

Scaling Spatial Intelligence with Multimodal Foundation Models

Supplementary Material

A. Details of Fig.1

A.1. Four Subfigures at the Corners

The four subfigures at the corners elaborate **SenseNova-SI**'s performance on four core spatial capabilities (*i.e.*, Perspective-taking, Spatial Relations, Metric Measurement, and Comprehensive Reasoning). Through data scaling, SenseNova-SI surpassing open-source models and even outperforms GPT-5 in specific spatial abilities, such as Perspective-taking. The lines denote the average performance across benchmark subtasks within each capability, while the shaded regions (confidence bands) represent ± 0.5 standard deviation. The detailed benchmark sub-tasks associated with each spatial capability are listed below.

- **Perspective-taking.** VSI-Bench: Obj. Rel. Direction; MMSI-Bench: Positional Relationship subtasks (Cam-Cam, Obj-Obj, Reg-Reg, Cam-Obj, Obj-Reg, Cam-Reg), Motion subtasks (Motion-Cam, Motion-Obj); SITE: multi-view & cross-image reasoning.
- **Spatial Relations.** VSI-Bench: Obj. Rel. Distance; SITE: 3d information understanding, spatial relationship reasoning.
- **Metric Measurement.** VSI-Bench: Obj. Size, Room Size, Obj. Abs. Distance; MMSI-Bench: Attribute Meas..
- **Comprehensive Reasoning.** VSI-Bench: Obj. Cnt., Obj. Appear Order, Route Plan; MMSI-Bench: MSR.

A.2. Normalization for Radar Chart Visualization

For the radar charts in Fig. 1, we apply normalization to enable a fair and intuitive comparison of relative performance. Specifically, for each metric, we first scale all values by dividing them by the maximum value observed across models. We normalize all metrics so that the best score among the models is mapped to 1.0 and the worst score is mapped to 0.2. The radar chart axes have a range of 0.0 to 1.0.

B. Additional Details in Data Curation

Our unified data pipeline collects data from diverse sources and efficiently converts them into reliable, high-quality QA and Chain-of-Thought (CoT) labels.

B.1. Data Processing

B.1.1. Unified Annotation.

We standardize heterogeneous raw data from source datasets into a unified set of spatial and multi-view annotations. Specifically, we convert existing formats and augment the data with additional labels to obtain: 3D camera

poses, 3D object poses including bounding boxes and orientations, 2D point and object visibility, and rich semantic labels of object and human-object interaction descriptions.

B.1.2. Dataset-specific Processing.

- **ScanNet [14], ScanNet++ [69].** These datasets provide 3D camera poses, 3D object bounding boxes, and 3D point clouds with object IDs. For each camera view, we project the 3D point cloud onto the image plane to establish correspondences between 2D pixels and 3D points, and to derive per-object projected and visible 2D bounding boxes.
- **SUN RGB-D [47], CA-1M [29].** These datasets provide 3D camera poses, 3D object poses, and 2D object bounding boxes. Building on this, we refine and standardize the 3D object orientations, discard object categories whose orientations are inconsistent across scenes, and, using the accurate orientation labels, further annotate possible human-object interactions with hypothetical 3D poses and rich textual descriptions.
- **MessyTable [6].** This dataset provides 3D camera poses, 2D object bounding boxes, and cross-view instance association labels (the same object instance is assigned the same instance ID in different viewpoints). We further employ a vision-language model (VLM) to annotate fine-grained textual descriptions of object appearance details.

B.2. Object Selection

We adopt a unified object selection pipeline to retain only recognizable objects with sufficient informative details captured within each frame. The resulting per-frame object sets provide a clean and reliable basis for cross-frame association analysis and QA construction.

B.2.1. Semantic Filtering.

We first filter out object categories with weak geometric structure and ambiguous 3D position, such as *floor*, *ceiling*, and *wall*, as well as objects with unclear or undefined semantic labels.

B.2.2. Visibility-based Filtering.

- **Minimum Visible Size.** We keep only the objects whose visible 2D bounding box (*i.e.*, the portion not occluded by other objects and lying within the camera view) occupies at least a certain fraction of the image area.
- **Visibility Ratio Threshold.** We further discard objects whose visible 2D bounding box area falls below a fixed ratio of their total projected 2D bounding box area.



[Corr Object] Which bbox in image 2 corresponds to object at bbox Ref in image 1? - B

Figure 4. Hard cases in MessyTable [6], where multiple instances of the same object class are present in the same scene.

B.3. Image Selection

To derive multi-view image sets that are well-posed, visually associated, and sufficiently challenging, we adopt a three-stage image selection pipeline.

B.3.1. Basic Pose Filtering.

We first discard views with extreme camera poses. In particular, we remove images whose camera pitch (severely top-down or bottom-up) or yaw deviates excessively from the typical viewing direction. This step eliminates degenerate or highly uninformative viewpoints.

B.3.2. Cross-view Association Control.

- **Connectivity.** We select images into sets in a manner such that, for any pair of images in the same set, there exists at least one connecting path, along which every pair of adjacent images satisfies a minimum association score. The calculation of the association score depends on the available annotation. For datasets with 2D point visibility, we compute the score based on the number of overlapping visible points. Otherwise, we compute the number of shared visually valid objects.
- **Difficulty.** To avoid trivial cross-view associations, we enforce that the association score between any pair of images does not exceed a specified maximum. We further exploit dataset-specific properties to design richer forms of difficulty control. For example, as shown in Fig. 4, we emphasize hard cases in MessyTable where multiple visually similar or identical objects exist in the same scene. In such cases, establishing cross-view associations requires fully understanding of the 3D spatial layout, while appearance-based shortcuts are impossible.

B.3.3. Full-scene Coverage Selection.

For scan-based datasets with point-level annotations, we further extend our selection to image sets with broader coverage of the scenes. Leveraging the temporal continuity of the scan videos, we design a time-efficient greedy algorithm that iteratively adds views to maximize point coverage, while maintaining the cross-view connectivity and difficulty constraints above. The resulting procedure is summarized in Algorithm 1.

Algorithm 1 Frame Selection with Overlap Control

Require: Video frames $\{f_1, \dots, f_T\}$, visible points V_{f_k} , minimum frames $N_{\min} = 16$

Ensure: Selected frame set S

- 1: $C \leftarrow \emptyset$ {Initialize covered point set}
 - 2: $S \leftarrow \{f_1\}$ {Start with first frame}
 - 3: $C \leftarrow V_{f_1}$
 - 4: **for** $k = 2$ **to** T **do**
 - 5: $\rho_k \leftarrow |V_{f_k} \cap C|/|V_{f_k}|$ {Compute overlap with coverage}
 - 6: **if** $0.03 \leq \rho_k \leq 0.20$ **then**
 - 7: $S \leftarrow S \cup \{f_k\}$
 - 8: $C \leftarrow C \cup V_{f_k}$ {Update covered points}
 - 9: **end if**
 - 10: **end for**
 - 11: **if** $|S| < N_{\min}$ **then**
 - 12: Insert additional frames uniformly in temporal gaps until $|S| = N_{\min}$
 - 13: **end if**
 - 14: **return** S
-

B.4. QA Selection

We apply quality control and quantity balancing to ensure reliable QA generation.

B.4.1. Ambiguity Reduction.

- To avoid ambiguous references when questions involve object names, we require that only a single instance of queried semantic object category is present within the image set.
- We discard cases in which the angular range of a referenced direction cannot be clearly mapped to a unique spatial sector (*e.g.*, front/left/right). Such ambiguous geometric configurations are removed to avoid confusion in answer interpretation.

B.4.2. Balanced Sampling.

We encourage both textual and visual diversity while maintaining balanced sampling.

- For questions with the same underlying intent, we randomly vary the textual descriptions (*e.g.*, paraphrased phrasings and directional expressions), while capping the total number of samples to avoid redundancy.
- Within each image set, we select diverse combinations of objects to construct QAs, while limiting the number of QAs per set to maintain a balanced distribution across different scenes.

B.5. Chain-of-Thought (CoT) Strategies

We explore three Chain-of-Thought (CoT) strategies for multi-frame reasoning.

Dataset	VSI-Bench		MMSI-Bench		MindCube		ViewSpatial		SITE		Avg.
	MRA, Acc	*	Acc	*	Acc	*	Acc	*	CAA	*	-
Random Choice	34.0	-	25.0	-	33.0	-	26.3	-	0.0	-	23.7
InternVL3-8B [74]	42.1	-	28.0	-	41.5	-	38.6	-	41.1	-	38.3
VLM-3R-DATA [17]	53.9	2	28.5	2	34.8	3	54.2	4	36.7	3	41.6
VSR [34]	41.1	3	28.3	3	37.9	2	55.9	2	40.9	1	40.8
Rel3D [22]	39.3	4	26.9	7	39.8	1	57.7	1	39.5	2	40.6
VSI590K [68]	64.0	1	29.0	1	26.7	7	48.1	5	34.7	5	40.5
SPEC [44]	38.7	5	27.6	5	33.6	4	54.7	3	34.8	4	37.9
SAT [45]	30.5	6	26.8	9	29.1	6	42.8	6	21.3	8	30.1
GQA [25]	26.6	9	27.4	6	24.7	8	42.2	7	23.4	7	28.9
MultiSPA [63]	21.8	10	27.7	4	22.8	9	34.4	9	32.2	6	27.8
CLEVR [27]	29.7	8	25.6	10	30.0	5	33.1	10	18.6	10	27.4
VQA [1]	30.0	7	26.9	7	15.3	10	39.4	8	20.2	9	26.4

Table 5. Evaluation on key spatial intelligence benchmarks using InternVL3-8B trained on each single dataset. *: ranking on each benchmark. **Dark purple** highlights the best result and **light purple** indicates the second-best result within models trained on different single datasets, respectively.

VLM-generated CoT. We provide QA pairs and step-wise instructions to GPT-5 for CoT annotations.

MindCube Aug-CGMap CoT. MindCube [70] uses discrete 2D cognitive map (CogMap) to describe top-down view of the scenes, projecting objects and cameras onto a 2D grid. The designed CoT contains two steps:

- **CGMap Inference.** Directly generate a JSON-formatted CogMap with a discretized grid (*e.g.*, 10×10), encoding approximate positions and four-direction orientations of objects and camera views.
- **Free-Form Reasoning.** Perform free-form reasoning on camera changes between consecutive frames, relate observations across views, and finally derive the answer from the aggregated observations.

Our Procedural CGMap CoT. We also adopt a top-down CogMap representation, but use continuous (non-gridded) coordinates and construct the CogMap in a procedural, step-by-step manner interleaved with textual reasoning. This CoT design exhibits improved geometric accuracy and more coherent reasoning, particularly in scenes with complex object layouts and diverse viewpoints. Experimental results can be found in Sec. 5.6. The detailed procedure is as follows:

- **Keyframe Localization.** Identify the keyframe set A in which the queried objects appear. These frames are emphasized during subsequent reasoning.
- **Incremental Relative Camera Estimation.** Traverse frames in temporal order. For each adjacent pair of frames, describe the shared objects and estimate coarse camera pose changes.
- **CogMap Construction.** Construct the CogMap along the keyframe set A , following an efficient path inferred from the previous step. We build a global CogMap and fix its origin and positive y -axis with the first keyframe

as reference. For each new frame, we perform metric 3D grounding of newly observed objects, estimate the camera rotation and translation relative to the reference frame, and then transform the placement of the new camera view and objects into the global CogMap.

- **Answer Derivation.** We may flexibly rotate the CogMap coordinate system according to any desired allocentric transformation, and reason about geometric relations (*e.g.*, distances, angles, relative ordering) to produce the final answer.

C. Full Results of Single Dataset Training

For each dataset, we train a model on its training set and evaluate its performance on five key spatial intelligence benchmarks: VSI, MMSI, MindCube, ViewSpatial, and SITE. As shown in Tab. 5, training on a single dataset often yields strong performance on a few benchmarks while sacrificing performance on others. For example, a model trained solely on VSI-590K achieves the best MRA accuracy on VSI (64.0), but its results on tasks like MMSI, MindCube, and SITE drop noticeably. This pattern highlights that no single dataset provides comprehensive spatial intelligence coverage, and therefore mixed-data training is crucial for building more balanced models. As different datasets tend to bias the model toward a particular subset of spatial reasoning skills, determining how to effectively balance datasets during training remains an open challenge. This table also provides the community with a useful reference for dataset selection, helping researchers understand which datasets contribute to which aspects of spatial reasoning and how to design more complementary training mixtures.

Train	MM	MR	SR	PT	CR
MM	+52.5%	-7.6%	+18.4%	+6.4%	+8.4%
SR	+10.6%	-61.5%	+18.7%	-19.2%	-6.0%
PT	+3.0%	+46.1%	-0.2%	+83.8%	+10.9%

Table 6. **Capability transfer matrix.** Each row shows the percentage change in model performance on other key spatial capabilities after training on data corresponding to a specific capability.

D. Capability Transfer Matrix

We study the capability transfer matrix by training SenseNova-SI-8B on MM, SR, and PT with sufficient training data (>1M), and observe that there exists cross-capability synergy in Tab. 6. For example, PT may act as a more comprehensive capability benefiting others.

E. Impact of Scaling on Benchmarks

In Fig. 1, we illustrate the effect of data scaling on model performance, grouped by core spatial capabilities. In contrast, Tab. 7 presents the scaling effects at the benchmark level. Across both views, we observe a consistent trend: model performance improves steadily as more data is introduced during training, validating that high-quality, diverse spatial data is effective in addressing key knowledge deficiencies in spatial intelligence.

# Data	VSI-Bench	MMSI-Bench	MindCube*	ViewSpatial	SITE
Metric	MRA, Acc	Acc	Acc	Acc	CAA
0M	42.1	28.0	41.5	52.0	42.1
1M	56.3	36.0	58.8	55.2	44.5
2M	60.3	39.6	76.2	56.7	47.1
3M	64.4	41.9	81.1	56.2	46.7
4M	62.7	41.9	83.9	55.6	45.2
5M	65.9	40.8	81.7	53.7	46.0
6M	66.3	41.8	85.0	55.2	47.0
7M	67.9	42.3	85.7	54.7	46.5
8M	68.7	43.3	85.6	54.6	47.7

Table 7. **Impact of scaling on key SI benchmarks.** MindCube* denotes MindCube-Tiny. 0M indicates base model (InternVL3-8B) whereas 8M indicates SenseNova-SI_{InternVL3-8B}.

F. Retention of General Capabilities

To evaluate whether SenseNova-SI retains its general understanding capabilities after continued training on spatial intelligence data (*i.e.*, SenseNova-SI-8M), we evaluate its performance on additional multimodal benchmarks: **MMBench-En** [35] and **MMStar** [10] for holistic multimodal understanding, **AI2D** [24] for scientific diagram reasoning, **OCRB** [36] and **DocVQA** [41] for text-rich image and document understanding, **MMVP** [50] and **V*** [61] for fine-grained visual perception and grounding,

MMMU [72] for multidisciplinary multimodal reasoning, and **Vid-MME** [19] for video understanding.

As shown in Tab. 8, SenseNova-SI_{Qwen3-VL-8B} and SenseNova-SI_{InternVL3-2B} exhibit minimal performance drops on MMBench-En relative to their respective base models, while SenseNova-SI_{Bagel-7B-MoT} and SenseNova-SI_{InternVL3-8B} even show slight improvements. Across the remaining three benchmarks, only marginal declines are observed.

Notably, compared with other open-source spatial-intelligence models, SenseNova-SI maintains competitive general visual understanding. Furthermore, prior studies (such as VST [66] and Cambrian-S [68]) suggest that incorporating additional general visual understanding data can further preserve or enhance this capability, a direction we plan to explore in future work.

G. Text-only Training

To further investigate the potential impact of language shortcuts, we conduct an experiment in which models are trained on text-only data, with all visual inputs removed (Tab. 9). The results show that text-only training yields substantially smaller gains, indicating that visual input is critical. Moreover, we observe that certain benchmarks are highly resistant to language shortcuts. For example, on MMSI-Bench, text-only training brings only minimal improvements.

H. Downstream Task

H.1. Embodied Agent for Manipulation.

To evaluate the practical utility of SenseNova-SI’s enhanced spatial intelligence, we conduct downstream robot manipulation experiments on EmbodiedBench [65], focusing specifically on its spatial subset. In this setting, SenseNova-SI is instantiated as an embodied agent that controls a virtual Franka Panda robot to execute user instructions containing rich spatial language such as “left”, “on top of”, “rear”, and “horizontal”. Importantly, *no finetuning* of SenseNova-SI is performed for this evaluation. Quantitative results for the spatial subset are shown in Tab. 10.

We report success rates under two prompting settings: the official prompt (OP) and a spatial-intelligence-oriented prompt (SIP). OP supplies bounding-box coordinates extracted from the input image, whereas SIP enriches OP with additional object-grounding cues to reduce ambiguity from object recognition and better isolate spatial-reasoning performance.

Across both OP and SIP, SenseNova-SI delivers substantial improvements, demonstrating that enhanced spatial intelligence directly benefits embodied manipulation. We observe that SenseNova-SI more reliably identifies key spatial

Models	MMB-EN [35]	MMStar [10]	AI2D [24]	OCRB [36]	DocVQA [41]	MMVP [50]	V* [61]	MMM U [72]	Vid-MME [19]
Open-source General Models									
Bagel-7B-MoT [15]	82.8	67.3	89.5	811	94.1	68.7	71.7	50.4	68.9
Qwen3-VL-8B [3]	84.6	68.5	81.3	906	95.8	58.7	82.7	60.9	78.1
InternVL3-2B [74]	79.7	56.9	77.8	853	87.3	56.7	68.6	43.2	69.9
InternVL3-8B [74]	81.7	68.2	85.2	880	92.1	59.3	57.1	55.6	76.1
Open-source SI Models									
VST-7B-SFT [66]	83.3 [†]	63.1 [†]	84.9 [†]	855 [†]	91.7	54.7	81.2	50.9	72.1
Cambrian-S-7B [68]	80.4 [†]	43.9	76.9 [†]	648 [†]	83.7	54.0	70.7	47.1	70.6
Ours									
SenseNova-SI _{Bagel-7B-MoT}	83.4	67.8	88.8	797	93.9	65.3	69.6	50.2	64.6
SenseNova-SI _{Qwen3-VL-8B}	83.5	65.7	84.2	863	95.4	57.3	81.1	57.6	76.4
SenseNova-SI _{InternVL3-2B}	78.9	57.0	76.8	817	85.2	47.3	81.2	44.4	66.6
SenseNova-SI _{InternVL3-8B}	84.9	65.4	79.0	815	84.9	56.0	71.2	49.4	72.2

Table 8. Evaluation on general understanding benchmarks. [†] denotes benchmark results directly cited from their papers. **Dark purple** highlights the best result and light purple indicates the second-best result within Open-source and SenseNova-SI models, respectively. MMB-EN: MMBench-En. OCRB: OCRBench.

Table 9. Training with text-only SSI-800K (10% of SSI-8M).

Model	VSI-Debiased	VSI-Bench	MMSI-Bench	MindCube-Tiny	ViewSpatial	SITE
InternVL3-8B	38.5	42.1	28.0	41.5	38.6	41.1
+SSI-800K	53.8	60.9	36.4	56.9	52.5	47.7
+SSI-800K (Text-Only)	42.4	50.2	28.2	44.2	44.1	42.5

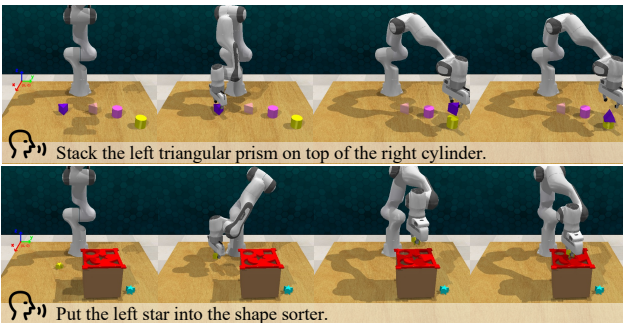


Figure 5. Visualization of the manipulation task rollout in EmbodiedBench [65], performed by the embodied agent powered by SenseNova-SI.

GPT-4o		InternVL3-8B		SenseNova-SI _{InternVL3-8B}	
OP	SIP	OP	SIP	OP	SIP
37.5	45.8	10.4	20.8	16.6 (+59.6%)	33.3 (+60.0%)

Table 10. Success rate on **Spatial** subset of EmbodiedBench [65]. **OP**: Official Prompt; **SIP**: Spatial-Intelligence-oriented Prompt.

cues, enabling more accurate reasoning and more consistent action planning.

Representative rollouts are visualized in Fig. 5. These examples illustrate that SenseNova-SI effectively integrates spatial information from both language instructions and visual observations, plans coherent motion trajectories, and generates action sequences that allow the robot to successfully complete the tasks.

H.2. Task Prompt for the Embodied Agent.

Following EmbodiedBench [65], we implement SenseNova-SI as an embodied agent in the downstream manipulation task to demonstrate its application. In this setting, the SenseNova-SI model controls a simulated Franka Panda robot with a parallel gripper. Conditioned on a language instruction and the visual state of the scene, the agent receives a symbolic description of the environment, where each object is represented by a discrete 3D position in a table-top coordinate frame. The model is required to output a sequence of low-level gripper actions in a structured action space, where each action specifies the target end-effector position, orientation, and a binary gripper state, all expressed in the same discretized coordinate system. This formulation enables direct execution of the predicted actions in the simulator without additional post-processing. This demonstrates the model’s ability to ground language and perception into coherent, executable manipulation trajectories that require spatial reasoning.

The official task prompt (OP) includes the role description, the definitions of the input space and output action space, the color space, example task conversations, and the instructions for the output JSON format. This design enables the SenseNova-SI model to perform reasoning and action planning while generating executable actions in the required format. For the spatial-intelligence-oriented prompt (SIP), instead of providing object bounding box coordinates with generic indices such as “object 1” and “object 2” as in OP, we provide the specific name of the objects, such as “first cylinder” or “second container”. This removes interference from object recognition and allows the model to focus on spatial reasoning. The official task prompt (OP) is shown below.

H.3. Case Study.

Fig. 6 illustrates how the SenseNova-SI_{InternVL3-8B} model behaves as an embodied agent in the manipulation task under the official task prompt. For each task instance, we show the task instruction, the scene observation provided to the model, the model output, and the resulting execution rollout in the simulator. Cases Fig. 6(a) and Fig. 6(b) demonstrate successful executions, where the model produces correct object recognition and accurate action sequences that lead to successful task completion. In contrast, cases Fig. 6(c) and Fig. 6(d) illustrate typical failure modes, including incorrect object recognition in the visual state description that leads to erroneous reasoning and execution planning, as well as limited manipulation precision that causes task failure even when the executable action plan is correct. Incorrect information in the output is highlighted in red. Please refer to the video demonstration in the Supplementary Material for the full rollout, more task instances and performance comparison with the base InternVL3-8B model.

The official task prompt (OP) for the embodied agent:

You are a Franka Panda robot with a parallel gripper. You can perform various tasks and output a sequence of gripper actions to accomplish a given task with images of your status. The input space, output action space and color space are defined as follows.

Input space

- Each input object is represented as a 3D discrete position in the following format: [X, Y, Z].
- There is a red XYZ coordinate frame located in the top-left corner of the table. The X-Y plane is the table surface.
- The allowed range of X, Y, Z is [0, 100].
- Objects are ordered by Y in ascending order.

Output action space

- Each output action is represented as a 7D discrete gripper action in the following format: [X, Y, Z, Roll, Pitch, Yaw, Gripper state].
- X, Y, Z are the 3D discrete position of the gripper in the environment. They follow the same coordinate system as the input object coordinates.
- The allowed range of X, Y, Z is [0, 100].
- Roll, Pitch, Yaw are the 3D discrete orientation of the gripper in the environment, represented as discrete Euler angles.
- The allowed range of Roll, Pitch, Yaw is [0, 120] and each unit represents 3 degrees.
- Gripper state is 0 for close and 1 for open.

Color space

- Each object can be described using one of the colors below: ["red", "maroon", "lime", "green", "blue", "navy", "yellow", "cyan", "magenta", "silver", "gray", "olive", "purple", "teal", "azure", "violet", "rose", "black", "white"].

Below are some examples to guide you in completing the task.

Example 0: Human Instruction: Pick up the star and place it into the yellow container.

Input: {"object 1": [45, 13, 18], "object 2": [72, 20, 18], "object 3": [50, 42, 17], "object 4": [36, 42, 18], "object 5": [70, 43, 17], "object 6": [50, 77, 17]}

Output: { "visual_state_description": "From left to right, I can see a purple cylinder at [45, 13, 18], a blue moon at [72, 20, 18], a green star at [50, 42, 17], a purple cube at [36, 42, 18], a yellow container at [70, 43, 17] and a purple container at [50, 77, 17].", "reasoning_and_reflection": "The task is to pick up the star located at [50, 42, 17] and place it into the yellow container located at [70, 43, 17]. The plan involves moving the gripper to the star's position, closing the gripper to pick it up, moving to the yellow container's position, and opening the gripper to release the object into the container.", "executable_plan": "[[50, 42, 17, 6, 61, 36, 1], [50, 42, 17, 6, 61, 36, 0], [47, 41, 26, 6, 61, 36, 0], [69, 39, 24, 88, 32, 77, 0], [69, 39, 24, 87, 32, 77, 1]]" }

Example 1: Human Instruction: Pick up the star and place it into the rose container.

Input: {"object 1": [78, 20, 17], "object 2": [28, 30, 17], "object 3": [39, 47, 17], "object 4": [50, 72, 17], "object 5": [39, 75, 18]}

Output: { "visual_state_description": "From left to right, I can see a rose container at [78, 20, 17], a black star at [28, 30, 17], a green container at [39, 47, 17], a green triangular prism at [50, 72, 17] and a grey cylinder at [39, 75, 18].", "reasoning_and_reflection": "The task is to pick up the star located at [28, 30, 17] and place it into the rose container located at [78, 20, 17]. The plan involves moving the gripper to the star's position, closing the gripper to pick it up, moving to the rose container's position, and opening the gripper to release the object into the container.", "executable_plan": "[[28, 32, 26, 0, 60, 94, 1], [27, 32, 18, 0, 60, 94, 0], [27, 32, 28, 0, 60, 94, 0], [76, 19, 28, 6, 66, 109, 1]]" }

Now you are supposed to follow the above examples to generate a sequence of discrete gripper actions that completes the human instruction below.

Human Instruction: <Task instruction>

Input: <Dict of object positions>

The output JSON format should be {"visual_state_description": str, "reasoning_and_reflection": str, "language_plan": str, "executable_plan": str}.

The fields in the above JSON follow the purposes below:

1. `visual_state_description`: Describe the color and shape of each object in the detection box in the numerical order in the image. Then provide the 3D coordinates of the objects chosen from the input.
 2. `reasoning_and_reflection`: Reason about the overall plan that needs to be taken on the target objects, and reflect on the previous actions taken if available.
 3. `language_plan`: A list of natural language actions to achieve the user instruction. Each language action is started by the step number and the language action name.
 4. `executable_plan`: A list of discrete actions needed to achieve the user instruction, with each discrete action being a 7-dimensional discrete action.
 5. Keep your plan efficient and concise.
- When generating content for JSON strings, avoid using any contractions or abbreviated forms (like 's, 're, 've, 'll, 'd, n't) that use apostrophes. Instead, write out full forms (is, are, have, will, would, not) to prevent parsing errors in JSON.
- Please do not output anything other than the above-mentioned JSON.

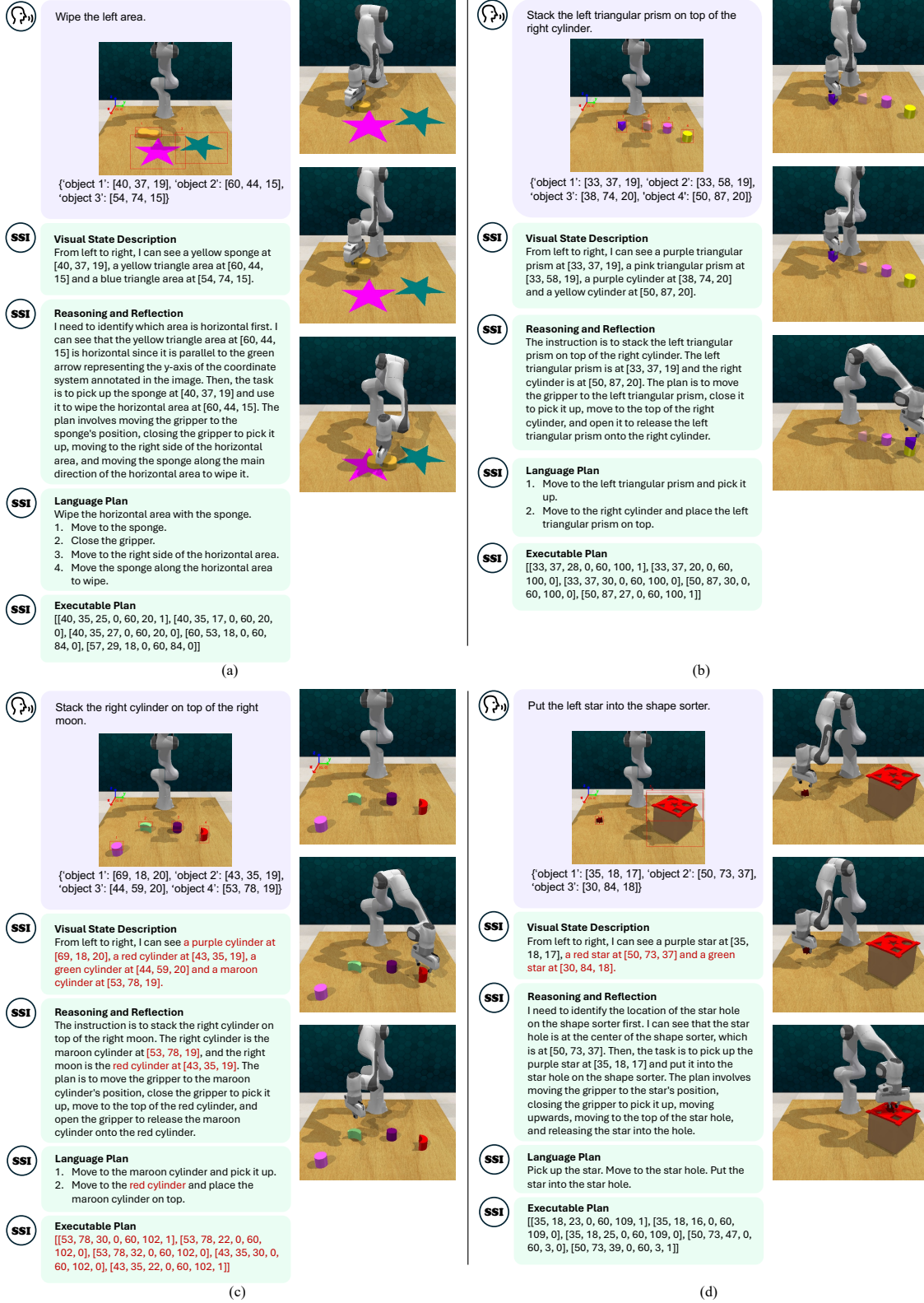


Figure 6. Demonstration of manipulation with SenseNova-SI_{InternVL3-8B} as an embodied agent. We present the task instruction, scene observation input, model output, and task execution rollout. Cases (a) and (b) are successful executions, while cases (c) and (d) are failure cases. Incorrect information in the output is highlighted in red. Please see the video in the Supplementary Material for the full rollout.

I. Detail Results on Key Benchmarks

In this section, we provide detailed per-benchmark results that complement the aggregated scores reported in the main text. The following tables report per-benchmark results for VSI-Bench (Tab. 11), MMSI-Bench (Tab. 12), MindCubeBench-Tiny (Tab. 13), ViewSpatial (Tab. 14), SITE (Tab. 15), BLINK (Tab. 16), 3DSR (Tab. 17), and EmbSpatial (Tab. 18), respectively. For each benchmark, we break down performance over all relevant subsets and question types, enabling a more fine-grained analysis of model strengths and failure modes than is possible from the single aggregated metrics in the main tables. On most subsets, our model attains the best or near-best accuracy among open-source models, and on several challenging subsets (*e.g.*, Rel.Dir in VSI), its performance is comparable to, or even surpasses, that of proprietary models.

Models	Avg.	Numerical Answer				Multiple-Choice Answer			
		Obj. Count	Abs. Dist	Obj. Size	Room Size	Rel. Dis	Rel. Dir	Route Plan	Appr. Order
		CR	MM	MM	MM	SR,MM	PT	CR	CR
Human	79.2	94.3	47.0	60.4	45.9	94.7	95.8	95.8	100.0
Random Choice(Frequency)	34.0	62.1	32.0	29.9	33.1	25.1	47.9	28.4	25.2
Proprietary Models									
Seed-1.6-2025-06-15 [48]	49.9	43.5	34.4	66.1	52.8	55.1	35.7	44.3	68.0
Gemini-2.5-pro-2025-06 [49]	53.6	46.0	37.4	68.7	54.4	62.0	43.9	47.4	68.8
Grok-4-2025-07-09 [62]	47.9	37.2	33.0	60.8	45.4	53.1	39.7	47.4	66.8
GPT-5-2025-08-07 [42]	55.0	53.3	34.5	73.3	47.5	63.7	48.7	50.3	68.9
Open-source General Models									
Bagel-7B-MoT [15]	31.4	30.1	29.2	35.5	25.8	34.9	41.4	30.4	24.1
Qwen2.5-VL-3B-Instruct [4]	27.0	19.2	21.2	24.3	27.3	33.8	42.1	27.3	20.9
Qwen2.5-VL-7B-Instruct [4]	32.3	32.9	18.2	43.9	31.7	38.0	37.4	28.4	28.0
Qwen3-VL-2B-Instruct [3]	50.4	62.2	40.3	71.5	49.8	52.3	42.0	30.4	54.5
Qwen3-VL-8B-Instruct [3]	57.9	67.6	47.0	76.3	61.9	58.0	51.0	35.1	66.3
InternVL3-2B [74]	33.0	64.8	30.9	32.4	23.0	32.3	34.9	33.0	12.6
InternVL3-8B [74]	42.1	66.1	34.9	43.6	47.5	48.0	39.3	26.3	31.4
Open-source Spatial Intelligence Models									
MindCube-3B-RawQA-SFT [70]	17.2	12.9	22.8	4.3	23.5	20.3	15.7	16.0	22.5
SpatialLadder-3B [32]	44.9	62.2	35.4	62.0	41.4	45.6	46.5	27.3	38.5
Spatial-MLLM-4B [59]	46.3	66.7	38.1	63.7	35.5	40.4	48.2	33.0	44.3
SpaceR-7B [43]	41.6	30.0	25.2	47.0	29.6	40.3	46.5	32.5	39.3
ViLaSR-7B [60]	44.6	58.1	33.9	61.4	28.9	45.1	46.5	29.9	53.2
VST-3B-SFT [66]	51.4	60.7	37.5	72.7	45.9	51.3	45.9	40.2	56.8
VST-7B-SFT [66]	55.5	68.9	37.3	74.5	62.2	55.2	48.7	41.8	55.5
Cambrian-S-3B [68]	56.1	69.4	38.7	66.3	52.7	61.8	58.3	28.4	73.1
Cambrian-S-7B [68]	62.9	68.2	45.8	72.5	67.6	66.8	69.6	39.2	73.8
Ours									
SenseNova-SI Bagel-7B-MoT	41.5	42.2	33.5	57.2	22.7	44.9	46.4	33.5	51.9
SenseNova-SI Qwen3-VL-8B	64.8	71.4	48.7	76.0	69.6	65.5	72.2	43.8	71.5
SenseNova-SI InternVL3-2B	63.7	70.1	47.2	74.5	67.1	61.0	73.1	41.2	75.6
SenseNova-SI InternVL3-8B	68.8	72.0	53.5	76.8	72.8	69.6	80.8	48.5	76.4

Table 11. **Evaluation on VSI-Bench [64]**. Numerical Answer uses **MRA** score, MCA uses **Acc** score, **Avg.** is the simple average across these metrics, following the original paper. All results are evaluated on EASI [7].

Models	Avg.	Positional Relationship						Attribute		Motion		MSR
		C-C	O-O	R-R	C-O	O-R	C-R	Meas.	Appr.	Cam.	Obj.	-
		PT	PT	PT	PT	PT	PT	MM	MR	PT	PT	CR
Human	97.2	95.7	98.9	97.5	94.2	98.8	96.4	95.3	98.5	98.6	98.7	97.0
Random Choice	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0
Proprietary Models												
Seed-1.6-2025-06-15 [48]	38.3	36.6	36.2	32.1	32.6	42.4	46.9	48.4	33.0	31.1	42.1	40.4
Gemini-2.5-pro-2025-06 [49]	38.0	38.7	34.0	40.7	44.2	38.8	41.0	62.5	30.3	39.2	25.0	33.3
Grok-4-2025-07-09 [62]	37.8	36.6	35.1	39.5	34.9	45.9	50.6	21.9	22.7	40.5	43.4	38.4
GPT-5-2025-08-07 [42]	41.8	41.9	33.0	35.8	49.8	42.4	68.7	54.7	37.4	28.3	40.8	36.4
Open-source General Models												
Bagel-7B-MoT [15]	31.0	34.4	35.1	29.6	32.6	42.4	31.3	34.4	21.2	18.9	27.6	30.3
Qwen2.5-VL-3B-Instruct [4]	28.6	36.6	30.9	28.4	26.7	28.2	31.3	31.2	16.7	16.2	35.5	28.8
Qwen2.5-VL-7B-Instruct [4]	26.8	28.0	26.6	19.8	32.6	38.8	28.9	23.4	21.2	20.3	30.3	24.8
Qwen3-VL-2B-Instruct [3]	28.9	26.9	29.8	30.9	38.4	35.3	33.7	23.4	28.8	29.7	28.9	21.2
Qwen3-VL-8B-Instruct [3]	31.1	28.0	37.2	32.1	31.4	35.3	38.5	37.5	15.2	27.0	28.9	29.8
InternVL3-2B [74]	26.5	31.2	22.3	28.4	30.2	28.2	28.9	25.0	22.7	16.2	28.9	26.8
InternVL3-8B [74]	28.0	22.6	22.3	34.6	31.4	42.4	33.7	25.0	19.7	20.3	34.2	24.8
Open-source Spatial Intelligence Models												
MindCube-3B-RawQA-SFT [70]	1.7	0.0	2.1	2.5	2.3	2.4	3.6	3.1	0.0	0.0	0.0	2.0
SpatialLadder-3B [32]	27.4	36.6	29.8	29.6	32.6	30.6	24.1	18.8	31.8	23.0	23.7	23.2
Spatial-MLLM-4B [59]	26.1	24.7	21.3	28.4	30.2	29.4	28.9	18.8	34.9	10.8	23.7	29.8
SpaceR-7B [43]	27.4	25.8	31.9	29.6	25.6	31.8	22.9	26.6	28.8	16.2	34.2	27.3
ViLaSR-7B [60]	30.2	29.0	35.1	28.4	39.5	40.0	44.6	31.2	16.7	17.6	31.6	23.2
VST-3B-SFT [66]	28.8	32.3	31.9	28.4	27.9	23.5	36.1	32.8	34.9	27.0	28.9	22.7
VST-7B-SFT [66]	32.5	39.8	36.2	35.8	37.2	29.4	33.7	29.7	47.0	36.5	35.5	18.2
Cambrian-S-3B [68]	27.0	25.8	28.7	24.7	48.8	24.7	33.7	29.7	22.7	20.3	28.9	18.7
Cambrian-S-7B [68]	27.1	24.7	26.6	24.7	47.7	22.4	31.3	32.8	24.2	12.2	30.3	24.2
Ours												
SenseNova-SI Bagel-7B-MoT	34.5	48.4	34.0	23.5	46.5	34.1	41.0	34.4	33.3	32.4	32.9	26.8
SenseNova-SI Qwen3-VL-8B	38.1	44.1	38.3	33.3	65.1	38.8	59.0	48.4	24.2	29.7	34.2	22.2
SenseNova-SI InternVL3-2B	34.2	39.8	45.7	33.3	46.5	30.6	39.8	31.2	30.3	29.7	32.9	24.8
SenseNova-SI InternVL3-8B	43.3	50.5	47.9	42.0	62.8	44.7	69.9	40.6	40.9	32.4	32.9	27.8

Table 12. **Evaluation on MMSI-Bench [67]**. Scores are Acc as in the original paper. Under Positional Relationship, C: Camera; O: Object; R: Region. All results are evaluated on EASI [7].

Models	Avg.	Rotation Among Around		
		PT	PT	PT
Human	94.5	-	-	-
Random Choice	33.0	33.3	31.8	35.7
Proprietary Models				
Seed-1.6-2025-06-15 [48]	48.8	89.0	36.4	45.6
Gemini-2.5-pro-2025-06 [49]	57.6	88.0	44.9	63.2
Grok-4-2025-07-09 [62]	63.6	93.0	54.4	61.6
GPT-5-2025-08-07 [42]	56.3	94.5	38.2	68.4
Open-source General Models				
Bagel-7B-MoT [15]	34.7	34.5	31.4	42.8
Qwen2.5-VL-3B-Instruct [4]	37.6	33.5	35.9	44.8
Qwen2.5-VL-7B-Instruct [4]	36.0	37.0	32.4	44.0
Qwen3-VL-2B-Instruct [3]	34.5	32.5	31.7	42.8
Qwen3-VL-8B-Instruct [3]	29.4	29.5	28.6	31.2
InternVL3-2B [74]	37.5	29.0	37.0	45.6
InternVL3-8B [74]	41.5	36.5	38.1	53.6
Open-source Spatial Intelligence Models				
MindCube-3B-RawQA-SFT [70]	51.7	34.0	51.0	67.6
SpatialLadder-3B [32]	43.5	35.0	43.2	50.8
Spatial-MLLM-4B [59]	33.5	39.0	30.5	36.0
SpaceR-7B [43]	38.0	35.0	34.2	49.2
ViLaSR-7B [60]	35.1	35.5	31.0	44.4
VST-3B-SFT [66]	36.0	32.0	34.9	41.6
VST-7B-SFT [66]	39.7	37.0	35.9	50.8
Cambrian-S-3B [68]	38.4	28.0	40.0	42.8
Cambrian-S-7B [68]	37.9	33.0	39.0	39.2
Ours				
SenseNova-SI _{Bagel-7B-MoT}	46.8	33.0	50.7	48.8
SenseNova-SI _{Qwen3-VL-8B}	73.8	79.5	73.2	70.4
SenseNova-SI _{InternVL3-2B}	41.8	30.5	46.4	40.0
SenseNova-SI _{InternVL3-8B}	85.7	82.0	84.9	90.4

Table 13. Evaluation on MindCube-Tiny [70]. All scores are Acc.

Models	Overall	Camera-based Tasks		Person-based Tasks		
		Rel. Dir.	Obj. Ori.	Obj. Ori.	Rel. Dir.	Sec. Sim.
		PT	PT	PT	PT	PT
Random Choice	26.3	25.2	26.1	24.6	31.1	26.3
Proprietary Models						
Seed-1.6-2025-06-15 [48]	43.9	55.8	26.9	54.8	48.5	26.6
Gemini-2.5-pro-2025-06 [49]	46.1	59.1	33.0	51.0	45.8	32.6
Grok-4-2025-07-09 [62]	43.2	57.1	23.9	47.6	51.7	24.9
GPT-5-2025-08-07 [42]	45.6	60.2	27.9	41.0	48.5	40.1
Open-source General Models						
Bagel-7B-MoT [15]	41.3	48.3	38.6	47.0	42.5	26.5
Qwen2.5-VL-3B-Instruct [4]	32.0	40.8	28.7	30.1	29.2	24.4
Qwen2.5-VL-7B-Instruct [4]	36.9	46.8	31.2	40.0	32.4	26.6
Qwen3-VL-2B-Instruct [3]	37.0	49.6	23.8	35.3	32.7	33.3
Qwen3-VL-8B-Instruct [3]	42.2	54.2	29.7	47.3	40.3	31.1
InternVL3-2B [74]	32.6	42.0	17.6	38.9	34.6	23.7
InternVL3-8B [74]	38.7	50.3	27.5	42.6	37.5	27.3
Open-source Spatial Intelligence Models						
MindCube-3B-RawQA-SFT [70]	24.1	30.9	22.1	23.7	22.4	16.9
SpatialLadder-3B [32]	39.9	46.2	25.7	56.2	31.9	33.6
Spatial-MLLM-4B [59]	34.7	35.0	23.4	40.4	40.4	34.6
SpaceR-7B [43]	35.9	43.2	28.9	37.5	34.1	30.2
ViLaSR-7B [60]	35.7	46.8	25.3	39.1	32.7	26.6
VST-3B-SFT [66]	52.9	46.9	35.4	70.3	52.6	62.8
VST-7B-SFT [66]	50.5	52.7	29.6	51.9	50.7	64.5
Cambrian-S-3B [68]	41.0	47.0	21.0	50.1	39.8	42.0
Cambrian-S-7B [68]	41.3	50.4	22.7	45.0	38.8	41.9
Ours						
SenseNova-SI <small>Bagel-7B-MoT</small>	46.9	54.7	33.5	45.9	43.9	49.5
SenseNova-SI <small>Qwen3-VL-8B</small>	51.2	60.3	22.0	67.8	41.5	55.6
SenseNova-SI <small>InternVL3-2B</small>	52.7	65.7	19.5	70.2	39.4	55.9
SenseNova-SI <small>InternVL3-8B</small>	54.7	66.3	43.2	38.4	43.1	70.0

Table 14. Evaluation on ViewSpatial-Bench [31]. All scores are Acc.

Models	Overall	Count		Loc	3D Inf	MultiV	Rel	Mov
		-	-					
Human	67.5	66.0	83.3	54.7	87.5	73.0	52.5	
Random Choice	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
Proprietary Models								
Seed-1.6-2025-06-15 [48]	54.6	62.0	66.5	60.4	37.1	70.6	32.4	
Gemini-2.5-pro-2025-06 [49]	57.1	61.3	69.2	55.2	38.5	71.5	48.6	
Grok-4-2025-07-09 [62]	47.0	50.4	60.3	51.6	26.2	61.2	37.4	
GPT-5-2025-08-07 [42]	61.9	63.1	57.0	73.1	49.9	72.2	60.7	
Open-source General Models								
Bagel-7B-MoT [15]	37.0	50.0	64.2	22.3	4.3	63.2	10.8	
Qwen2.5-VL-3B-Instruct [4]	33.1	48.4	43.6	16.1	8.5	56.3	12.4	
Qwen2.5-VL-7B-Instruct [4]	37.6	54.3	45.6	19.1	12.0	63.8	15.7	
Qwen3-VL-2B-Instruct [3]	35.7	43.6	45.8	26.1	14.4	57.0	18.9	
Qwen3-VL-8B-Instruct [3]	45.8	55.8	60.9	28.5	22.5	67.8	30.0	
InternVL3-2B [74]	30.0	46.5	40.4	13.8	5.7	50.9	9.8	
InternVL3-8B [74]	41.1	57.9	52.4	30.3	9.9	61.9	26.1	
Open-source Spatial Intelligence Models								
MindCube-3B-RawQA-SFT [70]	6.3	20.8	10.6	-6.9	-6.8	15.9	-8.9	
SpatialLadder-3B [32]	28.0	45.0	32.0	17.4	2.9	47.3	11.3	
Spatial-MLLM-4B [59]	18.0	30.3	24.2	7.0	7.3	20.0	10.9	
SpaceR-7B [43]	34.3	49.6	39.0	16.4	9.1	60.2	16.0	
ViLaSR-7B [60]	38.7	54.9	43.2	19.4	12.4	64.4	23.0	
VST-3B-SFT [66]	35.9	47.4	35.7	27.7	16.2	57.8	18.6	
VST-7B-SFT [66]	39.7	54.7	37.1	33.1	14.0	63.9	23.2	
Cambrian-S-3B [68]	31.0	47.1	46.2	16.4	1.4	44.0	24.6	
Cambrian-S-7B [68]	36.1	50.0	50.3	24.1	8.7	51.2	26.6	
Ours								
SenseNova-SI _{Bagel-7B-MoT}	42.0	47.4	62.9	30.6	27.8	59.2	17.4	
SenseNova-SI _{Qwen3-VL-8B}	49.6	53.1	56.1	39.1	40.3	64.9	24.8	
SenseNova-SI _{InternVL3-2B}	36.8	51.0	45.8	23.1	16.1	48.9	26.8	
SenseNova-SI _{InternVL3-8B}	47.7	56.2	49.2	40.6	35.1	57.6	40.5	

Table 15. **Evaluation on SITE [57]**. All scores are **CAA**. We follow the **SITE** paper’s original evaluation protocol, where MCQ are answered by *direct QA*.

Models	Avg.	Art	Count	Forens.	FuncCr	IQ	Jigsaw	MV-R	ObjLoc	Depth	Ref.	SemCr	SpatR	VisCr	VisSm
	-	-	-	-	-	CR	-	PT	-	SR	-	-	SR	PT	-
Random Choice	37.7	50.0	25.0	25.0	25.0	25.0	50.0	50.0	50.0	50.0	33.3	25.0	50.0	25.0	50.0
Proprietary Models															
Seed-1.6-2025-06-15 [48]	65.9	80.3	74.2	62.9	53.1	28.7	59.3	55.6	64.8	85.5	32.1	64.8	86.7	90.7	84.4
Gemini-2.5-pro-2025-06 [49]	73.5	86.0	78.0	90.0	62.0	44.0	85.0	51.0	76.0	86.0	34.0	68.0	90.0	91.0	86.0
Grok-4-2025-07-09 [62]	56.4	82.0	65.8	78.8	47.7	8.0	60.0	42.1	54.1	68.5	40.3	38.1	83.2	47.7	85.2
GPT-5-2025-08-07 [42]	68.0	85.5	69.2	79.5	60.0	32.7	70.0	40.6	61.5	81.5	32.8	66.2	88.8	94.2	87.4
Gemini-3-Pro-Preview [21]	76.0	86.3	77.5	98.5	66.9	42.0	77.3	57.9	85.2	90.3	32.1	72.7	93.0	95.9	88.9
Open-source General Models															
Bagel-7B-MoT [15]	63.6	78.6	75.0	37.9	40.0	32.0	85.3	45.9	72.1	89.5	31.3	46.0	82.5	82.0	92.6
Qwen2.5-VL-3B-Instruct [4]	48.7	57.3	68.3	32.6	22.3	20.0	46.0	43.6	55.7	74.2	38.8	36.7	82.5	45.4	65.2
Qwen2.5-VL-7B-Instruct [4]	55.9	71.8	71.7	47.7	23.9	26.7	56.0	55.6	52.5	79.0	38.1	36.7	88.8	52.3	88.2
Qwen3-VL-2B-Instruct [3]	53.2	65.8	62.5	34.1	33.1	1.3	54.0	55.6	59.8	71.0	30.6	59.0	81.1	63.4	77.8
Qwen3-VL-8B-Instruct [3]	66.7	77.8	67.5	87.9	53.9	12.0	67.3	54.1	69.7	88.7	32.1	59.7	87.4	89.5	88.2
InternVL3-2B [74]	50.8	56.4	73.3	18.9	26.9	24.0	75.3	59.4	62.3	71.0	34.3	23.7	79.0	34.3	80.0
InternVL3-8B [74]	53.5	67.5	71.7	34.9	28.5	23.3	78.0	44.4	53.3	69.3	34.3	28.1	86.7	44.2	89.6
Open-source Spatial Intelligence Models															
MindCube-3B-RawQA-SFT [70]	35.1	53.0	36.7	26.5	6.9	26.0	47.3	55.6	56.6	41.1	31.3	0.0	67.1	3.5	51.1
SpatialLadder-3B [32]	43.0	53.9	53.3	22.7	23.9	26.7	49.3	44.4	50.0	58.1	31.3	32.4	75.5	29.1	57.8
Spatial-MLLM-4B [59]	40.5	47.0	40.8	22.0	20.8	23.3	54.0	43.6	54.9	63.7	29.1	34.5	54.5	35.5	46.7
SpaceR-7B [43]	49.6	76.1	65.0	32.6	20.0	4.0	60.0	43.6	52.5	66.1	31.3	35.2	80.4	51.2	83.7
ViLaSR-7B [60]	51.4	68.4	70.0	55.3	26.1	18.0	63.3	48.1	40.2	74.2	12.7	19.4	86.7	55.2	85.9
VST-3B-SFT [66]	58.8	65.8	69.2	53.0	23.1	33.3	68.0	51.1	54.1	86.3	35.8	30.9	74.1	89.5	84.4
VST-7B-SFT [66]	61.9	67.5	70.8	55.3	26.9	23.3	80.0	63.2	54.1	91.9	34.3	33.1	81.1	94.2	85.9
Cambrian-S-3B [68]	37.7	32.5	63.3	15.2	20.0	27.3	50.0	6.8	50.8	54.0	31.3	31.6	66.4	34.3	46.7
Cambrian-S-7B [68]	37.9	42.7	63.3	0.0	20.8	24.0	37.3	44.4	51.6	69.3	28.4	28.8	77.6	28.5	21.5
Ours															
SenseNova-SI _{Bagel-7B-MoT}	65.4	81.2	71.7	33.3	49.2	28.7	77.3	44.4	77.0	87.1	33.6	66.9	76.2	94.8	91.8
SenseNova-SI _{Qwen3-VL-8B}	61.9	74.4	61.7	69.7	50.8	13.3	64.0	32.3	65.6	88.7	30.6	52.5	79.7	93.6	88.2
SenseNova-SI _{InternVL3-2B}	52.4	61.5	65.8	12.9	35.4	21.3	63.3	6.0	58.2	80.7	35.1	36.0	74.1	93.0	83.7
SenseNova-SI _{InternVL3-8B}	63.9	68.4	76.7	32.6	33.9	21.3	80.0	90.2	54.9	93.5	32.8	46.0	86.0	95.9	77.8

Table 16. Evaluation on BLINK [20]. All scores are Acc.

Models	Overall	Height		Location			Multi-Object				Orientation		
		Higher	Above	Closer	Next To	Closer	Facing	Parall.	Same D.	ViewPt	Front	Left	ViewPt
		SR	SR	SR	SR	PT,SR	PT,SR	PT,SR	PT,SR	PT,SR	PT	PT	PT
Random Choice	45.8	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	25.0	50.0	50.0	25.0
Proprietary Models													
Seed-1.6-2025-06-15 [48]	56.9	67.7	64.3	74.9	67.8	67.1	64.5	56.6	48.8	17.5	61.9	32.7	23.6
Gemini-2.5-pro-2025-06 [49]	59.3	68.1	66.3	78.8	81.1	66.9	63.3	56.9	45.4	15.2	67.2	31.2	37.3
Grok-4-2025-07-09 [62]	54.9	63.5	68.6	71.5	66.1	52.6	61.6	46.3	46.5	21.9	62.8	36.4	23.0
GPT-5-2025-08-07 [42]	60.3	66.1	67.8	82.7	72.9	64.9	63.9	54.6	45.1	22.2	74.4	40.1	33.8
Gemini-3-Pro-Preview [21]	68.9	69.0	73.1	80.1	72.9	74.9	73.4	66.1	63.4	35.3	82.3	72.8	48.4
Open-source General Models													
Bagel-7B-MoT [15]	50.2	47.0	64.0	75.5	49.6	56.6	58.1	49.0	49.1	5.0	58.1	29.2	26.5
Qwen2.5-VL-3B-Instruct [4]	43.5	45.8	54.5	69.6	47.2	44.0	44.5	23.6	39.8	14.6	43.9	34.4	20.4
Qwen2.5-VL-7B-Instruct [4]	47.5	44.9	50.9	74.3	56.0	48.3	51.2	37.8	47.7	9.9	56.1	34.4	31.8
Qwen3-VL-2B-Instruct [3]	47.5	50.0	57.8	74.9	57.8	56.6	50.0	48.7	45.4	4.4	47.1	26.1	11.1
Qwen3-VL-8B-Instruct [3]	53.9	47.8	62.6	87.6	57.8	62.0	65.6	46.0	47.1	13.1	56.7	30.4	35.0
InternVL3-2B [74]	47.7	50.0	54.2	61.8	55.8	53.4	45.4	50.4	52.6	12.5	53.2	34.1	26.2
InternVL3-8B [74]	44.2	49.4	58.2	76.1	59.3	46.3	37.6	17.4	34.0	13.7	43.6	21.8	23.3
Open-source Spatial Intelligence Models													
MindCube-3B-RawQA-SFT [70]	2.8	0.0	12.3	0.0	5.3	6.9	4.3	0.0	0.0	0.0	0.0	0.0	0.0
SpatialLadder-3B [32]	42.8	45.1	39.9	65.3	47.5	46.9	59.0	45.1	48.8	8.8	39.8	34.4	11.1
Spatial-MLLM-4B [59]	36.2	42.5	41.3	45.7	46.0	40.6	61.0	36.9	47.7	5.2	41.3	3.7	1.8
SpaceR-7B [43]	40.5	36.4	47.2	63.1	44.5	42.9	52.3	24.2	48.0	3.8	54.4	27.2	16.3
ViLaSR-7B [60]	46.6	38.3	65.8	66.4	50.4	45.4	59.5	47.8	45.4	3.5	52.3	33.5	20.1
VST-3B-SFT [66]	48.7	47.7	55.4	86.3	46.3	62.0	57.8	39.2	39.2	16.6	41.0	32.1	18.4
VST-7B-SFT [66]	53.1	55.8	53.3	84.5	55.5	72.9	64.2	43.7	45.6	17.2	53.8	32.7	24.2
Cambrian-S-3B [68]	41.4	33.5	46.2	52.8	61.6	53.4	55.8	44.0	50.0	10.5	39.0	26.6	15.2
Cambrian-S-7B [68]	45.0	43.3	46.1	58.3	64.9	61.4	48.3	35.4	48.8	19.5	50.3	25.2	26.8
Ours													
SenseNova-SI _{Bagel-7B-MoT}	42.4	33.5	47.4	67.7	47.8	43.4	53.5	43.4	46.5	2.9	47.7	28.9	23.6
SenseNova-SI _{Qwen3-VL-8B}	53.2	52.2	55.9	85.0	58.4	66.6	61.9	41.0	49.1	20.7	49.7	34.1	30.9
SenseNova-SI _{InternVL3-2B}	50.5	50.0	51.6	66.8	64.9	66.9	66.2	49.0	50.0	21.3	45.9	28.9	27.7
SenseNova-SI _{InternVL3-8B}	55.5	49.1	51.5	83.8	59.3	75.7	59.2	46.9	53.2	16.9	60.2	60.5	32.6

Table 17. Evaluation on 3DSRBench [39]. All scores are Acc (circ_eval setting).

Models	Overall	MP3D	AI2-THOR	ScanNet
		SR	SR	SR
Random Choice	25.0	25.0	25.0	25.0
Proprietary Models				
Seed-1.6-2025-06-15 [48]	75.4	74.3	75.1	76.7
Gemini-2.5-pro-2025-06 [49]	78.8	77.8	78.0	80.8
Grok-4-2025-07-09 [62]	75.5	75.3	75.4	75.8
GPT-5-2025-08-07 [42]	81.6	81.3	80.5	83.0
Gemini-3-Pro-Preview [21]	84.3	83.3	84.5	85.0
Open-source General Models				
Bagel-7B-MoT [15]	73.1	72.2	71.4	75.8
Qwen2.5-VL-3B-Instruct [4]	62.3	63.2	56.7	67.1
Qwen2.5-VL-7B-Instruct [4]	71.8	74.6	70.5	70.4
Qwen3-VL-2B-Instruct [3]	70.1	72.2	67.5	70.8
Qwen3-VL-8B-Instruct [3]	77.7	79.6	75.1	78.6
InternVL3-2B [74]	60.1	60.8	55.8	63.9
InternVL3-8B [74]	76.3	76.3	74.2	78.5
Open-source Spatial Intelligence Models				
MindCube-3B-RawQA-SFT [70]	37.0	36.7	33.0	41.3
SpatialLadder-3B [32]	58.2	60.0	54.1	60.8
Spatial-MLLM-4B [59]	50.0	51.5	46.9	51.5
SpaceR-7B [43]	66.9	67.8	66.7	66.3
ViLaSR-7B [60]	67.3	69.5	67.9	64.5
VST-3B-SFT [66]	69.0	71.7	63.3	72.2
VST-7B-SFT [66]	73.7	76.2	71.1	74.0
Cambrian-S-3B [68]	63.5	63.5	60.2	67.0
Cambrian-S-7B [68]	72.8	71.3	72.9	74.3
Ours				
SenseNova-SI _{Bagel-7B-MoT}	69.0	68.2	63.8	75.1
SenseNova-SI _{Qwen3-VL-8B}	72.5	75.8	66.4	75.4
SenseNova-SI _{InternVL3-2B}	62.8	64.2	56.7	67.6
SenseNova-SI _{InternVL3-8B}	72.0	71.0	70.5	75.1

Table 18. Evaluation on EmbSpatialBench [16]. All scores are Acc.

J. Case Study on Text-based CoT

As discussed in Sec. 5.6, we observe that textual CoT does not appear to yield genuinely stronger 3D reasoning capabilities. With supporting case studies in Fig. 7 and Fig. 8, we tentatively consider several hypotheses rather than definitive explanations, and we leave a more thorough investigation to future work. Note that green indicates correct reasoning steps, while red indicates wrong reasoning steps.

Train–test mismatch and error accumulation. Textual CoT enforces a long, linear per-token reasoning process. During training, tokens are learned with ground-truth context, whereas at inference time the model conditions on its own generated tokens. Early local errors in the reasoning (as in Fig. 8) may therefore accumulate along the sequence and become increasingly difficult to correct; tasks that require a globally coherent 3D structure may be particularly sensitive to this discrepancy.

Bias toward human-preferred explanation paths. Compared to direct QA supervision, textual CoT shifts the optimization focus from task accuracy towards reproducing specific intermediate verbal rationalizations, which may not align with the geometry-dominant nature of multi-view 3D reasoning. Supervision along a single, human-preferred explanatory path may effectively shrink the model’s reasoning space.

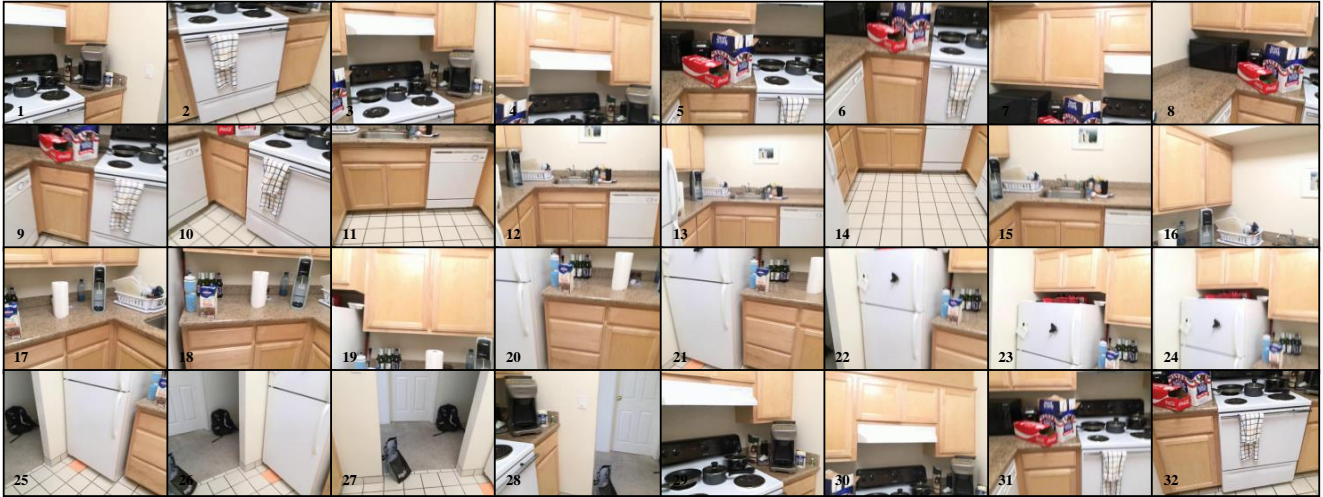


Figure 7. Successful case of trained *CoT-SenseNova-SI-CGMap* output on VSI-Bench [64] Rel. Dir subset. The model maintains consistent and accurate reasoning through images and concludes to a **correct final answer**.

Question:

If I am standing by the refrigerator and facing the backpack, is the microwave to my front-left, front-right, back-left, or back-right?

The directions refer to the quadrants of a Cartesian plane (if I am standing at the origin and facing along the positive y-axis).

Options: A. back-left, B. back-right, C. front-left, D. front-right

GT: A

Prediction (CoT-SenseNova-SI-CGMap):

<think>

I will reason through this question step by step.

For one object in one image coordination system, its location is described as "category.id at bbox [x1, y1, x2, y2], at direction, distance from camera and x, y".

x-axis is the right side of the camera view perspective. y-axis is the forward perspective.

The unit of distance is meters, and the unit of angle is degrees.

For describing camera motion, the movement from image 'a' to image 'b' is defined relative to the coordinate system of image 'a'.

Step 1: ****Clarify the queried objects in the related image****

From the question, I can infer that the queried objects are the refrigerator, the backpack and the microwave.

After looking over the provided images, I can get that the refrigerator.6 shows in image 21, specifically at bbox [0.09, 0.0, 0.67, 0.98], at 5 degrees(CCW), dist 1.5, x -0.1, y 1.4; the backpack.19 shows in image 27, specifically at bbox [0.54, 0.17, 0.76, 0.45], at 10 degrees(CW), dist 2.6, x 0.4, y 2.3; the microwave.28 shows in image 7, specifically at bbox [0.03, 0.72, 0.37, 1.0], at 15 degrees(CCW), dist 1.6, x -0.4, y 1.5.

Consequently, to solve this question, image [7, 21, 27] need to be paid more attentions to. In this step, I have also assigned an id to each queried object.

I will keep this custom to distinguish objects that belong to the same category but are not the same instance.

Step 2: ****Establish connections among the relevant images****

To analyze the relationships among the images identified in Step 1, I will select some images to connect them.

Two images are connected only if they share corresponding objects.

After looking over all these images, I derive the following chain that illustrates how the queried images are connected:

From image 1 to image 5 via image 2; image 3; image 4, the corresponding object is: stove.3. Based on the object appearance change, I can infer that in this phase, the camera translates right and front, rotates to left and looks down.

From image 5 to image 7 via image 6, the corresponding objects are: microwave.28, box.26. Based on the object appearance change, I can infer that in this phase, the camera does not move significantly, rotates to left slightly and looks up.

From image 7 to image 13 via image 8; image 11, the corresponding object is: sink_0. Based on the object appearance change, I can infer that in this phase, the camera translates right and back, rotates to left and looks down.

From image 13 to image 21 via image 15; image 17, the corresponding object is: (dish rack)_5. Based on the object appearance change, I can infer that in this phase, the camera translates right and front, rotates to left and looks down.

From image 21 to image 27 via image 25, the corresponding object is: refrigerator_6. Based on the object appearance change, I can infer that in this phase, the camera translates left, rotates to left and looks down.

Based on these analyse, I can infer that image 7 and image 21 can be related by image order [7, 8, 11, 13, 15, 17, 21]. Then image 27 can be further connected by image order [21, 25, 27].

Step 3: ****Construct Scene Layout****

The overall scene layout is constructed in the coordinate system of image 7 as follows.

How the main images are connected has been described in Step 1. The I will analyse the layout in detail.

**** From image7 to image13 ****

Image 7 contains objects: microwave_28 at bbox [0.03, 0.72, 0.37, 1.0], at 15 degrees(CCW), dist 1.6, x -0.4, y 1.5; sink_0 at bbox [0.43, 0.52, 0.56, 0.59], at 0 degrees(CW), dist 1.9, x 0.1, y 1.8; (dish rack)_5 at bbox [0.03, 0.26, 0.33, 0.47], at 15 degrees(CCW), dist 1.7, x -0.5, y 1.6.

Image 13 contains objects: sink_0 at bbox [0.41, 0.48, 0.64, 0.6], at 0 degrees(CW), dist 2.0, x 0.1, y 1.9.

The correspond objects between image 7 and image 13 is sink_0.

According to the objects motion changes, from image7 to image13, the camera motion is: translation (x 0.2, y -0.1), rotation (left 25 degrees, down 10 degrees).

**** From image13 to image21 ****

Image 13 contains objects: (dish rack)_5 at bbox [0.03, 0.26, 0.33, 0.47], at 15 degrees(CCW), dist 1.7, x -0.5, y 1.6.

Image 21 contains objects: (dish rack)_5 at bbox [0.85, 0.01, 1.0, 0.37], at 35 degrees(CW), dist 1.5, x 0.8, y 1.2.

The correspond objects between image 13 and image 21 is (dish rack)_5.

According to the objects motion changes, from image13 to image21, the camera motion is: translation (x 0.4, y 0.4), rotation (left 70 degrees, down 10 degrees).

As mentioned above, from image7 to image13, the camera motion is: translation (x 0.2, y -0.1), rotation (left 25 degrees, down 10 degrees).

Consequently, by accumulating the transformations, from image7 to image21, the camera motion is: translation (x 0.4, y 0.4), rotation (left 95 degrees, down 20 degrees).

**** From image21 to image27 ****

Image 21 contains objects: refrigerator_6 at bbox [0.34, 0.0, 0.79, 0.87], at 5 degrees(CW), dist 1.6, x 0.1, y 1.5.

Image 27 contains objects: backpack_19 at bbox [0.54, 0.17, 0.76, 0.45], at 10 degrees(CW), dist 2.6, x 0.4, y 2.3; refrigerator_6 at bbox [0.38, 0.0, 0.79, 0.88], at 5 degrees(CW), dist 1.6, x 0.1, y 1.4. The correspond objects between image 21 and image 27 is refrigerator_6.

According to the objects motion changes, from image21 to image27, the camera motion is: translation (x -0.2, y 0.0), rotation (left 50 degrees, down 15 degrees).

As mentioned above, from image7 to image21, the camera motion is: translation (x 0.4, y 0.4), rotation (left 95 degrees, down 20 degrees).

Consequently, by accumulating the transformations, from image7 to image27, the camera motion is: translation (x 0.4, y 0.2), rotation (left 145 degrees, down 35 degrees).

In this phase, image 27 has introduced some new objects in its own coordinate system: backpack_19 at bbox [0.54, 0.17, 0.76, 0.45]; refrigerator_6 at bbox [0.38, 0.0, 0.79, 0.88].

Transfer these objects from image 27 into image 7 coordination system: backpack_19 at 140 degrees(CCW), dist 2.5, x -1.2, y -1.5; refrigerator_6 at 70 degrees(CCW), dist 1.4, x -1.0, y 0.4;

Step 4: ****Infer object spatial relation and answer the question****

After localizing the objects from questions in the scene layout from step2, we can know that "In image 7 coordination system: refrigerator_6 at 70 degrees(CCW), dist 1.4, x -1.0, y 0.4; backpack_19 at 140 degrees(CCW), dist 2.5, x -1.2, y -1.5; microwave_28 at 15 degrees(CCW), dist 1.6, x -0.4, y 1.5".

Considering the ****view perspective**** mentioned in the question, the right answer this question is the right answer is 'A. back-left'

</think>

<answer>A</answer>

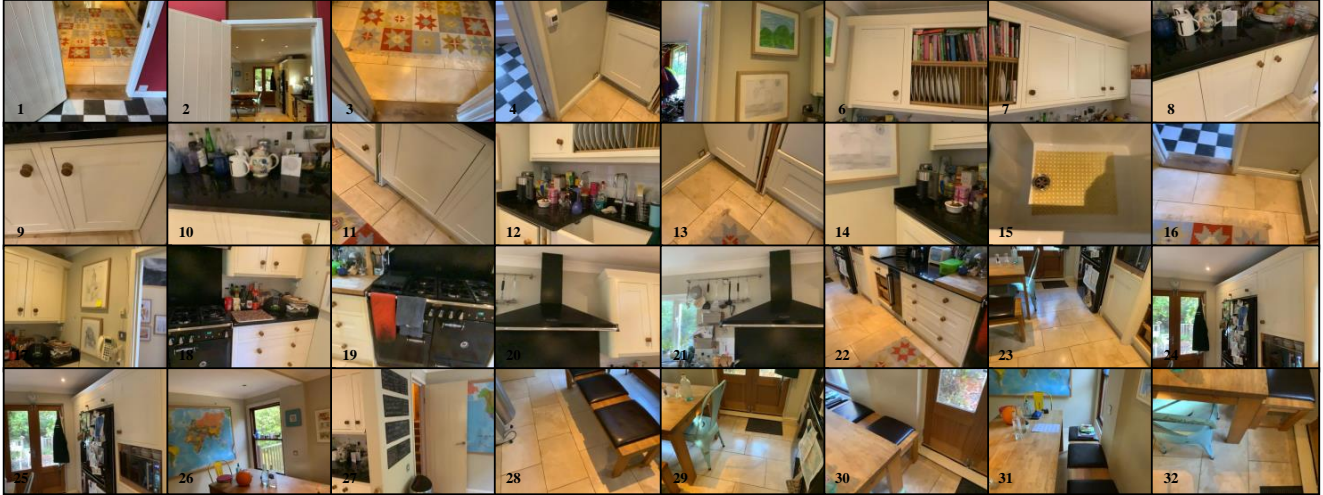


Figure 8. Failure case of trained *CoT-SenseNova-SI-CGMap* output on VSI-Bench [64] Rel. Dir subset. Multiple local inconsistencies accumulate through the reasoning trajectory and ultimately lead to an incorrect answer.

Question:

If I am standing by the stove and facing the table, is the refrigerator to my front-left, front-right, back-left, or back-right?

The directions refer to the quadrants of a Cartesian plane (if I am standing at the origin and facing along the positive y-axis).

Options: A. front-right, B. front-left, C. back-right, D. back-left

GT: A

Prediction (CoT-SenseNova-SI-CGMap) :

<think>

I will reason through this question step by step.

For one object in one image coordination system, its location is described as "category.id at bbox [x1, y1, x2, y2], at direction, distance from camera and x, y".

x-axis is the right side of the camera view perspective. y-axis is the forward perspective.

The unit of distance is meters, and the unit of angle is degrees.

For describing camera motion, the movement from image 'a' to image 'b' is defined relative to the coordinate system of image 'a'.

Step 1: **Clarify the queried objects in the related image**

From the question, I can infer that the queried objects are the stove, the table and the refrigerator. After looking over the provided images, I can get that the stove₁₈ shows in image 19, specifically at bbox [0.21, 0.01, 1.0, 1.0], at 5 degrees(CW), dist 1.6, x 0.1, y 1.4; the table₂₅ shows in image 21, specifically at bbox [0.02, 0.82, 0.2, 1.0], at 25 degrees(CCW), dist 2.6, x -0.9, y 2.1; the refrigerator₁₀ shows in image 25, specifically at bbox [0.41, 0.26, 0.6, 0.98], at 5 degrees(CW), dist 2.4, x 0.1, y 2.3.

Consequently, to solve this question, image [19, 21, 25] need to be paid more attentions to.

In this step, I have also assigned an id to each queried object.

I will keep this custom to distinguish objects that belong to the same category but are not the same instance.

Step 2: **Establish connections among the relevant images**

To analyze the relationships among the images identified in Step 1, I will select some images to connect them.

Two images are connected only if they share corresponding objects.

After looking over all these images, I derive the following chain that illustrates how the queried images are connected:

From image 1 to image 2 directly, the corresponding object is: doorframe₂₄. Based on the object appearance change, I can infer that in this phase, the camera translates right, rotates to right slightly and looks up.

From image 1 to image 4 directly, the corresponding object is: door₂₁. Based on the object appearance change, I can infer that in this phase, the camera translates left and front, rotates to left slightly and looks up.

From image 4 to image 16 directly, the objects are mostly not fully visible. But Based on the object appearance change, I can infer that in this phase, the camera translates left, rotates to left and looks down.

From image 16 to image 19 directly, the corresponding objects are: towel_16, stove_18. Based on the object appearance change, I can infer that in this phase, the camera translates right and front, rotates to back-left and looks up.

From image 2 to image 21 via image 19, the corresponding object is: (kitchen cabinets)_4. Based on the object appearance change, I can infer that in this phase, the camera translates right and front, rotates to left and looks up.

From image 21 to image 25 via image 24, the corresponding object is: doors_0. Based on the object appearance change, I can infer that in this phase, the camera translates left and rears, rotates to right and looks up.

Based on these analyse, I can infer that image 19 and image 21 can be related by image order [19, 16, 4, 1, 2, 19]. Then image 25 can be further connected by image order [19, 21, 24, 25].

Step 3: **Construct Scene Layout**

The overall scene layout is constructed in the coordinate system of image 19 as follows.

How the main images are connected has been described in Step 1. The I will analyse the layout in detail.

** From image19 to image4 **

Image 19 contains objects: stove_18 at bbox [0.21, 0.01, 1.0, 1.0], at 5 degrees(CW), dist 1.6, x 0.1, y 1.4; towel_16 at bbox [0.24, 0.35, 0.41, 0.78], at 10 degrees(CCW), dist 1.4, x -0.2, y 1.3; towel_15 at bbox [0.31, 0.32, 0.49, 0.77], at 5 degrees(CCW), dist 1.4, x -0.1, y 1.3.

Image 4 contains objects: towel_16 at bbox [0.9, 0.65, 1.0, 0.99], at 40 degrees(CW), dist 1.4, x 0.7, y 0.9; towel_15 at bbox [0.89, 0.34, 1.0, 0.83], at 35 degrees(CW), dist 1.4, x 0.7, y 1.0.

The correspond objects between image 19 and image 4 are towel_16, towel_15.

According to the objects motion changes, from image19 to image4, the camera motion is: translation (x -0.1, y 0.1), rotation (right 50 degrees, down 30 degrees).

** From image4 to image16 **

According to the objects motion changes, from image4 to image16, the camera motion is: translation (x -0.2, y 0.0), rotation (left 30 degrees, down 15 degrees).

As mentioned above, from image19 to image4, the camera motion is: translation (x -0.1, y 0.1), rotation (right 50 degrees, down 30 degrees).

Consequently, by accumulating the transformations, from image19 to image16, the camera motion is: translation (x -0.2, y 0.3), rotation (right 20 degrees, down 45 degrees).

** From image16 to image19 **

Image 16 contains objects: stove_18 at bbox [0.02, 0.0, 0.56, 0.51], at 20 degrees(CCW), dist 1.8, x -0.5, y 1.6; towel_16 at bbox [0.92, 0.0, 1.0, 0.11], at 30 degrees(CW), dist 1.6, x 0.7, y 1.3.

Image 19 contains objects: stove_18 at bbox [0.21, 0.01, 1.0, 1.0], at 5 degrees(CW), dist 1.6, x 0.1, y 1.4; towel_16 at bbox [0.24, 0.35, 0.41, 0.78], at 10 degrees(CCW), dist 1.4, x -0.2, y 1.3.

The correspond objects between image 16 and image 19 are stove_18, towel_16.

According to the objects motion changes, from image16 to image19, the camera motion is: translation (x 0.7, y 1.2), rotation (left 100 degrees, up 30 degrees).

As mentioned above, from image19 to image16, the camera motion is: translation (x -0.2, y 0.3), rotation (right 20 degrees, down 45 degrees).

Consequently, by accumulating the transformations, from image19 to image16, the camera motion is: translation (x 0.8, y 1.2), rotation (left 80 degrees, down 15 degrees).

** From image2 to image21 **

Image 2 contains objects: (kitchen cabinets)_4 at bbox [0.6, 0.7, 0.7, 0.96], at 10 degrees(CW), dist 2.9, x 0.5, y 2.6.

Image 21 contains objects: (kitchen cabinets)_4 at bbox [0.19, 0.71, 0.36, 1.0], at 10 degrees(CCW), dist 2.2, x -0.4, y 1.9; table_25 at bbox [0.02, 0.82, 0.2, 1.0], at 25 degrees(CCW), dist 2.6, x -0.9, y 2.1.

The correspond objects between image 2 and image 21 is (kitchen cabinets)_4.

According to the objects motion changes, from image2 to image21, the camera motion is: translation (x 2.0, y 1.2), rotation (left 45 degrees, up 25 degrees).

As mentioned above, from image19 to image2, the camera motion is: translation (x 0.1, y 0.1), rotation (right 50 degrees, down 30 degrees).

Consequently, by accumulating the transformations, from image19 to image21, the camera motion is: translation (x 2.4, y -0.5), rotation (right 5 degrees, down 0 degrees).

In this phase, image 21 has introduced some new objects in its own coordinate system: (kitchen cabinets)_4 at bbox [0.19, 0.71, 0.36, 1.0]; table_25 at bbox [0.02, 0.82, 0.2, 1.0].

Transfer these objects from image 21 into image 19 coordination system: (kitchen cabinets)_4 at 60 degrees(CW), dist 3.1, x 2.5, y 1.4; table_25 at 50 degrees(CW), dist 3.2, x 2.3, y 1.9;

** From image21 to image25 **

Image 21 contains objects: doors_0 at bbox [0.01, 0.23, 0.22, 0.85], at 25 degrees(CCW), dist 3.1, x -1.3, y 2.8.

Image 25 contains objects: doors_0 at bbox [0.0, 0.36, 0.32, 0.97], at 20 degrees(CCW), dist 3.7, x -1.2, y 3.5.

The correspond objects between image 21 and image 25 is doors_0.

According to the objects motion changes, from image21 to image25, the camera motion is: translation (x -1.0, y -0.2), rotation (right 30 degrees, up 10 degrees).

As mentioned above, from image19 to image21, the camera motion is: translation (x 2.4, y -0.5), rotation (right 5 degrees, down 0 degrees).

Consequently, by accumulating the transformations, from image19 to image25, the camera motion is: translation (x 1.4, y -0.7), rotation (right 35 degrees, up 10 degrees).

In this phase, image 25 has introduced some new objects in its own coordinate system: doors_0 at bbox [0.0, 0.36, 0.32, 0.97].

Transfer these objects from image 25 into image 19 coordination system: doors_0 at 40 degrees(CW), dist 3.6, x 2.3, y 2.8;

Step 4: **Infer object spatial relation and answer the question**

After localizing the objects from questions in the scene layout from step2, we can know that "In image 19 coordination system: the stove_18 at 5 degrees(CW), dist 1.6, x 0.1, y 1.4; the table_25 at 50 degrees(CW), dist 3.2, x 2.3, y 1.9; the refrigerator_10 at 5 degrees(CW), dist 2.4, x 0.1, y 2.3".

Considering the **view perspective** mentioned in the question, the right answer this question is the right answer is 'C. back-right'

</think> <answer>C</answer>