Appendix

I. Our Position and Advantages

Position. We position our work as a critical bridge between two well-studied paradigms (Fig. II a): (1) Visual reasoning on 2D images, which lacks any notion of active perception or physical action. (2) Embodied reasoning in 3D space, which requires a complete 3D simulator. This positioning makes our contribution ***modular and composable***. Our visual search policy is a direct and natural precursor to a downstream locomotion policy. The agent can first use our method to decide where to go at a junction, and then execute the movement. *In a word, our work provides a foundational layer for building generalized embodied agents across diverse human-centric environments.*

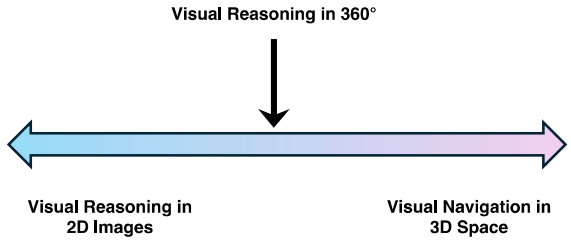

Figure I. Illustration of our position.

Advantages. While prior work typically needs 3D simulators that are hard to scale and limited to household scenes, we bypass the sim-to-real gap via real-world 360 panoramas, and enable the scalable study of visual reasoning across complex in-the-wild scenes. Most importantly, *our work fills the active vision gap for open-world embodied AI, and our dataset also serves as a valuable resource for building in-the-wild 3D simulators.*

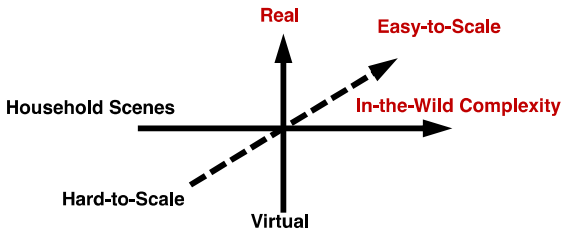

Figure II. Illustration of our advantages.

II. Geographical Distribution of *H*Bench*

As shown in Fig. III, our dataset exhibits broad geographical coverage, including thirteen cities across twelve countries and four continents. This diversity is reflected in the wide range of architectural styles, languages and scripts on signage, and environmental conditions.

III. HVS Task Difficulty Visualization

III.1. HOS Difficulty Visualization

Figure V illustrates the three difficulty levels of our ***HOS*** task with concrete examples. For each keyframe, we overlay the target object's area on the initial view and provide its visibility ratio d. From left to right, we show one example per level: **Easy** (mostly visible), **Medium** (partially visible), and **Hard** (invisible), demonstrating the model's initial observation.

III.2. HPS Difficulty Visualization

We define the four difficulty levels of the ***HPS*** task based on two criteria: whether the relevant cue aligns with the navigable path and whether textual information is provided. Examples are shown in Fig. VI-Fig. IX.

IV. Training and Inference Prompts

The natural-language prompts used for training and inference of the ***HOS*** and ***HPS*** tasks are shown in Fig. IV.

V. Objective Functions

SFT stage. The SFT objective function is the expected negative log-likelihood (cross-entropy loss) over the dataset $\mathcal{D}^{SFT}$ which consists of task input x and labeled trajectory $\mathcal{H}_T$:

$$\min_{\theta} \mathbb{E}_{(x,\mathcal{H}_T)\sim\mathcal{D}^{SFT}} \left[-\sum_{i=0}^{T-1} \log \pi_\theta(y_i, a_i \mid o_i, x, \mathcal{H}_i) \right].$$

RL stage. For each task, we sample G times and get outputs $\{\omega_1, \omega_2, \ldots, \omega_G\}$ where ω_i includes all the output tokens in the output sequence $\{y_0, a_0, y_1, a_1, \ldots, y_{T-1}, a_{T-1}\}$, then calculate the GRPO advantage to update the parameters. The GRPO objective function is:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[(s_o, x, y) \sim \mathcal{D}^{RL}, \{\omega_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\Omega|s_o, x)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|\omega_i|}\sum_{t=1}^{|\omega_i|}$$
$$\left\{ \min\left[\frac{\pi_\theta(\omega_{i,t}|s_o, x, \omega_{i,<t})}{\pi_{\theta_{\text{old}}}(\omega_{i,t}|s_o, x, \omega_{i,<t})} \hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\theta(\omega_{i,t}|s_o, x, \omega_{i,<t})}{\pi_{\theta_{\text{old}}}(\omega_{i,t}|s_o, x, \omega_{i,<t})}, 1-\epsilon, 1+\epsilon \right) \hat{A}_{i,t} \right] - \beta\, \mathbb{KL}(\pi_\theta \| \pi_{\text{ref}}) \right\},$$

where $\hat{A}_{i,t}$ denotes the group relative advantage at $\omega_{i,t}$:

$$\hat{A}_{i,t} = \frac{r_i - \text{mean}(r)}{\text{std}(r)}.$$

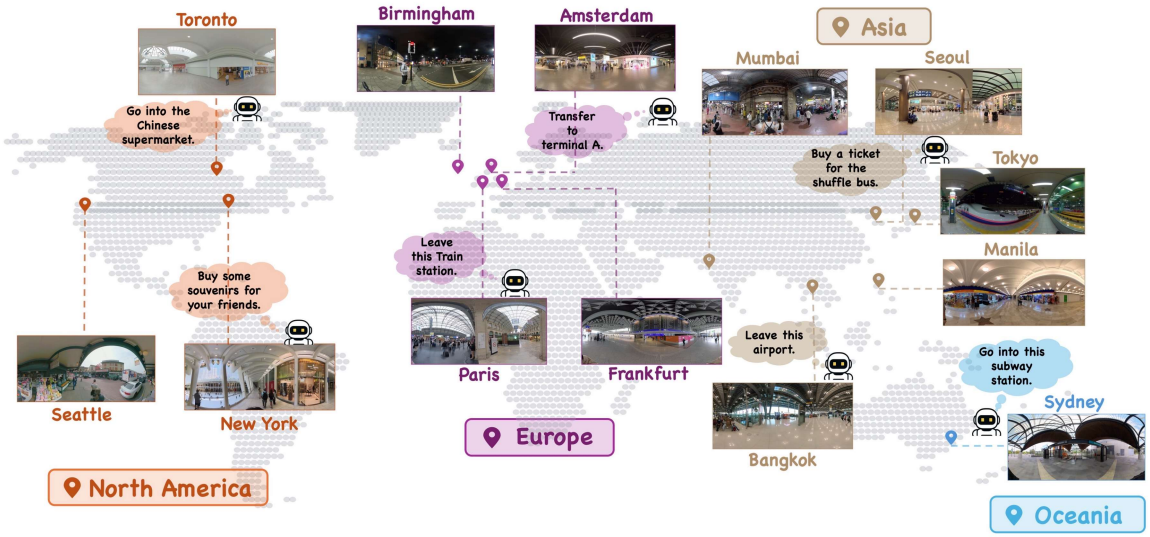

Figure III. ***H**Bench*** aggregates panoramic videos from diverse global locations, featuring visually cluttered environments.

VI. Reward Shaping

We use rule-based reward function to calculate the reward of the trajectory, which includes correctness reward, format reward.

$$r = r_{corr} + r_{form},$$

where:

$$r_{\text{corr}} = \begin{cases} 0.5, & \text{if the submitted answer satisfies the completion condition,} \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{\text{form}} = \begin{cases} 0.5, & \text{if the response is in <think></think> <answer><answer>format,} \\ 0, & \text{otherwise.} \end{cases}$$

Specially, we add a distance-to-goal reward for ***HPS***. It is calculated by the distance of the final direction to the target bounding box.

$$r_{dist} = \frac{\pi - d(\phi_T, \phi^*) + \pi - d(\gamma_T, \gamma^*)}{2\pi}.$$

Distance to bounding box is calculated by:

$$d(\alpha, \alpha^*) = |\alpha - (\alpha^* - \tau_\alpha)| + |\alpha - (\alpha^* + \tau_\alpha)|$$

which remains a constant minimum value when the direction is in the bounding box.

VII. Experimental Setup

Training Details. For the SFT stage, we use a learning rate of 1e-5. For the RL stage, we apply the GRPO algorithm with a batch size of 32, an actor learning rate of 1×10^{-7}, and a KL penalty coefficient $\beta = 0.01$. Rollouts are conducted under the ***H**Bench*** prompts with a temperature of 0.7, a maximum of 8 trajectories, and a dynamic turn limit (5 or 10) based on computational resources. Both stages use an input resolution of 1280×720 and are run on 8 NVIDIA H100 GPUs.

Benchmark Setting. The maximum number of inference turns is set to 10, as the step-cumulative success rate converges before this limit. At each step, the model processes up to five perspective images and uses the latest five dialogue turns as context. The image resolution is 1920×1080, and the sampling temperature is 0. Due to computational constraints, each episode is capped at 10 steps, with unfinished episodes counted as failures.

Train-Test Split. We annotated ∼3,000 task instances. These were divided into three mutually exclusive splits per task: a benchmark split, an SFT split, and an RL training split. Specifically, we reserved 1,000 instances (600 ***HOS*** and 400 ***HPS***) as the ***H**Bench***, resulting in 4,000 evaluation episodes. From the remaining data, we constructed the SFT dataset by randomly sampling 250 instances from both the ***HOS*** and ***HPS*** pools. All leftover instances were allocated exclusively for RL training.

VIII. Additional Qualitative Error Analysis

In Section 5.2, we identified two common-sense reasoning errors in the ***HPS*** task: (1) ***lack of physical commonsense*** and (2) ***lack of socio-spatial commonsense***. This section provides a detailed explanation of these error types, supported by additional qualitative examples.

Inference Prompt

SYSTEM PROMPT - HOS
You are a robot and perform object searching tasks according to instructions. Your goal is to rotate the camera to center the target object in the camera view, and then submit the task. The camera center is presented as a green cross in the picture.

SYSTEM PROMPT - HPS
You are a robot and perform navigation tasks according to instructions. Your goal is to turn your camera center to the target direction you need to move towards to reach the target location. The camera center is presented as a green cross in the picture. Don't move in the unavailable direction, such as obstacles or gaps.

Actions you can take: rotate(yaw:int,pitch:int), submit(yaw:int,pitch:int)
rotate(yaw:int,pitch:int): rotate the camera in the yaw and pitch direction relative to the current direction. Yaw is the rotation angle in the x-y plane, pitch is the rotation angle in the y-z plane. Yaw angle < 0 means rotate to the right, yaw angle < 0 means rotate to the left. Pitch angle > 0 means look up, pitch angle < 0 means look down.
submit(yaw:int,pitch:int): submit the task with the current camera view with the target object at the center, yaw and pitch are the angles of the current camera view, which is reported by the environment. You can only take one action at a time. The instruction will be provided with each observation. Look at the image carefully to complete the instruction.

FEW SHOT EXAMPLE (disabled after training)
Example:
Round 1: image_1
<think>I need to find the coffee machine. I can see a table on on my left, a couch in front of me, and a door to the right. The coffee machine is likely on the table, which is to my left.</think><answer>rotate(-45,0)</answer>
Round 2: Env_feedback: Last action is executed successfully, your current direction (yaw,pitch) is (315,0).
image_2
<think>From the secene, I see that by turning left 45 degrees, a kitchen table is in front of me. The coffee machine is on the left of the table and slightly lower than the camera center. I need to turn leftward and downward a little bit.</think> <answer>rotate(-30,-5)</answer>
Round 3: Env_feedback: Last action is executed successfully, your current direction (yaw,pitch) is (285,-5).
image_3
<think>The coffee machine is right now at the center of my camera, I think I can submit the task.</think> <answer>submit(285,-5)</answer>
Round 4: Env_feedback: Success
You can take 1 action(s) at a time. You should first give your thought process, and then your answer.
Your response should be in the format of: <think>...</think><answer>...</answer>
e.g. <think>I need to find the coffee machine. I can see a table on on my left, a couch in front of me, and a door to the right. The coffee machine is likely on the table, which is to my left.</think><answer>rotate(-45,0)</answer>

USER PROMPT
After your answer, the extracted valid action is {valid_action}.
The environment feedback is: {env_feedback}
done: {done}
After that, the observation is: {observation}
Human Instruction: {instruction}
Decide your next action.
You can take 1 action(s) at a time. You should first give your thought process, and then your answer.
Your response should be in the format of: <think>...</think><answer>...</answer>

Figure IV. Prompt used for inference and rollout.

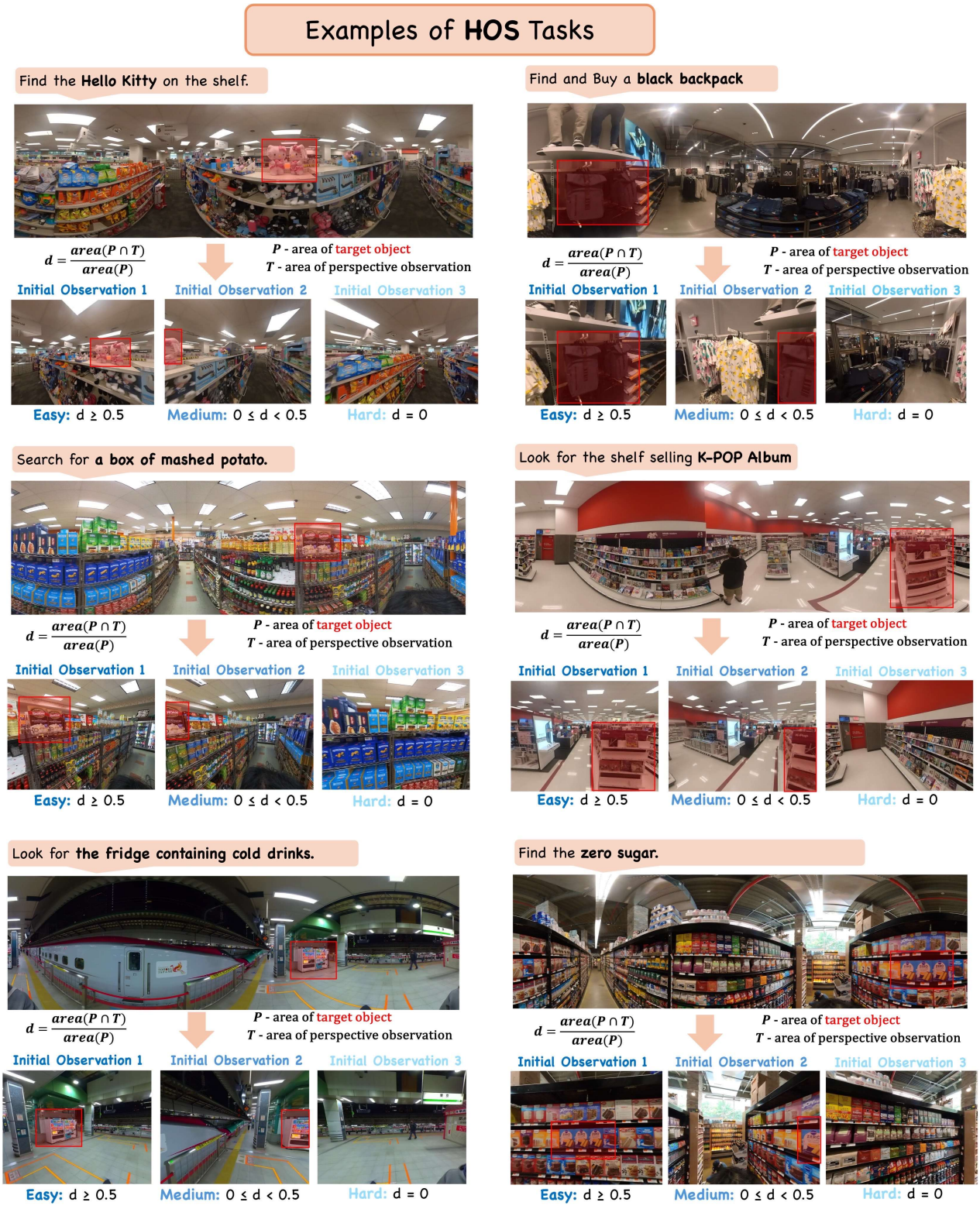

Figure V. Visualizations of ***HOS*** task instances.

Figure VI. Visualizations of easy-level ***HPS*** task instances.

Lack of Physical Commonsense. This error type denotes a failure in applying intuitive knowledge about 3D geometry and basic physics. Potential failures caused by the lack of physical commonsense include:

- **Ignoring Permanent Obstacles:** Attempting to move through non-traversable objects such as solid walls, glass barriers, or furniture.
- **Misjudging Vertical Connections:** Failing to understand how different floors are connected, *e.g.*, not recognizing that a staircase or elevator is required to change levels.
- **Direct Path Fallacy:** Heading in a straight-line towards the target without first identifying a feasible path, thereby ignoring the need to follow corridors, detour around obstacles, or use doorways.
- **Overlooking Drop-offs:** Proposing a path that would lead to falling from a significant height, such as walking off a balcony or over a ledge.

Lack of Socio-Spatial Commonsense. This error refers to the agent's inability to grasp the implicit social norms and functional roles of different areas in public spaces, leading to potential failures as follows:

- **Violating Traffic Norms:** Jaywalking across a busy driveway or road instead of using a nearby crosswalk or pedestrian lane.
- **Trespassing Restricted Zones:** Attempting to cut through behind a retail counter, through a staff-only area, across a floor hazard warning sign (*e.g.*, "Wet Floor"), or through a private property shortcut.
- **Disregarding Spatial Etiquette:** Violating implicit social norms that govern public behavior, *i.e.*, actions that are physically possible but socially disruptive, such as interrupting a queue or invading personal space.
- **Ignoring Functional Layouts:** Violating explicit architectural constraints and intended circulation paths, *i.e.*, attempting to navigate through physical obstructions (like tables or seats) rather than designated aisles.
- **Misusing Spaces:** Proposing to walk through a decorative fountain or flowerbed, or using an emergency exit as a routine shortcut.

Additional qualitative examples for all five error types (two for ***HOS*** and three for ***HPS***) are provided in Figures X and XIV.

IX. Case Study

In this section, we provide qualitative case studies (See Figs. XV-XVII) comparing the behavior of the model before and after post-training. For each case, we focus on one of the three key capabilities introduced by post-training, as described in Sec. 5.3.

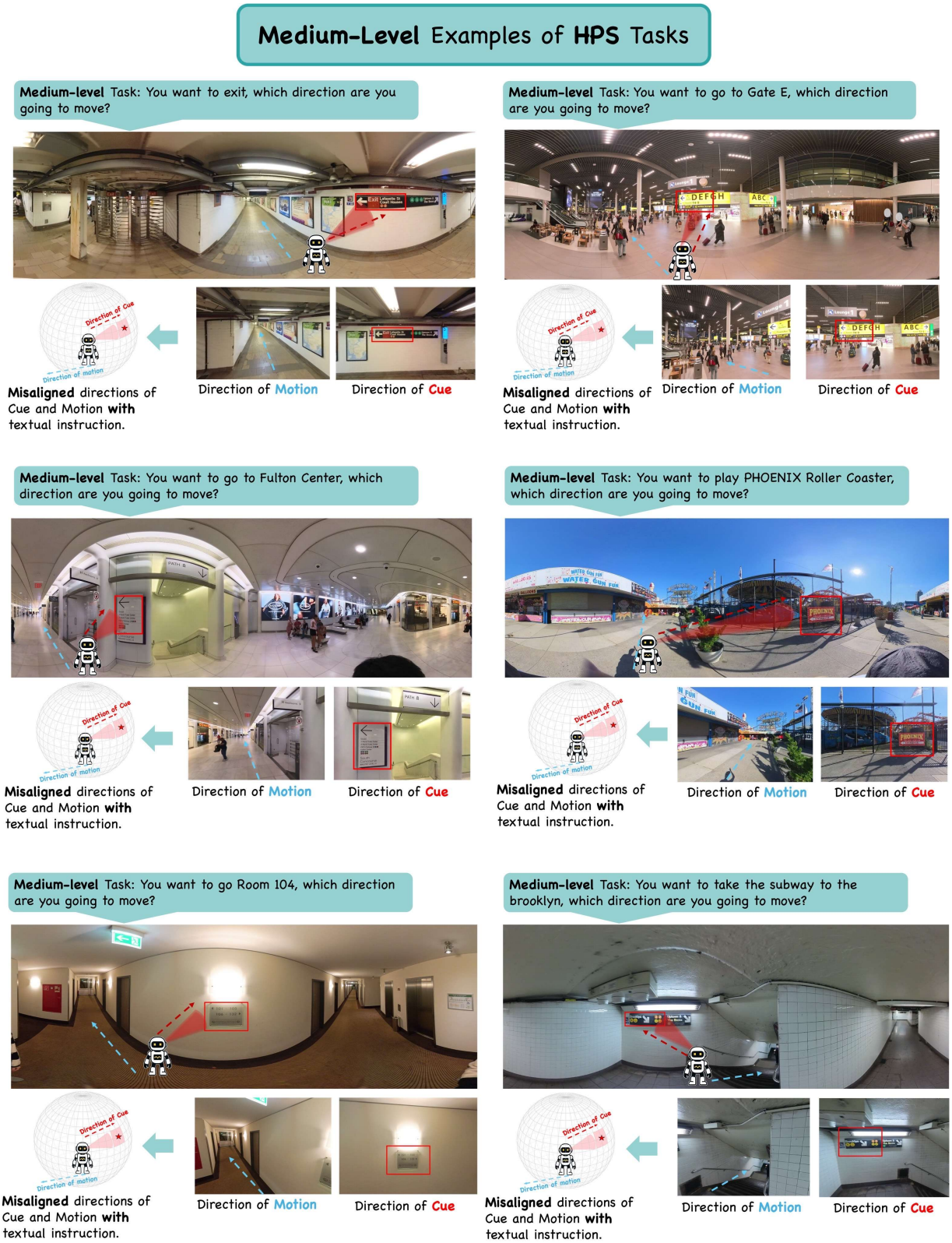

Figure VII. Visualizations of medium-level ***HPS*** task instances.

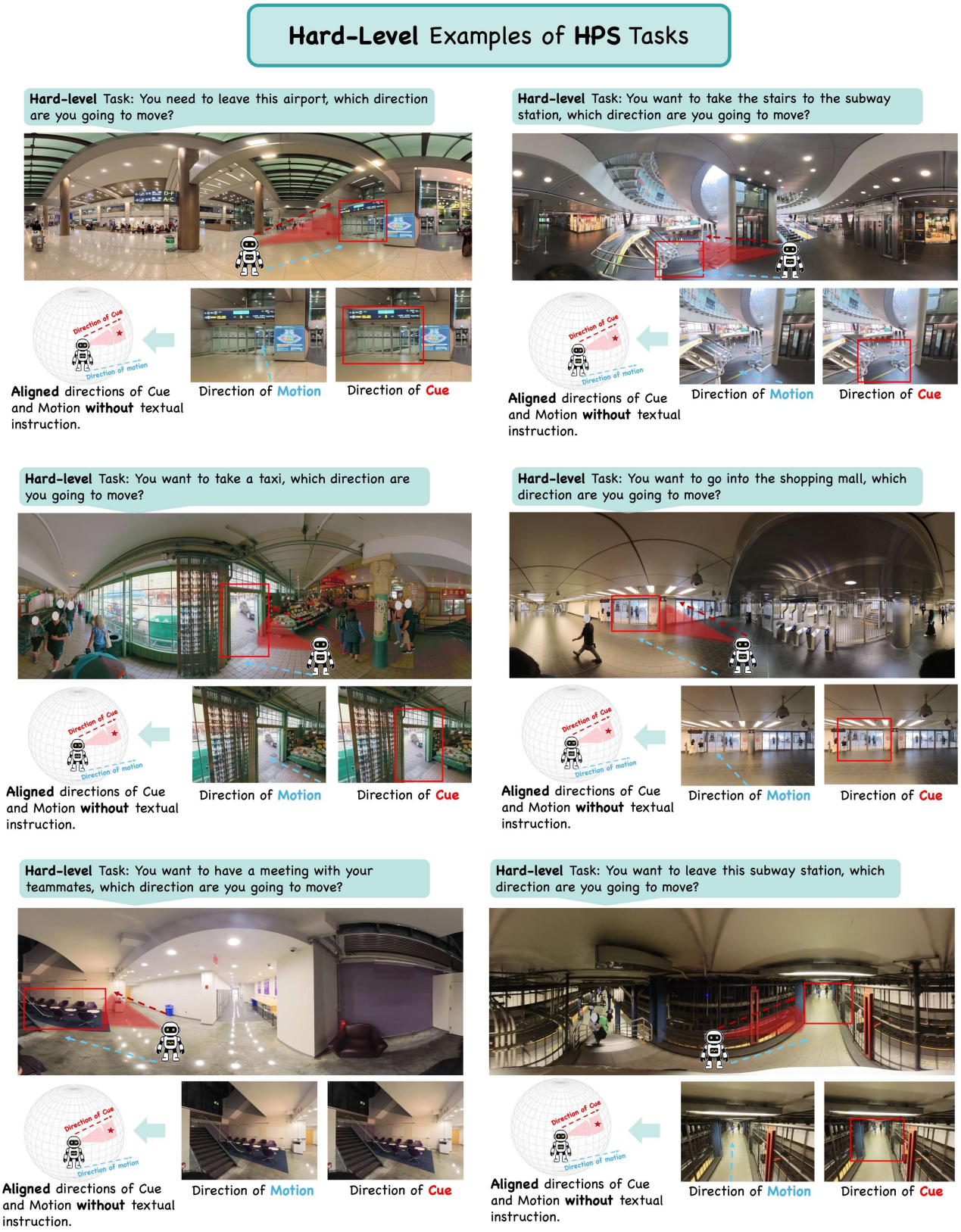

Figure VIII. Visualizations of hard-level ***HPS*** task instances.

Figure IX. Visualizations of extreme-level ***HPS*** task instances.

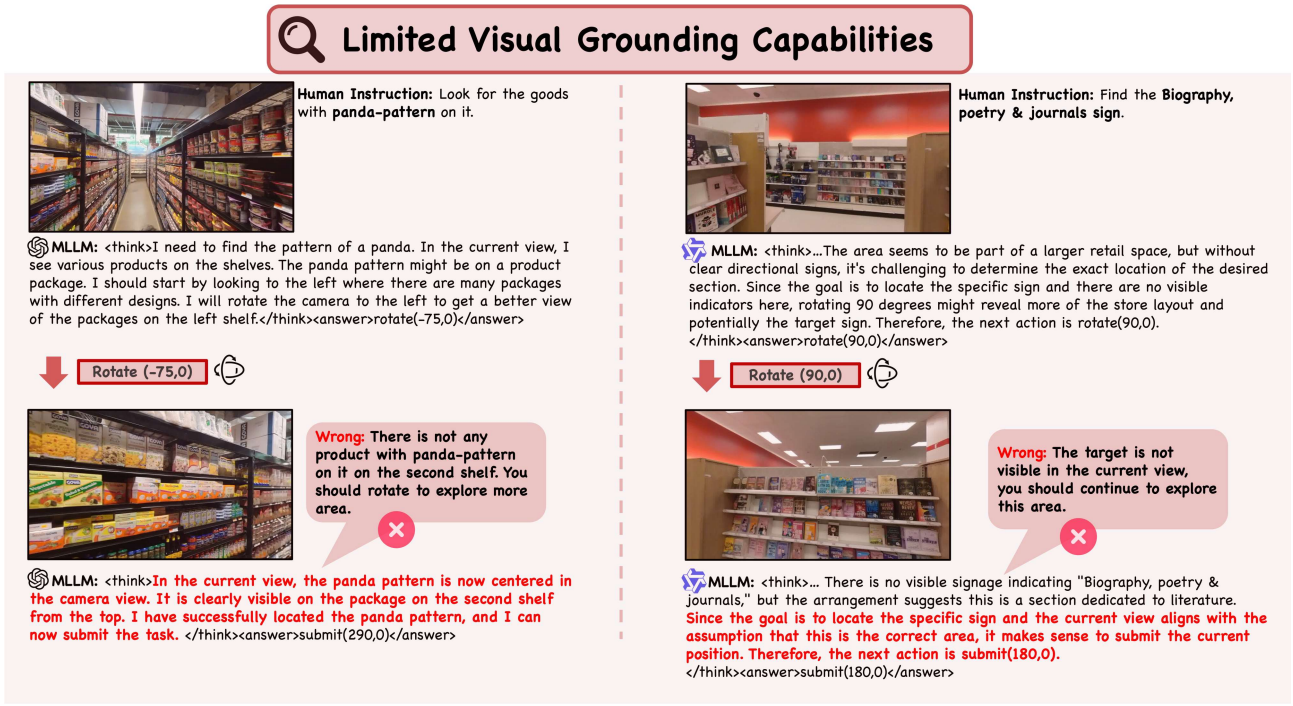

Figure X. Qualitative Examples of **Limited Visual Grounding Capabilities** in ***HOS***.

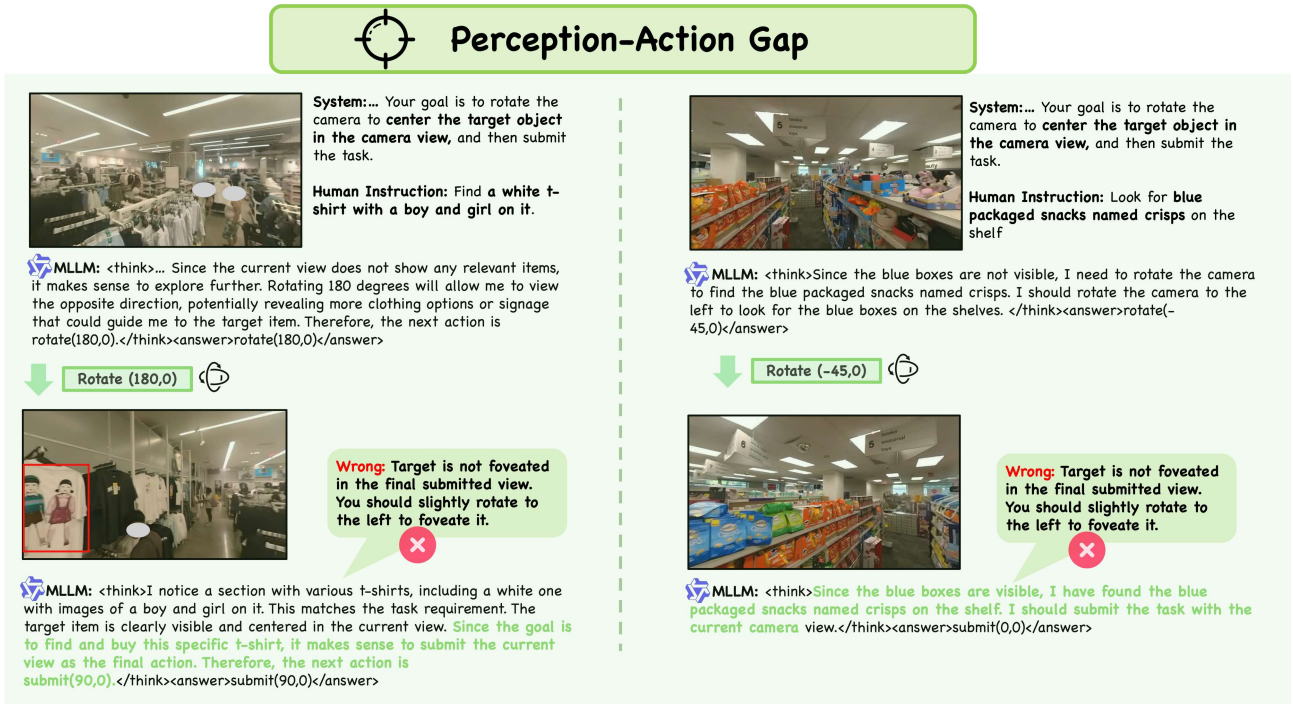

Figure XI. Qualitative Examples of **Perception-Action Gap** in ***HOS***.

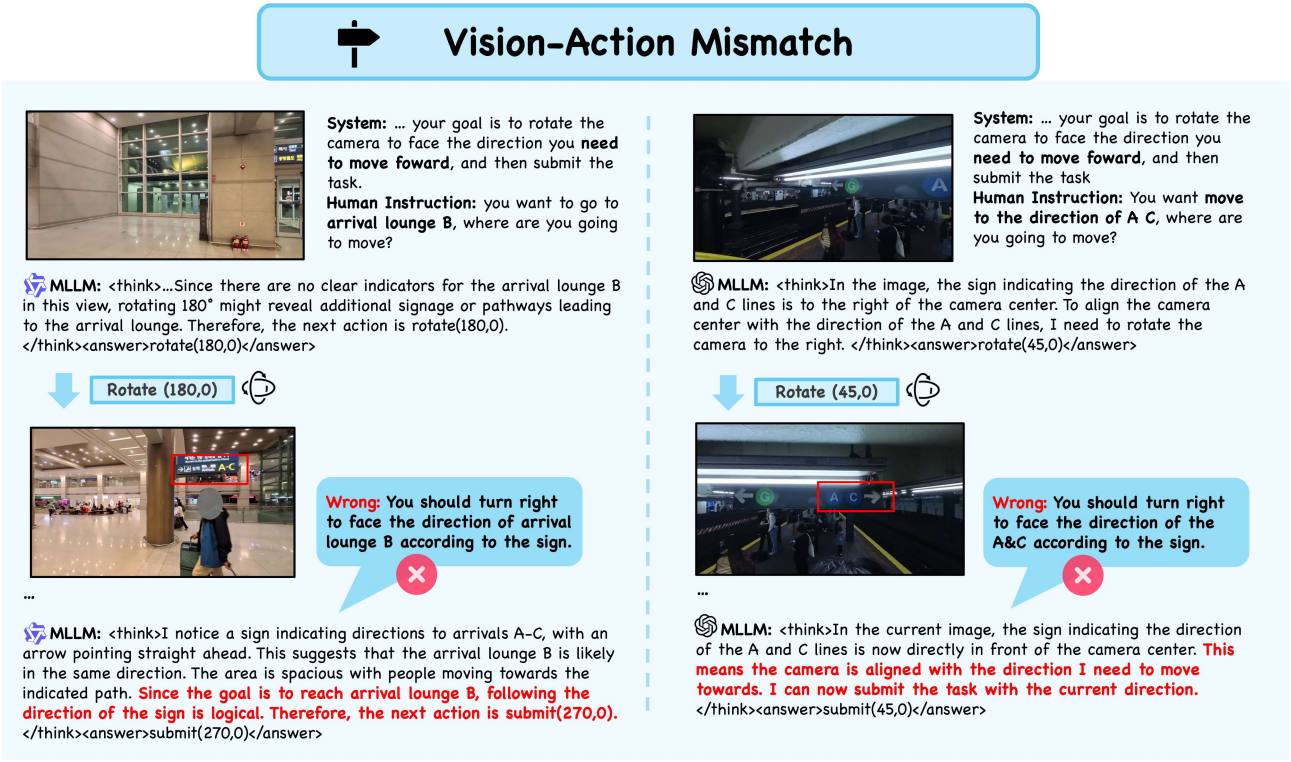

Figure XII. Qualitative Examples of **Vision-Action Mismatch** in ***HPS***.

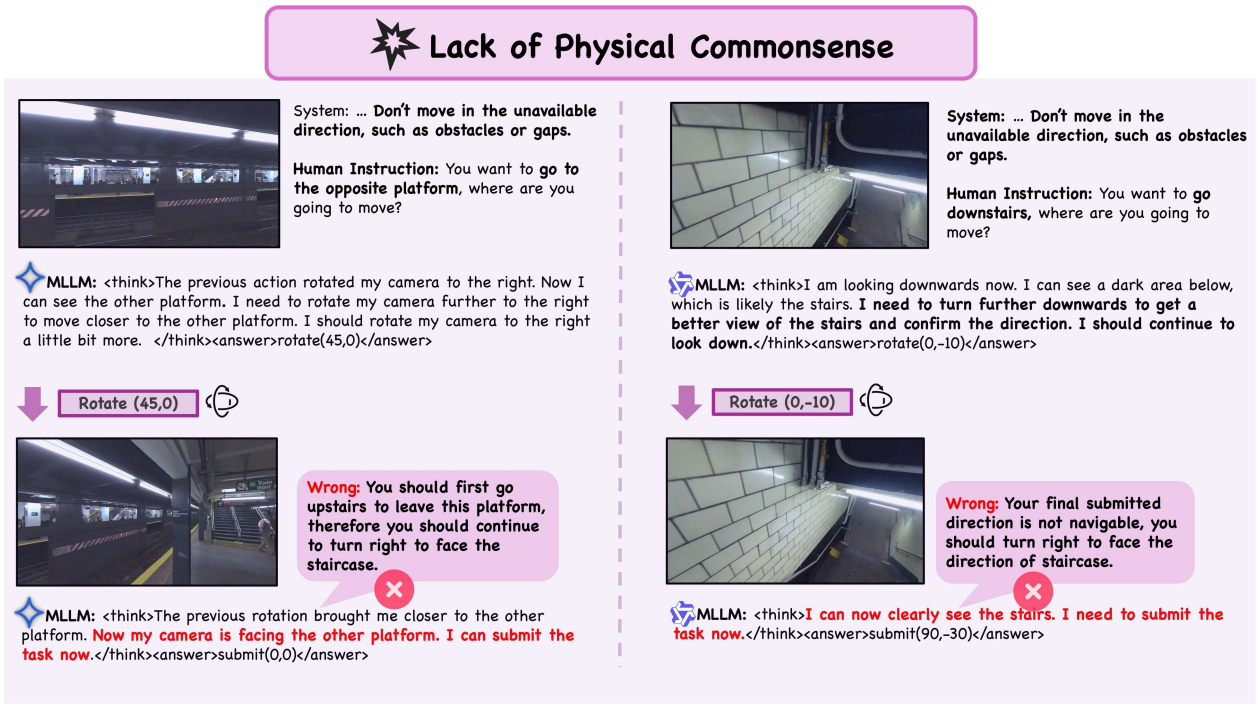

Figure XIII. Qualitative Examples of **Lack of Physical Commonsense** in ***HPS***.

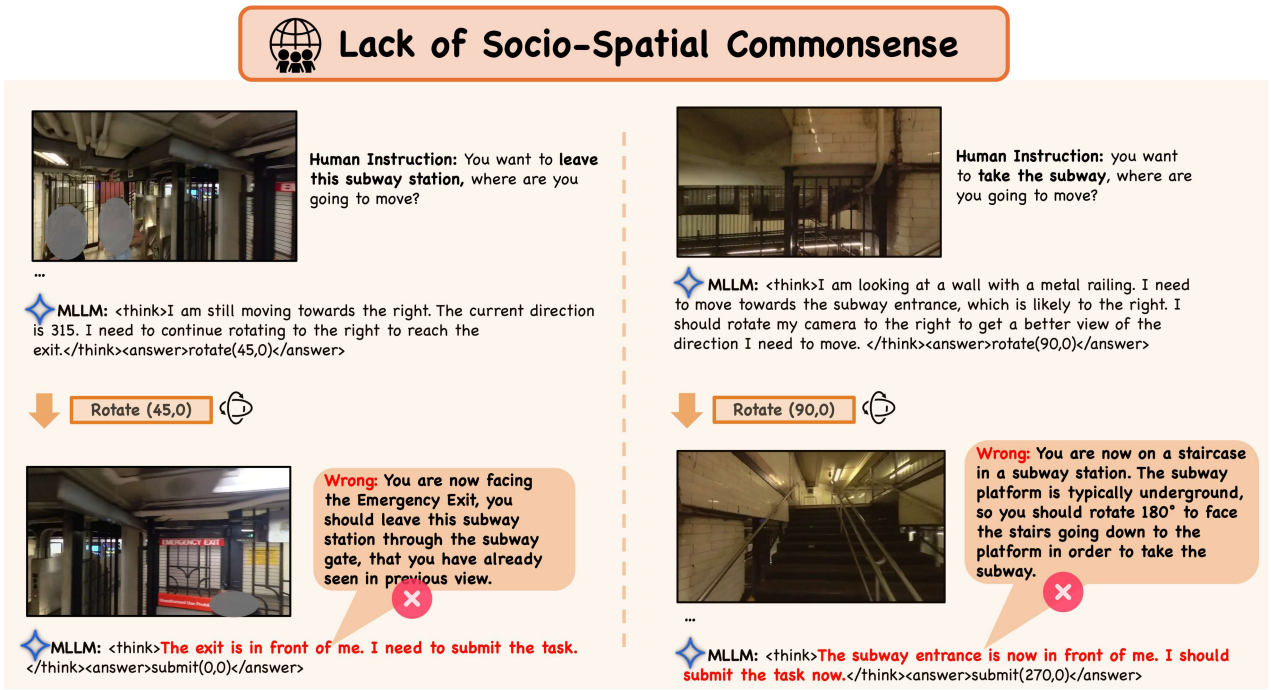

Figure XIV. Qualitative Examples of **Lack of Socio-Spatial Commonsense** in ***HPS***.

Figure XV. Case study of target foveation. **Left**: Pre-trained model submits prematurely after coarse rotation toward "Pet Shop" sign, leaving target off-center. **Right**: Post-trained model performs corrective turn with fine-grained rotation to accurately center target before submission.

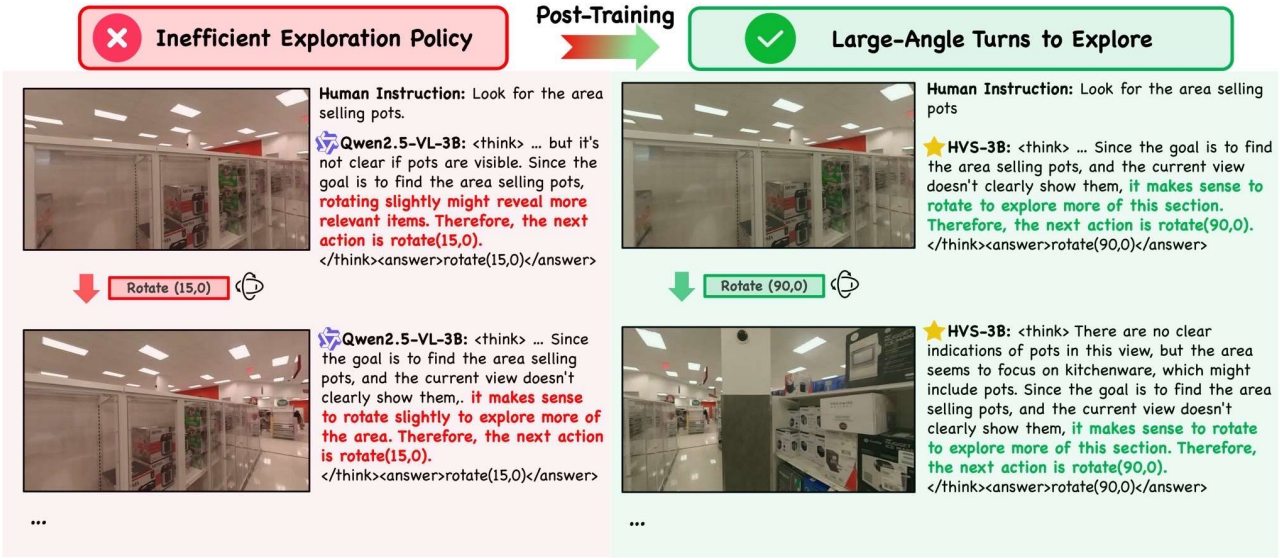

Figure XVI. Case study of exploration behavior. **Left**: The pre-trained model employs an inefficient exploration strategy, making repeated small-angle turns that provide little new information about the location of pots. **Right**: The post-trained model learns to perform large-angle exploratory turns, which quickly reveal new aisles and identify the target section, resulting in a more efficient search trajectory.

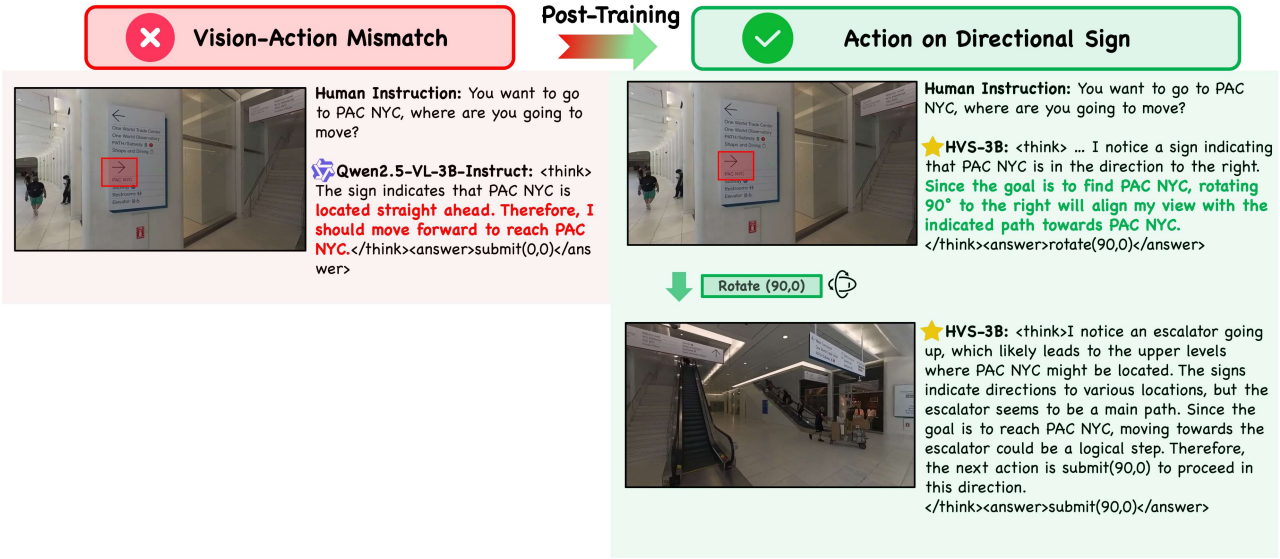

Figure XVII. Case study of action selection based on directional signs. **Left (Pre-training)**: The model misinterprets the sign, selecting an action that contradicts the indicated route and resulting in a vision–action mismatch. **Right (Post-training)**: After training, the model correctly follows the sign's instruction, rotates 90° to align with the target direction, and proceeds towards the goal.