SoPE: Spherical Coordinate-Based Positional Embedding for Enhancing Spatial Perception of 3D LVLMs

Supplementary Material

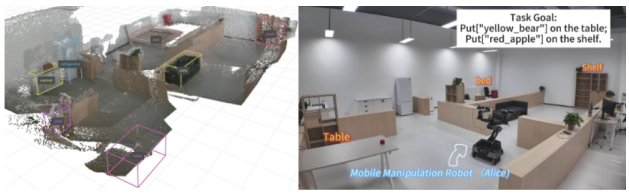

Figure 1. Detection visualization and task environments of SpatialSoPE in real-world reconstructed scenes.

1. Real-World Validation

1.1. Settings

In this section, we present a detailed account of the real-robot experimental validation process. Fig. 1 illustrates the reconstructed scene point cloud, the model's visual outputs, and the corresponding real-world setup. We build a simulated indoor home environment containing a table, sofa, bookshelf, refrigerator, and other common furniture. The experimental platform is the Galaxea R1 Lite mobile dual-arm robot, equipped with a wrist-mounted RealSense D435i, a RealSense D455, and a Mid-360 LiDAR. We design distinct atomic action sets for the robot to encapsulate its physical capabilities, including:

1. `[navigate] to <stand_pose_id> of <object>;`
2. `[open] <container>;`
3. `[pick] up <object>;`
4. `[place] <object> on/into <platform>;`
5. `[move] <delta_x> and <delta_y>;`

Following the methodologies of prior work [**?** **?**], we employ LLM-based role-playing to guide task planning through a Chain-of-Thought (CoT) framework. The robot selects and executes one action at a time from a predefined list. After each action, it receives feedback and, using this information together with its execution history, autonomously determines the next subtask or adjusts its plan. This multi-turn process enables effective coordination for long-horizon tasks. The detailed prompt design is provided in Subsection 1.2.

In the real-robot system, for manipulation tasks, we use AnyGrasp to predict grasp points and apply Grounded SAM to filter candidates based on semantic targets. The resulting grasp poses are executed via inverse-kinematics planning. For navigation tasks, we adopt a SLAM pipeline for mapping and localization, with A* as the global planner and DWA as the local planner.

We further integrate the multimodal *GPT-4o* model to obtain scene understanding and action-level feedback in real-world environments. After each action, the manipulator's wrist camera captures the current scene to assess execution success and overall task progress. For example, once the mobile robot reaches a new piece of furniture, *GPT-4o* identifies the objects on it to support exploration in unknown environments. Similarly, after each picking or placing operation, *GPT-4o* verifies whether the action was executed correctly. Additional feedback, such as pose estimation or inverse-kinematics results, can also be incorporated into the evaluation process.

1.2. Prompts

==== System Prompt ====

Role:

1. You are an intelligent robot named ${name}, configured with a wheeled chassis and a single manipulator arm.
2. You possess the ability to navigate across the ground and perform manipulation tasks, including transporting various objects and opening hinged objects.

Skills:

1. navigate(obj, pose); open(obj);
2. pick(obj); place(obj, loc); move(dx, dy);

Task Objective and Context:

1. The overall task is: ${target_task}.
2. Ingredients are scattered in an unknown indoor environment. The scene graph shows furniture locations but not their contents. Based on task goal, objects must be placed.

Principles:

1. Efficiently explore and navigate all locations in the scene graph without repetition.
2. Transport task-related items promptly.
3. Track task progress and adjust targets timely.
4. If grasp fails, try other stand poses or adjust base position.
5. Focus on completing the task without unrelated actions.

Output Response Format:

1. Thoughts: think step by step to analyze the problem;
2. Contents: choose and execute only one action from the action functions above.

CoT: Let's think step by step!

Examples: The following examples are provided for reference in decision-making. The related content involved has nothing to do with the actual task: [...]

==== User Prompt ====

Scene Graph:

drawer: (pos: [...], ori: [...], state: close, stand: [...]), cabinet: (...), sofa: (...), ... (It is generated by SpatialSoPE)

Robot Status:

Current robot states: pos: [...], ori: [...]. The gripper is empty.
Feedback History:
The historical feedbacks, from oldest to newest, are as follows: [... Robot successfully reached the target book case-stand pose 0. [‘book 0’, ‘book 1’] are found on the book case ...]
Action History:
The historical actions, from oldest to newest, are as follows: [... navigate(book case, stand pose 0) ...]