Supplementary Material for "Keep it SymPL: Symbolic Projective Layout for Allocentric Spatial Reasoning in Vision-Language Models"

A. SymPL Framework

A.1. Implementation Details

Object Detection. To extract 3D object information from an image, we used GroundingDINO [7] to obtain 2D bounding boxes. When using only the bounding box with the highest confidence score, the model frequently detected incorrect objects. To address this issue, we selected the top n bounding boxes based on their confidence scores and generated cropped images for each of them. As shown in Figure S1, these n cropped images were arranged sequentially with their corresponding ranks. To include minimal contextual information around the target object, we expanded the bounding box by a fixed $margin$ in all directions before cropping the image. We then prompted the VLM to identify the image that best matched the target object, and used the bounding box corresponding to the selected cropped image as the final detection result. In all experiments, we set $n = 5$ and $margin = 30$ for object detection.

Figure S1. Candidate cropped images fed to the VLM. The target object is the **dog**. Each image is indexed at the top according to its confidence score ranking, and the VLM returns the index of the image that best matches the target object through reasoning.

3D Position Estimation. To estimate the precise 3D coordinates of each object, we used the bounding box from GroundingDINO and the depth map predicted by DepthPro [2]. First, we cropped the corresponding region on the depth map using the bounding box and extracted the depth values within the masked area. Then, using the intrinsic parameters predicted by DepthPro, we unprojected each pixel coordinate within the mask into 3D space to generate a 3D point cloud. Next, we split the range of depth values into up to 20 bins of equal width and selected the bin that contained the most depth samples. Finally, we filtered depth values to those within a fixed ratio of the selected bin center and computed their median, which we used as the final estimate of the object's 3D coordinate. In all experiments, we set the ratio to 0.12 for filtering depth values.

The accuracy of the estimated 3D coordinates depended on the intrinsic parameters predicted by DepthPro. When these parameters were inaccurate, the z value of the 3D coordinates could become much larger or smaller than the x and y values. This issue was particularly critical in egocentric spatial reasoning, where the raw z values were directly used for *projection*. To reduce this distortion, we applied an additional correction to the z values before determining the final 3D coordinates. We first computed a z-axis scale as the mean absolute z value of the estimated 3D positions of all objects. We then computed an x–y scale as the mean distance of these 3D positions from the origin in the x–y plane. If the ratio between these two scales exceeded a predefined threshold, we multiplied all z values by a fixed scaling factor determined by this threshold so that the z-axis scale did not become excessively larger or smaller than the x–y scale. In the experiments, we set the threshold to 10 for z-axis correction.

B. Experiments

B.1. Hardware Settings

All main experiments were conducted on a shared server using a single NVIDIA RTX A6000 GPU. As an exception, the APC series was run on a local machine with two NVIDIA RTX 3090 GPUs, because APC-Vis image rendering was not supported on the shared server. For the additional ablation studies, we used both NVIDIA RTX A6000 and NVIDIA RTX 6000 Ada Generation GPUs across the study, while each subfigure and table was composed of results obtained on a single, consistent GPU model chosen from these two.

B.2. Dataset Configurations

We evaluated spatial reasoning abilities, including allocentric spatial reasoning, using five processed datasets: COMFORT#, 3DSRBench, COCOSPATIAL, COMFORT VI, and COMFORT Multi. All experiments were conducted using the evaluation toolkit provided by VLMEvalKit [4]. The preprocessing procedure for each dataset is as follows.

COMFORT#. As shown in Figure S2, COMFORT# is a dataset constructed to evaluate each model's allocentric spatial reasoning ability. We built the dataset in a Blender-based simulation environment [11] by composing scenes from a set of 12 assets (couch, chair, dog, duck, penguin, laptop, woman, cat, refrigerator, horse, camel, and snowman), randomly sampling objects from this list for each scene. The dataset consisted of four spatial reasoning categories: *closer*, *left/right*, *visibility*, and *facing*. For *visibility*, each scene contained two objects, while the other three categories used three objects per scene.

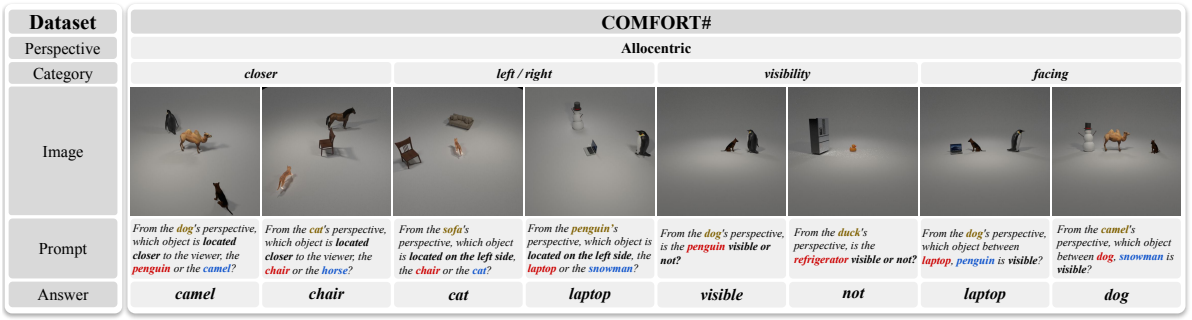


Figure S2. Examples of COMFORT#. This dataset consists of four spatial relation categories: *closer*, *left/right*, *visibility* and *facing*.

In the *closer* category, objects were arranged in a line such that the reference viewer was located at one end, and questions asked which object is closer to the reference viewer. In the *left/right* category, we placed two objects in front of a reference viewer object and constructed questions that require inferring which object is on the left from the reference viewer's perspective. In the *visibility* category, we randomly configured the target object to be either in front of or behind the reference viewer and constructed questions that require determining whether the target object was visible to the reference viewer in each case. Finally, in the *facing* category, objects were again arranged linearly, but the reference viewer was positioned at the center, and questions asked which object the reference viewer was facing. For all scenes, we injected noise into the camera position, object positions, and the facing directions of objects (except for the reference viewer), in order to generate diverse scenes.

Figure S3. Examples of 3DSRBench. This dataset consists of three spatial relation categories: *left/right*, *visibility* and *facing*.

3DSRBench. This dataset is a real-world dataset composed of allocentric spatial reasoning questions covering various spatial relationship categories [8]. As illustrated in Figure S3, we constructed an allocentric spatial reasoning dataset using the *left/right*, *visibility*, and *facing* categories from this dataset. The *left/right* and *facing* categories were used directly from the original dataset. For the *visibility* category, we modified the existing *front/behind* category such that questions with the answer *front* were labeled as *visible*, and those with the answer *behind* were labeled as *not*.

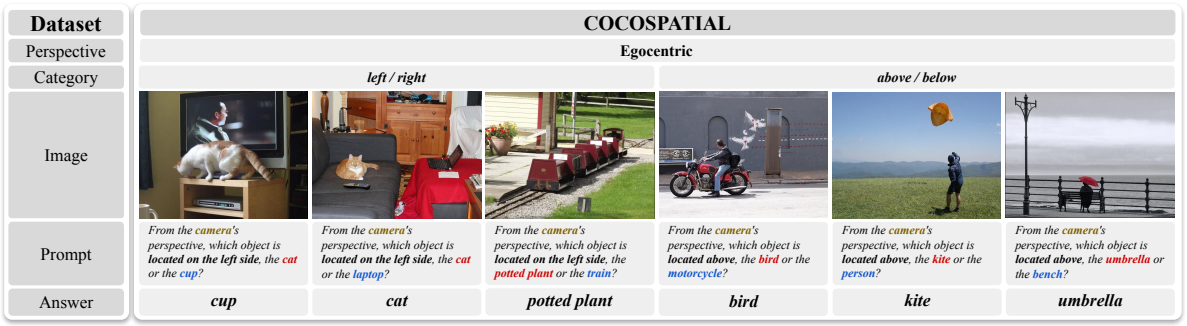


Figure S4. Examples of COCOSPATIAL. This dataset consists of two spatial relation categories: *left/right* and *above/below*.

COCOSPATIAL. This dataset is a real-world dataset that classifies spatial relations between various object pairs based on the positions of their bounding boxes in each image [10]. We refined this dataset to construct an egocentric spatial reasoning dataset that consisted of the *left/right* and *above/below* categories (see Figure S4). Specifically, for each image, we randomly selected one object pair from the *good pairs* with clearly defined spatial relations to form a question. Each question was designed to ask which object is located on the left side for the *left/right* category or on the upper side for the *above/below* category.

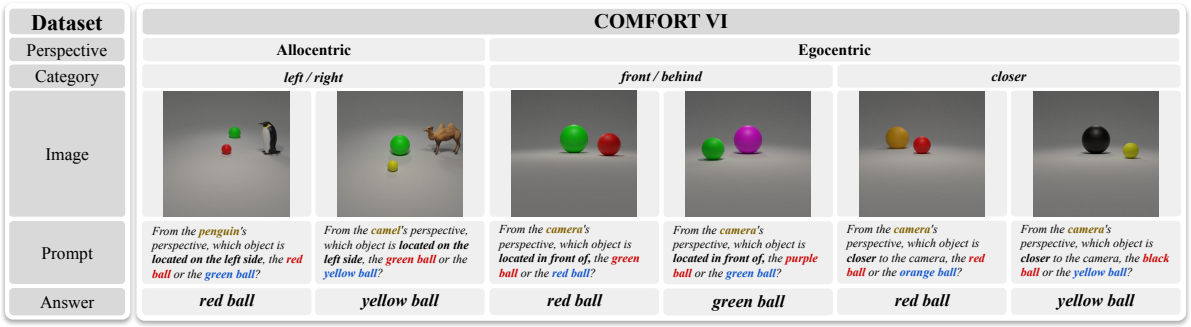


Figure S5. Examples of COMFORT VI. This dataset consists of three spatial relation categories: *left/right* for allocentric spatial reasoning, and *front/behind*, *closer* for egocentric spatial reasoning.

COMFORT VI. Figure S5 visualizes a spatial reasoning dataset designed for visual illusion scenarios, which was generated in the same Blender-based simulation environment as COMFORT#. The dataset consisted of the *left/right* category for allocentric spatial reasoning and the *front/behind* and *closer* categories for egocentric spatial reasoning. Each scene contained two spheres of different colors and sizes placed in 3D space. For the *left/right* category, a reference viewer was positioned to face the spheres from either the left or right side. To simulate visual illusion conditions, the sphere located farther from the camera was rendered significantly larger than the closer one, making the distant sphere appear larger in the image.

COMFORT Multi. As illustrated in Figure S6, COMFORT Multi was constructed in a Blender-based simulator to evaluate whether models can perform consistent allocentric spatial reasoning in the same scene from multiple viewpoints. For each category, we first created 10 scenes, and for each scene we extracted data from 20 viewpoints by varying the camera azimuth by 72° and the polar angle by 15° (see Figure S7). All viewpoints captured from the same scene shared an identical prompt as input for inference. The dataset was organized over the four categories: *left/right*, *closer*, *visibility*, and *facing*. The environment configuration defined for each category was identical to that of COMFORT#.

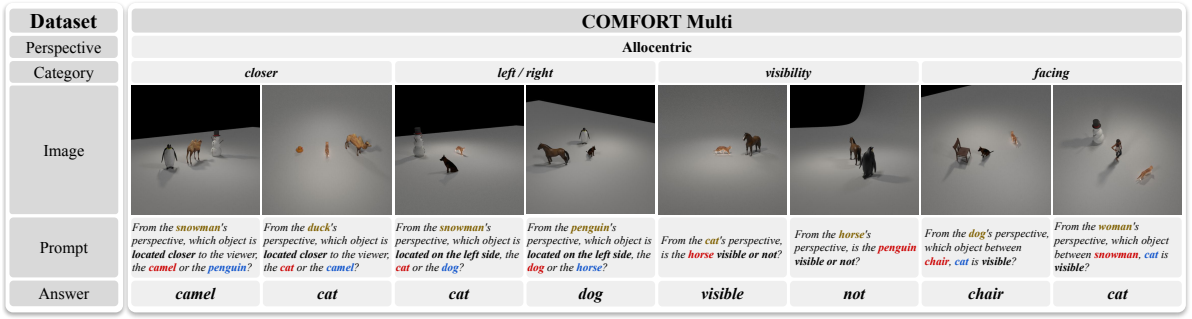


Figure S6. Examples of COMFORT Multi. This dataset consists of four spatial relation categories: *closer*, *left/right*, *visibility* and *facing*.

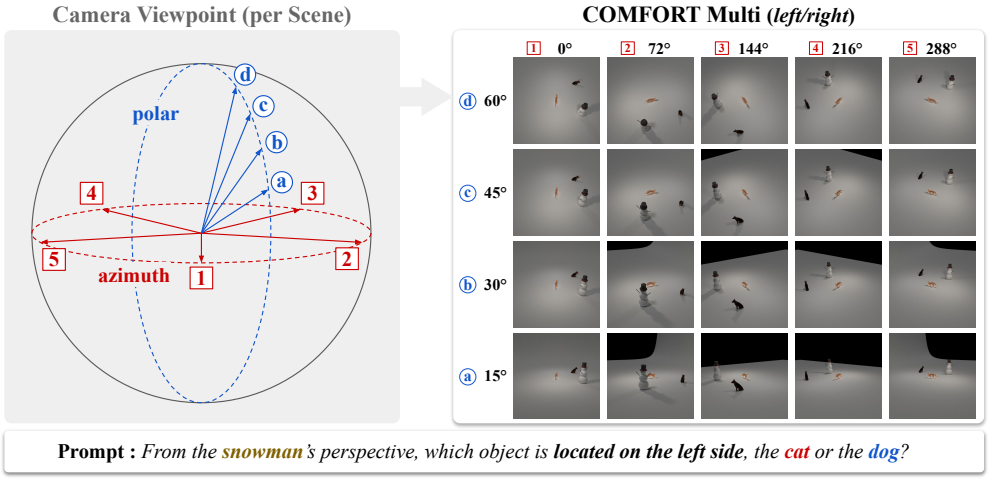


Figure S7. Dataset structure of COMFORT Multi. For each scene, on a spherical coordinate system centered at the scene, we captured 20 images by moving the camera viewpoint in steps of 15° in polar angle and 72° in azimuth.

B.3. Perspective-Aware Spatial Reasoning Examples

We evaluated the performance of the SymPL framework on a variety of spatial reasoning tasks using the COCOSPATIAL and COMFORT VI datasets, and the corresponding qualitative results were shown in Figure S8 and S9.

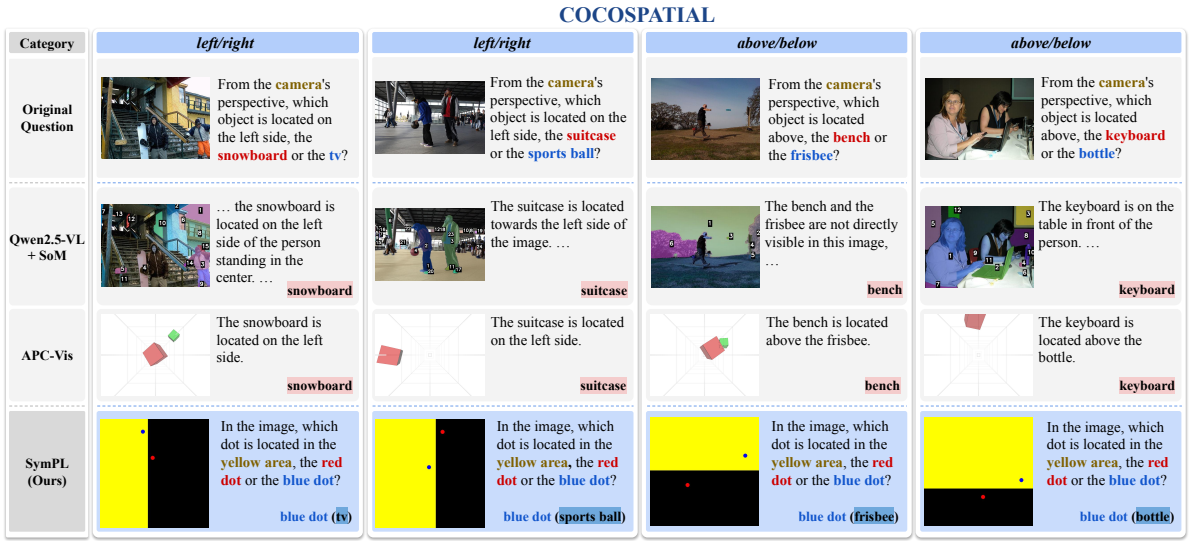


Figure S8. Egocentric spatial reasoning examples in the COCOSPATIAL dataset.

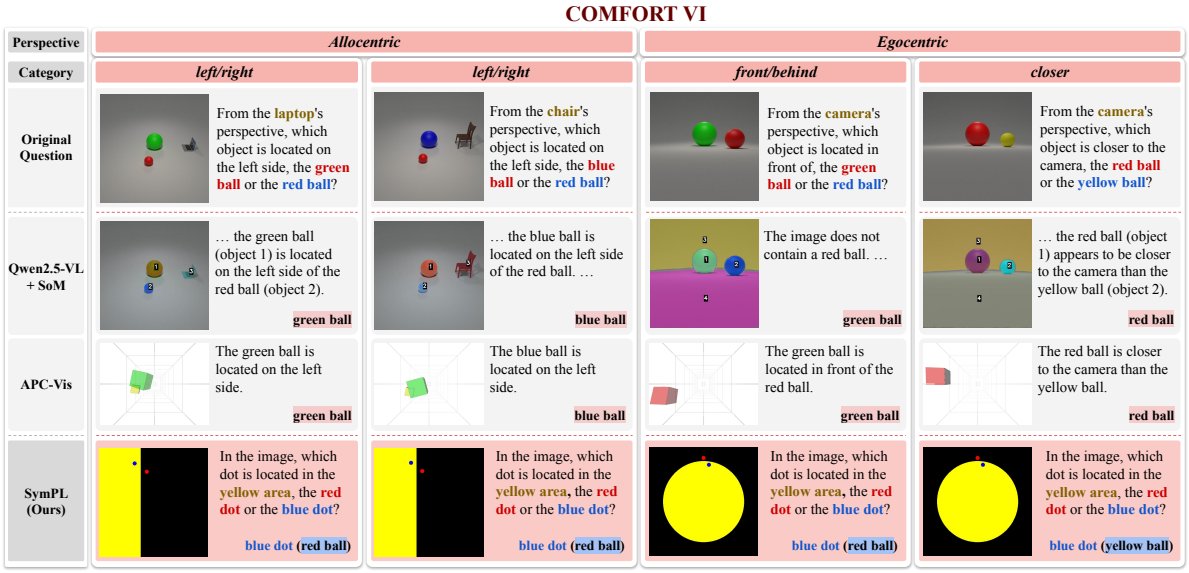


Figure S9. Perspective-aware spatial reasoning examples in the COMFORT VI dataset.

B.4. Ablation Study Details

B.4.1. Analysis of Each Key Factor

To validate the effectiveness of the four key factors (projection, abstraction, bipartition, and localization), we conducted separate analyses for each of them. All experiments were conducted separately for each of the five general purpose VLMs: Qwen2.5-VL [1], GPT-5 [9], LLaVA-NeXT [6], LLaVA-OneVision [5], and Molmo [3].

Projection. To examine how viewpoint affected spatial reasoning performance, we considered two categories of spatial relations: *left/right* and *above/below*. For each scene, we collected images by moving the camera in 10° increments: from a front view to a top view for the *above/below* relations, and from a top view to a side view for the *left/right* relations. For each category, we used data captured from 100 scenes. In each experiment, we used the same question format: for *left/right*, the question asked which object was located on the left side, and for *above/below*, it asked which object was located above.

Figure S10 shows that, for both categories, performance improved when the viewpoint was approximately orthogonal to the plane in which the spatial relation was expressed. Specifically, performance tended to increase as the viewpoint approached the front view for *above/below* relations and the top view for *left/right* relations. These findings indicated that the optimal viewpoint for spatial reasoning should depend on the type of spatial relation, and that this viewpoint was orthogonal to the plane in which the spatial relation was expressed.

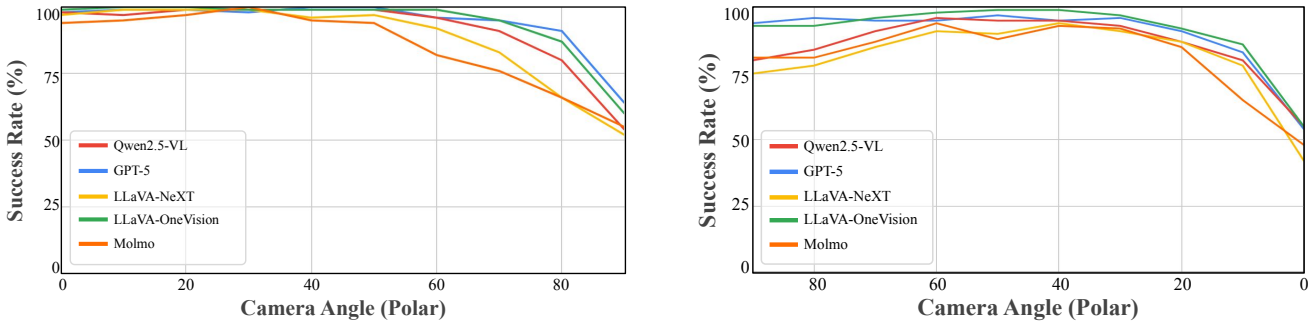


Figure S10. Spatial reasoning performance as a function of viewpoint position. The experiments were conducted on five general purpose VLMs. (a) shows the results for the *above/below* category as the viewpoint moves from front to top view, and (b) shows the results for the *left/right* category as it moves from top to side view.

Abstraction. We analyzed whether our abstraction factor had a meaningful effect on the reasoning performance of VLMs by evaluating performance under different forms of object representation. The experiment was conducted on a simple scene where three objects were arranged in a row, using a *closer* task that asked which object was closer to the rightmost object. For the analysis, we used 100 images for each of three conditions: the original image (original), the original image with segmentation masks annotated on the objects (seg mask), and an image in which the objects were abstracted into dot-shaped symbols (abstraction). When performing inferences with the abstraction image, we modified the prompt by replacing the original object names with the names of the abstracted symbols.

As shown in Figure S11, we observed that, for most models, the reasoning performance on abstraction images tended to be higher. This suggested that abstract representations of objects had a more positive effect than the original images and the segmentation masks commonly used for visual prompting.

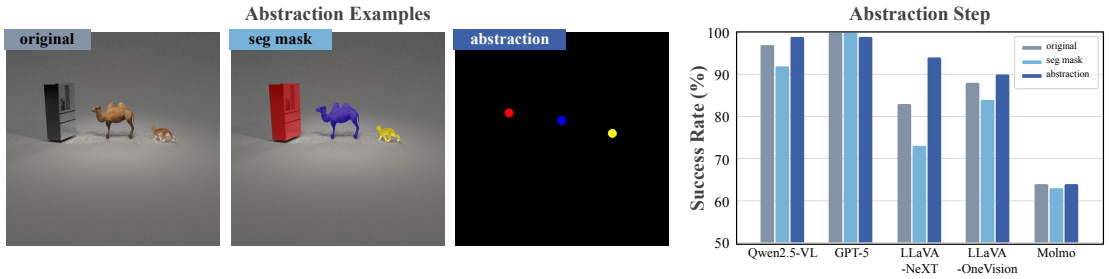


Figure S11. Spatial reasoning performance as a form of abstraction. The experiments were conducted on five general purpose VLMs. For each model, the results are shown from left to right in the following order: experiments on the original image (original), on the image with an annotated segmentation mask (seg mask), and on the abstracted image (abstraction).

Bipartition. We conducted an experiment to analyze whether partitioning the image was effective and how the number of partitions affected spatial reasoning performance. The experiment was performed using images generated by fixing the coordinates of each symbol and increasing the number of partitions from 1 to 4. For each task, we used 100 images, and all tasks were formulated as questions asking which symbol was closer to the yellow dot.

From the experimental results shown in Figure S12, for most VLMs, reasoning performance was higher when the image was partitioned (partition 2, 3, or 4) than when it was not (partition 1). This suggested that dividing the space into partitions provided useful guidelines that help VLMs reason about spatial relations. Regarding the number of partitions, performance was highest when the image was divided into two or three partitions, while four partitions caused a slight performance drop in some models, although it still remained above the partition 1 case. Overall, these trends indicated that the presence of partitions had a more direct impact on spatial reasoning than the exact number of partitions.

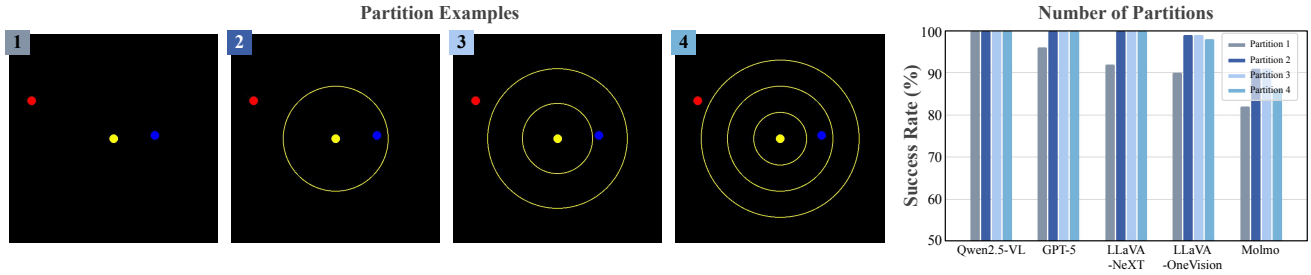


Figure S12. Spatial reasoning performance as a function of partition numbers. The experiments were conducted on five general purpose VLMs. For each model, the results are shown from left to right in the following order: images with 1, 2, 3, and 4 partitions.

Localization. Following the ablation conducted in the *bipartition* step, we further analyzed how the number of color-coded regions affected reasoning performance on the localization question by conducting additional experiments. The experimental data consisted of images generated by fixing the coordinates of the symbols and varying the number of differently colored regions from 2 to 4. For each task, we used 100 images, and each task was formulated as a question asking for the color of the region in which one of the two symbols in the image was located.

From the experimental results shown in Figure S13, we observed that reasoning performance consistently decreased across all VLMs as the number of color-coded regions increased. This suggested that, when answering the localization problem, VLMs performed better when the image was divided into fewer color-coded regions. Taken together with the ablation results from the *bipartition* step, these trends indicated that first splitting the space into two regions and then distinguishing them with different colors provided an effective way to solve the localization problem.

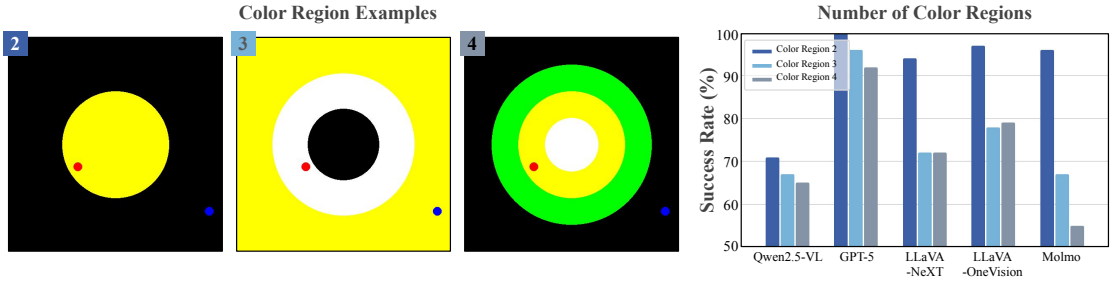


Figure S13. Spatial reasoning performance as a function of colored region numbers. The experiments were conducted on five general purpose VLMs. For each model, the results are shown from left to right in the following order: images with 2, 3, and 4 color regions.

B.4.2. Ablation on the Effectiveness of the Key Factor

Figure S14 presents the results of the ablation study, evaluating the reasoning performance of VLMs on the effectiveness of four key factors. In order, the graphs corresponded to the *left/right*, *closer*, *visibility*, and *facing* categories. We conducted experiments using five general purpose VLMs: Qwen2.5-VL, GPT-5, LLaVA-NeXT, LLaVA-OneVision, and Molmo. The experiments were conducted sequentially from Setting 1 to Setting 5. Each setting was obtained from the original allocentric question by sequentially applying the following factors: *projection*, *abstraction*, *bipartition*, and *localization*. Among these, the images in Setting 1 and Setting 2 were rendered directly in the simulation environment, while the remaining images were generated based on the object coordinates from the simulator.

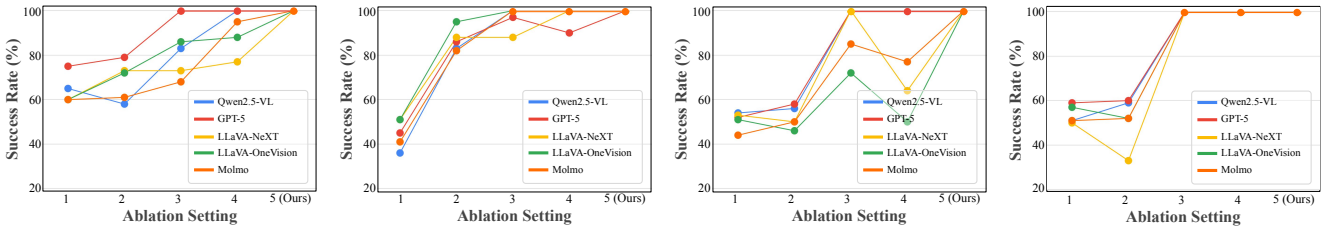


Figure S14. Reasoning performance of five general purpose VLMs on the effectiveness of four key factors. (a) *closer*, (b) *left/right*, (c) *visibility*, and (d) *facing*.

From the experimental results, we observed that the reasoning performance, which was low in Setting 1 (allocentric question), gradually increased as the key factors were incrementally introduced, and eventually reached 100% in Setting 5 (symbolic-layout question). This trend consistently appeared regardless of the type of general purpose VLM. These results showed that the robust reasoning capability of the proposed symbolic-layout question, based on the four key factors, was generally preserved across diverse VLMs. Examples and qualitative results for each setting are shown in Figure S15, S16, S17, and S18.

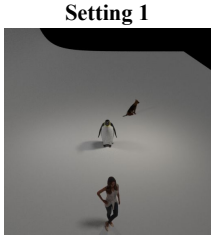
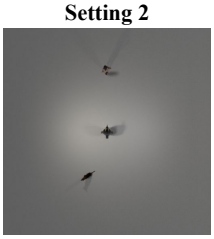
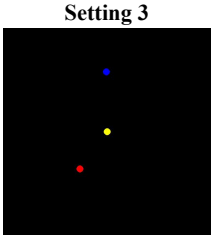
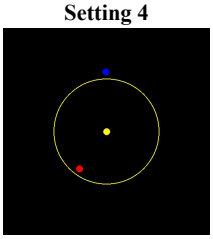
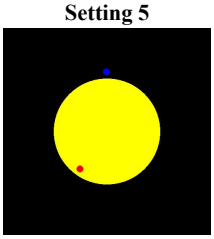
	Setting 1	Setting 2	Setting 3	Setting 4	Setting 5
Image					
Prompt	*From the **penguin**'s perspective, which object is **located closer**, the **dog** or the **woman**?*	*From the **penguin**'s perspective, which object is **located closer**, the **dog** or the **woman**?*	*Which dot is **located closer** to the **yellow dot**, the **red dot** or the **blue dot**?*	*Which dot is **located closer** to the **yellow dot**, the **red dot** or the **blue dot**?*	*Which dot is **located in** the **yellow area**, the **red dot** or the **blue dot**?*
Qwen2.5-VL	***dog***	***dog***	***blue dot***	***red dot***	***red dot***
GPT-5	***woman***	***dog***	***red dot***	***red dot***	***red dot***
LLaVA-NeXT	***woman***	***dog***	***blue dot***	***red dot***	***red dot***
LLaVA-OneVision	***woman***	***woman***	***red dot***	***blue dot***	***red dot***
Molmo	***woman***	***dog***	***red dot***	***red dot***	***red dot***

Figure S15. The qualitative results of the ablation study for the *closer* category.

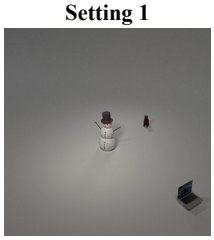
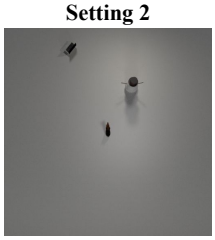
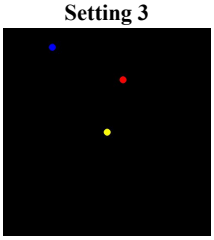
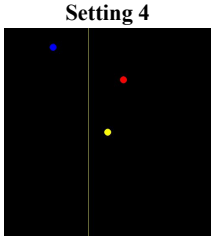
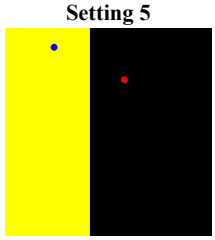
	Setting 1	Setting 2	Setting 3	Setting 4	Setting 5
Image					
Prompt	*From the **dog**'s perspective, which object is **located on the left side**, the **snowman** or the **laptop**?*	*From the **dog**'s perspective, which object is **located on the left side**, the **snowman** or the **laptop**?*	*Which dot is **located on the left side** of the **yellow dot**, the **red dot** or the **blue dot**?*	*Which dot is **located on the left side** of the **yellow dot**, the **red dot** or the **blue dot**?*	*Which dot is **located in** the **yellow area**, the **red dot** or the **blue dot**?*
Qwen2.5-VL	***snowman***	***laptop***	***blue dot***	***blue dot***	***blue dot***
GPT-5	***snowman***	***laptop***	***blue dot***	***blue dot***	***blue dot***
LLaVA-NeXT	***snowman***	***laptop***	***red dot***	***red dot***	***blue dot***
LLaVA-OneVision	***snowman***	***laptop***	***blue dot***	***blue dot***	***blue dot***
Molmo	***snowman***	***laptop***	***blue dot***	***blue dot***	***blue dot***

Figure S16. The qualitative results of the ablation study for the *left/right* category.

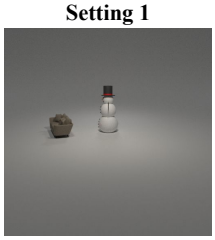
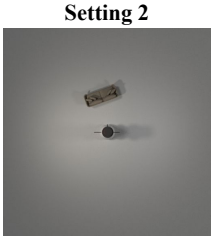
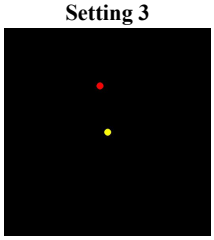
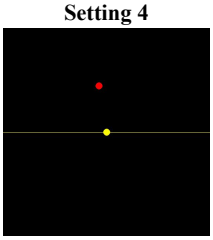
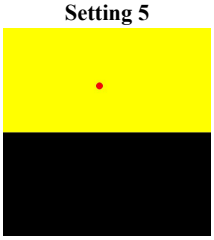
	Setting 1	Setting 2	Setting 3	Setting 4	Setting 5
Image					
Prompt	*From the **snowman**'s perspective, is the **sofa** **visible or not**?*	*From the **snowman**'s perspective, is the **sofa** **visible or not**?*	*Is the **red dot** **located above** the **yellow dot** or not?*	*Is the **red dot** **located above** the **yellow dot** or not?*	*Is the **red dot** located in the **yellow area** or the **black area**?*
Qwen2.5-VL	***visible***	***not***	***above***	***above***	***yellow area***
GPT-5	***visible***	***not***	***above***	***above***	***yellow area***
LLaVA-NeXT	***not***	***not***	***above***	***above***	***yellow area***
LLaVA-OneVision	***not***	***not***	***not***	***not***	***yellow area***
Molmo	***not***	***not***	***not***	***above***	***yellow area***

Figure S17. The qualitative results of the ablation study for the *visibility* category.

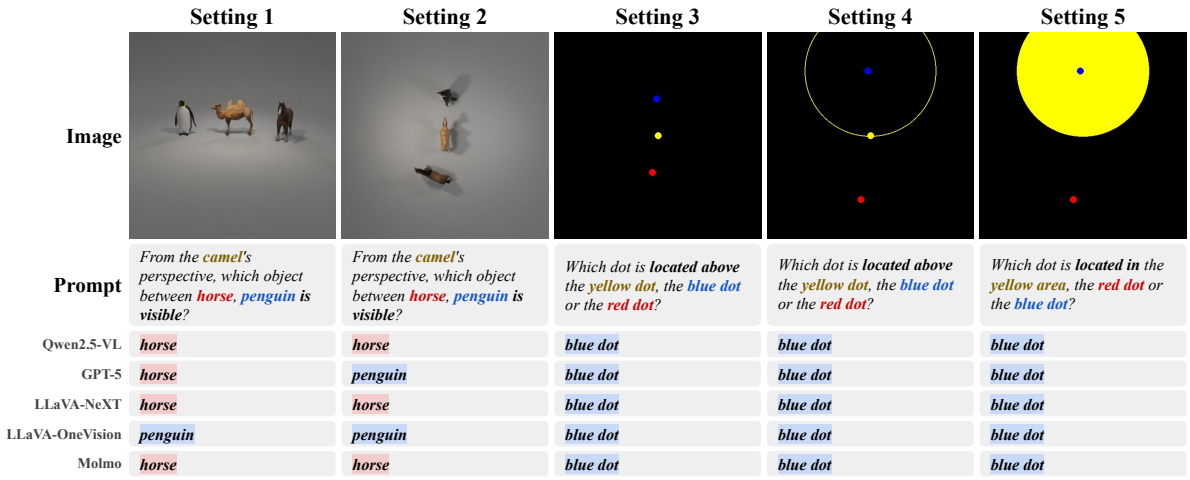


Figure S18. The qualitative results of the ablation study for the *facing* category.

B.5. Applying SymPL to Other VLM

We evaluated whether applying the SymPL framework with a VLM other than Qwen2.5-VL also led to consistent performance gains. To do this, we replaced Qwen2.5-VL with GPT-5 throughout the reasoning pipeline and analyzed allocentric spatial reasoning performance.

According to the results in Figure S19, similar to the trend observed with Qwen2.5-VL, applying the SymPL framework with GPT-5 improved performance across all categories. These results indicate that the proposed framework is not limited to the Qwen2.5-VL model alone.

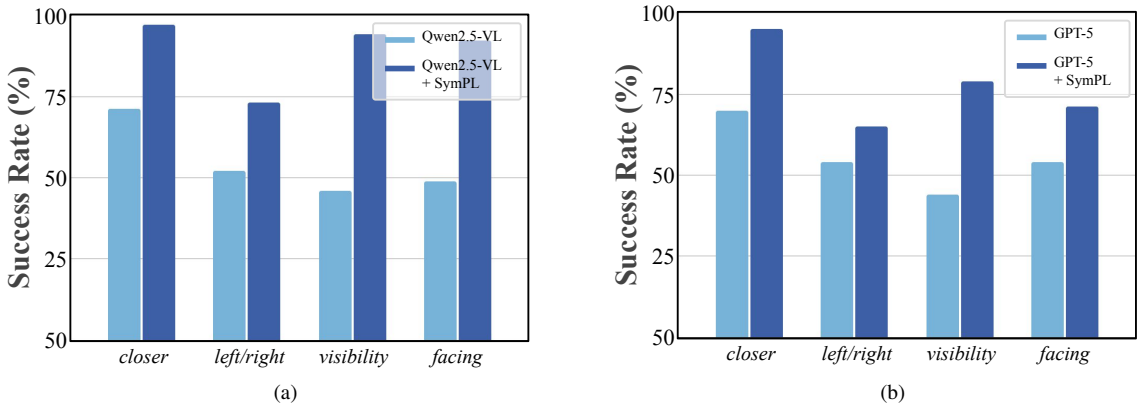


Figure S19. Allocentric spatial reasoning performance with applying SymPL to GPT-5. (a) shows the performance of Qwen2.5-VL before and after applying SymPL, and (b) shows the corresponding results for GPT-5.

C. Reasoning Prompt

The SymPL framework includes a stepwise procedure that utilizes the reasoning capability of the VLM at each step. This procedure consists of five steps: target object selection, detection refinement, reference viewer selection, category determination, and symbolic-layout reasoning. The text prompts used for reasoning in each step are as follows.

Prompt – Target Object Selection

 # Situation Description
Given a spatial reasoning question, please return all the words the represent the entities that are included in the question.

 # Example
[Question] From the old man's perspective, is the person wearing a hat on the left of the green car?
[Detect] [old man, person wearing a hat, green car]

 # Your Task
Now, given the question below, please identify the entities that are included in the question.
All the results return as a format [Detect] [object_1, object_2, ...].

[Question] {question}
[Detect]

Prompt – Detection Refinement

The input images are the cropped regions from the original image that correspond to description : 'category'.
Look at each of these images and select the one that best matches description : 'category'.

Your response should return only the index number of the image you selected.
Note : If multiple images are considered a match, select the one with the lowest index number.

Prompt – Reference Viewer Selection

 # Situation Description
Given a question about spatial reasoning, we want to extract the **perspective** of the question.
If the question is from the camera's perspective or cannot mention the perspective, return ++camera++. Never return anything else.

 # Example
[Question] If I stand at the shepherd's position facing where it is facing, is the sheep visible or not
[Perspective] ++shepherd++

 # Your Task
Given the question below, please specify the **perspective** from which the question is asked.
After ''[Perspective]'' at the end of this prompt, you must return the answer for the base object in the ''object_name'' field, following the format : ++object_name++
''object_name'' must be selected only from the [Option] list provided below.
Never return any answer outside of these options.
Just include ++ in front of and behind of the selected ''object_name'' candidate. Never change anything else.

[Question] {question}
[Options] {obj_str}, camera
[Perspective]

Prompt – Category Determination

 # Situation Description
Given a question about spatial reasoning, we want to extract the category of the question.
The words inside ** ** in the [Question] are the key elements of that [Category].
Depending on the expression, words such as ‘‘visible’’ or ‘‘facing’’ may appear in [Question].
However, the mere presence of these words does not determine that [Category] should be ‘‘visibility’’ or ‘‘facing.’’
Refer to the parts highlighted with ** ** in the examples and select the most appropriate [Category].

 # Example
[Question] If I stand at the man in cowboy hat’s position facing where it is facing, is the bus stop **on the left or right** of me?
[Category] --left_right--

 # Your Task
Given the question below, please specify the category from which the question is asked.
You must return in the format: [Category] --category_name--
‘‘object_name’’ is selected from [Options] below.
Never return a response that is not included in the given options.
Never change the format and capitalization from the option when returns response.

[Question] {question}
[Options] visibility / left_right / facing / closer / above_below / front_behind
[Category]

Prompt – Symbolic-Layout Reasoning (left/right1, visibility)

This is an image of a simple 2D Scene.

 # Task
Based on the image, please answer the following question.

[Question] In the image, is the {obj} dot located in the ‘yellow’ area or the ‘black’ area?

Please only return the answer.

Prompt – Symbolic-Layout Reasoning (left/right2, closer, facing, front/behind, above/below)

This is an image of a simple 2D Scene.

 # Task
Based on the image, please answer the following question.

[Question] In the image, which dot is located in the ‘yellow’ area, the {obj_1} dot or the {obj_2} dot?

Please only return the answer.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 5

[2] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R. Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *International Conference on Learning Representations*, 2025. 1

[3] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104, 2025. 5

[4] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM international conference on multimedia*, pages 11198–11201, 2024. 2

[5] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 5

[6] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 5

[7] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *European Conference on Computer Vision*, 2023. 1

[8] Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6924–6934, 2025. 3

[9] OpenAI. Gpt-5 system card. Technical report, OpenAI, 2025. Version as of Aug 2025. 5

[10] Kanchana Ranasinghe, Satya Narayan Shukla, Omid Poursaeed, Michael S Ryoo, and Tsung-Yu Lin. Learning to localize objects improves spatial reasoning in visual-llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12977–12987, 2024. 3

[11] Zheyuan Zhang, Fengyuan Hu, Jayjun Lee, Freda Shi, Parisa Kordjamshidi, Joyce Chai, and Ziqiao Ma. Do vision-language models represent space and how? evaluating spatial frame of reference under ambiguities. In *The Thirteenth International Conference on Learning Representations*, 2025. 2