivery task in PRS scenarios.