REASONMAP: Towards Fine-Grained Visual Reasoning from Transit Maps

Supplementary Material

Appendix

We provide a comprehensive overview in the Appendix, covering key details of our dataset, methodology, evaluation, training baseline, analysis, and further discussions. Specifically, we include the question templates, quality control details, a fine-grained taxonomy of difficulty, and sources of transit maps from 30 cities for REASONMAP construction in Appendix A. We then report detailed descriptions of the evaluation algorithm, experimental setup, and GRPO training in Appendix B. In Appendix C, we include supplementary results and conduct more experiments, including supplementary results, evaluation of symbolic representation and an ablation study about languages. We also provide the results of fine-grained error analysis metrics and systematically analyze failure causes. In Appendix D, we further extend case analysis by providing more classical cases. In addition, we further discuss the stated limitations, future directions, and potential broader impacts of our work in Appendix E. We finally present public implementation for the MLLMs used in our experiments, LLM usage statement, and ethical statement (see Appendix F).

A. Dataset Construction Details

A.1. Question Template Summary

We present one short question template and two long question templates as follows.

Short Question Template

According to the subway map, how do I get from `[Stop 1]` to `[Stop 2]`? Provide only one optimal route, with only the line name and the departure and arrival stations. The format should be strictly followed:

```
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
--
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
```

Long Question Template 1

According to the subway map, how do I get from `[Stop 1]` to `[Stop 2]`? Provide only one optimal route, and include the number of via stops for each route section (excluding the departure and arrival stops). The format should be strictly followed:

```
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
Number of Via Stops: x
--
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
Number of Via Stops: x
```

Long Question Template 2

According to the subway map, how do I get from `[Stop 1]` to `[Stop 2]`? Provide only one optimal route, including all the via stops. The format should be strictly followed:

```
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
Via Stops: xx Station, xx Station
--
Route Name: Line x
Departure Stop: xx Station
Arrival Stop: xx Station
Via Stops: xx Station
```

A.2. A More Fine-grained Taxonomy of Difficulty

Beyond the easy, middle, and hard categorization for map and question difficulty, we provide three additional difficulty aware labels: 1) $city\_line\_count$, the total number of lines in a city (*i.e.*, a proxy for map difficulty); 2) $city\_transfer\_count$, the total number of transfer stations in a city (*i.e.*, a proxy for map difficulty); and 3) $question\_transfer\_count$, the number of transfers in the queried route (*i.e.*, a proxy for question difficulty). These labels enable fine-grained category design and filtering in subsequent analyses.

A.3. Quality Control Details

Our quality control combines automated checks with manual refinement. Specifically, we first validate route correctness (*e.g.*, start stop, arrival stop, and connectivity), followed by manual checks to ensure visual consistency (*i.e.*, routes can be inferred from maps). Questions or GTs with issues are corrected or removed. Three domain experts reviewed the data and identified an error rate of $\sim$16%, after which all questions/GTs were corrected and verified to be accurate. Finally, we systematically adjust the difficulty distributions to prevent bias and ensure a balanced evaluation benchmark.

A.4. Map Source

We provide the sources of all maps included in REASONMAP for further reference (Table A1).

Table A1. Source links to the city transit maps used in the REASONMAP dataset. We present a total of 30 cities sourced from 13 countries.

City	Source	City	Source	City	Source
Budapest	[Link]	Oslo	[Link]	Rome	[Link]
Lisboa	[Link]	Geneva	[Link]	Dubai	[Link]
Auckland	[Link]	Sydney	[Link]	Singapore	[Link]
Kuala Lumpur	[Link]	Los Angeles	[Link]	Miami	[Link]
New York	[Link]	Toronto	[Link]	Washington	[Link]
Guiyang	[Link]	Shanghai	[Link]	Huhehaote (Hohhot)	[Link]
Nanchang	[Link]	Nanning	[Link]	Shenzhen	[Link]
Hangzhou	[Link]	Dalian	[Link]	Kunming	[Link]
Hefei	[Link]	Beijing	[Link]	Changzhou	[Link]
Jinan	[Link]	Xi'an	[Link]	Changshang	[Link]

B. Details of Evaluation and Training Baseline

B.1. Correctness and Quality Evaluation

We present the detailed algorithms for evaluating answer correctness and quality (Algorithm 1 for correctness evaluation and Algorithm 2 for quality evaluation).

For matching (*e.g.*, $=$) in the algorithms, we apply rule-based corrections on top of string matching to account for semantically irrelevant formatting variations (*e.g.*, Line 1" $=$ Route 1" $=$ "1"), preventing evaluation failures caused solely by stylistic or linguistic differences. These corrections are deliberately limited to remain consistent with the format requirements of each question.

Additionally, for multilingual maps, the pipeline is identical to that of English maps, except that route and station names are retained in their local language. During evaluation, we accept both the local language and its English translation as correct, prioritizing semantic correctness.

Algorithm 1: Correctness Evaluation

```
Initialize acc ← 1;
if departure stop of first segment ≠ stop_1 or arrival
  stop of last segment ≠ stop_2 then
    acc ← 0;
foreach segment in predicted route do
    if route name not in the Metro Data then
        acc ← 0;
    if departure or arrival stop not in the stop list of
      the route then
        acc ← 0;
    if not the last segment then
        if arrival stop of current segment ≠
          departure stop of next segment then
            acc ← 0;
return acc
```

B.2. High-Resolution Image Preprocessing.

We compare how different Multimodal Large Language Models (MLLMs) handle high-resolution image inputs in Table A2. Specifically, we examine three key components in their preprocessing pipelines: dynamic resolution han-

Algorithm 2: Quality Evaluation

Initialize map_score ← 0;
if *departure stop of first segment = stop*$_1$ ***and*** *arrival stop of last segment = stop*$_2$ **then**
  map_score ← map_score + 1;
  /* *Long-question-specific part* */
  Initialize $\mathcal{V}_{\text{union}}$, $\mathcal{V}_{\text{intersection}}$ ← ∅;
  Initialize via_stop_score, num_via_stop_score ← 0;
  foreach *segment pair (answer route, reference route)* **do**
    if *answer route name = reference route name* **then**
      map_score ← map_score + 2;
    if *answer departure stop = reference departure stop* **then**
      map_score ← map_score + 1;
    if *answer arrival stop = reference arrival stop* **then**
      map_score ← map_score + 1;
    /* *Long-question-specific part* */
    Calculate absolute difference (*error*) in the number of via stops;
    num_via_stop_score ← num_via_stop_score + max(0, 4 − *error*/ max(number of answer via stops, number of reference via stops) × 4);
    if *answer route name = reference route name* **then**
      Update $\mathcal{V}_{\text{union}}$, $\mathcal{V}_{\text{intersection}}$ with answer and reference via stops respectively;
    via_stop_score ← via_stop_score + number of correctly matched via stops;
  /* *Long-question-specific part* */
  via_stop_score ← min(10, via_stop_score);
  num_via_stop_score ← min(10, num_via_stop_score);
  via_stop_score ← average($|\mathcal{V}_{\text{intersection}}|/|\mathcal{V}_{\text{union}}| \times 10$, via_stop_score)
  map_score ← map_score + Option(via_stop_score *or* num_via_stop_score);
/* 10 for short question; 20 for long question */
map_score ← min(10, map_score)/*min*(20, map_score) ;
if *correctness evaluation (acc) = 1* **then**
  map_score ← map_score + 10/map_score + 20;
return map_score;

dling, positional encoding, and token compression.

1. **Dynamic resolution handling** refers to whether the model can directly accept images of arbitrary sizes without resizing or cropping. Most recent models support native resolution processing, enabling them to preserve fine-grained spatial information. In contrast, models like Gemini [9] rely on image tiling and resizing to fit fixed input constraints.
2. **Positional encoding** helps the model retain spatial structure among visual tokens. Common strategies include 2D Rotary Positional Encoding (2D-RoPE) [12], as seen in Qwen2.5-VL [2] and Doubao [3], or flexible alternatives like V2PE [8] in InternVL3 [41]. Some models (*e.g.*, Gemini, Skywork-R1V [24, 34]) do not explicitly disclose their positional encoding scheme, which we mark as "–" in the table.
3. **Token compression** aims to reduce the number of visual tokens for more efficient processing. Different models adopt different strategies: Qwen2.5-VL and QVQ [25] compress tokens via 2×2 patch concatenation followed by an MLP; InternVL3 [41] and Kimi-VL [31] utilize spatial transformations like pixel unshuffle or shuffle, also followed by MLPs; Doubao averages over 2×2 patches before projection. Models without token compression may incur higher memory and computation costs when processing high-resolution inputs.

Table A2. Comparison of high-resolution image preprocessing strategies across different MLLMs. We use "−" to denote unspecified or unclear content.

Model	Dynamic Resolution Handling	Positional Encoding	Token Compression
Qwen2.5-VL series [2]	✓	2D-RoPE	✓ (2×2 Concat + MLP)
QVQ-72B-Preview [25]	✓	2D-RoPE	✓ (2×2 Concat + MLP)
InternVL3 series [41]	✓	V2PE	✓ (Unshuffle + MLP)
Kimi-VL series [31]	✓	2D-RoPE	✓ (Shuffle + MLP)
Skywork-R1V-38B [24, 34]	✓	-	✗
Gemini [9]	✗ (Tiling+Resize)	-	✗
Doubao-1.5-Pro series [3]	✓	2D-RoPE	✓ (2×2 Pooling + MLP)

B.3. Details about Difficulty-Aware Weighting.

Each difficulty pair is assigned a predefined weight that reflects its relative challenge level. The full weight matrix is shown below, where the first element in each pair denotes the question difficulty and the second denotes the map difficulty:

Difficulty Pair	Weight
("easy", "easy")	1.0
("medium", "easy")	1.5
("hard", "easy")	2.0
("easy", "medium")	1.5
("medium", "medium")	2.0
("hard", "medium")	2.5
("easy", "hard")	2.0
("medium", "hard")	2.5
("hard", "hard")	3.0

This weighting scheme rewards models more for correctly solving harder question–map combinations, reflecting the increased reasoning complexity they entail, while maintaining moderate differences between buckets to prevent excessive score variance and preserve evaluation stability.

B.4. Details of GRPO RL Training

GRPO [28] extends standard policy gradient methods by normalizing rewards within a sampled group, which stabilizes optimization and encourages relative preference learning. Specifically, given an input x and a group of K sampled outputs $G = \{y_i\}_{i=1}^{K}$ with their corresponding scalar rewards $\{r_i\}_{i=1}^{K}$, the centered group advantage $\hat{A}_i$ is computed as the deviation of each sample's reward from the group mean:

$$\hat{A}_i = r_i - \frac{1}{K}\sum_{j=1}^{K} r_j. \tag{1}$$

The policy parameters θ are then updated to maximize the following objective:

$$\max_{\theta} \mathcal{L}(\theta) = \sum_{i=1}^{K} \hat{A}_i \log \pi_\theta(y_i \mid x), \tag{2}$$

where $\pi_\theta(y_i \mid x)$ denotes the model likelihood of generating y_i under parameters θ. This objective encourages the model to increase the probability of outputs with above-average rewards while suppressing those with below-average ones. In our implementation, the reward r_i is composed of an accuracy component and a format component.

C. Supplementary Experiments

C.1. Supplementary Results

We provide additional visualization results in Figure A1 and A2 to better illustrate our evaluation on REASONMAP as follows. Figure A1 and A2 illustrate the model's accuracy across different difficulty levels and cities, respectively. As shown, accuracy decreases with increasing difficulty and varies considerably across cities.

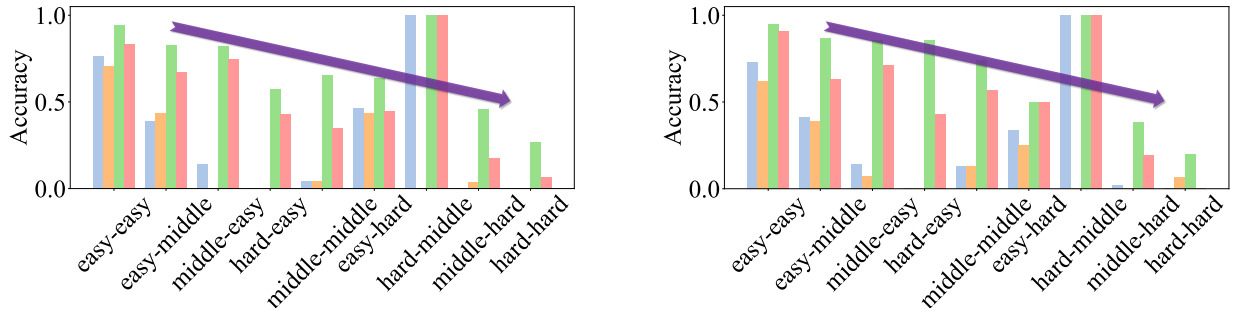
C.2. Fine-grained Error Analysis Metric Summary

We report multiple fine-grained error analysis metrics in Table A3 as follows: (1) $dep - arr\ score$: $+1$ if both the start and end stations are correct; (2) $route\ name\ score$: $+2$ for each correctly identified line name along the route; (3) $stops\ score$: $+1$ for each correctly identified intermediate stop; (4) $num\_via\_stop\_score$ (only for long questions): computed by taking the absolute difference between the number of via stops in the answer and the reference route, and mapping it to a score from 0 to 4; (5) via_stop_score (only for long questions): calculated by averaging the number of correctly matched via stops (up to 10) and the Intersection-over-Union (IoU) between the via stop sets of the answer and reference route (scaled to 10).

We further provide a sensitivity analysis with two additional weighting schemes for the components of the map score (C1 & C2 in Table A4). Based on the results in Tab. A3 & A4, performance ranking remains consistent across different weighting schemes, while all components of the map score exhibit similar increasing or decreasing trends.

C.3. Further Experiments about Languages

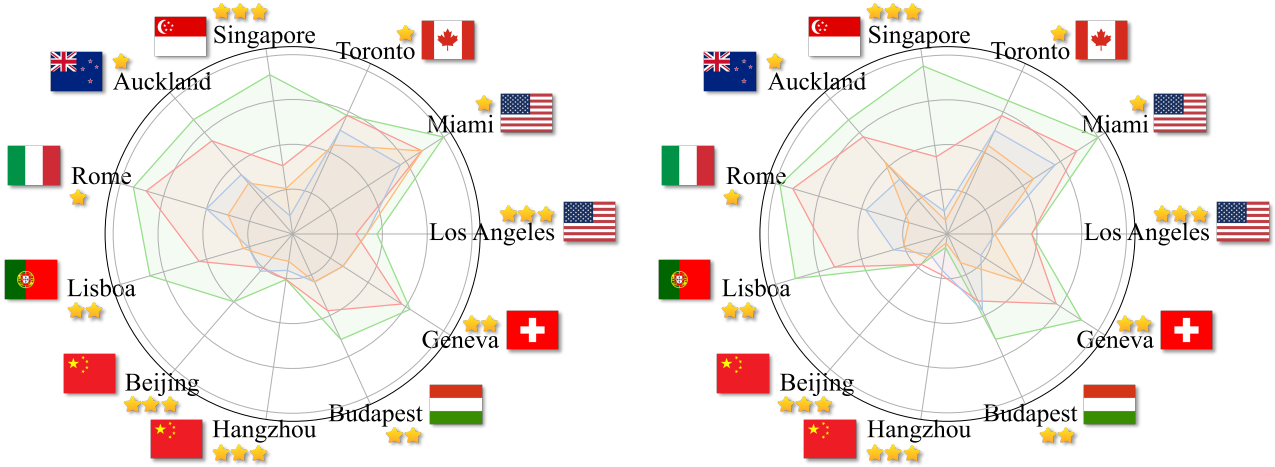
We conduct an ablation study under the textualized representation paradigm (as mentioned in Appendix C.4). In this setting, visual images are not involved, which allows us to safely replace all non-English station names with unique English aliases without introducing visual inconsistencies. This approach isolates the language prior factor and avoids



(a) Accuracy for short questions

(b) Accuracy for long questions

Figure A1. Accuracy across difficulty combinations for four representative MLLMs (Qwen2.5-VL-72B-I, InternVL3-78B, OpenAI o3, and Doubao-415). Each difficulty combination is denoted by a pair (*e.g.*, *easy-hard*), where the first term indicates question difficulty and the second term represents map difficulty. The pair (*hard-middle*) contains only one sample, leading to an accuracy of 100%. We summarize the number of evaluation samples in each difficulty bucket: 55 samples for *easy-easy*, 46 for *easy-middle*, 28 for *middle-easy*, 7 for *hard-easy*, 23 for *middle-middle*, 80 for *easy-hard*, 1 for *hard-middle*, 57 for *middle-hard*, and 15 for *hard-hard*.



(a) Accuracy for short questions

(b) Accuracy for long questions

Figure A2. Accuracy across different cities for four representative MLLMs (Qwen2.5-VL-72B-I, InternVL3-78B, OpenAI o3, and Doubao-415). Each city is marked with the corresponding map difficulty and the country flag. Each city in the test set provides a specific number of samples per model: 32 samples for Auckland, 34 for Los Angeles, 7 for Miami, 35 for Lisboa, 18 for Geneva, 40 for Beijing, 39 for Hangzhou, 17 for Budapest, 39 for Singapore, 40 for Rome, and 11 for Toronto.

any potential confounding effects from visual modifications. Concretely, we manually replace all Chinese station names in Beijing and Hangzhou with unique English station names (*e.g.*, mapping them to New York stops: 'zhichunli' <-> 86 St), preserving the original transit map structure. The results under this setting are as follows.

Overall, we observe from the results in Table A5 that using English labels leads to performance improvements, particularly for long-form questions. This suggests that the model indeed exhibits a language bias, with English showing an advantage over Chinese, which may be attributed to differences in pre-training data distributions.

C.4. Further Experiments about Symbolic Representation of Maps

We conduct further experiments about deterministic baselines derived from symbolic representations of the maps. This setting can serve as a theoretical performance ceiling, independent of perceptual challenges faced by MLLMs. We replace the visual input with symbolic representations extracted from the underlying map structure. Specifically, we convert all routes and station information into textual form to represent the topological structure of the map. This textualized representation is then used for evaluation. Specifically, we provide the model with textualized representations and the question as input, without including any visual maps.

Table A3. Fine-grained error analysis metrics of various MLLMs. $S.$ represents results for short questions, while $L.$ denotes results for long questions. **Bold** indicates the best results among open-source and closed-source models, respectively.

Model	Type	Dep-Arr Score ($S.$ / $L.$)	Route Name Score ($S.$ / $L.$)	Stops Score ($S.$ / $L.$)	Num. Via Stop Score ($L.$)	Via Stop Score ($L.$)
Open-source Models						
Qwen2.5-VL-3B-Instruct [2]	Base	0.86 / 0.78	0.03 / 0.02	1.03 / 0.96	0.42	0.00
Qwen2.5-VL-32B-Instruct [2]	Base	0.95 / 0.92	0.09 / 0.10	1.16 / 1.19	1.57	0.01
Qwen2.5-VL-72B-Instruct [2]	Base	**0.96 / 0.95**	**0.22 / 0.24**	**1.23 / 1.22**	1.56	**0.04**
InternVL3-38B [41]	Base	0.87 / 0.84	0.06 / 0.10	1.08 / 1.12	**1.63**	0.00
InternVL3-78B [41]	Base	0.96 / 0.89	0.15 / 0.17	1.15 / 1.12	1.46	0.01
Kimi-VL-A3B-Instruct [31]	Base	0.89 / 0.88	0.07 / 0.07	1.06 / 1.11	0.91	0.02
Kimi-VL-A3B-Thinking [31]	Reasoning	0.80 / 0.65	0.08 / 0.10	0.99 / 0.79	0.50	0.00
Skywork-R1V-38B [34]	Reasoning	0.60 / 0.62	0.06 / 0.09	0.74 / 0.71	1.00	0.00
QvQ-72B-Preview [25]	Reasoning	0.35 / 0.22	0.03 / 0.02	0.42 / 0.29	0.20	0.01
Closed-source Models						
Doubao-115 [3]	Base	0.78 / 0.96	0.08 / 0.18	1.08 / 1.31	1.94	0.06
OpenAI 4o [22]	Base	0.97 / 0.95	0.22 / 0.29	1.49 / 1.53	2.22	0.04
Doubao-415 [3]	Reasoning	0.98 / **0.98**	**0.33 / 0.30**	1.57 / 1.65	2.37	**0.08**
Doubao-428 [3]	Reasoning	0.73 / 0.75	0.00 / 0.03	1.19 / 1.27	2.27	0.00
Gemini-2.5-Flash [9]	Reasoning	0.93 / 0.67	0.27 / 0.29	1.67 / 1.22	1.82	0.05
OpenAI o3 [23]	Reasoning	**0.99** / 0.91	0.32 / 0.16	**1.77 / 1.73**	**3.31**	0.03

Table A4. Ablation on the map score. MS denotes the map score. C1 uses the avg scheme, while C2 excludes the Dep-Arr score part.

Model	Weighted MS (*S./L.*)	C1 MS (*S./L.*)	C2 MS (*S./L.*)
Qwen2.5-VL-32B-I	3.88 / 6.84	3.85 / 5.80	2.90 / 4.88
Qwen2.5-VL-72B-I	**5.09 / 8.80**	**5.08 / 8.83**	**4.12 / 7.88**
OpenAI GPT-4o	6.84 / 13.57	6.80 / 13.59	5.82 / 12.64
OpenAI o3	**9.53 / 17.96**	**9.38 / 17.95**	**8.39 / 17.04**

By comparing the results in Table A6 with those in Table 2 of the main paper, we observe a clear performance improvement. This is expected, as replacing the visual map with textualized representations substantially reduces task difficulty, as it removes the need to assess visual capabilities such as OCR and grounding. We further note that prior works, such as MapBench [35] and CityBench [5], also focus on visual map interpretation without constructing explicit symbolic baselines.

C.5. Further Systematic Analysis on Failure Causes

Case 1: grounding

Q: Hollywood/Highland Station -> Wilshire/Vermont Station? <image:los-angeles.png>
A (Kimi-VL-A3B-Thinking): ... Let me start by locating both stations on the map. Hollywood/Highland is on ... A Line (✗)... Wilshire/Vermont is on the C Line (✗) ... (✗)
A (Doubao-415): ... First, I need to see where these two sites are located on a map. Hollywood/Highland ... on B Line (✓) ... Wilshire/Vermont ... is on D Line and B Line (✓) ... Route Name: B Line Departure Stop: Hollywood/Highland Station Arrival Stop: Wilshire/Vermont Station (✓)

We systematically analyze failure causes, focusing on three MLLM capabilities pertinent to fine-grained visual reasoning (*e.g.*, OCR, grounding, and spatial reasoning). To assess OCR capabilities, we collect metrics of 9 representative MLLMs on OCRBench [20]. Comparing these with their performance on REASONMAP as shown in Table A7 in the paper, we observe no clear correlation between OCR ability and REASONMAP accuracy. Notably, this trend holds across both open-source and closed-source models, suggesting that stronger OCR performance alone does not lead to better fine-grained visual reasoning. For instance, among open-source models, InternVL3-78B achieves the highest OCRBench scores, but underperforms Qwen2.5-VL-72B-Instruct on REASONMAP.

Case 2: spatial reasoning

Q: Ponte Mammolo Station -> Sta. Maria del Soccorso Station? <image:rome.png>
A (Skywork-R1V-38B): ... Line A: Orange Line B: Blue Line C: Green (✓) ... Ponte Mammolo Station ... on Line B (✓) ... Let me trace the path of Line B from Ponte Mammolo towards the center of Rome (✗) ... But I need to go from Ponte Mammolo to Sta. Maria del Soccorso. Looking at the map, I don't see a direct connection between these two stations via Line B. (✗) So, I might need to transfer to another line ... Let me check if there's another line that connects to Sta. Maria del Soccorso ... (✗)
A (Gemini-2.5-Flash): Route Name: Line B Departure Stop: Ponte Mammolo Station Arrival Stop: Sta. Maria del Soccorso Station Number of Via Stops: 0 (✓)

We further conduct more in-depth case analyses, which reveal that the main causes of failure are grounding and spatial reasoning, as illustrated in the following example. We

Table A5. Evaluations on Beijing and Hangzhou (with and without English). $S.$ represents results for short questions, while $L.$ denotes results for long questions. **Bold** indicates performance improvements, while *italicized* values represent performance degradation.

Model	Beijing (S. / L.)	Beijing (w. English) (S. / L.)	Hangzhou (S. / L.)	Hangzhou (w. English) (S. / L.)
Kimi-VL-A3B-Instruct [31]	36.76% / 17.30%	*23.78%* / **20.81%**	40.00% / 42.22%	**42.22%** / **45.95%**
Doubao-115 [10]	64.86% / 50.51%	*45.95%* / **52.70%**	82.22% / 64.44%	*67.78%* / **65.56%**
Doubao-415 [10]	84.86% / 74.05%	**88.65%** / **85.95%**	94.44% / 97.22%	*87.78%* / **100%**

Table A6. Evaluations of various MLLMs using symbolic representation. $S.$ represents results for short questions, while $L.$ denotes results for long questions. **Bold** indicates the best results among open-source and closed-source models, respectively.

Model	Type	Weighted Acc. (S. / L.)	#Tokens (S. / L.)
Open-source Models			
Qwen2.5-VL-3B-Instruct [2]	Base	22.83% / 19.79%	51 / 162
Qwen2.5-VL-32B-Instruct [2]	Base	25.52% / 18.77%	97 / 297
Kimi-VL-A3B-Instruct [31]	Base	**39.58%** / **34.81%**	43 / 55
Closed-source Models			
Doubao-115 [3]	Base	81.16% / 72.66%	41 / 82
OpenAI 4o [22]	Base	82.38% / 78.91%	40 / 70
Doubao-415 [3]	Reasoning	**95.31%** / **93.66%**	563 / 1561

Table A7. Evaluations of various MLLMs on OCRBench. **Bold** indicates the best results among open-source and closed-source models, respectively. The references in the table indicate the result sources. All results are collected from the technical reports.

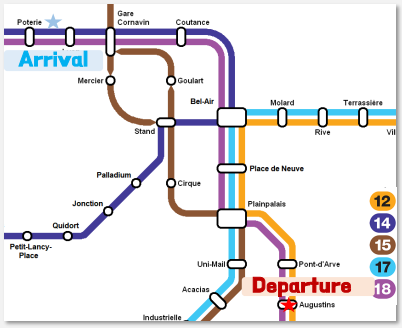
Model	Type	OCRBench
Open-source Models		
Qwen2.5-VL-3B-Instruct [2]	Base	797
Qwen2.5-VL-72B-Instruct [2]	Base	885
InternVL3-38B [41]	Base	886
InternVL3-78B [41]	Base	**906**
Kimi-VL-A3B-Instruct [31]	Base	864
Kimi-VL-A3B-Thinking [31]	Reasoning	864
Closed-source Models		
OpenAI 4o [31]	Base	815
Doubao1.5-VL (non-thinking) [10]	Base	**881**
Doubao1.5-VL (thinking) [10]	Reasoning	861

observe that OCR errors rarely occur, and most failure cases are instead caused by grounding or spatial reasoning issues.

For instance, in Case 1, Kimi-VL-A3B-Thinking incorrectly identifies the line of the departure station, indicating a grounding error that leads to subsequent reasoning failures. In Case 2, Skywork-R1V-38B correctly performs OCR and grounding in the initial steps, but fails in the reasoning stage (*i.e.*, it does not prioritize locating the arrival station and instead attempts to construct incorrect indirect paths). Such failures reflect deficiencies in spatial reasoning, particularly in planning and executing core steps of pathfinding. These cases further indicate that the principal capability gap between open-source and closed-source models lies in grounding and spatial reasoning.

D. Case Analysis

We provide additional case analyses covering both correct and incorrect predictions, along with detailed comparisons of their respective reasoning processes. We first compare Doubao-415 and Doubao-428 (Figure A3), both of which reach the correct destination (from Augustins Station to Poterie Station) but via distinct reasoning paths. Doubao-415 correctly identifies early that both stations are on Line 18 and efficiently converges on the optimal, direct route without transfers. In contrast, Doubao-428 misclassifies Augustins as being on Line 12 and, assuming Poterie is on Line 18, proposes a transfer route via Plainpalais—functionally correct but suboptimal due to unnecessary complexity. Both models engage in extensive self-correction , highlighting the significant downstream impact of early-stage misjudgments. Moreover, visual reasoning limitations persist: despite correctly recognizing Augustins on Line 12, Doubao-415 commits to a transfer path and fails to re-evaluate the possibility of a direct connection. This indicates room for improvement in both early visual grounding and global route optimality awareness. We then analyze the observed pattern when comparing the full input and text-only variants in the case (in Figure A4). The model with full visual access accurately identifies both stations on the Yellow Line and outputs the optimal direct route with the correct number of via stops. In contrast, the text-only variant makes an early misclassification, placing both stations on the Blue Line (Azul) and constructing a plausible but entirely incorrect sequence of intermediate stops. Although the final answer format appears coherent, the underlying logic is flawed due to the initial error in line recognition. This further illustrates the importance of visual input in spatial reasoning tasks, where even minor misinterpretations can lead to fundamentally incorrect conclusions. Additionally, some models, such as the InternVL3 series, default to rejection when visual input is absent. We further present several error cases in Figure A5, where Doubao-415 still exhibits visual confusion. In contrast, Qwen2.5-VL-32B-I, when lacking visual input, behaves differently from the InternVL3 series: rather than rejecting the query outright, it attempts to reason over the available information without producing a final answer, while explicitly notifying the missing visual input.



Q: Augustins Station to Poterie Station + Question Template

Doubao -428

A: Route Name: Line 18
Departure Stop: Augustins Station
Arrival Stop: Poterie Station ...

Doubao -428

A: <think> ... Is Augustins on Line 12? No. Again, is it on Line 12? Still no. Let's check the stations on Line 12. Is Augustins among them? Line 12 includes stations like Blanche, Augustins, Pont-d'Arve...
Is Augustins really on Line 12? ... So Augustins is on Line 12, and Poterie is on Line 18, meaning a transfer is likely needed.
So the route could be: Augustins (Line 12) → Transfer to Line 18 → Poterie. Or maybe there's a direct connection?
No, Augustins is on Line 12, and Poterie is on Line 18. So we must go from Augustins to a transfer station—maybe Place de Neuve? ...
Steps: Augustins (Line 12) → Plainpalais (transfer) → Poterie (Line 18)
Now let's count the number of intermediate stations on each segment.
Line 12: ...
So Augustins is or isn't on Line 12? ...
... (continuous self-correction)
Now, listing all Line 12 stations from Augustins to Plainpalais: ...
Eventually, ...
</think>
Route Name: Line 12 Departure Stop: Augustins Station Arrival Stop: Plainpalais Station ...
Route Name: Line 18 Departure Stop: Plainpalais Station Arrival Stop: Poterie Station ...

Correction
Visual Confusion

Doubao -115

A: <think> First, observe the stations passed by each line. Augustins Station is on Line 14 and Line 18...But Poterie Station is on the purple Line 18...
Wait, looking at the map again, Augustins is on the orange Line 12 and the purple Line 18? No, carefully see the station connections. In the map, Augustins is located on the line that ... Line 18. So Augustins is on Line 18, and Poterie is also on Line 18.
... (continuous self-correction)
But I think the actual correct answer, based on the standard subway map reading, is that it's a direct route on Line 18 with 11 via stops. ... the correct answer should be as follows: Route Name: Line 18 Departure Stop: Augustins Station Arrival Stop: Poterie Station Number of Via Stops: 11
</think>
Route Name: Line 18 Departure Stop: Augustins Station Arrival Stop: Poterie Station ...

Correction
Visual Confusion

Figure A3. Case analysis of various MLLMs using REASONMAP (Case N1). For reasoning models, the reasoning process is explicitly marked with `<think>` and `</think>` tags. We highlight error contents in the answers with red and correct contents in green.

E. Further Discussions

E.1. Limitations and Future Work

While REASONMAP provides a carefully curated benchmark for evaluating fine-grained visual reasoning with high-resolution transit maps, we acknowledge that it represents only one type of structured visual diagram. As such, caution should be taken when generalizing observations to other domains that involve different types of visual content or reasoning styles. Additionally, although efforts were made to ensure diversity across cities and languages, the current version may not fully capture all geographic or linguistic variations. Future iterations could further expand coverage and explore additional forms of reasoning [7] to enhance generality.

Furthermore, we note that GeoGuessr-style localization tasks [11, 14, 21] are compelling, as they emphasize detailed visual understanding of natural scenes and signage. We plan to pair transit maps with street view imagery to support cross-view reasoning and localization within REASONMAP, thereby expanding beyond static map inputs. In parallel, we will explore agent-based training and evaluation that moves from single-turn prediction to iterative planning with feedback, including reward designs for correctness, calibration, and format [39]. Finally, we will extend toward embodied settings [13] where agents perceive and act in interactive environments, enabling assessment of instruction following, route planning, and navigation under real-world constraints. Together, these directions broaden the benchmark from fine-grained visual reasoning to context-aware spatial intelligence and practical decision making.

Our REASONMAP can further evaluate the efficient

Q: Ameixoeira Station to Picoas Station + Question Template

InternVL3 Onlytext

A: r (reject)

Refusal

Doubao -415 Onlytext

A: <think> ... Ameixoeira and Picoas are stations there...Ameixoeira is on the Blue Line (Linha Azul), and Picoas is also on the Blue Line. Let me check the order of stations on the Blue Line. The Blue Line goes from ... Wait, no, actually, from Ameixoeira, the next stations towards the city center would be ... No, wait, Picoas is on the Blue Line between Entre Campos and Marquês de Pombal? Wait, maybe I should list the stations in order. Let's see: ...
... (*continuous self-correction*)
Wait, maybe I got the direction wrong ... Let me confirm: Ameixoeira (departure), next is São Sebastião (1), Campo Grande (2), Alameda (3), Entre Campos (4), then Picoas (arrival). So number of via stops is 4. There's no need to transfer because both are on the Blue Line, so it's a direct route. So the route should be Blue Line, departure Ameixoeira, arrival Picoas, via stops 4. Route Name: Azul Departure Stop: Ameixoeira Station Arrival Stop: Picoas Station Number of Via Stops: 4
</think>
Route Name: Azul...

Visual Confusion

Doubao -415

A: <think> The user is asking for the optimal route from Ameixoeira Station to Picoas Station. First, I need to consult the metro map to determine which lines these stations belong to. Ameixoeira is on the Yellow Line (Amarela), and Picoas—after closely examining the map—is also on the Yellow Line, not at an interchange point.
Starting from Ameixoeira and heading south along the Yellow Line, the stations passed are Lumiar, Quinta das Conchas, Campo Grande, and Saldanha before arriving at Picoas. This gives us 4 via stops (excluding the departure and arrival stations). No transfer is needed, as the entire trip is on the Yellow Line.
Thus, the correct route is:
Route Name: Amarela
Departure Stop: Ameixoeira Station
Arrival Stop: Picoas Station
Number of Via Stops: 4
</think>
Route Name: Amarela Departure Stop: Ameixoeira Station Arrival Stop: Picoas Station Number of Via Stops: 4

Figure A4. Case analysis of various MLLMs using REASONMAP (Case N2). For reasoning models, the reasoning process is explicitly marked with `<think>` and `</think>` tags. We highlight error contents in the answers with red and correct contents in green.

models from multiple efficiency strategies [6, 26, 27, 29, 30, 40]. Additionally, more fields [1, 4, 15–19, 32, 33, 36–38] require corresponding reasoning-centered benchmarks for proper evaluation.

E.2. Broader Impact

Advancing the capabilities of MLLMs in fine-grained visual reasoning has the potential to benefit a wide range of real-world applications, including navigation systems, urban planning tools, and assistive technologies for visually impaired individuals. By offering a structured and rigorous benchmark, REASONMAP encourages the development of MLLMs that can more effectively interpret complex visual artifacts and perform spatial reasoning. This could contribute to the long-term goal of building intelligent agents that interact more naturally and safely with human environments. Furthermore, the dataset's emphasis on high-resolution, globally sourced transit maps promotes research that is inclusive of diverse visual formats and geographic contexts. We hope REASONMAP can serve as a step toward more transparent, robust, and generalizable multimodal systems.

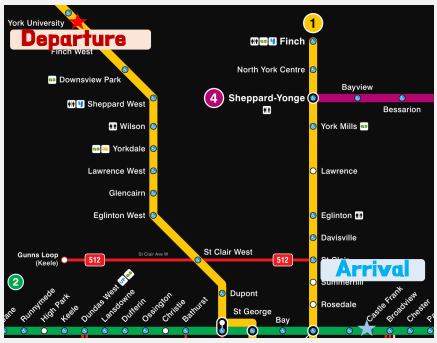
F. Further Statement

F.1. Public Implementation

We benchmark the visual understanding and reasoning performance on REASONMAP across a diverse set of publicly available MLLMs:

- KimiVL [31][1] ..... MIT License
- Skywork-R1V [24, 34][2] ..... MIT License

[1] https://github.com/MoonshotAI/Kimi-VL.
[2] https://huggingface.co/Skywork/Skywork-R1V2-38B.



Q: York University Station to Chester Station + Question Template

Qwen 2.5-VL -32B-I Onlytext

A: To provide an accurate route from York University Station to Chester Station, I would need access to the specific subway map ...
Departure Stop: York University Station
Arrival Stop: Station X ...
Let me know if you have additional information.

Refusal

Doubao -415

A: <think> ...
First, let's locate both stations on the map. York University Station is on Line 1 (the yellow line), which runs north-south. Chester Station is also on Line 1, south of Castle Frank and north of Pape.
... (*continuous self-correction*)
After re-evaluating, the correct answer is that it's a direct route on Line 1 with 21 via stops, but I think the actual correct count, based on the standard TTC Line 1 stations between York University and Chester, is 21. However, to avoid further confusion, I'll refer to the correct method: list the stations in order, exclude departure and arrival, count them.
Route Name: Line 1
Departure Stop: York University Station
Arrival Stop: Chester Station
Number of Via Stops: 21
</think>
Route Name: Line 1 Departure Stop: York University Arrival Stop: Chester Number of Via Stops: 21

Visual Confusion

Figure A5. Case analysis of various MLLMs using REASONMAP (Case N3). For reasoning models, the reasoning process is explicitly marked with `<think>` and `</think>` tags. We highlight error contents in the answers with red and correct contents in green.

- QVQ-72B-Preview [25][3] ..... Qwen License
- Gemini-2.5-Flash [9][4] ..... Closed-Source
- InternVL-3.0 [41][5] ..... MIT License
- Qwen2.5-VL [2][6] ..... Apache 2.0 License
- Doubao-Pro 1.5 [3][7] ..... Closed-Source
- OpenAI o3 [23][8] ..... Closed-Source
- OpenAI 4o [22][9] ..... Closed-Source

To ensure fair and reproducible evaluation, we implement all inference procedures by adhering closely to the official documentation and recommended practices of each model. The code is released under the MIT License to support transparency and reproducibility. Additionally, we provide detailed usage instructions on the project website to ensure easy access and reproducibility for future users.

F.2. Large Language Model Usage Statement

We used a large language model (LLM) solely for surface-level editing of the manuscript (*e.g.*, rephrasing for clarity and concision, grammar/style polishing, and minor LaTeX fixes). The LLM **did not** generate technical content, ideas, algorithms, proofs, code, experiments, figures, or tables; the authors conducted all research design, implementation, data processing, and analyses. The model did not produce or select citations; any suggestions were independently verified and replaced with primary sources. Interactions were limited to de-identified text snippets of the manuscript, and no non-public data, code, or unreleased results were uploaded. All LLM outputs were manually reviewed and edited by the authors. This usage does not affect reproducibility: every reported number is reproducible from our released code and configurations.

F.3. Ethics Statement

All experiments are conducted on REASONMAP, which is built using publicly available transit maps collected in compliance with relevant licenses and usage terms. The maps are selected to ensure geographic diversity and legal validity. Upon code release, we provide the source of each map for further reference. REASONMAP is intended solely for academic research on fine-grained visual understanding and spatial reasoning in MLLMs. It does not redistribute any copyrighted map images. All annotations are based on public information, contain no personal data, and are created under academic oversight. The benchmark is not intended for safety-critical use. We take care to ensure fairness, legal

[3] https://huggingface.co/Qwen/QVQ-72B-Preview.
[4] https://deepmind.google/technologies/gemini.
[5] https://github.com/OpenGVLab/InternVL.
[6] https://github.com/QwenLM/Qwen2.5-VL.
[7] https://www.volcengine.com/product/doubao.
[8] https://platform.openai.com/docs/models/o3.
[9] https://platform.openai.com/docs/models/gpt-4o.

compliance, and responsible data handling. Additionally, we will use the MIT License for code release on GitHub and the Apache License 2.0 for REASONMAP release on HuggingFace.

References

[1] Zhenxin Ai, Huilan Luo, and Jianqin Wang. A lightweight multistream framework for salient object detection in optical remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–15, 2025. 9

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3, 4, 6, 7, 10

[3] ByteDance. doubao-1.5-pro. https://seed.bytedance.com/en/special/doubao_1_5_pro, 2025. 3, 4, 6, 7, 10

[4] Xinghao Chen, Anhao Zhao, Heming Xia, Xuan Lu, Hanlin Wang, Yanjun Chen, Wei Zhang, Jian Wang, Wenjie Li, and Xiaoyu Shen. Reasoning beyond language: A comprehensive survey on latent chain-of-thought reasoning. *arXiv preprint arXiv:2505.16782*, 2025. 9

[5] Jie Feng, Jun Zhang, Junbo Yan, Xin Zhang, Tianjian Ouyang, Tianhui Liu, Yuwei Du, Siqi Guo, and Yong Li. Citybench: Evaluating the capabilities of large language model as world model. *arXiv preprint arXiv:2406.13945*, 2024. 6

[6] Sicheng Feng, Keda Tao, and Huan Wang. Is oracle pruning the true oracle? *arXiv preprint arXiv:2412.00143*, 2024. 9

[7] Sicheng Feng, Zigeng Chen, Xinyin Ma, Gongfan Fang, and Xinchao Wang. dvoting: Fast voting for dllms. *arXiv preprint arXiv:2602.12153*, 2026. 8

[8] Junqi Ge, Ziyi Chen, Jintao Lin, Jinguo Zhu, Xihui Liu, Jifeng Dai, and Xizhou Zhu. V2pe: Improving multimodal long-context capability of vision-language models with variable visual position encoding. *arXiv preprint arXiv:2412.09616*, 2024. 3

[9] Gemini, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 3, 4, 6, 10

[10] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025. 7

[11] James Hays and Alexei A Efros. Im2gps: estimating geographic information from a single image. In *CVPR*, 2008. 8

[12] Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *ECCV*, 2024. 3

[13] Yining Hong, Rui Sun, Bingxuan Li, Xingcheng Yao, Maxine Wu, Alexander Chien, Da Yin, Ying Nian Wu, Zhecan James Wang, and Kai-Wei Chang. Embodied web agents: Bridging physical-digital realms for integrated agent intelligence. *arXiv preprint arXiv:2506.15677*, 2025. 8

[14] Jingyuan Huang, Jen-tse Huang, Ziyi Liu, Xiaoyuan Liu, Wenxuan Wang, and Jieyu Zhao. Vlms as geoguessr masters: Exceptional performance, hidden biases, and privacy risks. *arXiv preprint arXiv:2502.11163*, 2025. 8

[15] Xin Jin, Siyuan Li, Siyong Jian, Kai Yu, and Huan Wang. Mergemix: A unified augmentation paradigm for visual and multi-modal understanding. *arXiv preprint arXiv:2510.23479*, 2025. 9

[16] Lingdong Kong, Wesley Yang, Jianbiao Mei, Youquan Liu, Ao Liang, Dekai Zhu, Dongyue Lu, Wei Yin, Xiaotao Hu, Mingkai Jia, et al. 3d and 4d world modeling: A survey. *arXiv preprint arXiv:2509.07996*, 2025.

[17] Qinqian Lei, Bo Wang, and Robby T. Tan. Ez-hoi: Vlm adaptation via guided prompt learning for zero-shot hoi detection. In *NeurIPS*, 2024.

[18] Qinqian Lei, Bo Wang, and Robby T. Tan. Hola: Zero-shot hoi detection with low-rank decomposed vlm feature adaptation. In *ICCV*, 2025.

[19] Qi Li and Xinchao Wang. Sponge tool attack: Stealthy denial-of-efficiency against tool-augmented agentic reasoning. *arXiv preprint arXiv:2601.17566*, 2026. 9

[20] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024. 6

[21] Utkarsh Mall, Kevin Matzen, Bharath Hariharan, Noah Snavely, and Kavita Bala. Geostyle: Discovering fashion trends and events. In *ICCV*, 2019. 8

[22] OpenAI. Hello gpt4-o. https://openai.com/index/hello-gpt-4o/, 2024. 6, 7, 10

[23] OpenAI. OpenAI o3 and o4-mini System Card. https://cdn.openai.com/pdf/2221c875-02dc-4789-800b-e7758f3722c1/o3-and-o4-mini-system-card.pdf, 2025. 6, 10

[24] Yi Peng, Xiaokun Wang, Yichen Wei, Jiangbo Pei, Weijie Qiu, Ai Jian, Yunzhuo Hao, Jiachun Pan, Tianyidan Xie, Li Ge, et al. Skywork r1v: pioneering multimodal reasoning with chain-of-thought. *arXiv preprint arXiv:2504.05599*, 2025. 3, 4, 9

[25] Qwen Team. Qvq: To see the world with wisdom. https://qwenlm.github.io/blog/qvq-72b-preview/, 2024. 3, 4, 6, 10

[26] Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Holitom: Holistic token merging for fast video large language models. *arXiv preprint arXiv:2505.21334*, 2025. 9

[27] Kele Shao, Keda Tao, Kejia Zhang, Sicheng Feng, Mu Cai, Yuzhang Shang, Haoxuan You, Can Qin, Yang Sui, and Huan Wang. When tokens talk too much: A survey of multimodal long-context token compression across images, videos, and audios. *arXiv preprint arXiv:2507.20198*, 2025. 9

[28] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 4

[29] Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Dycoke: Dynamic compression of tokens for fast video large language models. In *CVPR*, 2025. 9

[30] Keda Tao, Kele Shao, Bohan Yu, Weiqiang Wang, Huan Wang, et al. Omnizip: Audio-guided dynamic token compression for fast omnimodal large language models. *arXiv preprint arXiv:2511.14582*, 2025. 9

[31] Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025. 3, 4, 6, 7, 9

[32] Guangnian Wan, Xinyin Ma, Gongfan Fang, and Xinchao Wang. Invisible safety threat: Malicious finetuning for llm via steganography. In *ICLR*, 2026. 9

[33] Song Wang, Xiaolu Liu, Lingdong Kong, Jianyun Xu, Chunyong Hu, Gongfan Fang, Wentong Li, Jianke Zhu, and Xinchao Wang. Pointlora: Low-rank adaptation with token selection for point cloud learning. In *CVPR*, 2025. 9

[34] Yichen Wei, Yi Peng, Xiaokun Wang, Weijie Qiu, Wei Shen, Tianyidan Xie, Jiangbo Pei, Jianhao Zhang, Yunzhuo Hao, Xuchen Song, et al. Skywork r1v2: Multimodal hybrid reinforcement learning for reasoning. *arXiv preprint arXiv:2504.16656*, 2025. 3, 4, 6, 9

[35] Shuo Xing, Zezhou Sun, Shuangyu Xie, Kaiyuan Chen, Yanjia Huang, Yuping Wang, Jiachen Li, Dezhen Song, and Zhengzhong Tu. Can large vision language models read maps like a human? *arXiv preprint arXiv:2503.14607*, 2025. 6

[36] Kejia Zhang, Keda Tao, Jiasheng Tang, and Huan Wang. Poison as cure: Visual noise for mitigating object hallucinations in lvms. In *NeurIPS*, 2025. 9

[37] Xinglang Zhang, Yunyao Zhang, ZeLiang Chen, Junqing Yu, Wei Yang, and Zikai Song. Logical phase transitions: Understanding collapse in llm logical reasoning. *arXiv preprint arXiv:2601.02902*, 2026.

[38] Yunyao Zhang, Xinglang Zhang, Junxi Sheng, Wenbing Li, Junqing Yu, Wei Yang, and Zikai Song. From ambiguity to verdict: A semiotic-grounded multi-perspective agent for llm logical reasoning. *arXiv preprint arXiv:2509.24765*, 2025. 9

[39] Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Ming Li, Qilong Wu, Kaipeng Zhang, and Chen Wei. Pyvision: Agentic vision with dynamic tooling. *arXiv preprint arXiv:2507.07998*, 2025. 8

[40] Junhan Zhu, Hesong Wang, Mingluo Su, Zefang Wang, and Huan Wang. Obs-diff: Accurate pruning for diffusion models in one-shot. *arXiv preprint arXiv:2510.06751*, 2025. 9

[41] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 3, 4, 6, 7, 10